Publication Series of the John von Neumann Institute for Computing (NIC)
NIC Series Volume 3

John von Neumann Institute for Computing (NIC)

Johannes Grotendorst (Editor)

Modern Methods and Algorithms of Quantum Chemistry

Winterschool, 21 - 25 February 2000

Forschungszentrum Jülich, Germany

Proceedings, Second Edition

organized by

John von Neumann Institute for Computing

in cooperation with

Arbeitsgemeinschaft für Theoretische Chemie

NIC Series Volume 3

ISBN 3-00-005834-6

Die Deutsche Bibliothek – CIP-Cataloguing-in-Publication-Data
A catalogue record for this publication is available from Die Deutsche Bibliothek.

Publisher: NIC-Directors

Distributor: NIC-Secretariat
Research Centre Jülich
52425 Jülich
Germany

Internet: www.fz-juelich.de/nic

Printer: Graphische Betriebe, Forschungszentrum Jülich

NIC Series Volume 3
ISBN 3-00-005834-6

PREFACE

Computational quantum chemistry has long promised to become a major tool for the study of molecular properties and reaction mechanisms. The fundamental methods of quantum chemistry date back to the earliest days of quantum mechanics in the first decades of the twentieth century. However, widespread quantitative applications have only become common practice in recent times, primarily because of the explosive developments in computer hardware and the associated achievements in the design of new and improved theoretical methods and computational techniques. The significance of these advances in computational quantum chemistry is unterlined by the 1998 chemistry Nobel prize to Walter Kohn and John Pople; this award also documents the increasing acceptance of computer simulations and scientific computing as an important research method in chemistry.

Nearly one third of the projects which use the supercomputing facilities provided by the John von Neumann Institute for Computing (NIC) pertain to the area of computational chemistry. For projects in quantum chemistry the Central Institute for Applied Mathematics (ZAM) which runs the supercomputers and networks at the Research Centre Jülich offers several extensive software packages running on its Cray supercomputer complex. The computational requirements of large quantum-chemical calculations are enormous. They have made the use of parallel computers indispensable and have led to the development of a broad range of advanced algorithms for these machines.

This interdisciplinary Winterschool brought together experts from the fields of quantum chemistry, computer science and applied mathematics in order to present recent methodological and computational advances to research students in the field of theoretical chemistry and their applications. The participants were also informed about new software developments and about implementation issues that are encountered in quantum chemistry codes, particularly in the context of high-performance computing (topics not yet included in typical university courses). The major focus of the Winterschool was on method development and algorithms, but state-of-the-art applications were also demonstrated for illustration. The following topics were covered by twenty lectures:

- Density functional theory
- Ab initio molecular dynamics
- Post-Hartree-Fock methods
- Molecular properties
- Heavy-element chemistry
- Linear scaling approaches
- Semiempirical and hybrid methods
- Parallel programming models and tools
- Numerical techniques and automatic differentiation
- Industrial applications

The programme was compiled by Johannes Grotendorst (Research Centre Jülich), Marius Lewerenz (Université Pierre et Marie Curie, Paris), Walter Thiel (Max-Planck-Institut für Kohlenforschung, Mülheim an der Ruhr) and Hans-Joachim Werner (University of Stuttgart).

Fostering education and training in important fields of scientific computing by symposia, workshops, schools and courses is a major objective of NIC. This Winterschool continued a series of workshops and conferences in the field of computational chemistry organized by the ZAM in the last years; it provided a forum for the scientific exchange between young research students and experts from different academic disciplines. More than two hundred participants from sixteen countries took part in the NIC Winterschool, and more than fifty contributions were submitted for the poster session. This overwhelming international resonance clearly reflects the attractiveness of the programme. The excellent support of the Arbeitsgemeinschaft für Theoretische Chemie in preparing the Winterschool is highly appreciated.

As in previous conferences, many people have made significant contributions to the success of this Winterschool. The local organization at Research Centre Jülich was perfectly done by Elke Bielitza, Rüdiger Esser, Bernd Krahl-Urban, Monika Marx, Renate Mengels, and Margarete Reiser. We are grateful for the generous financial support by the Federal Ministry for Education and Research (BMBF) and by the Research Centre Jülich and for the help provided by its Conference Service. We thank the authors for their willingness to provide a written version of their lecture notes. Special thanks go to Monika Marx for her commitment concerning the compilation and editing of this book. Finally, we are indebted to Beate Herrmann who supported the difficult typesetting with professionalism and great care.

This second edition of the proceedings includes the material of all the lectures held at the NIC Winterschool and now gives a broad review on modern methods and computational techniques in quantum chemistry. In order to increase the circulation among research students and scientists working in the field of computational quantum chemistry the lecture notes are also available on the web (www.fz-juelich.de/nic-series).

Jülich
November 2000

Johannes Grotendorst

CONTENTS

INDUSTRIAL CHALLENGES FOR QUANTUM CHEMISTRY

ANSGAR SCHÄFER
BASF Aktiengesellschaft
Scientific Computing
ZDP/C - C13
67056 Ludwigshafen
Germany
E-mail: ansgar.schaefer@basf-ag.de

The current fields of application of quantum chemical methods in the chemical industry are described. Although there are a lot of questions that already can be tackled with modern algorithms and computers, there are still important problems left which will need further improved methods. A few examples are given in this article.

1 Introduction

Already in the 1970's and 80's, quantum chemical methods were very successful in describing the structure and properties of organic and main-group inorganic molecules. The Hartree-Fock (HF) method and its simplified semi-empirical modifications became standard tools for a vivid rationalization of chemical processes. The underlying molecular orbital (MO) picture was, and still is, the most important theoretical concept for the interpretation of reactivity and molecular properties. Nevertheless, quantum chemical methods were not used extensively for industrial problems, although most of the industrial chemistry produces organic compounds. One reason can be found in the fact that almost all industrial processes are catalytic. The catalysts are predominantly transition metal compounds, which in general have a more complicated electronic structure than main-group compounds, since their variability in the occupation of the d orbitals results in a subtle balance of several close lying energy levels. HF and post-HF methods based on a single electron configuration are not able to describe this situation correctly. Furthermore, the catalyst systems were generally too big to be handled. The usual approach was to choose small model systems, e.g. with PH3 substituting any phosphine ligand or a cluster representing a solid surface. Such investigations provided only a basic understanding of the catalytic reaction, but no detailed knowledge on steric and electronic dependencies.

With the improvement of both the methodology and the algorithms of density functional theory (DFT) in the last two decades, the situation changed significantly. DFT appears to be less sensitive to near degeneracy of electronic states, and furthermore incorporates some effects of electron correlation. The development of new functionals with improved description of non-uniform electron distributions in molecules or on surfaces, paved the way for a qualitative or even quantitative quantum chemical treatment of a large variety of transition metal compounds and their reactions. When functionals without partial inclusion of HF exchange contributions are used, an approximate treatment of the Coulomb interaction of the electrons (density fitting, resolution of identity (RI) approach) allows for a very ef-

ficient treatment of large systems. Therefore, with efficiently parallelized programs of this kind, it is routinely possible today to calculate the structure of molecules with 100-200 atoms.

Although the chemical processes in many cases involve transition metal compounds, calculations on pure organic molecules are still important to predict properties like thermodynamic data or various types of spectra. However, for a quantitative agreement between calculated and experimental results, DFT very often is not reliable enough, and approaches going beyond the HF approximation are necesary to assess the effects caused by electron correlation. These methods are computationally very demanding, and highly accurate calculations are still limited to small molecules with not more than about 10 atoms. Therefore, still only few problems of industrial relevance can be tackled by these methods at the moment.

2 Application fields of quantum chemistry in industry

2.1 Catalysis

As already mentioned, many of the industrial chemical processes involve catalysts. Most of the catalysts are in the solid state (heterogeneous catalysis), but, with the extensive developments in organometallic chemistry in the last decades, catalytic processes in the liquid phase become more and more important (homogeneous catalysis). For the development of a catalyst, besides economic considerations, three issues are of central importance:

- Activity: The catalyst must be efficient (high turnover numbers).
- Selectivity: By-products should be avoided.
- Stability: Deactivation or decomposition of the catalyst must be slow.

To improve a catalyst with respect to these criteria, a detailed understanding of the reaction pathways is necessary. This is one point, where quantum chemical methods can be of enormous value, since the experimental investigation of steps in a complicated mechanism is rather difficult. Once the crucial parameters are found, new guesses for better performing catalysts can be deduced and immediately be tested in the calculations. Thus, when theory is used for a rough screening and only the most promising candidates have to be tested in experiment, the development process for new catalysts can be shortened significantly. DFT is used almost exclusively for both homogeneous and heterogeneous applications. In the latter case, solids are treated with preriodic boundary conditions or QM/MM approaches.

2.2 Process design

The design of chemical processes and plants requires the knowledge of accurate thermodynamic and kinetic data for all substances and reactions involved. The experimental determination of such data is rather time-consuming and expensive. The substances have to be prepared in high purity and precise calorimetric or kinetic measurements have to be done for well-defined reactions. This effort must

be invested before the actual technical realization of a new process, because for economic and safety reasons, the data has to be as accurate as possible. However, for an early assessment of the practicability and profitability of a process, a fast but nevertheless reliable estimate for the thermodynamics usually is sufficient. A number of empirical methods based on group contributions are used for this purpose, but they are not generally applicable and often not reliable enough. For example, these methods often can not discriminate between isomers containing the same number and type of groups. Alternatively, reaction enthalpies and entropies can also be calculated by quantum chemical methods in combination with statistical mechanics. For the rotational and vibrational energy levels, a correct description of the molecular structure and the shape of the energy surface is needed, which can very efficiently be obtained with DFT. The crucial point for the overall accuracy, however, are the differences in electronic energies, and high level ab initio methods like coupled cluster theories are often needed to get the error down to a few kcal/mol. That such a high accuracy is needed can be illustrated by the fact, that a change in the Gibbs free energy of only 1.4 kcal/mol already changes an equilibrium constant by an order of magnitude at room temperature.

2.3 Material properties

When a desired property of a material can be connected to quantities on the atomic or molecular scale, quantum chemistry can be a useful tool in the process of improving such materials. Typical examples are dyes and pigments, for which color and brilliance depend on the energies and nature of electronic excitations. Organic dyes typically have delocalized pi systems and functional groups chosen appropriately to tune the optical properties. Since the molecules normally are too big for an ab initio treatment with the desired accuracy, semi-empirical methods are currently used to calculate the excited states.

3 Unsolved problems

3.1 Treatment of the molecular environment

Quantum chemical methods are predominantly applied to isolated molecules, which corresponds to the state of an ideal gas. Most chemical processes, however, take place in condensed phase, and the interaction of a molecule with its environment can generally not be neglected. Two important examples are given here.

3.1.1 Solvent effects

Solvent molecules can directly interact with the reacting species, e.g. by coordination to a metal center or by formation of hydrogen bonds. In such cases it is necessary to explicitely include solvent molecules in the calculation. Depending on the size of the solvent molecules and their number needed to get the calculated properties converged, the overall size of the molecular system and the resulting computational effort can significantly be increased. Currently, only semi-empirical methods are able to handle several hundred atoms, but the developments towards

linear scaling approaches in DFT are very promising. An alternative would be a mixed quantum mechanical (QM) and molecular mechanical (MM) treatment (QM/MM method).

If there are no specific solute-solvent interactions, the main effect of the solvent is electrostatic screening, depending on its dielectric constant. This can be described very efficiently by continuum solvation models (CSM).

3.1.2 Enzymes

Biomolecules like enzymes usually consist of many thousands of atoms and therefore can not be handled by quantum chemical methods. Although there exist several very elaborate force field methods which quite reliably reproduce protein structures, the accurate description of the interaction of the active site with a ligand bound to it still is an unsolved problem. Quantum mechanics is needed for a quantitative assessment of polarization effects and for the description of reactions. Treating only the active site quantum mechanically usually does not give the correct results because of the strong electrostatic influence of the whole enzyme and the water molecules included. QM/MM approaches can be used for the treatment of the whole system, but the QM/MM coupling still needs improvement.

A very important quantity of an enzyme/ligand complex is its binding free energy. It determines the measured equilibrium constant for a ligand exchange and is therefore important for the development of enzyme inhibitors. A reliable calculation of such binding constants is currently not possible. It certainly must involve quantum chemistry and molecular dynamics.

3.2 Accurate thermochemistry and kinetics

The highly accurate calculation of thermochemical data with ab initio methods is currently possible only for small molecules up to about 10 atoms. However, many of the data for molecules of this size are already known, whereas accurate experiments for larger compounds are quite rare. Therefore, efficient ab initio methods are needed which are able to treat molecules with 30-50 atoms with the same level of accuracy. The currently developed local treatments of electron correlation are very promising in this direction.

Another problem arises for large molecules. They often have a high torsional flexibility, and the calculation of partition functions based on a single conformer is therefore not correct. Quantum molecular dynamics could probably give better answers, but is in many cases too expensive.

For the assessment of catalytic mechanisms, kinetic data are needed to discriminate between different reaction pathways. As described above, ab initio methods are often not applicable when transition metals are involved. The DFT results for reaction energies are usually reliable enough to give correct trends, but calculated activation barriers easily can be wrong by a factor of two or more. The problem lies in the functionals, which best describe the electron distribution in the molecular ground states.

3.3 Spectroscopy for large molecules

Calculated molecular spectroscopic properties are very helful in the assignment and interpretation of measured spectra, provided that the accuracy is sufficiently high. In many cases IR, Raman and NMR spectra can be obtained with reasonable accuracy on DFT or MP2 level, but UV/VIS spectra normally require more elaborate theories like configuration interaction, which are only applicable to very small molecules. To improve on the currently applied semi-empirical approaches it would be necessary to calculate accurate excitation energies also for, e.g., organic dyes with 50 or even more atoms.

4 Conclusion

The improvement of the efficiency of the algorithms and the enormous increase of the available computer power already made quantum chemistry applicable to a lot of industrial problems. However, there are still many aspects concerning the accuracy, completeness and efficiency of the quantum chemical treatment, which will need more attention in the future.

Currently, DFT is the most widely used quantum theoretical method in industry. Also of high importance are semi-empirical methods for certain applications. Both approaches have in common that they do not offer a systematic way for the improvement of results, if they are found to be not reliable enough. Therefore, there is still need for efficient ab initio methods, at least as a reference.

AB INITIO TREATMENT OF LARGE MOLECULES

REINHART AHLRICHS, SIMON D. ELLIOTT AND UWE HUNIAR
Lehrstuhl für Theoretische Chemie, Universität Karlsruhe
Institut für Physikalische Chemie und Elektrochemie
Kaiserstr. 12, 76128 Karlsruhe
Germany
E-mail: ramail@tchibm3.chemie.uni-karlsruhe.de, simon.elliott@tcd.ie
uwe@tchibm3.chemie.uni-karlsruhe.de

We give a brief exposition of the fundamental approximations of ab initio calculations, the SCF, DFT and MP2 methods for calculating molecular wavefunctions and energies are introduced. The Resolution of Identity (RI) approach is described with a view to the treatment of large molecules (100 atoms and more). Scaling behavior of the various computational methods is compared; here, aluminium clusters represent a demanding application. The utility of analytical energy gradients is shown, in particular using the example of a theoretical study of the structural isomers of sulfur-bridged copper clusters. A strategy to compute electronic excitation energies for large molecules is sketched and applications to fullerenes and cadmium-selenide nanoclusters are presented. Developments in scientific computing hardware are considered, with emphasis on the emergence of PC's. The opportunities and difficulties inherent in the parallelization of quantum chemical code are also discussed, and the performance of parallel TURBOMOLE is presented.

1 Introduction

Computing in the natural sciences and engineering has gained considerably in importance as a result of the ever increasing power of available hardware. As a consequence it is justified to speak of the tripod of science: experiment, theory and scientific computing. The first two of these branches should not require an explanation. The third, scientific computing, comprises "computer experiments" which give exact results - up to rounding errors and errors in computer programs - within a model ultimately defined by the computational procedures applied, i.e. by the assumptions on which they are based. The better the model and the justification for assumptions, the better and the more realistic the results. Scientific computing thus permits the prediction and checking of results of experiments and especially their interpretation. This clearly will never make experiments superfluous - but it will change and has already changed the way in which experiments are designed and selected. We just remind the reader of the importance of computer modelling in the design of air planes, jet engines, cars, and computers, in oil exploration and weather forecasting, to name just a few fields essentially shaped by modern simulation techniques.

In the present article we deal with molecular electronic structure theory [1,2], the application of Quantum Mechanics to determine properties - "observables" - of molecular systems. The central problem is here the approximate but sufficiently accurate solution of the molecular Schrödinger equation. This is a formidable task, a real grand challenge, simply because of the dimensionality: the wavefunction describing a system of 100 electrons and nuclei is a function of 300 cartesian coordinates (all problems mentioned above are three-dimensional cases), which is simply

not manageable in an accurate way. Despite these apparently insurmountable difficulties, chemists now almost routinely perform calculations for systems with a few hundred atoms, which give results that are competitive in accuracy and effort with measurements. Some aspects of this methodology will be sketched in this article, where "large molecules" are those with of the order of 100 atoms and more.

2 The Zoo of Methods

2.1 The Standard Approximations: MO-LCAO-CGTO [2]

We separate the treatment of electrons and nuclei by means of the Born Oppenheimer approximation and are then left with the electronic Schrödinger equation. Approximate ab initio treatments are almost invariably based on the variation principle which requires an evaluation of

$$E[\psi] = \langle\psi|H|\psi\rangle/\langle\psi|\psi\rangle \tag{1}$$

$$H = \sum_i h(i) + \sum_{i<j} 1/r_{ij} \tag{2}$$

The one-electron terms h include the kinetic energy of electrons and their interaction with the nuclei; in the two-electron terms r_{ij} denotes the distance between electrons i and j. We have sufficient knowledge of properties of exact wavefunctions to guess a high quality ansatz for ψ, but it is then impossible to evaluate the $3n$-dimensional integrals in (1), n = number of electrons, with sufficient accuracy and reasonable effort. The structure of high quality wavefunctions has been extensively probed in treatments of small atoms and molecules, especially He and H_2.

An application of the variation principle requires an ansatz for ψ for which integration is easy in general: all integrals should factorize into low dimensional cases. This is achieved by building ψ from the one-electron functions φ_i called molecular orbitals (MO), and this leads to the general configuration interaction (CI) ansatz

$$\psi = \sum_I C_I \Phi_I \tag{3}$$

$$\Phi_I = [\varphi_{i1}...\varphi_{in}] \tag{4}$$

$$\langle\varphi_i|\varphi_j\rangle = \delta_{ij} \tag{5}$$

The square brackets in (4) indicate the assignment of spins and antisymmetrization, the formation of Slater determinants.

As a result of (3)-(5), the evaluation of $E[\psi]$ is reduced to simple three and six-dimensional integrals

$$h_{ij} = \int \varphi_i h \varphi_j d\tau^3 \tag{6}$$

$$(ij|kl) = \int \varphi_i(r_1)\varphi_j(r_1)1/r_{12}\varphi_k(r_2)\varphi_l(r_2)d\tau^6 \tag{7}$$

This solves the problem of integration. However, the number of configurations to be included in (3) to achieve a prescribed accuracy basically grows exponentially with the size of the system: CI treatments are in general not feasible for large molecules at present.

The MO ansatz makes an application of the variation principle possible; efficiency of an implementation depends on the actual representation of MO's φ_i. The success of Quantum Chemistry is essentially based on the use of Gaussian functions: the MO's φ_i are approximated as "linear combinations of atomic orbitals" (LCAO) f_μ

$$\varphi_i = \sum_\mu C_{\mu i} f_\mu \tag{8}$$

and the atom centered basis functions f_μ are "contracted Gauss type orbitals" (CGTO)

$$f_\mu = \left\{ \sum_\gamma d_\gamma exp(-\eta_\gamma r^2) \right\} \mathrm{Mon}(x, y, z). \tag{9}$$

$\mathrm{Mon}(x, y, z)$ denotes a monomial, i.e. Mon = 1 for an s function, Mon = x or y or z for p functions and so on. The fixed parameters in (9) - contraction coefficients d_γ and exponents η_γ - have typically been optimized in calculations of atoms. If (8) is plugged into (6) and (7) one is left with integrals $h_{\nu\mu}$ and $(\nu\mu|\kappa\lambda)$, defined in complete analogy to (6) and (7) by replacing φ_i by f_ν, etc. These integrals $h_{\nu\mu}$ and $(\nu\mu|\kappa\lambda)$ over Gaussians are simple analytic expressions which can be evaluated very efficiently: around 20 additions and multiplications per primitive integral.

2.2 Single Determinant Wavefunctions: SCF and DFT

The simplest MO ansatz includes only a single term, $\psi_{SCF} = \Phi_1$, in the CI expansion (3). This is a crude approximation and reasonable results can only be expected if the MOs occupied in ψ_{SCF} are variationally optimized: the Self-Consistent Field (SCF) approach. Despite its simplicity this is a very useful approximation. Equilibrium structure constants of most main group compounds are obtained with errors of about 2% in bond distances and a few degrees in bond angles; simple examples are discussed in [2]. Exceptions are mainly found for weak bonds like those in F_2 or the N-N bond in N_2O_4 where errors can be considerably larger. We leave it to the reader to compare this with the accuracy of experimental structure determinations typically achieved in the gas phase or in crystals for large molecules. The SCF approach is useless for binding energies but reaction energies can be computed to within 10 kJ/mol for isodesmic and especially homodesmic reactions. The SCF approximation also fails for molecules containing transition metals; only cases that are formally d^0 can be treated since they more or less behave as main group elements.

Many deficiencies of the SCF approach are rectified by density functional theory (DFT) methods. DFT is easy to characterize if one takes a very pragmatic attitude. Choosing for simplicity a closed shell system, we write the SCF energy expression as

$$E_{SCF} = E^{(1)} + J - E_x \tag{10}$$

$$E^{(1)} = 2\sum_i h_{ii} \tag{11}$$

$$J = 2\sum_{ij}(ii|jj) = \frac{1}{2}\int \rho(r_1)1/r_{12}\rho(r_2)d\tau^6 \tag{12}$$

$$E_x = \sum_{ij}(ij|ij) \tag{13}$$

$$\rho(r) = 2\sum_i |\varphi_i(r)|^2 = \sum_{\nu\mu} D_{\nu\mu} f_\nu(r) f_\mu(r)\ . \tag{14}$$

In a remarkable paper Kohn and Sham established - a correct proof was given later by others - the existence of an "exchange correlation" functional $E_{xc}[\rho]$ such that [3,4]

$$E_{DFT} = E^{(1)} + J - E_{xc}[\rho] \tag{15}$$

yields the exact density ρ and the exact ground state energy, $E_{DFT} = \mathrm{E}_0$, provided (15) is minimized with respect to the MO's φ_i. The exact functional $E_{xc}[\rho]$ is unknown, also unknown are systematic procedures to derive (better and better) functionals from the exact Schrödinger equation. Approximate functionals are thus derived from a study of the underlying physics, by considerations of model systems, and by simple fits (e.g fitting E_x to SCF results for atoms or to experimental data). The dominant contribution to E_{xc} is, of course, the Dirac approximation for exchange: $E_x \approx \mathrm{const} \cdot \int \rho^{4/3} d\tau^3$.

This pragmatic approach to determining exchange correlation functionals has been very successful. DFT yields much improved energetics for main group compounds [5]. Even more importantly, DFT methods also yield reasonable results for transition metal compounds and metal clusters for which SCF is essentially useless. There are still problems: DFT quite consistently overestimates bond distances and the density vanishes too slowly at larger distances from nuclei.

DFT has repeatedly been characterized as an essentially semiempirical method, quite recently by Handy and coworkers [6]. If one accepts this judgement, one has to add that DFT introduces a new quality of semiempirical procedures since the level of reasoning and the foundation of the approximations are on a much higher level than for example for CNDO or MNDO and its variants.

2.3 MP2: Second Order Perturbation Corrections to SCF

The only other method presently applicable to large molecules is MP2 (Møller-Plesset second order pertubation correction) which adds the dominant effects of electron correlation to the SCF treatment. Two comments about MP2 should be made. MP2 is the first term of a perturbation expansion based on SCF as zeroth order. There is no way to establish or estimate the convergence properties of this series in general. MP2 typically improves SCF if the latter is already a useful approximation, and errors with respect to experiment or reliable high level calculations are reduced to roughly 30% (compared to SCF), e.g. in NMR chemical shifts [7]. Applicability is thus restricted to most of main group chemistry, transition metals with d^0 and in addition with d^{10} atomic states. The other comment concerns basis sets: MP2 requires use of (much) larger basis sets than needed for SCF. MP2 treatments require, in addition to a preceeding SCF calculation, the computation of two-electron integrals $(ia|jb)$, e.g. (7), for all pairs of occupied (i,j) and virtual (a,b) MO's. The computational effort is thus much larger than for SCF.

2.4 Gradient Techniques and Properties [8]

The development of techniques to evaluate analytical gradients, i.e.

$$E_\lambda^{(1)} = \partial E/\partial\lambda \tag{16}$$

$$E_{\lambda\mu}^{(2)} = \partial^2 E/\partial\lambda\partial\mu \tag{17}$$

of the electronic energy (SCF, DFT, MP2, etc.) with respect to external parameters (λ,μ) has been essential for the success of Quantum Chemistry. The programs for (16) and (17) are clearly more complicated than those for the corresponding energy - but one gets used to it: last year's sensation is this year's calibration. Analytical gradients have a direct relationship to molecular properties or are at least very useful in their determination, as the following examples show.

$\lambda = X_\mu$, a nuclear coordinate: The gradient $E_\lambda^{(1)}$ is then the (negative) force component acting on the nucleus. Knowledge of the gradient is very useful for structure relaxations to locate the minima of the potential energy hypersurface which define molecular isomers and conformers.

$\lambda = X_\nu, \mu = X_\kappa$: The second derivatives $E_{\lambda\kappa}^{(2)}$ completely specify the potential for the treatment of nuclear dynamics in the harmonic approximation, i.e. infrared and Raman spectra.

λ,μ = components of external electric or magnetic fields: $E_{\lambda\mu}^{(2)}$ is the polarizability or susceptibility tensor.

λ = component of the magnetic field, μ = component of the nuclear magnetic moment: $E_{\lambda\mu}^{(2)}$ gives the chemical shielding of NMR, and chemical shifts by comparison with a standard. We note in passing that the treatment of magnetic properties has caused many problems which are connected with achieving invariance with respect to the gauge of (magnetic) vector potentials [9].

The advantage of analytical gradient evaluations is again efficiency: the evaluation of the complete gradient with respect to all nuclear coordinates is faster than

the computation of the energy for SCF and DFT, and only about a factor 3 more expensive for MP2.

2.5 Electronic Excitations

The treatment of electronic excitation generally requires extended CI calculations which are not feasible for large molecules. If a computational procedure is envisaged for this purpose which is comparable in effort to SCF or DFT, one is led to consider Time-Dependent SCF and especially Time-Dependent DFT (TDDFT) [10]. Considering the molecule under the influence of an electric field with frequency ω, one computes $\alpha(\omega)$, the frequency dependent polarizability - technically a modification of the static polarizability computations mentioned above. Electronic excitations are then obtained as the poles of $\alpha(\omega)$, i.e. by putting $\alpha(\omega) = \infty$ in the corresponding equations. TDDFT appears to be more accurate than the SCF analogue [11] - usually called SCF-RPA (random phase approximation) - which in turn is closely related to a CI with the SCF reference and all single excitations (SCI). A final note: TDDFT can be expected to be of use (relative errors of a few tenths of an eV) only if excited states are sufficiently well described by single excitations - at least for the DFT functionals presently in use.

2.6 RI Methods (Resolution of the Identity)

Electronic structure calculations are essentially a struggle with the consequences of interelectronic interactions, i.e. the computation and processing of two-electron integrals $(\nu\mu|\kappa\lambda)$, Eq. (7), within the LCAO-MO approximation. In attempts to reduce the large number of two-electron integrals, one has tried to approximate products of basis functions, $f_\nu f_\mu$, by a set of atom-centered auxiliary (or fitting) basis functions g, ususally labelled by indices α or β:

$$\Delta_{\nu\mu} = f_\nu f_\mu - \sum_\alpha C_\alpha g_\alpha \approx 0 \ . \tag{18}$$

The late Jan Almlöf and coworkers have shown that the appropriate metric for (18) in the context of interelctronic interactions is given as [12]

$$||\Delta_{\nu\mu}||^2 = (\Delta_{\nu\mu}|\Delta_{\nu\mu}) = \int \Delta_{\nu\mu}(r_1) 1/r_{12} \Delta_{\nu\mu}(r_2) d\tau^6 = \min \ . \tag{19}$$

The condition (19) leads, after standard manipulations, to the following replacement

$$(\nu\mu|\kappa\lambda) \approx \sum_{\alpha\beta} (\nu\mu|\alpha)(\alpha|\beta)^{-1}(\beta|\kappa\lambda) \tag{20}$$

where $(x|y)$ is as in (7) or (19) and where $(\alpha|\beta)^{-1}$ denotes the inverse matrix. This approximation formally resembles the Resolution of the Identity in Hilbert space theory for non-orthogonal basis functions: hence the acronym RI.

The r.h.s. of (20) looks more complicated than the l.h.s., but using (20) may still be advantageous since it only involves two- and three-index quantities and no four-index quantities. The most pronounced gain in efficiency results if (20) is used for the approximate treatment of the Coulomb term J, eq. (12), giving the RI-J approximation

$$J \approx \sum_{\nu\mu} D_{\nu\mu}(\nu\mu|\alpha) \left[\sum_{\beta} (\alpha|\beta)^{-1} \left\{ \sum_{\kappa\lambda} (\beta|\kappa\lambda) D_{\kappa\lambda} \right\} \right] \tag{21}$$

This is employed profitably in DFT treatments where only J is present and E_{xc} is treated separately (Eq. (15)), which leads to the RI-DFT method discussed in [13]. No way has been found so far to exploit (20) in connection with exchange terms (13). On the other hand, quite pronounced gains in efficiency are achieved for MP2 calculations: the RI-MP2 approximation [14].

The RI approximation is of use only if auxiliary basis sets g_α are available for which the loss of accuracy incurred by (20) can be controlled. This is fortunately the case for the RI-DFT [13,15] and the RI-MP2 methods [16]: the errors introduced are without consequences, since they are much smaller than those that result anyway from typical basis sets. (For the experts: one achieves an accuracy of obout 50 $\mu H \approx 0.1$ kJ/mol per atom and the corresponding errors largely cancel for energy differences.)

2.7 Scaling Behavior

For treatments of large molecules it is essential to establish the so called "scaling behavior": the way in which computational effort increases with increasing molecular size. Theoreticians usually take the number of basis functions, N, as a measure of molecular size (the number of atoms would also do).

It is important to distinguish between *formal* and *asymptotic* scaling, as will be explained in this example. The total number of two-electron integrals $(\nu\mu|\kappa\lambda)$ clearly increases with N^4, the formal scaling. However in practice one can neglect sufficiently small integrals. Since the differential overlap $f_\nu f_\mu$ vanishes when the corresponding centers of basis functions are far apart, the number of $f_\nu f_\mu$ to be considered grows only in proportion to N, and the number of $(\nu\mu|\kappa\lambda)$ in proportion to N^2. This is the asymptotic scaling behavior for large molecules.

Analoguous considerations lead to the following results

Computational step	Scaling: formal $\rightarrow$ asymptotic
$(\nu\mu\|\kappa\lambda)$	$N^4 \rightarrow N^2$
RI-J (Eq. (21))	$N^3 \rightarrow N^2$
Quadrature	$N^3 \rightarrow N$
Linear algebra	$N^3 \rightarrow N^3$
MP2	$N^5 \rightarrow N^3$

Quadrature denotes here the numerical integration necessary for DFT to evaluate E_{xc} (which cannot be integrated analytically). Linear algebra includes the usual matrix operations: product, inversion, and diagonalization. Since the investigation of scaling behavior is a very active field of research [17,18,19,20] and since discussions are sometimes confusing, a few comments appear appropriate.

Quantum chemists have always tried to avoid unnecessary computations and thus achieve better scaling behavior. As an example we mention the timings for a series of Al clusters which include between 13 and 147 atoms, i.e. 200 to 3000 CGTO basis functions, for a gradient calculation within the RI-DFT method [13,21]. To save computer time O_h symmetry was used (in the comparison) although this is distorted by Jahn Teller effects in most cases. The total times never show the formal N^3 behavior. The CPU times increase like $N^{2.3}$ for the smaller cases and $N^{1.7}$ for the larger cases. Clusters of Al are densely packed systems and certainly cannot be considered as fortunate examples.

The scaling behavior is only one aspect of an algorithm. The exact Coulomb term, Eq. (7), and the RI-J approximation, Eq. (21), have the same asymptotic scaling - but the latter can be evaluated 10 to 100 times faster [13,15]. The N^3 scaling for linear algebra is not seen for these cases although matrix calculations are carried out. If computational effort is approximated as cN^x, the factor c can be as important or even more important as the exponent x over a wide range of N values.

Very impressive progress has been made in the development of 'linear scaling' methods [17,18,19,20] where one attempts to achieve $t_{CPU} \propto N$ asymptotically for large molecules. There are now algorithms available which scale better than N^2 (and often reach N) for all steps of energy or gradient calculations. An exception is exact exchange E_x, Eq.(13), for delocalized systems, which still appears to require N^2 effort [22]. Although it is not clear at present for which system size these methods offer an advantage as compared to the best procedures already available, they will play an important role in the future.

3 Computer Hardware: Another Zoo

3.1 PC's and Workstations

The computers that can be afforded by research groups, institutes, faculties, and actually most computation centers are based on microprocessors. These are truly impressive devices: with about 10 million transistors packed on 1 cm^2 they constitute almost complete computers (with CPU and memory, a primary cache with a few 100 kB) and deliver a performance of roughly 1 GIPS (10^9 instructions per second), dwarfing the best mainframes of three decades ago. The performance of microprocessors has been increased by a factor of two every 18 to 24 months over the last decade - and this trend will more or less continue over the next decade.

Microprocessors are only partly suited to scientific computing. Because of the considerable investment costs, microprocessor development has been driven by consumer electronics (games), PC's for homes and offices, graphics requirements and transaction machines in business. So far these computers could always also be used

Table 1. Timings of TURBOMOLE for representative molecules on a HP workstation (180 MHz, PA 8000); BF denotes the number of basis functions, MO the number of occupied MO's, G the molecular symmetry group, and N_{atom} the number of atoms.

Molecule / basis	G	Method	BF/MO	N_{atom}		
C_5H_5N / TZP	C_{2v}	SCF	150/21	11	energy	6.3 min
					gradient	1.4 min
$Ni(CO)_4$ / SVP	T_d	DFT	136/42	9	energy	1.7 min
					gradient	13 sec
$C_{19}H_{40}$ / SV	C_{2v}	MP2	251/77	59	energy +grad.	96 min
$(CoPH_3)_6$ As_{12}/ SVP	D_{3d}	RI-DFT	726/333	42	energy	33 min
					gradient	4.3 min
Fullerene C_{60} / 3-21G	I_h	DFT	540/180	60	energy	7.8 min
					gradient	58 sec
Acetyl Salecylic Acid / SV(P)	C_1	SCF	198/47	21	energy	25.0 min
					gradient	4.5 min
$Fe(C_5H_5)_2$ / SV	D_{5d}	MP2	131/48	21	energy	59 sec
/ SVP		SCF	214/48		energy	4.4 min
					gradient	39 sec
/ SVP		RI-DFT			energy	1.9 min
					gradient	21 sec
$Cu_{12}[P(C_2H_5)_3]_8S_6$/ SVP+ECP	C_{4h}	RI-DFT	1260/426	194	energy	210 min
					gradient	32.4 min
C_{70} / TZVP	D_{5h}	RI-MP2	1330/210	70	energy +grad.	49.3 h

for number crunching purposes in science and engineering.

The big success of decentralized computing – as opposed to using mainframes – came with UNIX workstations. These were open systems (hardware and software): all interfaces were standardized and their specifications published. One could combine parts from different manufacturers, opening the way to heavy competition and price reductions. PC's have recently become competitive with workstations in performance, even for typical number crunching applications. This has been greatly facilitated by the public domain system LINUX, a UNIX system for PC's and workstations. Windows and WindowsNT are so far hardly used for scientific computing. It is to be expected, however, that they will replace UNIX, first for the "low end" computers and later also for larger systems. This process is well under way and may proceed faster than expected.

A desktop system for scientific computing is typically equipped with 256 MB memory (at least 128 MB, up to 1 GB is usually possible) and disks with 4 GB or 9 GB capacity. Prices for these systems start below 10 000 DM. We have carried out computations with various small computers with the program system TURBOMOLE [23]. Some representative timings are collected in the Table 1 for a typical high end workstation. The same set of test cases (with very few exceptions since access was limited) has been run on other hardware which permits to compare their relative speed. A newer version is available via [23].

Relative Performance of some Workstations:
(larger numbers mean faster machines)

IBM RISC 3CT (67 MHz)	0.49
Pentium Pro (200 MHz)	0.52
IBM RISC SP2 node (120 MHz)	0.85
SGI (R 10 000, 190 MHz)	0.98
Pentium II (375 MHz)	0.96
DEC (personal workstation, 433 MHz)	0.96
HP PA8000 (180 MHz)	1.0
DEC (workstation, 600 MHz)	1.25

The reader should not forget that the above comparison is for TURBOMOLE, and here only on average with a standard deviation around 15%. Running other programs may lead to quite different relative efficiencies of the machines considered. For Pentium-based PC's we employed the "Portland Group" compiler, for all others those provided by the respective companies. The timings for the '375 MHz Pentium II' have been obtained with a 333 MHz processor driven at higher clock frequency.

3.2 Multiprocessor Machines

There is a clear trend towards machines with 2 to 8 CPUs in a single frame since this reduces costs for production and maintenance. Even larger machines contain 512 (or even more) processors and are designed for parallel processing of big problems, e.g. the "grand challenges". Parallel computers are at present the only way to reduce turn-around times for very demanding computations. However, scientists do not always get the hardware they would like. One has to work with the machines the engineers can construct or are told to construct by their companies: i.e. basically standard workstations with a fast connection network. Although these are very powerful and potentially useful tools, they are only reluctantly accepted since the burden of parallelizing codes to exploit this machinery lies entirely with the scientific programmer. It is also embarrassing that different computer architectures may require different parallelization strategies. In Quantum Chemistry the situation is further complicated since the methods and algorithms employed have not settled down and are still in a state of flux.

Despite all these problems a variety of codes is available in parallelized versions. In the next section it will be shown that impressive improvements in turn-around time have been achieved.

4 Calculating some Large Systems

4.1 Methods

We will try to give the reader a broad idea of applications carried out in Karlsruhe. This is appropriate for a feature article; a comprehensive survey of computational Quantum Chemistry would in any case be impossible in a few pages.

Let us first put the methods in order according to the effort typically required:

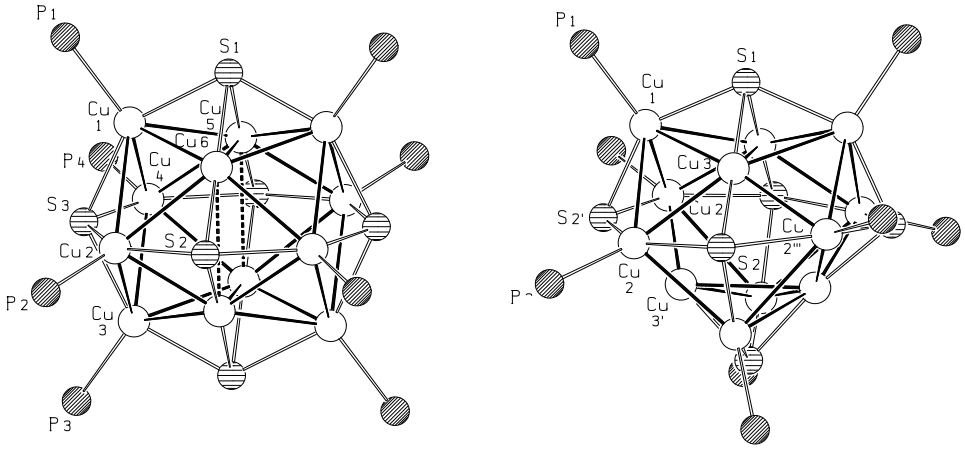
$$\text{Effort of MP2} > \text{RI-MP2} > \text{DFT}(J_{exact}) \geq \text{SCF} > \text{RI-DFT}.$$

This ordering applies to TURBOMOLE and most other programs where corresponding methods are implemented. DFT(J_{exact}) denotes usage of the exact Coulomb energy based on two-electron integrals, Eqs. (7), (8), and (12). Such calculations are more expensive than SCF since a quadrature is necessary in addition. RI-DFT is considerably more efficient than DFT(J_{exact}) or SCF – typically by a factor of 3 to 5 [13,15] – and is the method of choice for large systems. RI-MP2 is about 5 to 7 times faster than MP2 [14].

Molecular symmetry reduces computer times roughly according to the order of the molecular symmetry group, for TURBOMOLE at least. This makes it possible to calculate large molecules on small and inexpensive machines.

4.2 Ligand-stabilized Sulfur-bridged Copper Clusters [24]

Numerous sulfur or selenium bridged copper clusters have been synthesized and structurally characterized in the group of Dieter Fenske; accompanying theoretical treatments have been carried out especially to establish the energetics. S. Dehnen synthesized two structural isomers, type 1 and 2, Fig. 1, for $[Cu_{12}S_6(PR_3)_8]$. Whereas the type 1 structure is found for R=Et, and appears to be quite common, type 2 is known for R=^{n}Pr only. The calculations were started with the idea that a pronounced ligand effect would be responsible for this state of affairs: type 2 structures can better accomodate the bulkier ligands P^{n}Pr$_3$ (as compared to P Et$_3$). The following results were obtained.



1 2

Figure 1. Molecular structure types 1 and 2 for $[Cu_{12}S_6(PR_3)_8]$, the organic groups R attached to P are not shown.

(i) MP2 and DFT lead to virtually identical results for the model compound with R=H (larger R could not be treated by MP2). Type 1 is slightly more stable than type 2: by 2.4 kJ/mol (MP2) and 1.2 kJ/mol (DFT).

(ii) The type 1 and type 2 structure can easily accomodate either ligand, R=Et or R=^{n}Pr, Fig. 2. Type 1 is 22 kJ/mol (R=Et) and 24 kJ/mol (R=^{n}Pr) more stable than type 2.

Since the authors of the theoretical study see no compelling reasons to doubt the reliability of their findings, it was concluded that the structural types found in experiment are probably not determined by the energetics of isolated clusters alone and rather that kinetics plays the decisive role. The cluster with R=^{n}Pr precipitates already at -80°C and dissolves at higher temperature; for R=Et crystals form at -24°C. (The thermodynamically stable Cu_2S is obtained at room temperature in either case). To reconcile all aspects mentioned, it was proposed that structure 2, R=^{n}Pr, is kinetically stabilized at the low temperature where it exists.

This example involved very demanding calculations with structure determinations (in C_{4h} symmetry) for clusters with up to 266 atoms; timings are given in Table 1.

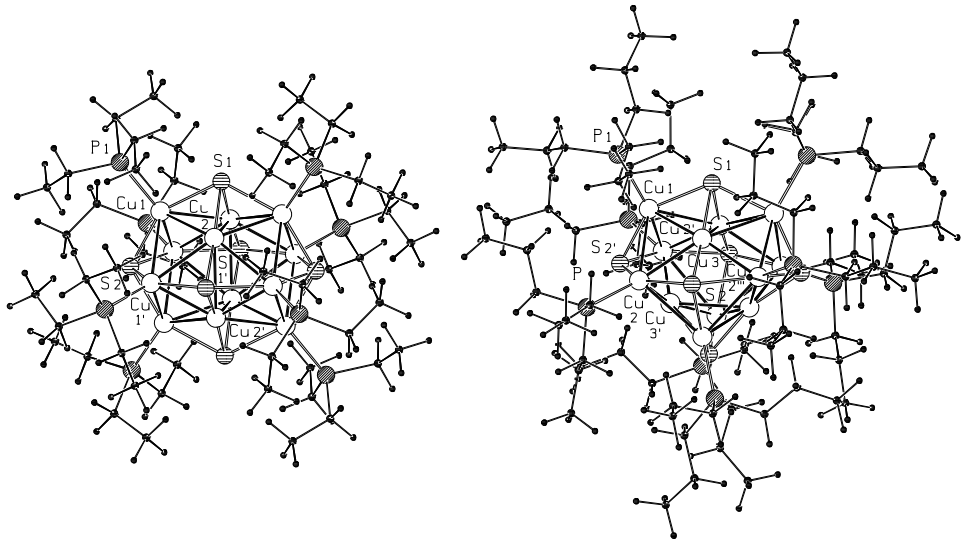
4.3 Cadmium Selenide Nanoparticles [25]

Clusters of semiconductor compounds such as CdSe often exhibit strongly size dependent electrical and optical properties. These materials could form the basis for new devices with considerable technological importance, e.g. light emitting diodes or electronic devices operating above GHz frequency. The behavior of the clusters is mainly governed by their electronic structure. Since the electrons are spatially confined to the cluster volume one expects and finds a pronounced "quantum size effect" which causes the size dependence of properties.

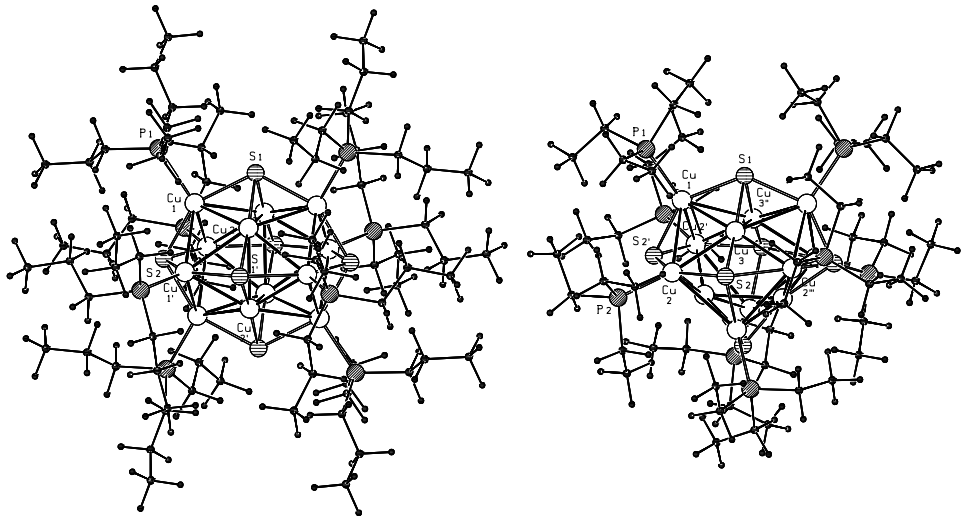
As a first step towards a detailed understanding of the quantum size effect the clusters shown in Fig. 3 have been investigated. The compounds have been mainly synthesized in the group of Fenske and are available as monodisperse materials in crystalline form (ionic clusters with the necessary counterions, of course). The stabilization of the clusters is achieved by protecting phenyl groups bonded to outer layer Se atoms.

Dipole allowed electronic excitation spectra are an important aspect of the electronic spectra. These have been computed within the TDDFT method employing the RI approximation [26]. The results are presented in Fig. 4. Since the two largest cluster could not be computed with the phenyl groups, these groups were replaced by H throughout for better comparison. Structure optimization - which show good agreement with experiment - has been carried out for the 294-atom cluster $[Cd_{10}Se_4(SePh)_{12}(PPh_3)_4]$; requiring 7 hours for an RI-DFT structure optimization cycle (energy + gradient) on a HP-PA 8000 (180 MHz).

The largest clusters treated have a diameter exceeding 1 nm: they are large enough to be "nanoclusters". However they are still too small to allow electronic excitations to be treated by band structure theory (with effective masses for holes and electrons). Quantum chemical calculations, although expensive, thus provide

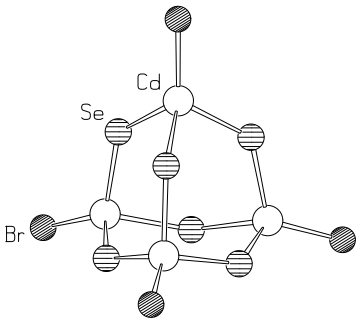


a) $[Cu_{12}S_6(PEt_3)_8]/C_{4h}$ $[Cu_{12}S_6(P^nPr_3)_8]/S_4$

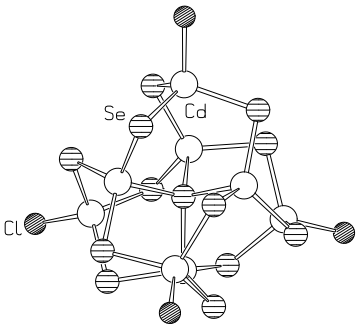


b) $[Cu_{12}S_6(P^nPr_3)_8]/C_{4h}$ $[Cu_{12}S_6(PEt_3)_8]/S_4$

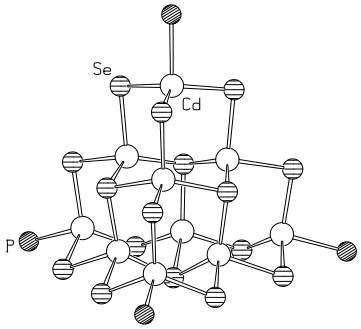
Figure 2. Computed RI-DFT structures of $[Cu_{12}S_6(PR_3)_8]$ for R=Et and ^{n}Pr. Type 1 clusters are on the left side (symmetry C_{4h}), type 2 on the right side (symmetry S_4); a) known compounds, b) 'hypothetical' clusters.



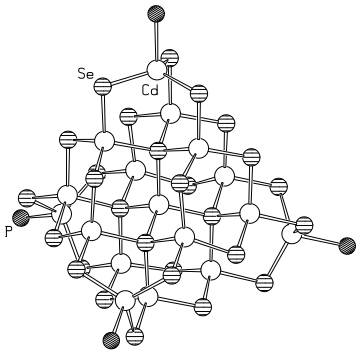
$[Cd_4(SePh)_6Br_4]^{2-}$



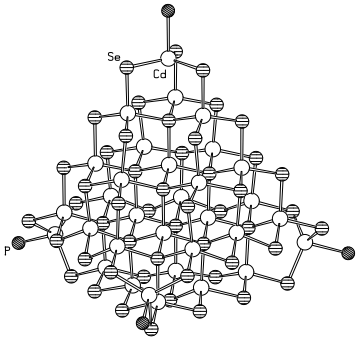
$[Cd_8Se(SePh)_{12}Cl_4]^{2-}$



$[Cd_{10}Se_4(SePh)_{12}(PPh_3)_4]$



$[Cd_{17}Se_4(SeH)_{24}(PH_3)_4]^{2+}$



$[Cd_{32}Se_{14}(SeH)_{36}(PH_3)_4]$

Figure 3. RI-DFT cluster structures of $[Cd_4(SePh)_6Br_4]^{2-}$, $[Cd_8Se(SePh)_{12}Cl_4]^{2-}$, $[Cd_{10}Se_4(SePh)_{12}(PPh_3)_4]$, $[Cd_{17}Se_4(SeH)_{24}(PH_3)_4]^{2+}$, $[Cd_{32}Se_{14}(SeH)_{36}(PH_3)_4]$; the ligands H and Ph attached to Se and P atoms are not shown.

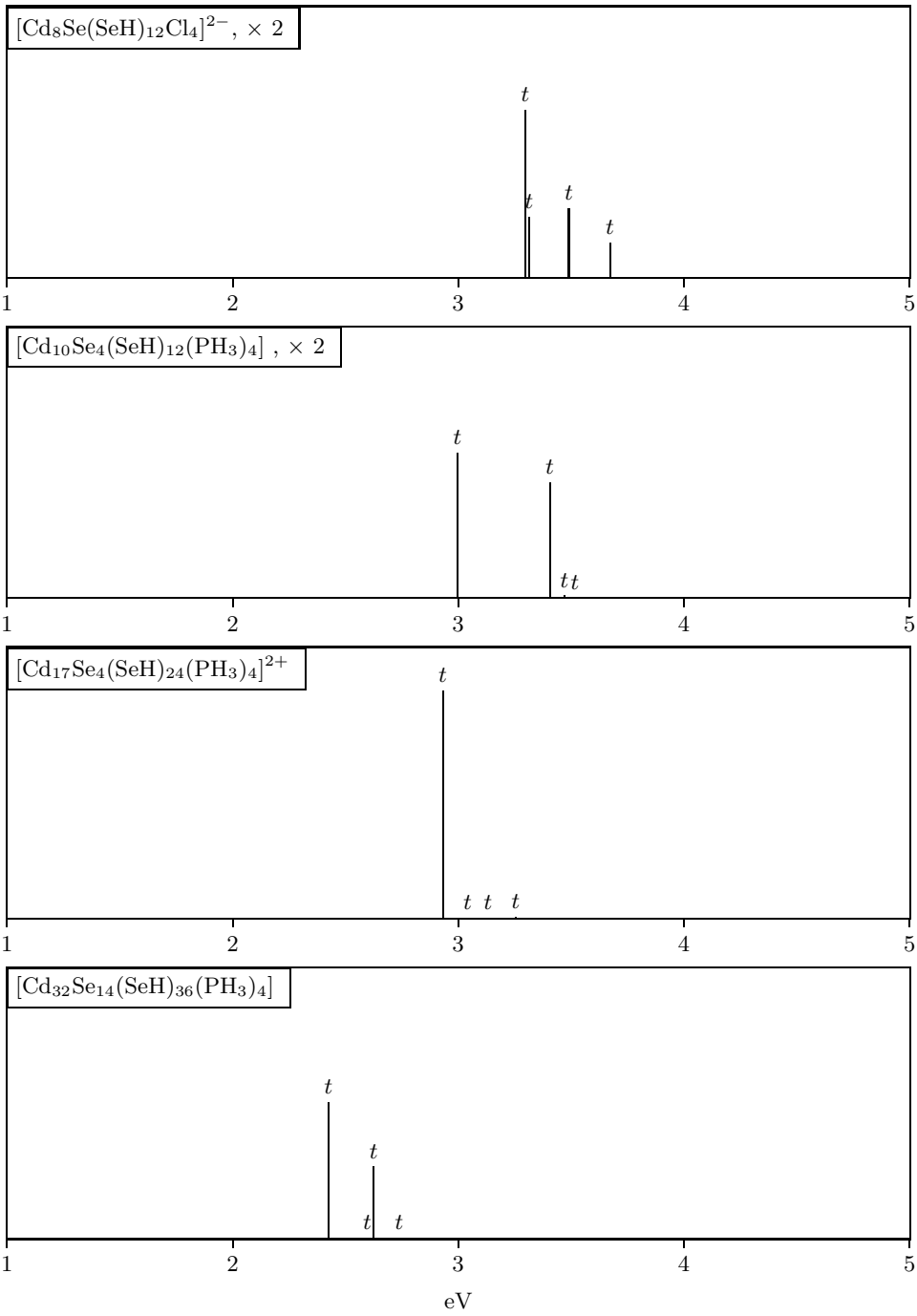


Figure 4. TDDFT electronic excitation spectra of $[Cd_8Se(SeH)_{12}Cl_4]^{2-}$, $[Cd_{10}Se_4(SeH)_{12}(PH_3)_4]$, $[Cd_{17}Se_4(SeH)_{24}(PH_3)_4]^{2+}$, $[Cd_{32}Se_{14}(SeH)_{36}(PH_3)_4]$; the first dipole allowed transitions are shown.

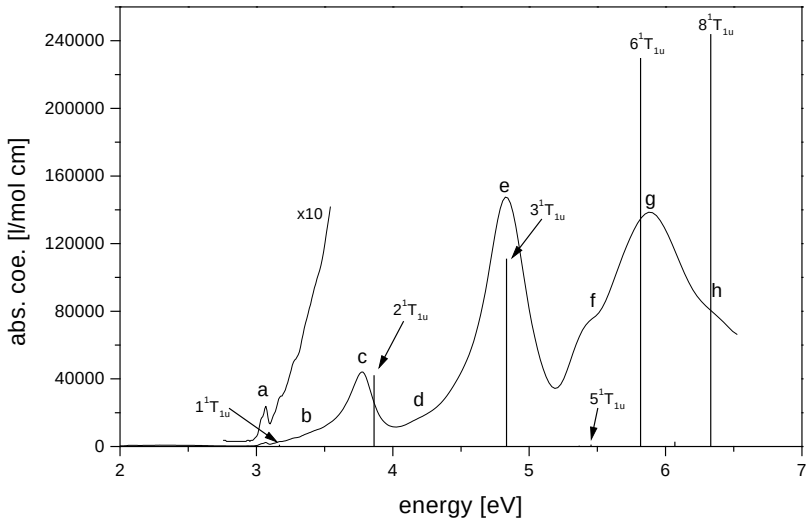


Figure 5. Absorption spectrum of C_{60} (I_h) in n-hexane at room temperature in comparison with the computed TDDFT excitation energies.

the only way so far to calculate such spectra for nanoclusters of this size; measurements could not be carried out so far. The calculated spectra are shown in Fig. 4. We observe that such large-scale calculations start to show convergence to the known bulk properties.

4.4 Fullerenes [27]

TDDFT [11,26] has been applied to compute electronic excitations of fullerenes C_{60} (I_h), C_{70} (D_{5h}), C_{76} (D_2), C_{78} (C_{2v}, D_3 and D_{3h}) and C_{80} (D_2). The fullerenes had been isolated and their spectra measured in solution (toluene, hexane). The calculations were carried out to assign the spectra as far as possible. Despite various uncertainties (solution versus gas phase, neglect of vibration), the level of agreement is good enough to allow assignment of the dominant spectral features. As a simple example we present a comparison of theory and experiment in Fig. 5. The computed excitation energies have all been increased by 0.35eV to compensate for the underestimation typical of TDDFT.

4.5 Parallelized Programs

It was only possible to calculate the large systems mentioned above on workstations and PC's by exploiting the molecular symmetry. Reducing the symmetry increases the computational demand; for example, in TURBOMOLE, going from D_{4h}, D_{4d}, T_d or O symmetry down to C_1 raises the time taken by a factor of 20. It is thus

impractical to compute systems of 300 atoms or so without symmetry on such workstations. Instead one needs parallel computers and parallel codes. The high symmetry cases treated on a single workstation could then be computed without symmetry restrictions but in comparable turn-around time with 8 to 32 parallel processors.

Parallel TURBOMOLE [28] achieves a very good speed-up in the dominant computational steps of SCF, DFT and RI-DFT calculations (energy and first order gradients). For the evaluation and processing of two-electron integrals as well as for the quadrature the speed-up is > 100 for 128 processors, ≈ 59 for 64 processors, and ≈ 30 for 32 processors. The RI-J algorithms even show a superlinear speed-up. The performance for a complete run is limited by linear algebra, however, which has not been efficiently parallelized.

Reliable estimates of the parallelization efficiency require a single processor run for comparison. The largest case for which this could be carried out was $[SiAl_{14}Cp^*_6]$, $Cp^* = (CH_3)_5$, with 165 atoms and 1365 CGTO basis functions. The following timings and speed-ups have been obtained in C_1 symmetry on an IBM SP2 (120 MHz) for a complete RI-DFT run.

Processors	Time/min	Speed-up
1	1131	1
8	135	9
16	71	16
32	43	26

At around 32 parallel processors one reaches the point of diminishing returns for the present program version. Even the large cases $[Cd_{10}Se_4(SePh)_{12}\ (PPh_3)_4]$ and $[Cd_{32}Se_{14}(SeH)_{36}\ (PH_3)_4]$ with up to 294 atoms and 2754 CGTO basis functions can be treated at the RI-DFT level in less than 4 hours on 32 processors.

5 Outlook

The ongoing dynamic development of methods and the continuing improvement of hardware have consequences which are easy to discern: Quantum Chemistry is rapidly becoming a standard tool of chemistry. In this respect it is following other techniques like NMR spectroscopy or X-ray scattering. There is an important difference which should be kept in mind. NMR and X-ray are the indispensable methods for qualitative and quantitative structure analysis. Quantum Chemitry has the capacity to be a universal tool to simulate all properties: spectra (ir, Raman, electronic, NMR), structures, energetics, intermolecular interactions, reactions, and so on.

Of course, this universal tool is still far off. Despite the progress theoreticians have made, the applicability of Quantum Chemistry is still restricted to specific cases. We just mention one problem: it would be highly desirable to develop efficient and reliable procedures to generate potential energy surfaces globally for larger molecules. This would allow the routine simulation of reactions and of the

temperature dependence of properties.

The progress in ab initio Quantum Chemistry also affects the computationally less demanding semi empirical and force field procedures. A better parametrization of these methods and an assessment of their accuracy becomes possible with the aid of detailed and reliable ab initio results often not available from experiment. One can further combine various methods in the treatment of a large molecule by means of embedding procedures [29], which may extend the applicability of ab initio methods to molecules in the range of 1000 to 10 000 atoms.

Solutions to many of the problems still limiting the applicability of present methods are in fact in reach or have been worked out in principle. With qualified manpower and sufficient support these will become routine in the near future.

Acknowledgements

The work sketched in this article would not have been possible without the decisive contributions of the late Marco Häser. We remain greatly indebted to him. We also acknowledge the generous support of the University of Karlsruhe, the Fonds der Chemische Industrie and the DFG through SFB 195.

References

1. This is not a review and is not intended to provide a survey of the literature. Given the pace of development there are hardly any up to date textbooks. A useful account of current activity is given by W. Klopper, W. Thiel, H.-J. Werner, and G. Jansen in: Theoretische Chemie 1997, Nachr. Chem. Tech. Lab. **46**, 196 (1998).
2. A. Szabo and N.S. Ostlund, Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory, Macmillan Publishing Co., Inc., New York, 1982.
3. R.G. Parr and W. Yang, Density-Functional Theory of Atoms and Molecules, Oxford University Press, 1989.
4. R.M. Dreizler and E.K.U. Gross, Density Functional Theory, Springer, Berlin, 1990.
5. A.D. Becke, J. Chem. Phys. **107**, 8554 (1997).
6. D.J. Tozer and N.C. Handy, J. Chem. Phys. **108**, 2545 (1998).
7. J. Gauss, J. Chem. Phys. **99**, 3629 (1993).
8. P. Pulay, in: Ab Initio Methods in Quantum Chemistry II, Ed. K.P. Lawley, John Wiley & Sons, New York, 1987.
9. M. Schindler and W. Kutzelnigg, J. Chem. Phys. **76**, 1919 (1982).
10. M.E. Casida, in: Recent advances in density functional methods, Vol. **1**, Ed. D.P. Chong, World Scientific, Singapore, 1995.
11. R. Bauernschmitt, M. Häser, O. Treutler, and R. Ahlrichs, Chem. Phys. Lett. **264**, 573 (1997).
12. O. Vahtras, J. Almlöf, and M.W. Feyereisen, Chem. Phys. Lett. **213**, 514 (1993).
13. K. Eichkorn, O. Treutler, H. Öhm, M. Häser, and R. Ahlrichs, Chem. Phys.

Lett **242**, 652 (1995).
14. F. Weigend and M. Häser, Theor. Chem. Acc. **97**, 331 (1997).
15. K. Eichkorn, F. Weigend, O. Treutler, and R. Ahlrichs, Theor. Chem. Acc. **97**, 119 (1997).
16. F. Weigend, M. Häser, H. Patzelt, and R. Ahlrichs, Chem. Phys. Lett. **294**, 143 (1998). The basis sets are available via `WWW,` Ref. 23.
17. C.A. White, B.G. Johnson, P.M.W. Gill, and M. Head-Gordon, Chem. Phys. Lett. **253**, 268 (1996).
18. M.C. Strain, G.E. Scuseria, and M.J. Frisch, Science **271**, 51 (1996).
19. R.E. Stratmann, G.E. Scuseria, and M.J. Frisch, Chem. Phys. Lett. **257**, 213 (1996).
20. B.G. Johnson, C.A. White, Q. Zhang, B. Chen, R.L. Graham, P.M.W. Gill, and M. Head-Gordon, Advances in Methodologies for Linear Scaling Density Functional Calculations, in: Recent Developments in Density Functional Theory, J.M. Seminario, Ed., Elsevier Science, Amsterdam, 1996, Vol. **4**, p. 441.
21. O. Treutler and R. Ahlrichs, J. Chem. Phys., **102**, 346 (1995).
22. E. Schwegler, M. Challacombe, and M. Head-Gordon, J. Chem. Phys. **106**, 9708 (1997).
23. To obtain more informations about TURBOMOLE please refer to `http://www.chemie.uni-karlsruhe.de/PC/TheoChem/` and `ftp://ftp.chemie.uni-karlsruhe.de/pub/`
24. K. Eichkorn, S. Dehnen, and R. Ahlrichs, Chem. Phys. Lett **284**, 287 (1998), and references therein.
25. K. Eichkorn and R. Ahlrichs, Chem. Phys. Lett. **288**, 235 (1998), and references therein.
26. R. Bauernschmitt and R. Ahlrichs, **256**, 454 (1996).
27. R. Bauernschmitt, R. Ahlrichs, F. H. Hennrich, and M. Kappes, J. Am. Chem. Soc. **120**, 5052 (1998).
28. M. v. Arnim and R. Ahlrichs, J. Comp. Chem. **19**, 1746 (1998); an older program version is described in TURBOMOLE, Parallel Implementation of SCF, Density Functional and Chemical Shift Modules, in: Methods and Techniques in Computational Chemistry METECC-95, STEF, Cagliari 1995, E. Clementi and G. Corongiu, Eds.
29. F. Maseras and K. Morokuma, J. Comp. Chem., **16**, 1170 (1995); D. Bakowies and W. Thiel, J. Phys. Chem., **100**, 10580 (1996).

PARALLEL PROGRAMMING MODELS, TOOLS AND PERFORMANCE ANALYSIS

MICHAEL GERNDT
John von Neumann Institute for Computing
Central Institute for Applied Mathematics
Research Centre Jülich, 52425 Jülich
Germany
E-mail: m.gerndt@fz-juelich.de

The major parallel programming models for scalable parallel architectures are the message passing model and the shared memory model. This article outlines the main concepts of those models as well as the industry standard programming interfaces MPI and OpenMP. To exploit the potential performance of parallel computers, programs need to be carefully designed and tuned. We will discuss design decisions for good performance as well as programming tools that help the programmer in program tuning.

1 Introduction

Although the performance of sequential computers increases incredibly fast, it is insufficient for a large number of challenging applications. Applications requiring much more performance are numerical simulations in industry and research as well as commercial applications such as query processing, data mining, and multi-media applications. Architectures offering high performance do not only exploit parallelism on a very fine grain within a single processor but apply a medium to large number of processors concurrently to a single application. High end parallel computers deliver up to 3 Teraflop/s (10^{12} floating point operations per second) and are developed and exploited within the ASCI (Accelerated Strategic Computing Initiative) program of the Department of Energy in the USA.

This article concentrates on programming numerical applications on distributed memory computers introduced in Section 1.1. Parallelization of those applications centers around selecting a decomposition of the data domain onto the processors such that the workload is well balanced and the communication is reduced (Section 1.2)[7].

The parallel implementation is then based on either the message passing or the shared memory model (Section 2). The standard programming interface for the message passing model is MPI (Message Passing Interface) [13,10], offering a complete set of communication routines (Section 2.1). OpenMP [5,12] is the standard for directive-based shared memory programming and will be introduced in Section 2.2.

Since parallel programs exploit multiple threads of control, debugging is even more complicated then for sequential programs. Section 3 outlines the main concepts of parallel debuggers and presents TotalView, the most widely available debugger for parallel programs.

Although the domain decomposition is key to good performance on parallel architectures, program efficiency depends also heavily on the implementation of the

communication and synchronization required by the parallel algorithm and the implementation techniques chosen for sequential kernels. Optimizing those aspects is very system dependent and thus, an interactive tuning process consisting of measuring performance data and applying optimizations follows the initial coding of the application. The tuning process is supported by programming model specific performance analysis tools. Section 4 presents basic performance analysis techniques and introduces the three performance analysis tools for MPI programs available on CRAY T3E.

1.1 Parallel Architectures

Parallel computers that scale beyond a small number of processors circumvent the main memory bottleneck by distributing the memory among the processors. Current architectures [4] are composed of single-processor nodes with local memory or of multiprocessor nodes where the node's main memory is shared among the node's processors. In the following it is assumed that nodes do have only a single CPU and the terms node and processor will be used interchangeably.

The most important characteristic of this *distributed memory architecture* is that access to the local memory is faster than to remote memory. It is the challenge for the programmer to assign data to the processors such that most of the data accessed during the computation are already in the node's local memory.

Three major classes of distributed memory computers can be distinguished:

No Remote Memory Access (NORMA) computers do not have any hardware support to access another node's local memory. Processors obtain data from remote memory only by exchanging messages between processes on the requesting and the supplying node.

Remote Memory Access (RMA) computers allow to access remote memory via specialized operations implemented by hardware. The accessed memory location is not determined via an address in a shared linear address space but via a tuple consisting of the processor number and the local address in the target processor's address space.

Cache-Coherent Non Uniform Memory Access (ccNUMA) computers do have a shared physical address space. All memory locations can be accessed via usual load and store operations. Access to a remote location results in a copy of the appropriate cache line in the processor's cache. Coherence algorithms ensure that multiple copies of a cache line are kept coherent, i.e. the copies do have the same value.

While most of the early parallel computers were NORMA systems, today's systems are either RMA or ccNUMA computers. This is because remote memory access is a light-weight communication protocol that is more efficient than standard message passing since data copying and process synchronization are eliminated. In addition, ccNUMA systems offer the abstraction of a shared linear address space resembling physical shared memory systems. This abstraction simplifies the task of program development but does not necessarily facilitate program tuning.

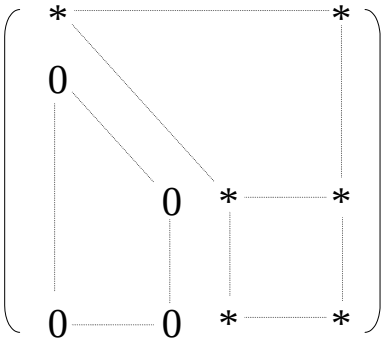


Figure 1. Structure of the matrix during Gaussian elimination.

Typical examples of the three classes are clusters of workstations (NORMA), CRAY T3E (RMA), and SGI Origin 2000 (ccNUMA).

1.2 Data Parallel Programming

Applications that scale to a large number of processors usually perform computations on large data domains. For example, crash simulations are based on partial differential equations that are solved on a large finite element grid and molecular dynamic applications simulate the behavior of a large number of atoms. Other parallel applications apply linear algebra operations to large vectors and matrices. The elemental operations on each object in the data domain can be executed in parallel by the available processors.

The scheduling of operations to processors is determined according to a selected *domain decomposition* [8]. Processors execute those operations that determine new values for local elements (owner-computes rule). While processors execute an operation, they might need values from other processors. The domain decomposition has thus to be chosen so that the distribution of operations is balanced and the communication is minimized. The third goal is to optimize single node computation, i.e. to be able to exploit the processor's pipelines and the processor's caches efficiently.

A good example for the design decisions taken when selecting a domain decomposition is Gaussian elimination [2]. The main structure of the matrix during the iterations of the algorithm is outlined in Figure 1.

The goal of this algorithm is to eliminate all entries in the matrix below the main diagonal. It starts at the top diagonal element and subtracts multiples of the first row from the second and subsequent rows to end up with zeros in the first column. This operation is repeated for all the rows. In later stages of the algorithm the actual computations have to be done on rectangular sections of decreasing size.

If the main diagonal element of the current row is zero, a pivot operation has to be performed. The subsequent row with the maximum value in this column is selected and exchanged with the current row.

A possible distribution of the matrix is to decompose its columns into blocks, one

block for each processor. The elimination of the entries in the lower triangle can then be performed in parallel where each processor computes new values for its columns only. The main disadvantage of this distribution is that in later computations of the algorithms only a subgroup of the processes is actually doing any useful work since the computed rectangle is getting smaller.

To improve load balancing, a cyclic column distribution can be applied. The computations in each step of the algorithm executed by the processors differ only in one column.

In addition to load balancing also communication needs to be minimized. Communication occurs in this algorithm for broadcasting the current column to all the processors since it is needed to compute the multiplication factor for the row. If the domain decomposition is a row distribution, which eliminates the need to communicate the current column, the current row needs to be broadcast to the other processors.

If we consider also the pivot operation, communication is necessary to select the best row when a rowwise distribution is applied since the computation of the global maximum in that column requires a comparison of all values.

Selecting the best domain decomposition is further complicated due to optimizing single node performance. In this example, it is advantageous to apply BLAS3 operations for the local computations. Those operations make use of blocks of rows to improve cache utilization. Blocks of rows can only be obtained if a block-cyclic distribution is applied, i.e. columns are not distributed individually but blocks of columns are cyclically distributed.

This discussion makes clear, that choosing a domain decomposition is a very complicated step in program development. It requires deep knowledge of the algorithm's data access patterns as well as the ability to predict the resulting communication.

2 Programming Models

The two main programming models, *message passing* and *shared memory*, offer different features for implementing applications parallelized by domain decomposition.

The message passing model is based on a set of processes with private data structures. Processes communicate by exchanging messages with special send and receive operations. The domain decomposition is implemented by developing a code describing the local computations and local data structures of a single process. Thus, global arrays have to be split up and only the local part be allocated in a process. This handling of global data structures is called *data distribution*. Computations on the global arrays also have to be transformed, e.g. by adapting the loop bounds, to ensure that only local array elements are computed. Access to remote elements have to be implemented via explicit communication, temporary variables have to be allocated, messages be setup and transmitted to the target process.

The shared memory model is based on a set of threads that are created when parallel operations are executed. This type of computation is also called *fork-join*

parallelism. Threads share a global address space and thus access array elements via a global index.

The main parallel operations are *parallel loops* and *parallel sections*. Parallel loops are executed by a set of threads also called *team*. The iterations are distributed onto the threads according to a predefined strategy. This scheduling strategy implements the chosen domain decomposition. Parallel sections are also executed by a team of threads but the tasks assigned to the threads implement different operations. This feature can for example be applied if domain decomposition itself does not generate enough parallelism and whole operations can be executed in parallel since they access different data structures.

In the shared memory model, the distribution of data structures onto the node memories is not enforced by decomposing global arrays into local arrays, but the global address space is distributed onto the memories on system level. For example, the pages of the virtual address space can be distributed cyclically or can be assigned on a first touch basis. The chosen domain decomposition thus has to take into account the granularity of the distribution, i.e. the size of pages, as well as the system-dependent allocation strategy.

While the domain decomposition has to be hardcoded into the message passing program, it can easily be changed in a shared memory program by selecting a different scheduling strategy for parallel loops.

Another advantage of the shared memory model is that automatic and incremental parallelization is supported. While automatic parallelization leads to a first working parallel program, its efficiency typically needs to be improved. The reason for this is that parallelization techniques work on a loop-by-loop basis and do not globally optimize the parallel code via a domain decomposition. In addition, dependence analysis, the prerequisite for automatic parallelization, is limited to statically known access patterns.

In the shared memory model, a first parallel version is relatively easy to implement and can be incrementally tuned. In the message passing model instead, the program can be tested only after finishing the full implementation. Subsequent tuning by adapting the domain decomposition is usually time consuming.

2.1 MPI

The Message Passing Interface (MPI) [13,10] was developed between 1993 and 1997. It includes routines for point-to-point communication, collective communication, one-sided communication, and parallel IO. While the basic communication primitives are already defined since May 1994 and implemented on allmost all parallel computers, remote memory access and parallel IO routines are part of MPI 2.0 and are only available on few machines.

2.1.1 MPI basic routines

MPI consists of more than 120 functions. But realistic programs can already be developed based on no more than six functions:

MPI_Init initializes the library. It has to be called at the beginning of a parallel

operation before any other MPI routines are executed.

MPI_Finalize frees any resources used by the library and has to be called at the end of the program.

MPI_Comm_size determines the number of processors executing the parallel program.

MPI_Comm_rank returns the unique process identifier.

MPI_Send transfers a message to a target process. This operation is a blocking send operation, i.e. it terminates when the message buffer can be reused either because the message was copied to a system buffer by the library or because the message was delivered to the target process.

MPI_Recv receives a message. This routines terminates if a message was copied into the receive buffer.

2.1.2 MPI communicator

All communication routines depend on the concept of a *communicator*. A communicator consists of a process group and a communication context. The processes in the process group are numbered from zero to process count - 1. The process number returned by MPI_Comm_rank is the identification in the process group of the communicator which is passed as a parameter to this routine.

The communication context of the communicator is important in identifying messages. Each message has an integer number called a *tag* which has to match a given selector in the corresponding receive operation. The selector depends on the communicator and thus on the communication context. It selects only messages with a fitting tag and having been sent relative to the same communicator. This feature is very useful in building parallel libraries since messages sent inside the library will not interfere with messages outside if a special communicator is used in the library. The default communicator that includes all processes of the application is MPI_COMM_WORLD.

2.1.3 MPI collective operation

Another important class of operations are *collective operations*. Collective operations are executed by a process group identified via a communicator. All the processes in the group have to perform the same operation. Typical examples for such operations are:

MPI_Reduce performs a global operation on the data of each process in the process group. For example, the sum of all values of a distributed array can be computed by first summing up all local values in each process and then summing up the local sums to get a global sum. The latter step can be performed by the reduction operation with the parameter MPI_SUM. The result is delivered to a single target processor.

MPI_Comm_split is an administration routine for communicators. It allows to create multiple new communicators based on a given coloring scheme. All processes of the original communicator have to take part in that operation.

MPI_Barrier synchronizes all processes. None of the processes can proceed beyond the barrier until all the processes started execution of that routine.

2.1.4 MPI IO

Data parallel applications make use of the IO subsystem to read and write big data sets. These data sets result from replicated or distributed arrays. The reasons for IO are to read input data, to pass information to other programs, e.g. for visualization, or to store the state of the computation to be able to restart the computation in case of a system failure or if the computation has to be split into multiple runs due to its resource requirements.

IO can be implemented in three ways:

1. Sequential IO

   A single node is reponsible to perform the IO. It gathers information from the other nodes and writes it to disc or reads information from disc and scatters it to the appropriate nodes. While the IO is sequential and thus need not be parallelized, the full performance of the IO subsystem might not be utilized. Modern systems provide high performance IO subsystems that are fast enough to support multiple IO requests from different nodes in parallel.

2. Private IO

   Each node accesses its own files. The big advantage of this implementation is that no synchronization among the nodes is required and very high performance can be obtained. The major disadvantage is that the user has to handle a large number of files. For input the original data set has to be splitted according to the distribution of the data structure and for output the process-specific files have to be merged into a global file for postprocessing.

3. Parallel IO

   In this implementation all the processes access the same file. They read and write only those parts of the file with relevant data. The main advantages are that no individual files need to be handled and that reasonable performance can be reached. The disadvantage is that it is difficult to reach the same performance as with private IO. The parallel IO interface of MPI provides flexible and high-level means to implement applications with parallel IO.

Files accessed via MPI IO routines have to be opened and closed by collective operations. The open routine allows to specify hints to optimize the performance such as whether the application might profit from combining small IO requests from different nodes, what size is recommended for the combined request, and how many nodes should be engaged in merging the requests.

The central concept in accessing the files is the *view*. A view is defined for each process and specifies a sequence of data elements to be ignored and data elements

to be read or written by the process. When reading or writing a distributed array the local information can be described easily as such a repeating pattern. The IO operations read and write a number of data elements on the basis of the defined view, i.e. they access the local information only. Since the views are defined via runtime routines prior to the access, the information can be exploited in the library to optimize IO.

MPI IO provides blocking as well as nonblocking operations. In contrast to blocking operations, the nonblocking ones only start IO and terminate immediately. If the program depends on the successful completion of the IO it has to check it via a test function. Besides the collective IO routines which allow to combine individual requests, also noncollective routines are available to access shared files.

2.1.5 MPI remote memory access

Remote memory access (RMA)operations allow to access the address space of other processes without participation of the other process. The implementation of this concept can either be in hardware, such as in the CRAY T3E, or in software via additional threads waiting for requests. The advantages of these operations are that the protocol overhead is much lower than for normal send and receive operations and that no polling or global communication is required for setting up communication such as in unstructured grid applications and multiparticle applications.

In contrast to explicit message passing where synchronization happens implicitely, accesses via RMA operations need to be protected by explicit synchronization operations.

RMA communication in MPI is based on the *window concept*. Each process has to execute a collective routine that defines a window, i.e. the part of its address space that can be accessed by other processes.

The actual access is performed via a *put* and *get operation*. The address is defined by the target process number and the displacement relative to the starting address of the window for that process.

MPI provides also special synchronization operations relative to a window. The MPI_Win_fence operation synchronizes all processes that make some address ranges accessible to other processes. It is a collective operation that ensures, that all RMA operations started before the fence operation terminate before the target process executes the fence operation and that all RMA operations of a process executed after the fence operation are executed after the target process executed the fence operation.

2.2 OpenMP

OpenMP [5,12] is a directive-based programming interface for the shared memory programming model. It is the result of an effort to standardize the different programming interfaces on the target systems. OpenMP is a set of directives and runtime routines for Fortran 77 (1997) and a corresponding set of pragmas for C and C++ (1998). An extension of OpenMP for Fortran 95 is under investigation.

Directives are comments that are interpreted by the compiler. Directives do have the advantage that the code is still a sequential code that can be executed on

sequential machines and thus no two versions, a sequential and a parallel version, need to be maintained.

Directives start and terminate parallel regions. When the master thread hits a parallel region a team of threads is created or activated. The threads execute the code in parallel and are synchronized at the beginning and the end of the computation. After the final synchronization the master thread continues execution after the parallel region. The main directives are:

PARALLEL DO specifies a loop that can be executed in parallel. The DO loop's iterations can be distributed in various ways including STATIC(CHUNK), DYNAMIC(CHUNK), and GUIDED(CHUNK) onto the set of threads (as defined in the OpenMP standard). STATIC(CHUNK) distribution means that the set of iterations are consecutively distributed onto the threads in blocks of CHUNK size (resulting in block and cyclic distributions). DYNAMIC(CHUNK) distribution implies that iterations are distributed in blocks of CHUNK size to threads on a first-come-first-served basis. GUIDED (CHUNK) means that blocks of exponentially decreasing size are assigned on a first-come-first-served basis. The size of the smallest block is determined by CHUNK size.

PARALLEL SECTION starts a set of sections that are executed in parallel by a team of threads.

PARALLEL REGION introduces a code region that is executed redundantly by the threads. It has to be used very carefully since assigments to global variables will lead to conflicts among the threads and possibly to nondeterministic behavior.

PDO is a work sharing construct and may be used within a parallel region. All the threads executing the parallel region have to cooperate in the execution of the parallel loop. There is no implicit synchronization at the beginning of the loop but a synchronization at the end. After the final synchronization all threads continue after the loop in the replicated execution of the program code.

The main advantage of this approach is that the overhead for starting up the threads is eliminated. The team of threads exists during the execution of the parallel region and need not be built before each parallel loop.

PSECTION is also a work sharing construct that enforces that the current team of threads executing the surrounding parallel region cooperates in the execution of the parallel section.

Program data can either be shared or private. While threads do have an own copy of private data, only one copy exists of shared data. This copy can be accessed by all threads. To ensure program correctness, OpenMP provides special synchronization constructs. The main constructs are *barrier synchronization* enforcing that all threads have reached this synchronization operation before execution continues and *critical sections*. Critical sections ensure that only a single thread can enter the section and thus, data accesses in such a section are protected from race conditions. A common situation for a critical section is the accumulation of values. Since an

accumulation consists of a read and a write operation unexpected results can occur if both operations are not surrounded by a critical section.

3 Parallel Debugging

Debugging parallel programs is more difficult than debugging sequential programs not only since multiple processes or threads need to be taken into account but also because program behaviour might not be deterministic and might not be reproducible. These problems are not attacked by current state-of-the-art commercial parallel debuggers. Only the first reason is eliviated by current debuggers. They provide menus, diplays, and commands that allow to inspect individual processes and execute commands on individual or all processes.

The widely used debugger is TotalView from Etnus Inc.[14] TotalView provides breakpoint definition, single stepping, and variable inspection via an interactive interface. The programmer can execute those operations for individual processes and groups of processes. TotalView also provides some means to summarize information such that equal information from multiple processes is combined into a single information and not repeated redundantly.

4 Performance Analysis

Performance analysis is an iterative subtask during process development. The goal is to identify program regions that do not perform well. Performance analysis is structured into four phases:

1. Measurement

   Performance analysis is executed based on information on runtime events gathered during program execution. The basic events are, for example, cache misses, termination of a floating point operation, start and stop of a subroutine or message passing operation. The information on individual events can be summarized during program execution or individual trace records can be collected for each event.

   Summary information has the advantage to be of moderate size while trace information tends to be very large. The disadvantage is that it is not that fine grained, the behavior of individual instances of subroutines can for example not be investigated since all the information has been summed up.

2. Analysis

   During analysis the collected runtime data are inspected to detect *performance problems*. Performance problems are based on *performance properties*, such as the existence of message passing in a program region, which do have a condition for identifying it and a severity function that specifies its importance for program performance.

   Current tools support the user in checking the conditions and the severity by visualizing program behavior. Future tools might be able to automatically detect performance properties based on a specification of possible properties.

During analysis the programmer applies a threshold. Only performance properties whose severity exceeds this threshold are considered to be performance problems.

3. Ranking

   During program analysis the severest performance problems need to be identified. This means that the problems need to be ranked according to the severity. The most severe problem is called the *program bottleneck*. This is the problem the programmer tries to resolve by applying appropriate program transformations.

4. Refinement

   The performance problems detected in the previous phases might not be precise enough to allow the user to start optimization. At the beginning of performance analysis, summary data can be used to identify critical regions. Those summary data might not be sufficient to identify why, for example, a region has high message passing overhead. The reason, e.g. very big messages or load imbalance, might be identified only with more detailed information. Therefore the performance problem should be refined into hypotheses about the real reason and additional information be collected in the next performance analysis cycle.

Current techniques for performance data collection are *profiling* and *tracing*. Profiling collects summary data only. This can be done via *sampling*, the program is regularly interrupted, e.g. every 10 ms, and the information is added up for the source code location which was executed in this moment. For example, the UNIX profiling tool *prof* applies this technique to determine the fraction of the execution time spent in individual subroutines.

A more precise profiling technique is based on *instrumentation*, i.e. special calls to a *monitoring library* are inserted into the program. This can either be done in the source code by the compiler or specialized tools, or can be done in the object code. While the first approach allows to instrument more types of regions, for example, loops and vector statements, the latter allows to measure data for programs where no source code is available. The monitoring library collects the information and adds it to special counters for the specific region.

Tracing is a technique that collects information for each event. This results, for example, in very detailed information for each instance of a subroutine and for each message sent to another process. The information is stored in specialized trace records for each event type. For example, for each start of a send operation, the time stamp, the message size and the target process can be recorded, while for the end of the operation, the timestamp and bandwidth are stored.

The trace records are stored in the memory of each process and are output either because the buffer is filled up or because the program terminated. The individual trace files of the processes are merged together into one trace file ordered according to the time stamps of the events.

The following sections describe performance analysis tools available on CRAY T3E.

4.1 The CRAY T3E Performance Analysis Environment

The programming environment of the CRAY T3E supports performance analysis via interactive tools. Cray itself provides two tools, Apprentice and PAT, which are both based on summary information. In addition, Apprentice accesses source code information to map the performance information to the statements of the original source code. Besides these two tools, programmers can use VAMPIR, a trace analysis tool developed at our institute. Within a collaboration with Cray, instrumentation and trace generation for VAMPIR is being integrated into the next version of PAT.

4.2 Compiler Information File

The F90 and C compilers on CRAY T3E generate for each source file on request a *compiler information file (CIF)*. This file includes information about the compilation process (applied compiler options, target machine characteristics, and compiler messages) and source information of the compilation units (procedure information, symbol information, information about loops and statement types, cross-reference information for each symbol in the source file).

Apprentice requires this information to link the performance information back to the source code. CIFs are initially in ASCII format but can be converted to binary format. The information can be easily accessed in both formats via a library interface.

4.3 Apprentice

Apprentice is a post-execution performance analysis tool for message passing programs [3]. Originally it was designed to support the CRAFT programming model on the CRAY T3E predecessor system, the CRAY T3D. Apprentice analyzes summary information collected at runtime via an instrumentation of the source program.

The instrumentation is performed by the compiler and is triggered via an appropriate compiler switch. To reduce the overhead of the instrumentation, the programmer can selectively compile the source files with and without instrumentation. The instrumentation is done in a late phase of the compilation after all optimizations already occurred. This prevents that instrumentation affects the way code is compiled. During runtime, summary information is collected at each processor for each basic block. This information comprises:

- execution time
- number of floating point, integer, and load/store operations
- instrumentation overhead

For each subroutine call the execution time as well as the pass count is determined. At the end of a program run, the information of all processors is summed up and written to the *runtime information file (RIF)*. In addition to the summed up execution times and pass counts of subroutines calls, their mean value and standard deviation, as well as the minimum and maximum values are stored.

The size of the resulting RIF is typically less than one megabyte. But the overhead due to the instrumentation can easily be a factor of two which results from instrumenting each basic block. This severe drawback of the instrumentation is partly compensated in Apprentice by correcting the timings based on the measured overhead.

When the user starts Apprentice to analyze the collected information, the tool first reads the RIF as well as the CIFs of the individual source files. The performance data measured for the optimized code are related back to the original source code. Apprentice distinguishes between:

- parallel work: user-level subroutines
- I/O: system subroutines for performing I/O
- communication overhead: MPI and PVM routines, SHMEM routines
- uninstrumented code

The available barcharts allow the user to identify critical code regions that take most of the execution time or with a lot of I/O and communication overhead. Since all the values have been summed up, no specific behavior of the processors can be identified. Load balance problems can be detected by inspecting the execution times of calls to synchronization subroutines, such as global sums or barriers. Based on the available information, the processors with the least and the highest execution time can be identified.

While Apprentice does not evaluate the hardware performance counters of the DEC Alpha, it estimates the loss due to cache misses and suboptimal use of the functional units. Based on the number of instructions and a very simple cost model (fixed cycles for each type of instruction) it determines the loss as the difference between the estimated optimal and the measured execution time.

4.4 VAMPIR

VAMPIR (Visualization and Analysis of MPI Resources) is an event trace analysis tool [11] which was developed by the Central Institute for Applied Mathematics of the Research Centre Jülich and now is commercially distributed by a German company named PALLAS. Its main application area is the analysis of parallel programs based on the message passing paradigm but it also has been successfully used for other areas (e.g., for SVM-Fortran traces to analyze shared virtual memory page transfer behavior [9] or to analyze CRAY T3E usage based on accounting data). VAMPIR has three components:

- The VAMPIR tool itself is a graphical event trace browser implemented for the X11 Window system using the Motif toolkit. It is available for any major UNIX platform.
- The VAMPIR runtime library provides an API for collecting, buffering, and generating event traces as well as a set of wrapper routines for the most commonly used MPI and PVM communication routines which record message traffic in the event trace.

- In order to observe functions or subroutines in the user program, their entry and exit has to be instrumented by inserting calls to the VAMPIR runtime library. Observing message passing functions is handled by linking the program with the VAMPIR wrapper function library.

  VAMPIR comes with a source instrumenter for ANSI Fortran 77. Programs written in other programming languages (e.g., C or C++) have to be instrumented manually. To improve this situation, our institute in collaboration with CRAY Research is currently implementing an object code instrumenter for CRAY T3E. This is described in the next section.

During the execution of the instrumented user program, the VAMPIR runtime library records entry and exits to instrumented user and message passing functions and the sending and receiving of messages. For each message, its tag, communicator, and length is recorded. Through the use of a configuration file, it is possible to switch the runtime observation of specific functions on and off. This way, the program doesn't have to be re-instrumented and re-compiled for every change in the instrumentation.

Large parallel programs consist of several dozens or even hundreds of functions. To ease the analysis of such complex programs, VAMPIR arranges the functions into groups, e.g., user functions, MPI routines, I/O routines, and so on. The user can control/change the assignment of functions to groups and can also define new groups.

VAMPIR provides a wide variety of graphical displays to analyze the recorded event traces:

- The dynamic behavior of the program can be analyzed by timeline diagrams for either the whole program or a selected set of nodes. By default, the displays show the whole event trace, but the user can zoom-in to any arbitrary region of the trace. Also, the user can change the display style of the lines representing messages based on their tag/communicator or the length. This way, message traffic of different modules or libraries can easily be visually separated.

- The parallelism display shows the number of nodes in each function group over time. This allows to easily locate specific parts of the program, e.g., parts with heavy message traffic or I/O.

- VAMPIR also provides a large number of statistical displays. It calculates how often each function or group of functions got called and the time spent in there. Message statistics show the number of messages sent, and the minimum, maximum, sum, and average length or transfer rate between any two nodes. The statistics can be displayed as barcharts, histograms, or textual tables.

  A very useful feature of VAMPIR is that the statistic displays can be linked to the timeline diagrams. By this, statistics can be calculated for any arbitrary, user selectable part of the program execution.

- If the instrumenter/runtime library provides the necessary information in the event trace header, the information provided by VAMPIR can be related back

to source code. VAMPIR provides a source code and a call graph display to show selected functions or the location of the send and the receive of a selected message.

In summary, VAMPIR is a very powerful and highly configurable event trace browser. It displays trace files in a variety of graphical views, and provides flexible filter and statistical operations that condense the displayed information to a manageable amount. Rapid zooming and instantaneous redraw allow to identify and focus on the time interval of interest.

4.5 PAT

PAT (Performance Analysis Tool) is the second performance tool available from CRAY Research for CRAY T3E. The two main differences to Apprentice are that no source code instrumentation or special compiler support is necessary. The user only needs to re-link his/her application against the PAT runtime library (because CRAY Unicos doesn't support dynamic linking). Second, PAT aims at keeping the additional overhead to measure/observe program behavior as low as possible. PAT is actually three performance tools in one:

1. PAT allows the user to get an rough overview about the performance of the parallel program through a method called sampling, i.e., interrupting the program at regular intervals and evaluating the program counter. PAT can calculate then the percentage of time spent in each function. The sampling rate can be changed by the user to adapt it to the execution time of the program and to keep overhead low. Because the sampling method provides only a statistical estimate of the actual time spent in a function, the tool also provides a measure of confidence in the sampling estimate.

   In addition PAT determines the total, user, and system time of the execution run and the number of cache misses and the number of either flointing point, integer, store, or load operations. These are measured through the DEC Alpha hardware counters. The user can select the hardware counter by setting an environment variable. All this statistical information is stored after the execution in a so-called *Performance Information File* (PIF).

2. If a more detailed analysis is necessary, PAT can be used to instrument and analyze a specific function or set of functions in a second phase. PAT can instrument object code (however only on the function level). This is a big advantage especially for large complex programs because they do not have to be re-compiled for instrumentation. In addition, it is possible to analyze functions contained in system or 3rd-party libraries. A third advantage is that programs written in more than one language can be handled. The big disadvantage is that it is more difficult to relate the results back to the source code.

   This detailed investigation of function behavior is called *Call Site Report* by PAT. It records for each call site of the instrumented functions how often it got called and time spent in this instantiation of the function. Execution times are

measured with a high-resolution timer. The results are available for each CPU used in the parallel program. The next version of PAT will allow to gather hardware counter statistics for instrumented functions as well.

3. Last, if a very detailed analysis of the program behavior is necessary, PAT also supports event tracing. The object instrumenter of PAT can also be used to insert calls to entry and exit trace routines around calls to user or library routines. Entry and exit trace routines can be provided in two ways:

   - The user can supply function-specific wrapper functions. The routines must be written in C, they must have the same number and same types of arguments as the routine they are tracing, and finally, the wrapper function name for a function *func* must be *func*_trace_entry for entry trace routines and *func*_trace_exit for exit trace routines.
   - If specific wrapper routines for the requested function are not available, PAT uses generic wrapper code which just records the entry and exit of the function in the event trace.

   In addition, PAT provides extra tracing runtime system calls, which can be inserted in the source code and allow to switch tracing on and off, and to insert additional information into the trace (e.g., information unrelated to functions).

   The tracing features of PAT were developed in a collaboration of Research Centre Jülich with CRAY Research. Our institute implemented all the necessary special wrapper functions for all message passing functions available on the T3E (MPI, PVM, and SHMEM and for both the C and Fortran interfaces) which record the message traffic in the event trace. In addition, we implemented a tool for converting the event traces contained in PIF files into VAMPIR trace format.

The major drawback of PAT's object instrumentation is the very low-level interface for specifying the functions to be instrumented. The user has to specify the function names as they appear in the object code, i.e., C++ functions or F90 functions which are local or contained in modules have to be specified in the mangled form (e.g., "__0FDfooPd" instead of the C++ function name "int foo(double *)"). Clearly, a more user friendly or automatic way for the instrumenter interface needs to be added to PAT.

In addition, the combination of three different instrumentation/analysis techniques into a single tool is very confusing for users. This confusion is further increased since the supported techniques overlap with the techniques applied in the other tools.

4.6 Summary

The previous sections pointed out that the CRAY T3E has a programming environment that includes the most advanced performance analysis tools. On the other hand, each of these tools comes with its own instrumentation, provides partially overlapping information, and has a totally different user interface. The programmer has to understand the advantages and disadvantages of all the tools to be able

	Apprentice	VAMPIR	PAT		
data collection	instrumentation via compiler	source instrumentation by preprocessor	sampling	object code inst.	
intrusion	high / corrected	high	low	high	
selective instr.	not required, optional	required	—	required	
selection interface	compiler switch	GUI or ASCII file	—	ASCII interface or file	
level of detail	summary information	event trace	statistics	summary information	event trace
information	total time and oper. counts for basic blocks and call sites	subroutine start/stop and send/recv events	statistical distr. of time (subroutines)	total time and pass counts for call sites	subroutine start/stop and send/recv events
strength	source-level, analysis of loops	many displays for message passing history and for statistics of arbitrary execution phases	low overhead profiling	total time for call sites, no recompilation	object code instr. for VAMPIR, no recompilation

Table 1. User interface and properties of performance analysis tools on CRAY T3E

to select and apply the right ones. Table 1 summarizes the main features of these three tools.

5 Summary

This article gave an overview of parallel programming models as well as programming tools. Parallel programming will always be a challenge for programmers. Higher-level programming models and appropriate programming tools only facilitate the process but do not make it a simple task.

While programming in MPI offers the greatest potential performance, shared memory programming with OpenMP is much more comfortable due to the global style of the resulting program. The sequential control flow among the parallel loops and regions matches much better with the sequential programming model all the programmers are trained for.

Although program tools were developed over years, the current situation seems not to be very satisfiable. Program debugging is done on a per thread basis, a technique that does not scale to larger numbers of processors. Performance anal-

ysis tools do also suffer scalability limitations and, in addition, those tools are complicated to use. The programmers have to be experts for performance analysis to understand potential performance problems, their proof conditions and their serverity. In addition they have to be experts for powerful but also complex user interfaces.

Future research in that area has to try to automate performance analysis tools, such that frequently occuring performance problems can be identified automatically. It is the goal of the ESPRIT IV working group APART on *Automatic Performance Analysis: Resources and Tools* to investigate base technologies for future more intelligent tools [1]. A first result of the work ia a collection of performance problems for parallel programs that have been formalized with the ASL, the *APART Specification Language* [6]. This approach will lead to a formal representation of the knowledge applied in the manually executed performance analysis process and thus will make this knowledge accessible for automatic processing.

A second important trend that will effect parallel programming in the future is the move towards clustered shared memory systems. Within the GoSMP study executed for the Federal Ministry for Education and Research its influence on program development is investigated. Clearly, a hybrid programming approach will be applied on those systems for best performance, combining message passing between the individual SMP nodes and shared memory programming in a node. This programming model will lead to even more complex programs and program development tools have to be enhanced to be able to help the user in developing those codes.

References

1. APART: *ESPRIT Working Group on Automatic Performance Analysis Resources and Tools*, `www.fz-juelich.de/apart`, 1999.
2. D.P. Bertsekas, J.N. Tsitsiklis: *Parallel and Distributed Computation: Numerical Methods*, Prentice-Hall, ISBN 0-13-648759-9, 1989.
3. Cray Research: *Cray MPP Fortran: Reference Manual*, Cray SR-2504, 1994.
4. D.E. Culler, J.P. Singh, A. Gupta: *Parallel Computer Architecture - A Hardware/Software Approach*, Morgan Kaufmann Publishers, ISBN 1-55860-343-3, 1999.
5. L. Dagum, R. Menon: *OpenMP: An Industry-Standard API for Shared-memory Programming*, IEEE Computational Science & Engineering, 1998.
6. Th. Fahringer, M. Gerndt, G. Riley, J. Träff: *Knowledge Specification for Automatic Performance Analysis*, APART Technical Report, Research Centre Jülich, FZJ-ZAM-9918, 1999.
7. I. Foster: *Designing and Building Parallel Programs*, Addison Wesley, ISBN 0-201-57594-9, 1994.
8. G. Fox: *Domain Decomposition in Distributed and Shared Memory Environments*, International Conference on Supercomputing June 8-12, 1987, Athens, Greece Lecture Note in Computer Science **297**, edited by Constantine Polychronopoulos, 1987.
9. M. Gerndt, A. Krumme, S. Özmen: *Performance Analysis for SVM-Fortran*

with OPAL, Proceedings Int. Conf. on Parallel and Distributed Processing Techniques and Applications (PDPTA'95), Athens, Georgia, pp. 561-570, 1995.
10. MPI Forum: *Message Passing Interface*, `www.mcs.anl.gov/mpi`, 1999.
11. W.E. Nagel, A. Arnold, M. Weber, H-C. Hoppe, K. Solchenbach: *VAMPIR: Visualization and Analysis of MPI Resources*, Supercomputer 63, Vol. **12**, No. 1, pp. 69-80, 1996.
12. OpenMP Forum: *OpenMP Standard*, `www.openmp.org`, 1999.
13. M. Snir, St. Otto, St. Huss-Lederman, D. Walker, J. Dongarra: *MPI - The Complete Reference*, MIT Press, ISBN 0-262-69216-3, 1998.
14. Etnus Inc.: *Totalview*, `www.etnus.com/products/totalview`, 1999.

BASIC NUMERICAL LIBRARIES FOR PARALLEL SYSTEMS

INGE GUTHEIL
John von Neumann Institute for Computing
Central Institute for Applied Mathematics
Research Centre Jülich, 52425 Jülich
Germany
E-mail: i.gutheil@fz-juelich.de

Three public domain libraries with basic numerical operations for distributed memory parallel systems are presented: ScaLAPACK, PLAPACK, and Global Arrays. They are compared not only with respect to performance on CRAY T3E but also to user-friendliness.

1 Introduction

There are many projects for parallelization of numerical software. An overview of public domain libraries for high performance computing can be found on the HPC-Netlib homepage[1]. Often these libraries are very specialized either concerning the problem which is treated or the platform on which they run. Three of the more general packages based on message-passing with MPI will be presented here in some detail: ScaLAPACK[2], PLAPACK[3], and Global Arrays[4].

Often there is an MPI-implementation on shared-memory multiprocessor systems, hence libraries based on MPI can also be used there and often very efficiently as message-passing programs take great care of data locality.

1.1 ScaLAPACK, Scalable Linear Algebra PACKage

The largest and most flexible public domain library with basic numerical operations for distributed memory parallel systems up to now is ScaLAPACK. Within the ScaLAPACK project many LAPACK[5] routines were ported to distributed memory computers using message passing.

The communication in ScaLAPACK is based on the BLACS (Basic Linear Algebra Communication Subroutines)[6]. There are public domain versions of the BLACS based on MPI and PVM available. For CRAY T3E there is also a version of the BLACS in Cray scientific libraries (libsci)[7] using Cray shared memory routines (shmem) which is often faster than the public domain versions.

The basic routines of ScaLAPACK are the PBLAS (Parallel Basic Linear Algebra Subroutines). They contain parallel versions of the BLAS[a], which are parallelized using BLACS for communication and sequential BLAS for computation. As most vendors offer optimized sequential BLAS the BLAS and PBLAS deliver very good performance on most parallel computers.

Based on BLACS and PBLAS ScaLAPACK contains parallel solvers for dense linear systems and linear systems with banded system matrix as well as parallel

[a]BLAS 1 contains vector-vector operations, e.g. dotproduct, BLAS 2 matrix-vector operations, e.g. matrix-vector multiplication, BLAS 3 matrix-matrix operations, e.g. matrix-matrix multiplication

routines for the solution of linear least squares problems and for singular value decomposition. Routines for the computation of all or some of the eigenvalues and eigenvectors of dense real symmetric matrices and dense complex hermitian matrices and for the generalized symmetric definite eigenproblem are also included in ScaLAPACK.

ScaLAPACK also contains additional libraries to treat distributed matrices and vectors. One of them is the TOOLS library, which offers useful routines for example to find out which part of the global matrix a local process has in its memory or to find out the global index of a matrix element corresponding to its local index and vice versa. Unfortunately these routines are documented only in the source code of the routines and not in the Users' Guide. Another library is the REDIST library which is documented in the ScaLAPACK Users' Guide. It contains routines to copy any block-cyclicly distributed (sub)matrix to any other block-cyclicly distributed (sub)matrix.

ScaLAPACK is a Fortran 77 library which uses C subroutines internally to allocate additional workspace. Especially the PBLAS allocate additional workspace.

1.2 PLAPACK, Parallel Linear Algebra PACKage

PLAPACK does not offer as many black-box solvers as ScaLAPACK but is designed as a parallel infrastructure to develop routines for solving linear algebra problems. With PLAPACK routines the user can create global matrices, vectors, and multiscalars, and he may fill them with values with the help of an API (Application Programming Interface). To make the development of programs easier and to get good performance PLAPACK includes parallel versions of most real BLAS routines and solvers for real dense linear systems using LU-decomposition and for real symmetric positive definite systems applying Cholesky-decomposition which operate on the global data.

PLAPACK is a C library with a Fortran interface in Release 1.2. Unfortunately up to now Release 1.2 does not run on CRAY T3E.

1.3 Global Arrays

Like PLAPACK Global Arrays supplies the user with global linear algebra objects and an interface to fill them with data. For the solution of linear equations there is an interface to special ScaLAPACK routines which can be modified to use other ScaLAPACK routines, too. For the solution of the full symmetric eigenvalue problem Global Arrays contains an interface to PeIGS[8].

Global Arrays is a Fortran 77 library which uses a memory allocator library for dynamic memory management.

2 Data Distributions

There are many ways to distribute data, especially matrices, to processors. In the ScaLAPACK Users' Guide many of them are presented and discussed.

All of the three libraries described here distribute matrices to a two-dimensional processor grid, but they do it in different ways and the user can more or less influence

the way data are distributed. Users who only want to use routines from the libraries don't have to care for the way data are distributed in PLAPACK and Global Arrays. The distribution is done automatically. To use ScaLAPACK, however, the user has to create and fill the local parts of the global matrix on his own.

2.1 ScaLAPACK: Two-dimensional Block-Cyclic Distribution

For performance and load balancing reasons ScaLAPACK has chosen a two-dimensional block-cyclic distribution for full matrices (see ScaLAPACK Users' Guide). First the matrix is distributed to blocks of size $MB \times NB$. These blocks are then uniformly distributed across the $NP \times NQ$ processor grid in a cyclic manner. As a result, every process owns a collection of blocks, which are contiguously stored in a two-dimensional "column major" array.

This local storage convention allows ScaLAPACK software to efficiently use local memory by calling BLAS 3 routines on submatrices that may be larger than a single $MB \times NB$ block. Figure 1 shows the distribution of a 9×9-matrix subdivided into blocks of size 3×2 distributed across a 2×2-processor grid.

	0		1		0		1		0
0	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}	a_{19}
	a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}	a_{27}	a_{28}	a_{29}
	a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}	a_{37}	a_{38}	a_{39}
1	a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}	a_{47}	a_{48}	a_{49}
	a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}	a_{57}	a_{58}	a_{59}
	a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}	a_{67}	a_{68}	a_{69}
0	a_{71}	a_{72}	a_{73}	a_{74}	a_{75}	a_{76}	a_{77}	a_{78}	a_{79}
	a_{81}	a_{82}	a_{83}	a_{84}	a_{85}	a_{86}	a_{87}	a_{88}	a_{89}
	a_{91}	a_{92}	a_{93}	a_{94}	a_{95}	a_{96}	a_{97}	a_{98}	a_{99}

Figure 1. Block-cyclic 2D distribution of a 9×9-matrix subdivided into 3×2-blocks to a 2×2-processor grid. The numbers outside the matrix indicate processor row and column indices respectively.

2.2 PLAPACK: "Physically based Matrix Distribution"

For those who want to develop programs which operate on PLAPACK distributed vectors and matrices it may be interesting to know more about the way data are distributed.

The distribution of matrices is induced by the distribution of vectors in a linear system $A\vec{x} = \vec{y}$: Vectors are divided into blocks of length NB, where NB is a blocking factor chosen by the user. These blocks are distributed to the processor grid in "column-first-order", i.e. processor (0,0) gets the first block, processor (1,0) the second one and so on.

The distribution of the matrix is now induced by requiring a column of matrix A to be assigned to the same column of processors as the corresponding element of $\vec{x}$

and the rows of A to the same row of processors as the corresponding element of $\vec{y}$, e.g. processor (1,0) has x_2 and y_2, consequently the second row of A is distributed to the second row of processors and the second column of A is distributed to the first processor column.

Figure 2 first shows a 2×3-processor-grid and then a vector of length 7 distributed to it in "column-first-order" with block size 1. Below there is shown how a 7×7-matrix is distributed to the processor grid accordingly.

2×3
processor-grid

(0,0)	(0,1)	(0,2)
(1,0)	(1,1)	(1,2)

distribution of the vector,
$NB = 1$

$$
\begin{aligned}
x_1 &\rightarrow (0,0)\\
x_2 &\rightarrow (1,0)\\
x_3 &\rightarrow (0,1)\\
x_4 &\rightarrow (1,1)\\
x_5 &\rightarrow (0,2)\\
x_6 &\rightarrow (1,2)\\
x_7 &\rightarrow (0,0)
\end{aligned}
$$

induced distribution of the matrix

		0			1		2	
		1	2	7	3	4	5	6
0	1	a_{11}	a_{12}	a_{17}	a_{13}	a_{14}	a_{15}	a_{16}
	3	a_{31}	a_{32}	a_{37}	a_{33}	a_{34}	a_{35}	a_{36}
	5	a_{51}	a_{52}	a_{57}	a_{53}	a_{54}	a_{55}	a_{56}
	7	a_{71}	a_{72}	a_{77}	a_{73}	a_{74}	a_{75}	a_{76}
1	2	a_{21}	a_{22}	a_{27}	a_{23}	a_{24}	a_{25}	a_{26}
	4	a_{41}	a_{42}	a_{47}	a_{43}	a_{44}	a_{45}	a_{46}
	6	a_{61}	a_{62}	a_{67}	a_{63}	a_{64}	a_{65}	a_{66}

Figure 2. Physically based distribution of a 7×7-matrix with block size 1 to a 2×3-processor-grid. The outmost numbers indicate processor row and column indices respectively, the next ones are matrix row and column indices.

Thus processor (0,0) has elements of rows 1, 3, 5, and 7 and columns 1, 2, and 7 because elements 1, 3, 5, and 7 of the vector are assigned to processor row 0 and elements 1, 2, and 7 of the vector are assigned to processor column 0.

2.3 Global Arrays: Two-dimensional Block Distribution

The global objects in this library are distributed in a very simple way and the user has little influence on that. Matrices are distributed into contiguous blocks and each process gets one of these blocks. The user can only choose the minimum number of rows or columns in a block. With this he can force for example a column block distribution by setting the minimum number of rows per block to the total number of rows of the global matrix.

When routines from other libraries are called there is an interface where data are redistributed in the way the other library expects them.

3 User-Interfaces

The user-interface of a library influences the decision for or against it. An easy-to-use interface can significantly reduce parallelization time.

3.1 ScaLAPACK

ScaLAPACK as a parallel successor of LAPACK attempts to leave the calling sequence of the subroutines unchanged as much as possible in comparison to the corresponding sequential subroutine from LAPACK. The user should have to change only a few parameters in the calling sequence to use ScaLAPACK routines instead of LAPACK routines.

Therefore ScaLAPACK uses so-called descriptors, which are integer arrays containing all necessary information about the distribution of a matrix. This descriptor appears in the calling sequence of the parallel routine instead of the leading dimension of the matrix in the sequential one.
For example the sequential BLAS 3 routine for the computation of $C = \alpha AB + \beta C$, A an $M \times K$-matrix, B a $K \times N$-matrix, overwriting the original C with the result, has the following calling sequence:

```
...
CALL SGEMM(TRANSA,TRANSB,M,N,K,alpha,A(1,1),LDA, &
           B(1,1),LDB,beta,C(1,1),LDC)
...
```

whereas the ScaLAPACK routine PSGEMM is called

```
...
! Call of PSGEMM with descriptors and the global
! starting indices of the whole matrix
CALL PSGEMM(TRANSA,TRANSB,M,N,K,alpha,A,1,1,DESCA, &
           B,1,1,DESCB,beta,C,1,1,DESCC)
...
```

Instead of taking the whole matrix starting with $A(1,1)$, any contiguous submatrix starting with $A(I,J)$, I and J global indices, can be multiplied with a submatrix of B starting with $B(J,L)$ by calling

```
...
CALL PSGEMM(TRANSA,TRANSB,M-I+1,N-L+1,K-J+1,alpha,A,I,J,DESCA, &
           B,J,L,DESCB,beta,C,I,L,DESCC)
...
```

The main problem is that the user has to take care of the data distribution. He has to choose the processor grid by initializing MP, the number of processor rows, and NP, the number of processor columns and to determine the blocking by choosing MB and NB, the number of rows and the number of columns per block, respectively. For many routines, especially for the eigenvalue solvers and the Cholesky decomposition, $MB = NB$ is necessary.

The conversion of global to local indices and vice versa is supported only by some auxiliary routines in the TOOLS sublibrary. It is completely left to the user to put the correct local part of the matrix to the right places and to put the correct data to the descriptor. The Users' Guide and the comments at the beginning of all routines are sufficient to use ScaLAPACK correctly but for someone not familiar with parallel programming it can be rather difficult and time-consuming to learn how to use it.

The main steps the user has to perform for creating and filling a matrix A are (it is assumed that MB=NB and N=M=K):

```
...
! Create the MP * NP processor grid
CALL BLACS_GRIDINIT(ICTXT,'Row-major',MP,NP)
! Find my processor coordinates MYROW and MYCOL
! NPROW should return same value as MP,
! NPCOL should return same value as NP
CALL BLACS_GRIDINFO(ICTXT,NPROW,NPCOL,MYROW,MYCOL)
! Compute local dimensions with routine NUMROC from TOOLS
! N is dimension of the matrix, NB is block size
MYNUMROWS = NUMROC(N,NB,MYROW,0,NPROW)
MYNUMCOLS = NUMROC(N,NB,MYCOL,0,NPCOL)
! Local leading dimension of A,
! Number of local rows of A
MXLLDA = MYNUMROWS
! Allocate only the local part of A
ALLOCATE(A(MXLLDA,MYNUMCOLS))
! Fill the descriptors, P0 and Q0 are processor coordinates
! of the processor holding global element A(0,0)
CALL DESCINIT(DESCA,N,N,NB,NB,P0,Q0,ICTXT,MXLLDA,INFO)
! Fill the local part of the matrix with data
do j = 1, MYNUMCOLS, NB   ! Fill the local column blocks
   do jj=1,min(NB,MYNUMCOLS-j+1) ! All colums of one block
      jloc = j-1 + jj   ! local column index
      jglob = (j-1)*NPCOL + MYCOL*NB +jj ! global column index
      do i = 1, MYNUMROWS, NB ! The local row blocks in this column
         do ii=1,min(NB,MYNUMROWS-i+1) ! The rows in this row block
            iloc = i-1 + ii   ! local row index
            iglob = (i-1)*NPROW + MYROW*NB + ii ! global row index
            A(iloc,jloc) = function of global indices iglob, jglob
         enddo
      enddo
   enddo
enddo
...
```

The four nested loops show how local and global indices can be computed from block sizes, the number of rows and columns in the processor grid and the processor coordinates.

The calling sequences of PLAPACK routines are also very similar to the ones of BLAS and LAPACK, e.g. the matrix-matrix-multiplication routine PLA_Gemm is called in the following way:

```
...
/* Call PLA_Gemm with global objects */
PLA_Gemm ( PLA_NO_TRANSPOSE, PLA_NO_TRANSPOSE, alpha,
           A, B, beta, C );
...
```

With PLAPACK always the whole global matrix is treated as the sizes and the distribution are implicitly contained in the global object. If the user wishes to deal with a submatrix only he has to create a so-called view into the matrix which is a new distributed object using the data and the memory locations of the whole matrix.

To write to or read entries from the global linear algebra objects of PLAPACK there is an Application Program Interface (API) which must be started and finished and during which no other communication should take place. Within the API a global matrix can be filled columnwise or blockwise. On CRAY T3E large matrices must be filled by larger blocks as there is a limit in the number of MPI messages which can be open simultaneously, and columnwise filling of global matrices causes too many open messages.

To start PLAPACK and the API and fill a global matrix, e.g. column block wise, the user has to do the following after initializing MPI (again N=M=K):

```
...
/* Create a 2D-Communicator */
PLA_Comm_1D_to_2D(MPI_COMM_WORLD, mp, np, &comm);
/* Create an object distribution template */
PLA_Temp_create( nb_distr, ist0_1, &templ );
/* Create the global matrices */
PLA_Matrix_create( datatype, N, N, templ,
                PLA_ALIGN_FIRST, PLA_ALIGN_FIRST, &A );
...
/* Create a global scalar */
PLA_Mscalar_create( MPI_DOUBLE, PLA_ALL_ROWS,
                  PLA_ALL_COLS, 1, 1, templ, &alpha );
...
/* Initialize the matrices to equal zero */
  PLA_Obj_set_to_zero ( A );
...
/* Enter Application Interface mode */
PLA_API_begin();
/* Open object A, ... for read/write  */
PLA_Obj_API_open(A);
...
/* Create a work buffer for computing one column block */
```

```
/* of the global matrix A */
/* locA is a local array */
locA = pla_calloc(N*fill_blk_size, type_size);
/* Column blocks are computed by processors in a round-robin fashion */
for (j=me*fill_blk_size;j< N; j+=nprocs*fill_blk_size) {
   int jb, jj;
   jb = min( fill_blk_size, N-j );
   /* Fill column block j of width fill_blk_size */
   for (jj=0; jj<jb; jj++) {
     for (i=0; i < N; i++) {
        ((double *)locA)[jj*N+i] = function of (j+jj,i) ;
     }
   }
   /* Add the column block locA containing jb columns */
   /* to the global matrix A at the location starting with*/
   /* global index (ist0_1+0,ist0_1+j) */
   PLA_API_axpy_matrix_to_global( N, jb, &d_one, locA,
                                N, A, 0,j );
   /* synchronization after filling in a block of A */
   PLA_Obj_API_sync(A);
}
/* Close the objects */
PLA_Obj_API_close(A);
...
/* Free the workspace */
pla_free( locA );
/* leave Application Interface mode */
PLA_API_end();
```

3.3 Global Arrays

The usage of Global Arrays is described in Th. Steinke's[9] article in these proceedings.

4 Performance

All performance measurements for the matrix-multiplication routines and the routines for the solution of linear systems with LU-decomposition from ScaLAPACK/PBLAS, PLAPACK and Global Arrays and for the solution of the full symmetric eigenvalue problem with ScaLAPACK and Global Arrays (PeIGS) were done on a 256-node CRAY T3E-900 with 128 MB RAM on each node. Some diagrams with performance results are shown in the appendix.

Execution times were measured for various block sizes and grid shapes and in the diagrams collected in the appendix always the shortest time for each matrix size and processor number is shown.

Additionally we looked at the routines with the performance analysis tool PAT[10]

and the performance counter library PCL[11]. We used these tools to find out which part of the execution time was spent with communication and in different BLAS routines and to get operation counts per node to see how well the load was balanced.

4.1 Matrix-multiplication and LU-decomposition

We measured execution times for the multiplication $C = 2AB + 3C$ with A, B, and C square matrices of size n and for the LU-decomposition of a square matrix A of size n with the solution of the resulting triangular system with n right-hand-sides. The values of n were $n = 1200, \ldots, 6000$ on 12 nodes, $n = 6000, \ldots, 12500$ on 64 nodes and $n = 6000$ on 10 to 50 nodes.

Global Arrays uses an interface to ScaLAPACK for the solution of a linear system with LU-decomposition. The block size for matrix distribution is 64 for all blockings. This is fixed as a parameter in the interface routine.

For matrix-multiplication Global Arrays contains a routine which uses a blocked version of the usual nested loops, distributes matrix blocks to the processors and calls SGEMM[b] on each node. As this leads to high communication, we modified the ScaLAPACK interface for LU-decomposition to one for matrix-multiplication. This resulted in much better performance for all problems we measured.

4.2 Solution of the Full Symmetric Eigenvalue Problem

For the solution of the full symmetric eigenproblem we measured execution times for the computation of all eigenvalues and eigenvectors of a real full symmetric matrix of size n. The times were measured on 4, 8, 16, 25, 32, 36, and 64 nodes and problem sizes varied from $n = 400$ on four nodes and $n = 800$ on more than four nodes to the maximum n possible on that number of nodes.

A detailed study of the performance of the dense symmetric eigensolvers from ScaLAPACK and PeIGS called by Global Arrays can be found in an internal report of Research Centre Jülich[12].

ScaLAPACK contains two driver routines for the solution of the full symmetric eigenproblem, PSSYEVX, the so-called expert-driver, and PSSYEV, the simple driver. Both compute eigenvalues and optionally eigenvectors by a three-step-algorithm: Reduction of the full matrix to tridiagonal form via Householder transformations, computation of the eigenvalues and (optionally) the eigenvectors of the tridiagonal matrix and back transformation of the eigenvectors to those of the original matrix. Global Arrays' routine GA_DIAG_STD calls PDSPEV from PeIGS and uses the same three steps to compute all eigenvalues and eigenvectors of a real full symmetric matrix.

PSSYEVX and GA_DIAG_STD use parallel bisection and inverse iteration for the computation of the eigenvalues and eigenvectors of the tridiagonal matrix whereas in PSSYEV the eigenvalues of the tridiagonal matrix are computed redundantly (and sequentially) on all nodes via a modified QR-algorithm and only the computation of the eigenvectors is done in parallel.

[b] BLAS 3 routine for matrix-matrix-multiplication, contained in libsci

PSSYEV only allows to compute all eigenvalues of the matrix and optionally all eigenvectors whereas in PSSYEVX the user can choose a range of eigenvalues to be computed with or without the corresponding eigenvectors. GA_DIAG_STD always computes all eigenvalues and all eigenvectors.

If there are clusters of eigenvalues inverse iteration does not guarantee orthogonality of the corresponding eigenvectors and therefore they have to be reorthogonalized if orthogonal eigenvectors are required. This is done on one single processor for one cluster in PSSYEVX.

If there is one very large cluster of eigenvalues (more than 2000) there is not enough memory (128 MB RAM) for the reorthogonalization of the eigenvectors of this cluster on one node. There is an additional parameter ORFAC in the calling sequence of PSSYEVX which does not appear in SSYEVX, the corresponding LAPACK routine. If ORFAC is set to zero, no reorthogonalization is done and execution times of PSSYEVX are the same whether eigenvalues are clustered or not. Eigenvectors are no longer orthogonal to machine precision. However, the eigenvectors still are nearly orthogonal to an accuracy which might be sufficient in many cases.

In GA_DIAG_STD this problem is solved by a parallel version of the reorthogonalization. It is said in the Users' Guide that it does not guarantee to always deliver orthogonal eigenvectors but in our study we didn't find a case where it didn't work.

The modified QR-algorithm of PSSYEV guarantees orthogonal eigenvectors even for large clusters of eigenvalues and it is even a littlebit faster with one large cluster than without clusters. On the other hand, it needs about twice as many operations per node as PSSYEVX if no eigenvectors have to be reorthogonalized.

4.3 Factors that Influence Performance

There are many factors that affect performance on an MPP system. The user can influence some of them, but others are only influenced by choosing between the libraries.

4.3.1 Usage of BLAS Routines

All tested library routines use BLAS routines for single node computations, so vendor optimized BLAS routines, on CRAY T3E those from libsci, are an important factor for performance. Due to the small level 1 cache on T3E and BLAS 1 routines becoming very slow when data are not in level 1 cache, for all but very small problems performance of BLAS 1 routines is very poor. BLAS 2 routines still can not deliver high performance, so it is preferable to use BLAS 3 routines because cache reuse is possible here.

For matrix-multiplication and the solution of a linear system with LU-decomposition, it is no problem to use BLAS 3 routines. As Global Arrays utilizes an interface to ScaLAPACK for solving linear systems via LU-decomposition there is almost no difference in BLAS 3 usage between both libraries.

Although PAT shows higher communication overhead and lower BLAS 3 usage for PLAPACK than for ScaLAPACK, PLAPACK is faster for large problems. Therefore, we think that PAT results seem to be not significant in the case of linear

system solution. This is probably due to the fact that we could not find out how much of the communication overhead indicated for PLAPACK was due to the filling of the global matrix in the beginning and how much was due to the tested routine.

The usage of BLAS 3 routines plays an important role when comparing the routines for the solution of the symmetric eigenvalue problem. From Table 1 it can be seen that GA_DIAG_STD from Global Arrays, which calls PDSPEV from PeIGS, is based on BLAS 1 routines SDOT and SAXPY whereas PSSYEV and PSSYEVX from ScaLAPACK call the BLAS 3 routine SGEMM and the BLAS 2 routine SGEMV whenever possible. As a result if there is no large cluster of eigenvalues whose eigenvectors have to be reorthogonalized PSSYEVX is much faster than GA_DIAG_STD and reaches a higher MFLOPS rate per node for large problems(see Table 2), although they both use the same algorithm. We can up to now not explain why GA_DIAG_STD needs so much more operations than PSSYEVX. This surely also leads to higher execution times.

Table 1. Percentage of time spent in different BLAS routines. The ranges in percentage arise from different numbers of nodes. For large problems $\geq$ means that for larger problems the percentage is still higher.

no clusters	percentage of time spent in	small problem 200×200 elements per node	large problem $\geq 1000 \times 1000$ elements per node
PSSYEVX	BLAS 3 SGEMM	6 - 8%	$\geq$ 30 %
	BLAS 2 SGEMV	7 - 8 %	$\approx$ 27 %
	BLAS 1	-	-
PSSYEV	BLAS 3 SGEMM	3 - 4 %	$\geq$ 9 %
	BLAS 2 SGEMV	2 - 5 %	7 - 9 %
	BLAS 1 SROT	26 - 33 %	$\geq$ 56 %
GA_DIAG_STD	BLAS 2, 3	-	-
	BLAS 1 SDOT	28 - 41 %	39 - 58 %
	BLAS 1 SAXPY	7 - 8 %	19 - 27 %

If there is a large cluster of eigenvalues the usage of BLAS 3 and BLAS 2 in PSSYEV even is a little bit higher than in the case of non-clustered eigenvalues whereas BLAS 1 usage in GA_DIAG_STD remains almost the same. For large problems this leads to higher MFLOPS rates for PSSYEV than for GA_DIAG_STD and consequently to shorter execution times if there is a large cluster of eigenvalues.

4.3.2 Load Balance and Communication Overhead

The communication overhead is another important factor for MPP performance. Problems must not be too small for a larger number of nodes because more nodes usually mean more communication and less computation per node.

The algorithm choosen also influences communication overhead as can be seen with the matrix-multiplication routine contained in Global Arrays. For a 1000×1000-matrix on 4 nodes the original routine spends only 60 % in SGEMM and 24 %

in communication whereas with the ScaLAPACK interface 77 % of the time is spent in SGEMM and 11 % in communication. Also load balance is worse in the original routine, three of the four nodes perform about 1800 million floating point operations (MFLOP) and the fourth one only about 750 MFLOP,wheras with ScaLAPACK all processors perform about 1600 MFLOP.

Load imbalance leads to a high communication overhead as a lot of time is spent in waits for other processors to finish computation and send data needed to continue. From Table 2 it can be seen that the ScaLAPACK routines have better balanced operation counts than GA_DIAG_STD in the case with no clusters. For small problems the node with most operations has about 8 to 24 % more operations in PSSYEVX, about 3 to 40 % more in PSSYEV and about 50 to 68 % more operations in GA_DIAG_STD than the node with least operations. With large matrices this becomes more extreme. Whereas in both ScaLAPACK routines the difference is less than 10 % of the operation count of the node with least work, in GA_DIAG_STD the node with most operations has up to 70 % more operations to do than the one with least operations.

Table 2. Millions of floating point operations and MFLOPS per node, equally spread eigenvalues. The operation counts are the lowest and the highest value per node as delivered by PCL, the MFLOPS are computed by the times measured and these operation counts. Only the highest MFLOPS/node rate is shown. On the other nodes MFLOPS rates are lower mainly because of waits.

Million operations per node (MFLOPS per node)	no clusters	small problem 200×200 elements per node	large problem 1000×1000 elements per node
PSSYEVX	4 nodes	82-92 (70)	8190-8380 (180)
	32 nodes	219-245 (60)	22000-22700 (180)
	64 nodes	268-331 (55)	30700-32100 (180)
PSSYEV	4 nodes	175-184 (80)	16800-17000 (130)
	32 nodes	455-631 (65)	50400-54200 (130)
	64 nodes	674-945 (65)	74600-81100 (125)
GA_DIAG_STD	4 nodes	124-186 (100)	14500-22000 (115)
	32 nodes	349-585 (75)	37600-64800 (80)
	64 nodes	494-768 (60)	54300-92700 (75)

In the case of one large cluster of eigenvalues load balance remains almost the same for PSSYEV. For GA_DIAG_STD it becomes more imbalanced as reorthogonalization plays an important role.

The most extreme example for load imbalance is the reorthogonalization of eigenvectors belonging to a large cluster of eigenvalues which is done sequentially on one single node in PSSYEVX. There it can be seen that with 64 nodes and a problem size of $n = 1600$ and a cluster of 1333 eigenvalues about 94 % of the execution time summed up over all nodes is spent in communication/wait. 62 of the 64 nodes only have to execute about 260-320 MFLOP, one node about 560 MFLOP (orthogonalization of the eigenvectors belonging to one smaller cluster),

and one node has to perform about 21900 MFLOP.

4.3.3 Block Sizes and Grid Shapes

ScaLAPACK as well as PLAPACK allow the user to choose block sizes for distribution of vectors and matrices. This size can influence load balance and communication overhead. Small blocks lead to better load balance but to higher communication.

Block sizes have more influence on the performance of routines for solving linear systems than on the performance of routines for the solution of the full symmetric eigenvalue problem.

In ScaLAPACK the system matrix for LU-decomposition has to be distributed into square blocks, i.e. $MB = NB$, but the matrix of the right-hand-sides may be distributed to rectangular blocks with the columns distributed like the system matrix and the rows to blocks of size $NBRHS$.

We found out that for problems with small matrix parts per node ($n = 6000$ on 40 nodes) small block sizes for the system matrix (here $NB = 32$) were best whereas for problems with large matrix parts per node ($n = 6000$ on 12 nodes) larger blocks (here $NB = 64$) were better. For the blocking of the right-hand-sides always a large block size (here $NBRHS = 64$) was best. Powers of two often were slightly better than other numbers of block sizes even if the matrix size wasn't a power of two.

PLAPACK allows to choose only one block size, the one for the distribution of the template vector. Here we could find that as in ScaLAPACK problems with small parts per node perform better with small block sizes ($NB = 32$) and systems with large parts per node with large ($NB = 64$) ones.

For the solution of the symmetric eigenvalue problem with ScaLAPACK routines the system matrix also has to be distributed to square blocks. Here smaller blocks, $NB = 16$ or $NB = 20$ gave best results. For some of the largest problems tested $NB = 32$ delivered the fastest result.

Usually the differences were rather small, but there is one case where the difference is significant. We found out that for block sizes of $NB = 16$ or $NB = 32$ PSSYEVX needs up to twice the time as with a block size of $NB = 20$ if one of the nodes or all the nodes have a local matrix of size 1024×1024 (e.g. 2×2 nodes, $n = 2048, NB = 16$: 80 sec, $NB = 20$: 52 sec execution time; 6×6 nodes, $n = 6000, NB = 32$: 282 sec, $NB = 20$: 133 sec). For PSSYEV the difference is almost the same. But as the execution times are higher time increases only by 50 %. This is due to a performance problem of SGEMM from libsci with the first matrix not transposed and the second one transposed, which is called in the back transformation of the eigenvectors. Called with random matrices the time for SGEMM in the above situation is 4.6 sec for $n = 1000$, 54.8 sec for $n = 1024$ and 6.1 sec for $n = 1050$, hence it takes almost 9 times as long to multiply two 1024×1024 matrices, the second one transposed, than to multiply two 1050×1050 matrices. Therefore it is better not to use powers of two as block sizes even though sometimes the performance is better with those block sizes.

As mentioned in section 2, all libraries presented here distribute matrices to

a two-dimensional processor grid. ScaLAPACK and PLAPACK allow the user to explicitely choose the shape of this grid whereas Global Arrays only allows to determine the minimum number of rows or columns which must be in one block.

For the solution of the symmetric eigenvalue problem with ScaLAPACK the shape of the grid usually does not influence performance very much. If the number of nodes is a square, a square grid achieves highest performance. On rectangular grids, e.g. 8 nodes, sometimes a 2×4 grid and sometimes a 4×2 grid delivers slightly better performance results.

For the solution of linear systems via LU-decomposition grid shapes have more influence on performance than for the solution of the symmetric eigenvalue problem. The time for the solution of the triangular system with n right-hand-sides after LU-decomposition of the matrix with ScaLAPACK is very sensitive to grid shapes. E.g. we found out that in the case of 26 nodes for $n = 6000$, $NB = 32$, and $NBRHS = 64$ on a 2×13-grid the LU-decomposition time was about 19.6 sec and the solution time 139.2 sec whereas on a 13×2-grid the LU-decomposition time was about 18.8 sec and the solution time was only about 61.1 sec. This means that the total time (LU-decomposition + solution) was only half as high on a 13×2-grid than on a 2×13-grid.

4.3.4 Memory Requirements

As mentioned in section 4.3.2 problem sizes per node have to be large to get high performance. High additional memory requirements can therefore cause low performance because the problem size per node can't be made large enough.

Due to the necessity to hold at least a small part of the global data as a local copy when filling the global matrix, we were not able to solve as large linear systems with Global Arrays or PLAPACK as with ScaLAPACK.

On 64 nodes, the largest problem we could solve with the Global Arrays interface to ScaLAPACK was $n = 12288$, with PLAPACK $n = 12800$, and with ScaLAPACK $n = 19000$. Performance of ScaLAPACK was still increasing from less than 300 MFLOPS per node for $n = 12000$ to 350 MFLOPS per node for $n \geq 17000$.

For the symmetric eigenvalue problem we did not see a large difference in memory usage between ScaLAPACK and Global Arrays. This is because PSSYEVX needs additional space for reorthogonalization of eigenvectors as matrices larger than $n = 1000$ tend to have at least one very large cluster due to a nonscalable definition of clusters to remain consistent with LAPACK (see ScaLAPACK Users' Guide). PSSYEV needs additional space for the solution of the tridiagonal eigenvalue problem. For large problems the tridiagonal matrix must be stored on each node no matter how many nodes are used.

4.4 Performance Results

In the appendix we show some diagrams with results of performance measurements.

MFLOPS shown in the figures for matrix-multiplication and LU-decomposition were not taken from the MFLOP counts per node delivered by PAT but were based on the number of floating point operations necessary to solve the problem ($2n^3$ for multiplication of two $n \times n$ matrices and $\frac{8}{3}n^3 - \frac{1}{2}n^2$ for LU-decomposition and

solution of an $n \times n$ linear system with n right-hand-sides) divided by the number of nodes and divided by the time the slowest node needed to complete computation. The MFLOP counts shown by PAT are of course higher than the ones computed because there is always some parallelization overhead and PAT counts operations of the whole program including initialization and collection of results.

Figure 3 shows the results of matrix-matrix-multiplication routines. It can be seen that on 12 processors PLAPACK reaches stable performance of more than 500 MFLOPS/node. ScaLAPACK's PBLAS performance is in the same range and perhaps there could be some better block sizes for $n = 1800, 3000, 4200, 5400$ to get the same performance as with the other sizes. The original Global Arrays routine never reaches 500 MFLOPS/node, thus the ScaLAPACK interface is really necessary for performance.

From figures 4 and 5 it can be seen that for 12 nodes $n = 6000$ is large enough to get high MFLOPS rate of about 370 MFLOPS/node for the solution of linear systems with n right-hand-sides via LU-decomposition with PLAPACK and ScaLAPACK. On 64 nodes, however, any routine delivers poor performance of less than 250 MFLOPS/node with $n = 6000$. On 64 nodes the performance differences become higher. The PLAPACK routine already reaches 250 MFLOPS/node at problem sizes of less than $n = 7000$, whereas this was the highest performance we got with Global Arrays calling ScaLAPACK. ScaLAPACK performance is still increasing with problem sizes $n \geq 12000$ and reaches 350 MFLOS for $n \geq 17000$ as mentioned in section 4.3.4. For both matrix-multiplication and LU-decomposition with solution of the resulting triangular system PLAPACK delivers highest MFLOPS rates and therefore the shortest execution times.

Figures 6 and 7 show execution times for the computation of all eigenvalues and all eigenvectors of a real symmetric matrix of size $n = 2000, \ldots, 2500$. Figure 6 shows the times in the case where the eigenvalues are equally spread and reorthogonalization is not necessary. Figure 7 shows execution times for a matrix with one large cluster of $n - 267$ eigenvalues. In PSSYEVX and GA_DIAG_STD the eigenvectors belonging to this cluster are reorthogonalized. It can be seen that for equally spread eigenvalues PSSYEVX on 4 nodes is as fast as PSSYEV on 16 nodes. One reason, of course is, that PSSYEV needs twice as many operations as PSSYEVX. The other reason is, that the sequential QR-algorithm within PSSYEV uses a lot of BLAS 1 routines and consequently reaches a lower MFLOPS rate than PSSYEVX.

Although GA_DIAG_STD uses the same algorithm as PSSYEVX if there is no large cluster of eigenvalues the number of operations is much higher even for the node with the smallest number of operations (see Table 2). Due to the poor load balance of GA_DIAG_STD, see Table 2, it is even higher than that of PSSYEV on the node with the highest operation count. For large problems also MFLOPS rates per node reached with GA_DIAG_STD are significantly lower than the ones reached by PSSYEV or PSSYEVX, mainly because it is completely based on BLAS 1 routines and therefore performance is additionally reduced by cache misses.

If there is one large cluster of eigenvalues and the eigenvectors have to be reorthogonalized the situation changes dramatically. Now on four nodes the largest problem that could be solved with PSSYEVX was $n = 2000$. The execution times

for PSSYEVX are almost independent of the number of nodes.

It can be seen that the execution times for PSSYEV are slightly lower in the case of one large cluster of eigenvalues than in the case of equally spread eigenvalues. The execution times of GA_DIAG_STD on the other hand become higher as eigenvectors are reorthogonalized. Consequently the difference between PSSYEV and GA_DIAG_STD becomes larger than in the case without a cluster.

For the solution of the symmetric eigenvalue problem there is always one ScaLAPACK routine with highest performance: if eigenvalues are not clustered this is PSSYEVX, for one large cluster of eigenvalues whose eigenvectors have to be reorthogonalized PSSYEV is the fastest routine.

5 Conclusions

ScaLAPACK offers very good performance compared to the other libraries and a broad range of black box solvers but at the expense of a little more complicated user interface. Programmers willing to apply ScaLAPACK routines should become familiar with the data distribution used in ScaLAPACK and adapt their program to this distribution from the start. This will result in good performance and low memory usage.

PLAPACK achieves highest performance on those routines available in that library but these are only a few and the lack of a solver for the symmetric eigenvalue problem will prevent most people having to solve eigenvalue problems from using it. "Using PLAPACK"[3] explains how to use PLAPACK for writing linear algebra routines based on the PLAPACK distributed objects but we think it needs an experienced user to write an eigensolver based on PLAPACK.

Global Arrays mainly offers an infrastructure to treat global objects transparently. If routines from other libraries are to be used this costs some performance due to redistribution of data. There is only one interface routine to ScaLAPACK within Global Arrays, the one for LU decomposition and the solution of the resulting triangular system. This routine must be modified if other ScaLAPACK or PBLAS routines like matrix-matrix-multiplication shall be used. It seems to be much easier to write an interface to use PSSYEV(X) from Global Arrays than to write a new eigensolver for PLAPACK.

All libraries can help developers of new application programs or application packages to take advantage of work already done. They are not meant for those who only want to use software on the application level.

References

1. *HPC-Netlib Homepage*, `http://www.nhse.org/hpc-netlib/`
2. L.S. Blackford, J. Choi, A. Cleary *et al.*, *ScaLAPACK Users' Guide*, SIAM, Philadelphia (1997).
3. R.A. van de Geijn, *Using PLAPACK: Parallel Linear Algebra Package*, The MIT Press, Cambridge, Massachusetts (1997).
4. *Global Arrays User Guide*, `http://www.emsl.pnl.gov:2080/docs/global/`
5. E. Anderson, Z. Bai, C. Bischof *et al.*: *LAPACK Users' Guide, Second Edition*,

SIAM, Philadelphia (1995).
6. J.J. Dongarra and R.C. Whaley, *A User's Guide to the BLACS v1.1*, LAPACK Working Note 94, (1997), `http://www.netlib.org/lapack/lawns/lawn94.ps`
7. Silicon Graphics, CRAY Research: *Scientific Libraries Reference Manual, Vols. 1-2*, CRAY Research, Inc. (1997).
8. D. Elwood, G. Fann, R. Littlefield, *Parallel Eigensystem Solver PeIGS Version 2.1, rev. 0.0*, Pacific Northwest National Laboratory (July 28, 1995), `ftp://ftp.pnl.gov/pub/permanent/emsl/`
9. Th. Steinke, *Tools for parallel quantum chemistry software*, Modern Methods and Algorithms of Quantum Chemistry, Proceedings, Jülich (February 2000).
10. J. Galarowicz, B. Mohr, *Analyzing Message Passing Programs on the Cray T3E with PAT and VAMPIR*, Forschungszentrum Jülich GmbH, Zentralinstitut für Angewandte Mathematik, Interner Bericht, FZJ-ZAM-IB-9809, Mai 1998.
11. R. Berrendorf, H. Ziegler, *PCL The Performance Counter Library: A Common Interface to Access Hardware Performance Counters on Microprocessors (Version 1.2)*, Research Centre Jülich GmbH, 1999, `http://www.fz-juelich.de/zam/PCL/`
12. I. Gutheil and R. Zimmermann, *Performance of Software for the Full Symmetric Eigenproblem on CRAY T3E and T90 Systems*, Forschungszentrum Jülich, Zentralinstitut für Angewandte Mathematik, Interner Bericht, FZJ-ZAM-IB-2000-07, 2000.

Appendix

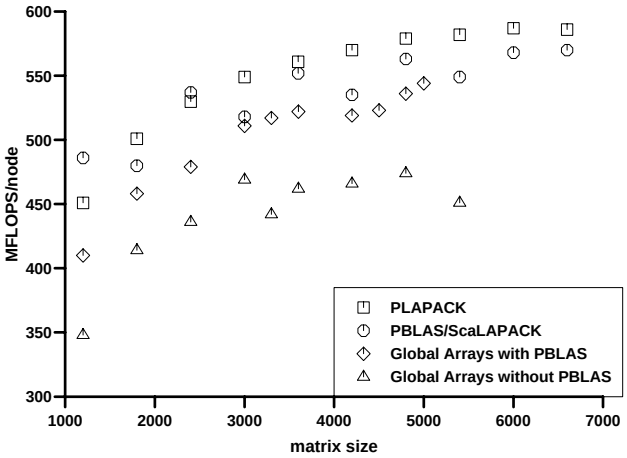


Figure 3. Matrix-matrix-multiplication $C = 2AB + 3C$, A, B, C square matrices, using different library routines on 12 nodes.

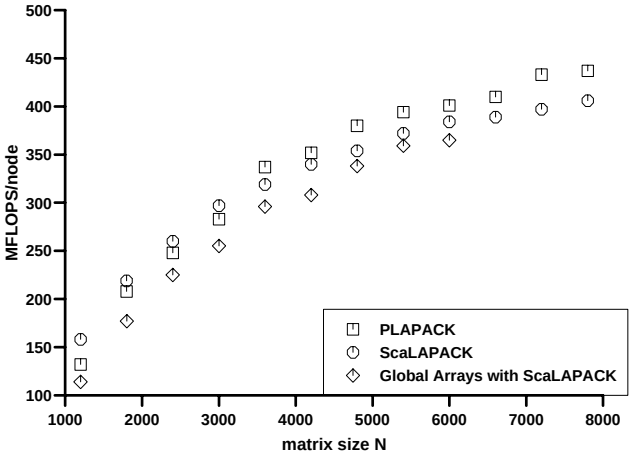


Figure 4. Solving a linear system with N right-hand-sides by means of LU-decomposition using different library routines on 12 nodes.

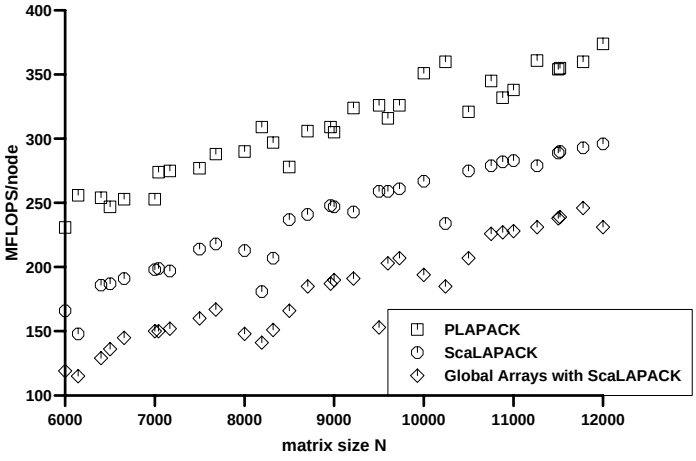


Figure 5. Solving a linear system with N right-hand-sides by means of LU-decomposition using different routines on 64 nodes.

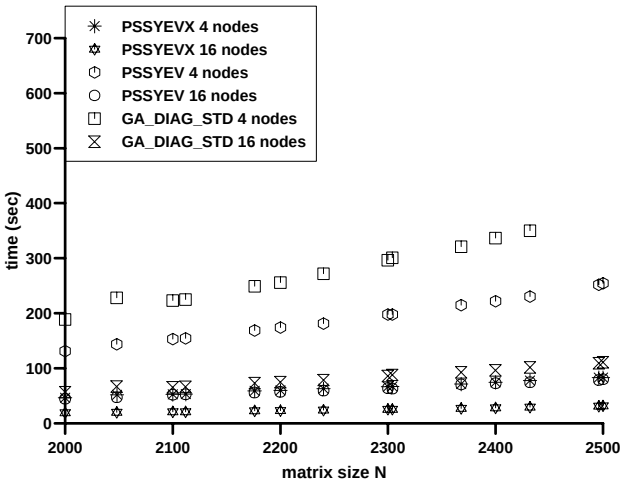


Figure 6. Computation of all eigenvalues and eigenvectors of an $N \times N$-matrix (no clusters of eigenvalues) using different library routines.

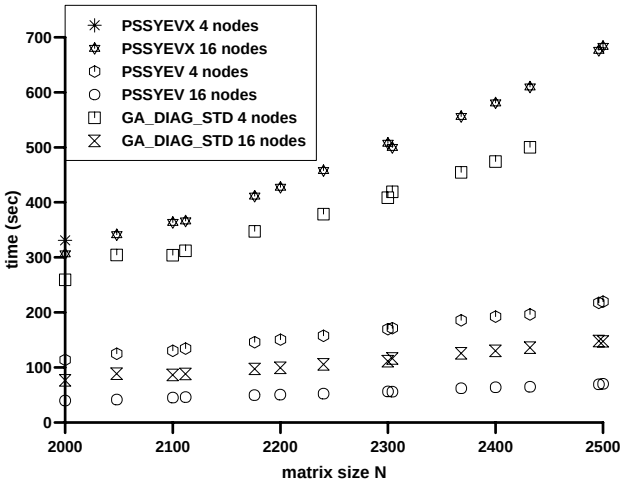


Figure 7. Computation of all eigenvalues and eigenvectors of an $N \times N$-matrix with one large cluster of eigenvalues using different library routines.

TOOLS FOR PARALLEL QUANTUM CHEMISTRY SOFTWARE

THOMAS STEINKE
Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB)
Takustr. 7
14195 Berlin-Dahlem
Germany
E-mail: steinke@zib.de

The lack of appropriate middleware for implementing parallel methods on true parallel computers like the Intel series in late 80s forced the design of suitable programming tools for application packages. The TCGMSG was one of the first examples which provides a portable and efficient message passing interface. This tool is still used in production codes in the field of quantum chemistry. Further developments led to programming interfaces (e.g. Global Array Toolkit) providing a shared memory programming model on distributed memory architectures. This article gives an introduction into the basic terminology and the application of these tools.

1 Introduction

First-principle quantum-chemical simulations of *large* molecular systems consisting of a few hundreds of atoms are going to be more and more feasible as common "daily" task as powerful parallel processor computers are becoming available. Massively-parallel processor platforms (e.g. CRAY T3E, IBM SP) as well as large scalable shared memory systems (e.g. SGI Origin) both providing some tens or hundreds of GigaFlop/s sustained performance are accessible for scientists at certain sites. Today, a scientist who is developing parallel applications for high-performance computer platforms can choose between some programming libraries, programming models and toolkits. But, the situation was quite different in the late 80s.

In order to take advantage of new hardware features provided by the first true parallel computers 15 years ago (e.g. Intel Hypercube, later Paragon etc.), the redesign of serial versions of existing quantum chemistry packages (e.g. COLUMBUS, GAMESS (US), GAMESS-UK, DGauss) was initiated as well as the new design and implementation of packages from scratch (e.g. NWChem, Q-Chem). Due to the lack of appropriate middleware for implementing software on such distributed memory architectures research groups in the chemistry community developed suitable communication libraries and user-friendly APIs for programming these parallel machines. For example, TCGMSG was developed to provide a portable communication library for quantum chemistry software. Although the functionality of TCGMSG is superceeded by standard messsage-passing libraries PVM and MPI this software is still in use in production codes.

This article gives an introduction into the fundamentals of some of the programming tools used in production codes for quantum chemical simulations. The part of informations provided are restricted to the basics of the topic. A more complete view of the capabilities of tools explained below will arise together with the overhead presentation given at the Winterschool at Forschungszentrum Jülich.

2 Basic Tasks in Typical Quantum Chemical Calculations

The major part of today's quantum chemical calculations carried out routinely is dealing with Hartree-Fock SCF, MP2, or DFT type calculations of more or less large molecules. To identify the computational steps which take advantage of parallelization we will give a coarse-grain analysis of such calculations. We will focus on common SCF type calculations for solving the stationary Schrödinger equation $\hat{H}\Psi = E\Psi$ within the common molecular orbital (MO) approach. The following summary lists the basic time consuming steps:

1. the computation of 2-electron integrals (ERI),
2. the construction of the Fock matrix **F** from ERIs and density matrix **P**,
3. the diagonalization of **F** delivering MO coeffients **C**, and
4. the calculation of a new density matrix **P** from **C**

In *large* molecular systems consisting of a few hundreds of atoms, the number of basis functions included is in the order of thousands. In the *direct* SCF scheme the steps 1 and 2 are combined with sophisticated integral screening techniques to reduce the number of ERI contributions to be considered. In this way, any storing of ERIs either on disk (conventional approach) or in memory (in-core scheme) is avoided by (re-)computing ERIs for the Fock matrix build as they needed.

From the picture given above one can imagine that

- the *work* distribution to calculate ERIs and to construct the Fock matrix **F** which itself should be distributed over all compute nodes,
- the diagonalization of **F**, or quasi-Newton orbital rotations, and finally
- subsequent operations on the wavefunction (**C** and/or **P**) like matrix operations

should be the subject of parallelization efforts. In the next subsection some basic issues related to parallel programming are discussed.

2.1 Issues related to parallel programming

Given the list of basic tasks above one can define essential features of a suitable toolkit required for quantum-chemical calculations. Below some key issues regarding support for parallelism, not only in the context of quantum chemistry, are explained:

Work sharing, task parallelism: That is the basic idea of parallelism. It is wanted to use as many compute nodes as possible without substantial overhead in order to reduce the wall-time for a given problem size. A given set of tasks, e.g. the calculation of ERIs or elements of **F**, should be distributed over all computing nodes.

Work sharing within an application may be at a high level (*coarse-grain parallelism*) or low level (*fine-grain parallelism*). The suitable approach depends

on the given hardware and software environment, e.g. latency and bandwith of the communication network or availability of thin low-level APIs.

Dynamic load balancing: This aspect is closely connected to the previous topic. Under certain conditions it may be impossible or unpredictable to split a problem into a set of evenly sized tasks. A suitable mechanism for *dynamic* load balancing is required to keep *all* compute resources as busy as possible. The load balancing problem becomes even more prominent if the (virtual) parallel computer system consists of compute nodes with different CPU performances (e.g. heterogeneous resources in network of workstations).

Replicated data model: Storing all data in a replicated fashion in memory on all nodes is the first approach to implement task parallelism. Its advantage is that in general less communication is required, but this programming approach shows problem size limitations.

Distributed data model: To offer solutions for *large* problems where per node memory resources are not sufficient, *data distribution* is the only way to tackle such *grand challenge* problems. It is desirable that the total memory of a parallel computer system can be allocated to keep large arrays. Then, the largest problem which can be handled scales with the number of compute nodes involved in the calculation. Large arrays like **F** and/or **P** are limiting actual realizations of simulations in quantum chemistry. These arrays need to be distributed.
A user-friendly support of the distributed data model is *not* provided in common message-passing libaries (MPI, PVM), and thus, it is the objective of software projects discussed in this tutorial.

Interface to numerical libraries: A high-level interface to common numerical libraries (e.g. ScaLAPACK) should be available. Desirable are routines for matrix operations of distributed arrays (matrix multiply) as well as an interface to parallel eigensolvers[a] .

One-sided communication: In a programming model supporting distributed data, a "lightweighted" access to remote data is important. Usual point-to-point connections implemented in message-passing environments require the interaction of CPU resources on both sides. Tools described in this tutorial provide methods to access remote data without interferring with application process running on a remote CPU.

A common classification scheme for communication operations considers to what extent compute nodes are involved:

Collective operations: These are operations where *all* computing nodes of a given set are involved. Typically, they distribute work across the nodes. Examples are global summation, broadcast, and synchronization operations.

[a] for details see contribution given by I. Gutheil[1] in this volume

Non-collective operations: These type of communication operations fall into synchronous and one-sided (or asynchronous) operations:

Synchronous operations: In common message passing environments (e.g. MPI) any point-to-point data transfer consists of a complementary pair of *send/receive* invocations. On both sides CPU resources are allocated to perform the data transfer. In this type of communication an implicit pairwise synchronization is involved.

One-sided operations: Within these operations only one node is active. Communication or data transfer is performed without interferring with the other node which keeps remote data in memory. Examples are one-sided read and write operations from/into remote memory or atomic update operations on remote locations.

Atomic operations: To update the contents of variables in a save way it is necessary to prevent concurrent read or write operations on the same memory locations. This is achieved by locking the critical code region, i.e. only *one* request is permitted to perform read/write actions. More complex atomic operations like *read-and-increment* or *accumulate* includes a locking mechanism in a transparent way.

3 Parallel Tools in Today's Production Codes

There are several program packages available to perform quantum chemical calculations of large molecules on massively-parallel platforms. Table 1 presents some important examples of programming tools developed in this context. For the sake of completeness, commonly used tools on shared-memory platforms are included[b] too.

The Global Array Toolkit[4,5] (GA) is the *de-facto* standard in the quantum chemistry community. The suite of tools in GA is primarily designed as middleware for quantum chemical program packages. Major application examples are those which are developed and maintained at PNNL like NWChem[9] and COLUMBUS[10] as well as GAMESS-UK[11], and applications coming from other fields. Beside providing a portable interface to global shared arrays the GA includes tools for dynamic memory allocation (MA), one-sided communication (AMRCI), and efficient I/O capablities for parallel I/O (e.g. ChemIO).

The Distributed Data Interface[13] (DDI) has been written for the GAMESS[12] (US) package, and provides a one-sided data access via communication with data-servers on top of MPI or TCP/IP sockets. Within the DFT code DGauss[15] a proprietary Distributed Matrix[16] library is used. A quite different programming approach is used in Gaussian[17] and MPQC[18]. In Gaussian, the parallelization is implemented using the LINDA programming model. The MPQC package follows an object-oriented design and parallelization is implemented by using the Scientific Computing Toolkit[19] C++ class library.

[b] for details regarding OpenMP see contribution given by M. Gerndt[2] in this volume

Table 1. Tools used by various quantum chemistry packages on multi-processor systems

Memory Architecture	Tool (Communication Lib)	Packages (examples)
distributed memory	Global Array Toolkit (TCGMSG, MPI, SHMEM, sockets)	NWChem, GAMESS-UK, COLUMBUS
	Distributed Data Interface (MPI, sockets)	GAMESS (US)
	Distributed Matrix Lib (SHMEM, PVM, MPI)	DGauss
	LINDA (MPI, SHMEM)	Gaussian
	Scientific Computing Toolkit (PVM, MPI, NX, IPC)	MPQC
shared memory	OpenMP Microtasking	MOLPRO, Gaussian DGauss

4 The TCGMSG Library

In the late 80s and early 90s portable message passing libraries like PVM or MPI did not exists yet. To provide a compact, user-friendly, and portable programmer's interface for message passing R. J. Harrison implemented the TCGMSG[3] (Theoretical chemistry group message-passing toolkit) toolkit. The programming model and interface is directly modelled after PARMACS[22] developed at Argonne National Lab.

In the UNIX environment communication is done via TCP sockets. If identical processes are running on shared memory machine the faster communication mechanism provided by the hardware is used. Thus, applications can be built to run on an entire network of machines with local communication running at memory bandwidth and remote communication running at the corresponding network speed. On true message-passing machines TCGMSG is just a thin layer over the system interface (e.g. SHMEM on CRAY T3E).

Asynchronous communication is available in TCGMSG on machines supporting it. Otherwise, send operations block until the message is explicitly received, and messages from a particular process can only be received in the order sent. As far as buffering provided by the transport layer permits, messages are actually sent without any synchronization between sender and receiver.

TCGMSG supports FORTRAN and C bindings, Table 2 gives an overview of the TCGMSG API.

One remarkable feature is the `integer function NXTVAL(MPROC)` which implements a shared counter by communicating with a dedicated server process. It returns the next counter associated with a single active loop (0,1,2,...). `MPROC` is the number of processes actively requesting values. This `NXTVAL` function allows the implementation of dynamic load balancing within loops (work parallelism).

Table 2. Basic functions of TCGMSG

Operation	FORTRAN	C
Initialization	call PBEGINF()	PBEGIN_(...)
Termination	call PEND()	PEND_()
Identification		
Number of nodes	NNODES()	NNODES_()
Node ID	NODEID()	NODEID_()
Communication Operations		
Send	call SND(...)	SND_(...)
Receive	call RCV(...)	RCV_(...)
Broadcast	call BRDCST(...)	BRDCST_(...)
Synchronization	call SYNCH(...)	SYNCH_(...)
Shared counter	NXTVAL(...)	NXTVAL_(...)
Global operation	call DGOP(...)	DGOP_(...)
Utilities		
Print statistics	call STATS()	STATS_()
Wall time	TCGTIME()	TCGTIME_()

A FORTRAN example may look like the following code snippet:

```
   next = NXTVAL(MPROC)

   do i = 0, big
      if (i .eq. next) then

!     ... do work for iteration i

         next = NXTVAL(MPROC)
      endif
   end do
```

A complete code example showing the usage of the `NXTVAL` function can be found in Appendix A.

5 The Global Array Toolkit

5.1 Overview

The Global Array Toolkit[4] (GA) is written and maintained at the William R. Wiley Environmental Molecular Sciences Laboratory (EMSL) at Pacific Northwest National Laboratory (PNNL). The principal development work is done by J. Nieplocha, the latest version is release 3[5,6].

The GA provides a portable shared memory style programming environment which consists of a certain *programming model* as well as a *logical view* of a virtual machine (machine model)[20]. The virtual machine might be a massively-parallel distributed memory or a scalable shared memory platform. It is viewed as an ensemble of nodes consisting of CPU(s) and local memory (Figure 1). The whole

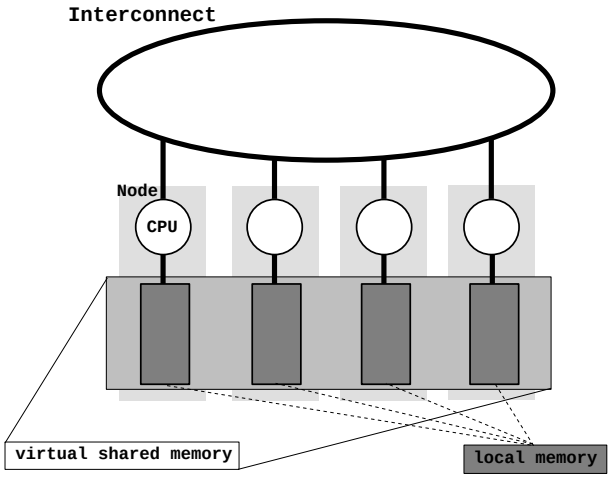


Figure 1. The Global Array machine model

memory is accessible via global arrays. These logically shared data are divided into local and remote portions, and the GA recognizes variable data transfer costs required to access the data (fast access to local data, slower access to remote data). In acknowledging the difference of local and remote data access, the GA model exposes to the programmer the *Non-Uniform Memory Access* (NUMA) architecture of computer systems found today.

5.2 Basic Features

The GA programming environment supports the development of parallel programs in C or/and FORTRAN. Main application areas are codes that operate on distributed dense multidimensional arrays and/or require shared memory programming style or one-sided communication. GA encapsulates many details of array distribution and addressing so that the programming effort is substantially reduced.

The basic shared memory operations supported include the one-sided communication operations *get*, *put*, *scatter*, and *gather*. The atomic operations *accumulate* and *read-and-increment* complete the list of fundamental functions. These operations are truely *one-sided* and will complete regardless of actions taken by a remote process that owns the data.

Details of the data distribution, addressing, and communication are encapsulated in the GA objects (internal data structures). The information on actual distribution patterns and locality of data can be obtained by library routines. Advantage of this information is taken whenever data locality is important.

At the user level, transfer operations of arrays or array patches are initiated using an array index interface rather than addresses. GA does not require the user to specify process IDs for accessing remote data. Any global array index-to-address translation and estimation of target IDs is done internally.

The GA is not built on top of any particular message-passing library but it

requires one to initialize the parallel environment and create processes. The one-sided communication required to support a shared memory programming model is facilitated by the ARMCI library included in the GA package[c] .

5.3 Using the GA Toolkit

Supported Plattforms

The current GA release supports *homogeneous* hardware platforms as follows:

- massively-parallel processor (MPP) systems (CRAY T3E, IBM SP)
- scalable shared memory systems (CRAY PVP, SGI Origin, Fujitsu VX/VPP)
- cluster of workstations (all UNIX flavours including Linux)
- SMP server (UNIX)
- multi-processor NT server

Selection of message-passing library

GA is working on top of a message-passing library which is selected at compile time on a given platform. For example, the message-passing library is used on some platforms to fork processes, implement broadcast or global operations (`ga_dgop`). Currently, either TCGMSG (s. section 4) or MPI can be used as an interface. Since TCGMSG is only a small set of routines and provides a convenient interface it is used per default on UNIX systems. On NT server the GA is build on top of WMPI, a NT implementation derived from MPICH.

Dependency on other software

Besides the message-passing interface there are additional software tools which are required by GA:

- MA (Memory Allocator) by Greg Thomas, a library for portable memory management;
- ARMCI, a one-sided communication library used by GA as its run-time system;
- BLAS library is required for the eigensolver and matrix multiply (`ga_dgemm`);
- GA eigensolver, `ga_diag`, is a wrapper for the eigensolver from PeIGS[8,d];
- LAPACK library is required for the eigensolver;
- MPI, SCALAPACK, PBBLAS, and BLACS libraries are required for some linear algebra functions (`ga_lu_solve`, `ga_cholesky`, `ga_llt_solve`, `ga_spd_invert`, `ga_solve`);

[c] In earlier versions of GA, the one-sided communication was implemented directly inside GA.
[d] application of PeIGS and performance issues are discussed by I. Gutheil[1]

5.4 Getting started with GA

Table 3 gives an overview about the most important functions implemented in GA, Table 4 summarizes some of the utility operations.

Table 3. Basic functionality provided by GA (only 2D array operations on whole arrays are listed)

Operation	FORTRAN	C
Initialization	ga_initialize()	GA_Initialize()
Termination	ga_terminate()	GA_Terminate()
Creation of arrays		
Regular dist.	ga_create(...)	NGA_Create(...)
Irregular dist.	ga_create_irreg(...)	NGA_Create_irreg(...)
Duplication	ga_duplicate(...)	GA_Duplicate(...)
Destroying	ga_destroy(...)	GA_Destroy(...)
One-sided operations		
Put	ga_put(...)	NGA_Put(...)
Get	ga_get(...)	NGA_Get(...)
Atomic accumulate	ga_acc(...)	NGA_Acc(...)
Atomic read and increment	ga_read_inc(...)	NGA_Read_inc(...)
Scatter	ga_scatter(...)	NGA_Scatter(...)
Gather	ga_gather(...)	NGA_Gather(...)
Interprocess synchronization		
Lock	ga_lock(...)	GA_lock(...)
Unlock	ga_unlock(...)	GA_unlock(...)
Fence	ga_init_fence()	GA_Init_fence()
	ga_fence()	GA_Fence()
Barrier	ga_sync()	GA_Sync()
Collective array operations		
Basic array operations		
Init zero	ga_zero(...)	GA_Zero(...)
Fill	ga_fill(...)	GA_Fill(...)
Scale	ga_scale(...)	GA_Scale(...)
Copy	ga_copy(...)	GA_Copy(...)
Linear algebra		
Matrix add	ga_add(...)	GA_Add(...)
Matrix multiply	ga_dgemm(...)	GA_Dgemm(...)
Dot product	ga_ddot(...)	GA_Ddot(...)
Symmetrization	ga_symmetrize(...)	GA_Symmetrize(...)
Transposition	ga_transpose(...)	GA_Transpose(...)

Table 4. Utility operations in GA

Operation	FORTRAN	C
Locality of data		
Locate	ga_locate(...)	NGA_Locate(...)
Find distrib.	ga_distribute(...)	NGA_Distribute(...)
Accessing	ga_acess(...)	NGA_Access(...)
Process information		
Process ID	ga_nodeid()	GA_Nodeid()
Nodes	ga_nnodes()	GA_Nnodes()
Memory availability		
Available memory	ga_memory_avail()	GA_Memory_avail()
Used memory	ga_inquire_memory()	GA_Inquire_memory()
Wrappers to reduction/broadcast operations		
Broadcast	ga_brdcst(...)	GA_brdcst(...)
Global operation	ga_dgop(...)	GA_Dgop(...)
	ga_igop(...)	GA_Igop(...)
Print detailed informations		
Print array	ga_print(...)	GA_Print(...)
Statistics	ga_print_stats()	GA_Print_stats()

The principal structure of a GA program should look like:

- when GA runs on top of MPI

```
   FORTRAN                      C

call mpi_init(...)          MPI_Init(...)       ! start MPI
call ga_initialize()        GA_Initialize()     ! start GA
status = ma_init(...)       MA_Init(...)        ! start MA

 .... do some work          ... do some work

call ga_terminate()         GA_Terminate()      ! tidy up GA
call mpi_finalize()         MPI_Finalize()      ! tidy up MPI
stop                        exit()              ! exit
```

- when GA runs on top of TCGMSG

```
   FORTRAN                      C

call pbeginf()              PBEGIN_(...)        ! start TCGMSG
call ga_initialize()        GA_Initialize()     ! start GA
status = ma_init(...)       MA_Init(...)        ! start MA

 .... do some work          ... do some work

call ga_terminate()         GA_Terminate()      ! tidy up GA
call pend()                 PEND_()             ! tidy up TCGMSG
stop                        exit()              ! exit
```

Examples: The Parallel Process Environment

The following two examples will demonstrate the first basic steps to employ the GA toolkit. In the first example the parallel process environment is provided and each node prints its unique node ID.

```
program example1
implicit none
include 'global.fh'

integer         :: n_nodes, me
logical         :: stat

call pbeginf()                                          ! init of TCGMSG
call ga_initialize()                                    ! init GA
n_nodes = ga_nnodes()                                   ! get number of nodes
me      = ga_nodeid()                                   ! who am i
print *, 'Hi, Iam node ', me, ' of ', n_nodes, 'nodes'

call ga_terminate()                                     ! tidy up GA
call pend()                                             ! tidy up TCGMSG
stop
end program example1
```

If the program is running on 4 nodes it generates the following output:

```
Hi, Iam node  3  of  4 nodes
Hi, Iam node  0  of  4 nodes
Hi, Iam node  1  of  4 nodes
Hi, Iam node  2  of  4 nodes
```

Beginning with release 3, the node numbering in GA and TCGMSG is coherent, e.g. a code snippet like the following reports identical node IDs in a TCGMSG and GA environment.

```
tcg_n_nodes = NNODES ()          ! TCGMSG nodes cntr
tcg_me      = NODEID ()          ! TCGMSG nodeid
ga_n_nodes  = ga_nnodes()        ! GA nodes cntr
ga_me       = ga_nodeid()        ! GA nodeid

print *, 'TCGMSG: #nodes:', tcg_n_nodes, ' on node', tcg_me
print *, '    GA: #nodes:', ga_n_nodes , ' on node', ga_me
```

Example: Creating a Regular Global 2D Array

The next example introduces the creation of a 2D array. The syntax of the corresponding GA function `ga_create` is as follows:

```
logical function ga_create(type, dim1, dim2, name, chunk1, chunk2, g_a)
        integer       type      - MA type                              [input]
        integer       dim1/2    - array(dim1,dim2) as in FORTRAN       [input]
        character*(*) name      - a unique character string            [input]
        integer       chunk1/2  - minimum size that dimensions should
                                  be chunked up into                   [input]
        integer       g_a       - handle for future references         [output]
```

Setting `chunk1 = dim1` gives distribution by vertical strips (`chunk2*columns`); setting `chunk2 = dim2` gives distribution by horizontal strips (`chunk1*rows`). Actual chunks will be modified so that they are at least the size of the minimum and each process has either zero or one chunk. Specifying `chunk1/2` as $\langle$ 1 will cause that dimension to be distributed evenly. `ga_create` is a *collective* operation.

The following code snippet demonstrates the creation of a regular 2D array. Assuming that the code is started on `4` nodes the distribution of the squared `100x100` array named `A` is done by horizontal strips. Figure 2 shows various distribution patterns depending on the chunk sizes in the `ga_create` call. (In the example below the data type `MT_DBL` means double-precision).

```
status = ma_init(...)                                    ! allocate memory
call ga_initialize()                                     ! init GA
status = ga_create ( MT_DBL, 100, 100, 'A', 25, 100, ga_hndl)
call ga_destroy ( ga_hndl )
call ga_terminate()
```

Complete code examples demonstrating the creation of a global regular and irregular 2D arrays, the usage of matrix multiply interface and some utility functions are listed in Appendices B-F.

5.5 About Performance: Matrix Multiply

To give some impression about the performance we present some numbers for matrix multiply operations obtained on our CRAY T3E-1200. Three matrix multiply routines are compared: the optimized BLAS routine `SGEMM` (64 bit, running on a single node), the parallel `GA_DGEMM` provided by GA (code example Appendix E), and a matrix multiply version implemented in High Performance Fortran (HPF).

Fig. 3 shows the single node performance for a $N = 1000$ matrix multiply. The performance degradation for `GA_DGEMM` is small, and is mainly due to the array index-to-address conversions. More in detail: using the native `SGEMM` routine one obtains 592 MFlop/s whereas `GA_GEMM` achieves 563 MFlop/s. Please note the poor performance of 6 MFlop/s for the HPF code. The dependence of the global performance on the number of nodes for a fixed problem size is showing in Fig. 4. For the given problem size $N = 1000$, the efficiency of `GA_DGEMM` ranges from 0.92 to 0.57 for two up to eight nodes, respectively. If one measures the performance with respect to the problem size on eight nodes (s. Fig. 5) one can show that the dimension of matrices should be at least in the order of some hundreds to obtain a reasonable speed on CRAY T3E.

5.6 Further components of the Global Array toolkit

The *Memory Allocator* (MA) is a library of routines that comprises a dynamic memory allocator for use by C, FORTRAN, or mixed-language applications. MA is designed to be portable across a variety of platforms. C applications can benefit from using MA instead of the ordinary `malloc()` and `free()` routines because of the extra features MA provides: both heap and stack memory management disciplines, debugging and verification support, usage statistics, and quantitative

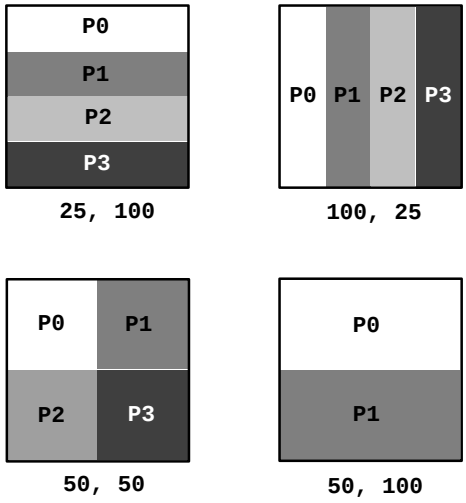


Figure 2. Creation of global 2D arrays: distribution patterns for various chunk sizes i, j

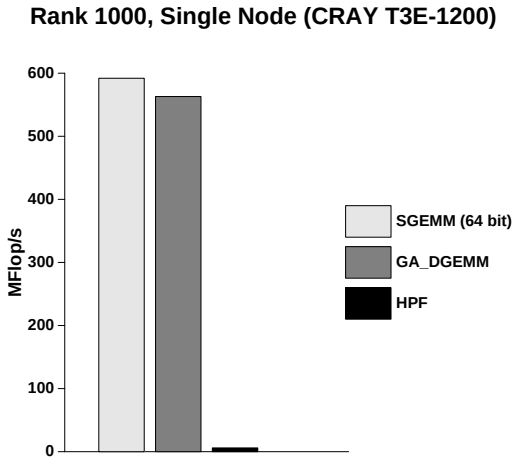


Figure 3. Matrix Multiply: Single Node Performance (CRAY T3E-1200)

memory availability information. FORTRAN applications can take advantage of the same features, and predecessors of FORTRAN 9x codes may in fact require a library such as MA because dynamic memory allocation is not supported. One important advantage of MA should be noted: memory leaks are avoided due to the management strategies of MA.

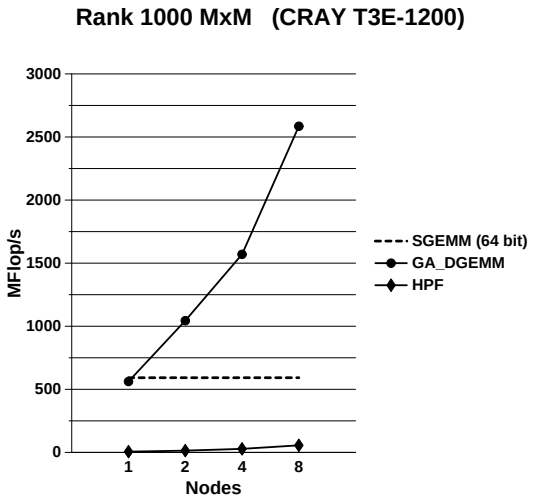


Figure 4. Matrix Multiply: Performance Scaling vs. Number of Nodes (CRAY T3E-1200)

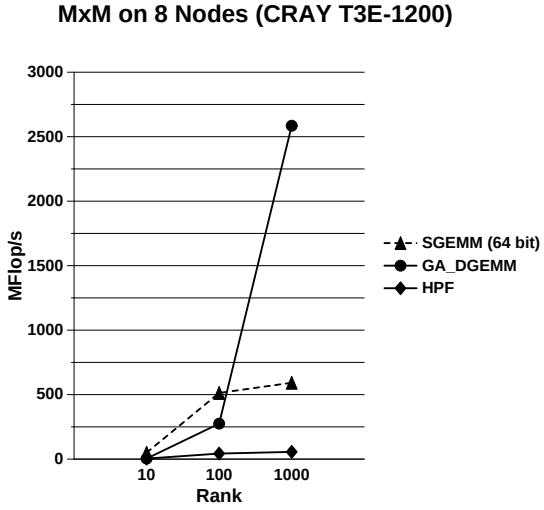


Figure 5. Matrix Multiply: Performance Scaling vs. Problem Size (CRAY T3E-1200)

A suite of independent parallel I/O libraries for high-performance computers building a chemistry I/O API, ChemIO[23], has been developed:

- Disk Resident Arrays (DRA),
- Shared Files (SF), and
- Exclusive Access Files (EAF).

All the components have an asynchronous API allowing the applications to overlap expensive I/O with computations. As an example, the DRA layer is employed in the RI-MP2 module in NWChem, thus some aspects of the DRA are discussed in the following.

Disk Resident Arrays (DRA) extend the Global Arrays (GA) programming model to disk. The library encapsulates the details of data layout, addressing and I/O transfer in disk arrays objects. Disk resident arrays resemble global arrays except that they reside on disk instead of in main memory. The main features of this model are:

- Data can be transferred between disk and global memory.
- I/O operations have a nonblocking interface to allow overlapping of I/O with computations.
- All I/O operations are collective.
- Either whole or sections of global arrays can be transferred between GA memory and the disk.

5.7 Guidelines for using GA

Some guidelines regarding a suitable application scenario for the GA toolkit may be summarized as follows[6]. Utilization of GA is preferred:

- for applications with dynamic and irregular communication patterns,
- for calculations gaining advantage from dynamic load balancing,
- if one-sided access to shared data is required,
- when coding in message-passing style is too complicated,
- if data locality is important.

Alternatives to GA – MPI, HPF, or OpenMP – might be considered,

- if there are regular communication patterns or only a few communication paths (e.g. nearest neighbour) often found in domain decomposition approaches,
- when synchronization after point-to-point message passing is needed, or
- if compiler parallelization is more effective e.g. OpenMP on shared memory platforms.

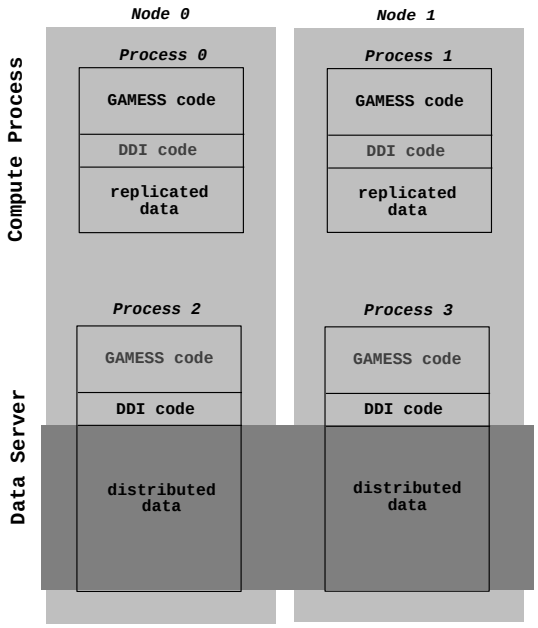


Figure 6. The Data Server model in DDI for two nodes (less active program components are shown in gray)

6 The Distributed Data Interface Used in GAMESS (US)

The Distributed Data Interface (DDI) was introduced in the public GAMESS[12] (US) in June 1999, and replaces the TCGMSG API used since 1991. The low level transport layers which DDI relies on are SHMEM, MPI-1, or TCP/IP sockets.

The DDI is similar to the GA concept in the sense that DDI attempts to exploit the entire machines memory as global shared memory. The implementation follows the concept of providing one-sided communication calls. There are three subroutine calls to access memory on remote nodes: `DDI_PUT`, `DDI_GET`, and `DDI_ACCUMULATE`.

At the present time, the DDI routines support only two dimensional FORTRAN arrays, organized in such a way that columns are kept on a single node's memory. Up to 10 matrices may be distributed in this fashion.

6.1 The Data Server concept

Since MPI-2 is still unavailable on most platforms, and the SHMEM[24] [e] is or will be available on a limited number of computer systems only, a data server model is implemented in DDI on other platforms (Figure 6).

On each node two processes are running: one is doing the chemistry work (compute process) and the other one acts as data server supporting DDI *get*, *put*, and

[e]The logically shared, distributed memory access (SHMEM) routines provide low-latency, high-bandwidth communication.

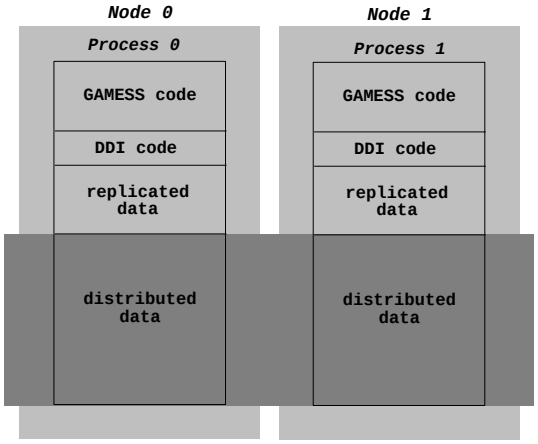


Figure 7. The DDI implementation on CRAY T3E for two nodes

Table 5. Some useful DDI Functions

Operation	Functions
Global Operations	
Initialization	DDI_PBEG(...)
Termination	DDI_PEND(...)
Number of nodes, ID	DDI_NPROC(...)
Allocate shared memory	DDI_MEMORY(...)
Create distrib. matrix	DDI_CREATE(...)
Destroy matrix	DDI_DESTROY(...)
Synchronization	DDI_SYNC(...)
Global sum (fp)	DDI_GSUMF(...)
Broadcast	DDI_BCAST(...)
Point-to-Point Operations	
Synchronous send	DDI_SEND(...)
Synchronous receive	DDI_RECV(...)
Get next index	DDI_DLBNEXT(...)
Get block of matrix	DDI_GET(...)
Put block of matrix	DDI_PUT(...)
Accumulate data into block	DDI_ACC(...)

accumulate requests. Interupts are handled by the operating system, as the data servers are distinct processes.

On CRAY MPP systems an efficient, hardware supported API for (fast) one-sided communication is provided (SHMEM). Therefore, the data server model can be dropped which allows for a more compact implementation. The memory image of a GAMESS process on T3E is shown in Figure 7.

DDI provides an interface for ordinary message passing parallel programming. A summary of important operations can be found in Table 5. There are functions to initialize and terminate the various processes, point-to-point send and receive, and collective operations like global sum and the broadcast. In addition, functions for distributed data manipulation, which include creation and destruction of the arrays, the put, get, and accumulation operations mentioned above, are available.

Note, that the current version of DDI is not intended to be a general parallel programming library, e.g. there are no portable memory management routines included. Thus, parts of the GAMESS distribution are still required.

6.2 DDI Example

To illustrate the DDI a very simple example[14] is given below. It shows the proper initialization and closure of the DDI library, requests replicated memory but not distributed memory.

```
  program bcast
  implicit double precision(a-h,o-z)
  parameter (maxmsg=500000)
  common /fmcom / xx(1)
  data exetyp/8hRUN     /

  nwdvar=2                              ! open DDI, tell it integer  ...
  call ddi_pbeg(nwdvar)                 ! ... word length is 32 bit
  memrep=maxmsg                         ! request allocation of only
  memddi=0                              ! replicated memory
  call ddi_memory(memrep,memddi,exetyp)
  call setfm(memrep)
  call ddi_nproc(nproc,me)              ! who am I
  master=0
  if(me.eq.master) print *, 'running ', nproc, ' processes'
  call valfm(loadfm)                    ! allocate a replicated array
  lbuff = loadfm + 1
  last  = lbuff  + maxmsg
  need = last - loadfm - 1
  call getfm(need)

  if (me.eq.master) then
     do i=1,maxmsg                      ! fill it up with ones
        xx(lbuff-1+i) = 1.0d+00
     end do
  end if

!                                 send it to all the other compute processes
  call ddi_bcast(102,'F',xx(lbuff),maxmsg,master)

  call retfm(need)                      ! replicated storage done
  istat=0
  call ddi_pend(istat)                  ! close the DDI library gracefully
  stop
  end
```

7 Further Reading

Students which are not necessarily familiar with parallel computational chemistry are encouraged to study the presentation *Parallel Computing in Chemistry* by Roland Lindh[21]. Access to further documentation regarding the Global Array Tookit can be found on the Web[5]. The *Global Array User's Manual* [6] explains most of the details required to implement parallel software using the GA. An excellent introduction to DDI as well as to the global shared memory programming model is given in the GAMESS (US) documentation[14]. Some of the ideas for the Global Array presentation were "borrowed" from the on-line slide-show *Global Arrays* by Krister Dackland[20].

8 Summary

Driven by practical requirements in the field of quantum chemistry, stable and robust tools for parallel programming were and are developed. The TCGMSG library was (and is) *the* "quantum chemist's" message-passing interface. Besides other tools and libraries, the Global Array Toolkit (GA) can be viewed as the *de-facto* standard for parallel applications in the quantum chemistry community. Using the GA one benefits from it since the Global Array Toolkit

- provides a shared memory programming style ...
- ... on distributed memory *and* true shared memory architectures,
- supports an efficient NUMA,
- is user-friendly, and
- is open-source.

Due to performance reasons, HPF based applications play only a minor role in the quantum-chemistry arena.

References

1. I. Gutheil, *Basic numerical libraries for parallel systems*, Modern Methods and Algorithms of Quantum Chemistry, Winterschool at Forschungszentrum Jülich, 21-25 February 2000 (this volume).
2. M. Gerndt, *Parallel programming models, tools and performance analysis*, Modern Methods and Algorithms of Quantum Chemistry, Winterschool at Forschungszentrum Jülich, 21-25 February 2000 (this volume).
3. R.J. Harrison, The TCGMSG Message-Passing Toolkit, Pacific Northwest National Laboratory, version 4.04, 1994.
   `http://www.emsl.pnl.gov:2080/docs/nwchem/tcgmsg.html`
4. J. Nieplocha, R.J. Harrison, R.J. Littlefield, Global Arrays: A portable "shared-memory" programming model for distributed memory computers, Proc. Supercomputing'94 , 340-349 (1994).

J. Nieplocha, R.J. Harrison, R.J. Littlefield, The Global Array programming model for high performance scientific computing, SIAM-News **September**, (1995).
J. Nieplocha, R.J. Harrison, R.J. Littlefield, Global Arrays: A nonuniform memory access programming model for high-performance computers, The Journal of Supercomputing **10**, 197-220 (1996).

5. J. Nieplocha, j_nieplocha@pnl.gov, Environmental Molecular Sciences Laboratory, Pacific Northwest National Laboratory.
   http://www.emsl.pnl.gov:2080/docs/global/ga.html
6. J. Nieplocha, J. Ju, Global Arrays User's Manual, Pacific Northwest National Laboratory, 1999.
7. further information: http://www.emsl.pnl.gov:2080/docs/parsoft/armci
8. please contact George Fann gi_fann@pnl.gov about PeIGS.
9. Northwest Computational Chemistry Package (NWChem) 3.3, Anchell, J.; Apra, E.; Bernholdt, D.; Borowski, P.; Clark, T.; Clerc, D.; Dachsel, H.; Deegan, M.; Dupuis, M.; Dyall, K.; Fann, G.; Früchtl, H.; Gutowski, M.; Harrison, R.; Hess, A.; Jaffe, J.; Kendall, R.; Kobayashi, R.; Kutteh, R.; Lin, Z.; Littlefield, R.; Long, X.; Meng, B.; Nichols, J.; Nieplocha, J.; Rendall, A.; Stave, M.; Straatsma, T.; Taylor, H.; Thomas, G.; Wolinski, K.; Wong, A.; "NWChem, A Computational Chemistry Package for Parallel Computers, Version 3.2.1" (1998), Pacific Northwest National Laboratory, Richland, Washington 99352-0999, USA.
   D.E. Bernholdt, E. Apra, H.A. Früchtl, M.F. Guest, R.J. Harrison, R.A. Kendall, R.A. Kutteh, X. Long, J.B. Nicholas, J.A. Nichols, H.L. Taylor, A.T. Wong, G.I. Fann, R.J. Littlefield and J. Nieplocha, Parallel Computational Chemistry Made Easier: The Development of NWChem, Int. J. Quantum Chem. Symposium **29**, 475-483 (1995).
   http://www.emsl.pnl.gov:2080/docs/nwchem/
10. COLUMBUS, H. Lischka, R. Shepard, I. Shavitt, F.B. Brown, R.M. Pitzer, R. Ahlrichs, H.-J. Böhm, C. Ehrhardt, R. Gdanitz, P. Scharf, H. Schiffer, M. Schindler, D.C. Comeau, M. Pepper, K. Kim, P.G. Szalay, J.-G. Zhao, E. Stahlberg, G. Kedziora, G. Gawboy, H. Dachsel, S. Irle, M. Dallos, Th. Müller, T. Kovar, M. Ernzerhof, A.H.H. Chang, V. Parasuk, M. Schüler, P. Höchtl.
    http://www.itc.univie.ac.at/~hans/Columbus/columbus_parallel.html
11. GAMESS-UK is a package of ab initio programs written by M.F. Guest, J.H. van Lenthe, J. Kendrick, K. Schöffel, and P. Sherwood, with contributions from R.D. Amos, R.J. Buenker, H.J.J. van Dam, M. Dupuis, N.C. Handy, I.H. Hillier, P.J. Knowles, V. Bonacic-Koutecky, W. von Niessen, R.J. Harrison, A.P. Rendell, V.R. Saunders, A.J. Stone and A.H. de Vries. The package is derived from the original GAMESS code due to M. Dupuis, D. Spangler and J. Wendoloski, NRCC Software Catalog, Vol. 1, Program No. QG01 (GAMESS), 1980.
    GAMESS 6.2, M.F. Guest, J. Kendrick, J.H. van Lenthe and P. Sherwood, Computing for Science Ltd., 1999.
    http://wserv1.dl.ac.uk/CFS/

12. GAMESS rel. June 1999, M.W. Schmidt, K.K. Baldridge, J.A. Boatz, S.T. Elbert, M.S. Gordon, J.H. Jensen, S. Koseki, N. Matsunaga, K.A. Nguyen, S.J. Su, T.L. Windus, M. Dupuis, J.A. Montgomery, J. Comput. Chem. **14**, 1347-1363 (1993).
`http://www.msg.ameslab.gov/GAMESS/GAMESS.html`
13. G.D. Fletcher, M.W. Schmidt, Distributed Data Interface - "SPMD" Data Server Model, 1999.
14. Programmer's Reference, GAMESS, June 1999.
15. DGauss 4.1, Oxford Molecular, 1998-1999.
`http://www.oxmol.com/software/unichem/dgauss`
16. E.A. Stahlberg, private communication, 1998.
17. Gaussian 98, Revision A.7, M.J. Frisch, G.W. Trucks, H.B. Schlegel, G.E. Scuseria, M.A. Robb, J.R. Cheeseman, V.G. Zakrzewski, J.A. Montgomery, Jr., R.E. Stratmann, J.C. Burant, S. Dapprich, J.M. Millam, A.D. Daniels, K.N. Kudin, M.C. Strain, O. Farkas, J. Tomasi, V. Barone, M. Cossi, R. Cammi, B. Mennucci, C. Pomelli, C. Adamo, S. Clifford, J. Ochterski, G.A. Petersson, P.Y. Ayala, Q. Cui, K. Morokuma, D.K. Malick, A.D. Rabuck, K. Raghavachari, J.B. Foresman, J. Cioslowski, J.V. Ortiz, A.G. Baboul, B.B. Stefanov, G. Liu, A. Liashenko, P. Piskorz, I. Komaromi, R. Gomperts, R.L. Martin, D.J. Fox, T. Keith, M.A. Al-Laham, C.Y. Peng, A. Nanayakkara, C. Gonzalez, M. Challacombe, P.M.W. Gill, B. Johnson, W. Chen, M.W. Wong, J.L. Andres, C. Gonzalez, M. Head-Gordon, E.S. Replogle, and J.A. Pople, Gaussian, Inc., Pittsburgh PA, 1998, `www.gaussian.com`
18. MPQC (Massively Parallel Quantum Chemistry) Limit Point Systems, Inc., Fremont, CA, 1999.
`http://aros.ca.sandia.gov/~cljanss/mpqc/`
19. C.L. Janssen, E.T. Seidl, and M.E. Colvin, *Object Oriented Implementation of Parallel Ab Initio Programs*, ACS Symposium Series 592, *Parallel Computers in Computational Chemistry*, T. Mattson, Ed., 1995.
`http://aros.ca.sandia.gov/ cljanss/sc/`
20. K. Dackland, *Global Array Toolkit*, slide show, 1999.
`http://www.hpc2n.umu.se/events/courses/slides/GA/index.htm`
21. R. Lindh, *Parallel Computing in Chemistry*.
`http://www.hpc2n.umu.se/events/ngssc/98/parallel/notes/lectures_26-27/`
22. J. Boyle, R. Butler, T. Disz, B. Glickfeld, E. Lusk, R. Overbeek, J. Patterson, R. Stevens. *Portable Programs for Parallel Processors*, Holt, Rinehart and Winston, Inc., 1987.
23. `http://www.emsl.pnl.gov:2080/docs/parsoft/chemio/chemio.html`
24. CRAY T3E Fortran Optimization Guide, SG-2518 3.0, Cray Res. Inc., 1997.

Please note that the following examples are written for tutorial purposes only, i.e. the author acknowledges that there exists more elegant and sophisticated solutions.

Appendix A: Shared Counter using TCGMSG NXTVAL Function

Program Code

```
program sc
implicit none

integer, parameter      :: NTASKS = 20
integer                 :: nodes, me
integer, external       :: NNODES, NODEID, NXTVAL
integer                 :: next, i, ntsks
logical                 :: master

nodes  = NNODES()
me     = NODEID()
master = me == 0

if ( master ) print *, 'run on', nodes, ' nodes'

ntsks = 0
next = NXTVAL ( nodes )

do i=0, NTASKS-1
   if ( next == i ) then
      print *, 'node', me, ' do task', next

      call work ( 5. )

      ntsks = ntsks + 1
      next = NXTVAL ( nodes )
   end if
end do

print *, 'node', me, ' processed', ntsks, ' tasks'

end program sc

!------------------------------------------------------------------------------

subroutine work ( fctr )
implicit none
real            :: fctr
real            :: tmp
integer         :: secs, left, error

call RANDOM_NUMBER ( tmp )
secs = fctr * tmp

call pxfsleep ( secs, left, error )             !sleep for <secs> seconds

return
end subroutine work
```

Example Output

```
 run on 6  nodes
 node 1  do task 3
 node 1  do task 7
 node 1  do task 14
 node 1  processed 3  tasks
 node 4  do task 2
 node 4  do task 9
 node 4  do task 15
 node 4  processed 3  tasks
...
 node 3  do task 0
 node 3  do task 6
 node 3  do task 13
 node 3  do task 18
 node 3  processed 4  tasks
 run on 6  nodes
 node 0  do task 5
 node 0  do task 8
 node 0  do task 12
 node 0  do task 19
 node 0  processed 4  tasks
```

Appendix B: Creation of Global 2D Array (I)

Program Code

```
program create_2d

implicit none

include 'mafdecls.fh'
include 'global.fh'

integer, parameter      :: MAXDIM = 100

integer         :: n_nodes, me, ga_hndl, owner
logical         :: status
integer         :: jnk_size, i

call pbeginf ()                                         ! init TCGMSG
call ga_initialize ()                                   ! init GA
n_nodes = ga_nnodes ()
me = ga_nodeid ()

jnk_size = MAXDIM/n_nodes

status = ma_init(MT_DBL, MAXDIM*jnk_size, MT_BYTE)     ! allocate memory
status = ga_create ( MT_DBL, MAXDIM, MAXDIM, 'A', jnk_size, MAXDIM, ga_hndl)

do i = 1, MAXDIM, MAXDIM/n_nodes
   if ( ga_locate( ga_hndl, i, 1, owner) ) then
      if ( me == owner ) &
         print *,'PE', ga_nodeid(), ' owns rows ', i, ' - ', i+MAXDIM/n_nodes-1
   end if
end do
```

```
call ga_sync ()
status = ga_destroy ( ga_hndl )
call ga_terminate()
call pend()
stop
end program create_2d
```

Example Output

```
 PE 3  owns rows  76  -  100
 PE 2  owns rows  51  -  75
 PE 0  owns rows  1  -  25
 PE 1  owns rows  26  -  50
```

Appendix C: Creation of Global 2D Array (II)

Program Code

```
program crea_2d

implicit none

include 'mafdecls.fh'
include 'global.fh'

integer, parameter      :: MAXDIM = 100

integer         :: n_nodes, me, ga_hndl, owner
logical         :: status
integer         :: jnk_size, j

call pbeginf ()                                         ! init TCGMSG
call ga_initialize ()                                   ! init GA
n_nodes = ga_nnodes ()
me = ga_nodeid ()

jnk_size = MAXDIM/n_nodes

status = ma_init(MT_DBL, MAXDIM*jnk_size, MT_BYTE)        ! allocate memory

status = ga_create ( MT_DBL, MAXDIM, MAXDIM, 'A', MAXDIM, jnk_size, ga_hndl)

if ( me == 0 ) call ga_summarize ( .true. )

call ga_sync()
status = ga_destroy ( ga_hndl )
call ga_terminate()
call pend()
stop
end program crea_2d
```

Example Output

```
 Summary of allocated global arrays
-----------------------------------
  array 0 => double precision A(100,100),  handle: -1000
    (1:100,1:25) -> 0
    (1:100,26:50) -> 1
    (1:100,51:75) -> 2
    (1:100,76:100) -> 3
```

Appendix D: Print a Distributed Array

Program Code Snippet

```
...
integer, parameter      :: MAXDIM = 16
logical                 :: status
integer                 :: ga_hndl, chunk_size
...
status = ga_create ( MT_DBL, MAXDIM, MAXDIM, 'A', MAXDIM, chunk_size, ga_hndl)

call ga_sync()
call ga_print ( ga_hndl )
...
```

Example Output

```
 global array: A[1:16,1:16],  handle: -1000
```

	1	2	3	4	5	6
1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
5	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
6	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
7	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
8	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
9	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
10	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
11	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
12	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
13	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
14	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
15	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
16	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

	7	8	9	10	11	12
1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
5	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
6	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

```
  7       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
  8       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
  9       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
 10       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
 11       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
 12       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
 13       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
 14       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
 15       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000
 16       0.00000     0.00000     0.00000     0.00000     0.00000     0.00000

         13          14          15          16
     ----------- ----------- ----------- -----------
  1       0.00000     0.00000     0.00000     0.00000
  2       0.00000     0.00000     0.00000     0.00000
  3       0.00000     0.00000     0.00000     0.00000
  4       0.00000     0.00000     0.00000     0.00000
  5       0.00000     0.00000     0.00000     0.00000
  6       0.00000     0.00000     0.00000     0.00000
  7       0.00000     0.00000     0.00000     0.00000
  8       0.00000     0.00000     0.00000     0.00000
  9       0.00000     0.00000     0.00000     0.00000
 10       0.00000     0.00000     0.00000     0.00000
 11       0.00000     0.00000     0.00000     0.00000
 12       0.00000     0.00000     0.00000     0.00000
 13       0.00000     0.00000     0.00000     0.00000
 14       0.00000     0.00000     0.00000     0.00000
 15       0.00000     0.00000     0.00000     0.00000
 16       0.00000     0.00000     0.00000     0.00000
```

Appendix E: Parallel Matrix Multiply using GA_DGEMM

Program Code

```
program mxm_1

implicit none

include 'mafdecls.fh'
include 'global.fh'

integer                 :: n

integer                 :: n_nodes, me, hndl_a, hndl_b, hndl_c, owner
logical                 :: status
integer                 :: jnk_size, mem_size, j
logical                 :: master
real                    :: s, alpha, beta

real                    :: usr_time(2), usrt(2), mflops

call pbeginf ()                                         ! init TCGMSG
call ga_initialize ()                                   ! init GA
n_nodes = ga_nnodes ()
me = ga_nodeid ()
master = me == 0
```

```
call CPU_TIME ( usr_time(1) )

if ( master ) then
   print *, 'Nodes: ', n_nodes
   read (*,*) n
   print *, 'Rank = ', n
end if

call ga_sync()
call ga_brdcst ( MT_INT, n, MA_sizeof(MT_INT,1,MT_BYTE), 0 )

jnk_size = n/n_nodes

mem_size = MAX(6*jnk_size*jnk_size, 2000000)

status = ma_init(MT_DBL, 0 , mem_size)        ! allocate memory

call CPU_TIME ( usrt(1) )

status = ga_create ( MT_DBL, n, n, 'A', jnk_size, n, hndl_a)
status = ga_create ( MT_DBL, n, n, 'B', n, jnk_size, hndl_b)
status = ga_create ( MT_DBL, n, n, 'C', jnk_size, jnk_size, hndl_c)

s = 2.
call ga_fill_patch ( hndl_a, 1, n, 1, n, s )
s = 0.5
call ga_fill_patch ( hndl_b, 1, n, 1, n, s )
call ga_zero ( hndl_c )

call CPU_TIME ( usrt(2) )
if ( master ) then
   write (*, '(''usr time for init'',t24,f9.3,1x,''s'')') usrt(2)-usrt(1)
end if

call ga_sync()
call CPU_TIME ( usrt(1) )

call ga_dgemm( 'N',  'N', n, n, n, 1.0, hndl_a, hndl_b, 0.0, hndl_c )

call ga_sync()
call CPU_TIME ( usrt(2) )
usrt(1) = usrt(2)-usrt(1)
mflops=1.e-6*2*n*n*n/usrt(1)
if ( master ) then
   write (*, '(''usr time for matmul'',t24,f9.3,1x,''s'')') usrt(1)
   write (*, '(''global performance of'',t24,f9.3,1x,''mflop/s'')')  mflops
end if

if ( master ) then
   call ga_summarize ( .true. )
end if
call ga_sync()

status = ga_destroy ( hndl_c )
status = ga_destroy ( hndl_b )
status = ga_destroy ( hndl_c )
call ga_terminate()
call pend()
```

```
call CPU_TIME ( usr_time(2) )
if ( master ) then
   write (*, '(''total usr time'',t24,f9.3,1x,''s'')') usr_time(2)-usr_time(1)
end if

stop
end program mxm_1
```

Example Output

```
 Nodes:  16
 Rank =  1000
usr time for init          0.006 s
usr time for matmul        0.522 s
global performance of   3830.552 mflop/s
 Summary of allocated global arrays
-----------------------------------
  array 0 => double precision A(1000,1000),  handle: -1000
    (1:63,1:1000) -> 0
    (64:126,1:1000) -> 1
    (127:189,1:1000) -> 2
    (190:252,1:1000) -> 3
    (253:315,1:1000) -> 4
    (316:378,1:1000) -> 5
    (379:441,1:1000) -> 6
    (442:504,1:1000) -> 7
    (505:567,1:1000) -> 8
    (568:630,1:1000) -> 9
    (631:693,1:1000) -> 10
    (694:756,1:1000) -> 11
    (757:819,1:1000) -> 12
    (820:882,1:1000) -> 13
    (883:945,1:1000) -> 14
    (946:1000,1:1000) -> 15
  array 1 => double precision B(1000,1000),  handle: -999
    (1:1000,1:63) -> 0
    (1:1000,64:126) -> 1
    (1:1000,127:189) -> 2
    (1:1000,190:252) -> 3
    (1:1000,253:315) -> 4
    (1:1000,316:378) -> 5
    (1:1000,379:441) -> 6
    (1:1000,442:504) -> 7
    (1:1000,505:567) -> 8
    (1:1000,568:630) -> 9
    (1:1000,631:693) -> 10
    (1:1000,694:756) -> 11
    (1:1000,757:819) -> 12
    (1:1000,820:882) -> 13
    (1:1000,883:945) -> 14
    (1:1000,946:1000) -> 15
  array 2 => double precision C(1000,1000),  handle: -998
    (1:250,1:250) -> 0
    (251:500,1:250) -> 1
    (501:750,1:250) -> 2
    (751:1000,1:250) -> 3
    (1:250,251:500) -> 4
```

```
    (251:500,251:500) -> 5
    (501:750,251:500) -> 6
    (751:1000,251:500) -> 7
    (1:250,501:750) -> 8
    (251:500,501:750) -> 9
    (501:750,501:750) -> 10
    (751:1000,501:750) -> 11
    (1:250,751:1000) -> 12
    (251:500,751:1000) -> 13
    (501:750,751:1000) -> 14
    (751:1000,751:1000) -> 15
```

Appendix F: Irregular Distributed 2D Array

Program Code

```
program irreg_2d

implicit none

include 'mafdecls.fh'
include 'global.fh'

integer, parameter      :: MAXDIM = 8

integer                 :: n_nodes, me, ga_hndl, owner
logical                 :: status

integer, parameter      :: NBLOCK1 = 1, NBLOCK2 = 3
integer                 :: map1(NBLOCK1) = 1, map2(NBLOCK2) = (/1,2,5/)
integer                 :: node, mem_sze

logical                 :: master

call pbeginf ()                                         ! init TCGMSG
call ga_initialize ()                                   ! init GA
n_nodes = ga_nnodes ()
if ( n_nodes /= NBLOCK2 ) then
   print *, 'need', NBLOCK2, ' nodes, have', n_nodes, '!'
   stop 'wrong no. of nodes'
end if
me      = ga_nodeid ()
master  = me == 0

status = ma_init(MT_DBL, MAXDIM*MAXDIM, MT_BYTE)

status = ga_create_irreg ( MT_DBL, MAXDIM, MAXDIM, 'A', &
                           map1, NBLOCK1, map2, NBLOCK2, ga_hndl)

if ( master ) call ga_summarize ( .true. )

call ga_sync()
status = ga_destroy ( ga_hndl )
call ga_terminate()
call pend()
stop
end program irreg_2d
```

Example Output

```
 Summary of allocated global arrays
-----------------------------------
  array 0 => double precision A(8,8),  handle: -1000
    (1:8,1:1) -> 0
    (1:8,2:4) -> 1
    (1:8,5:8) -> 2
```

AB INITIO METHODS FOR ELECTRON CORRELATION IN MOLECULES

PETER KNOWLES
School of Chemistry
University of Birmingham
Edgbaston
Birmingham, B15 2TT
United Kingdom
E-mail: P.J.Knowles@bham.ac.uk

MARTIN SCHÜTZ AND HANS-JOACHIM WERNER
Institute for Theoretical Chemistry
University of Stuttgart
Pfaffenwaldring 55
70569 Stuttgart
Germany
E-mail: {schuetz, werner}@theochem.uni-stuttgart.de

Reliable *ab initio* electronic structure calculations require high-level treatment of electron correlation effects. For molecules in electronic ground states, single-reference correlation methods, which are based on the Hartree-Fock self-consistent field (SCF) wavefunctions as zeroth order approximation, are usually sufficient. Møller-Plesset perturbation theory up to fourth order (MP2-MP4) and coupled-cluster methods with all single and double excitations followed by a perturbative treatment of triple excitations [CCSD(T)] are the most popular single-reference methods. All of these approaches can also be formulated in a local framework which gives a demand on computational resources that scales only linearly with system size; they can also be carried out using integral-direct techniques, that avoid the storage of large numbers of two electron integrals by recomputing them on demand. For computing electronically excited states or global potential energy functions, multiconfiguration self-consistent field (MCSCF) wavefunctions are required for a qualitatively correct representation of the wavefunction. The major part of dynamical electron correlation effects can then be accounted for by subsequent multireference correlation treatments, in which a large number of single and double excitations relative to the MCSCF reference configurations are taken into account. In multireference configuration interaction (MRCI) calculations the expansion coefficients are determined variationally. Alternatively, the coefficients can be obtained by first-order perturbation theory, and the energy be evaluated to second (MRPT2) or third (MRPT3) order. These lecture notes give a short review of all these methods.

1 Introduction

1.1 Electron correlation and the configuration interaction method

Hartree-Fock Self-Consistent Field (SCF) Theory enjoys considerable success in the first-principles determination of molecular electronic wavefunctions and properties. However, there are important situations where the underlying assumption of molecular orbital theory, that the electronic wavefunction can be approximated by an antisymmetrized product of orbitals, breaks down. There are still further situations where SCF does provide a reasonable qualitative description, but fails to

predict energetics to desired accuracy. We explore here the deficiencies of Hartree-Fock, and survey the various techniques available for going beyond SCF.

Hartree-Fock is a mean field theory, in which each electron has its own wavefunction (orbital), which in turn obeys an effective 1-electron Schrödinger equation. The effective hamiltonian (Fock operator) contains the average field (Coulomb and exchange) of all other electrons in the system. The total electronic wavefunction for the molecule, ignoring complications introduced by the Pauli principle, is a simple product of the orbitals. Following the Born interpretation of wavefunctions, this implies that if $P(r_1, r_2)$ is the probability density for finding electrons labelled 1 and 2 in regions of space around r_1 and r_2 respectively,

$$P(r_1, r_2) = P(r_1)P(r_2) \tag{1}$$

i.e., the probability density for a given electron is independent of the positions of all others.

In reality, however, the motions of electrons are more intimately correlated. Because of the direct Coulomb repulsion of electrons, the instantaneous position of electron 2 forms the centre of a region in space which electron 1 will avoid. This avoidance is more than that caused by the mean field, and is local; if electron 2 changes position, the Coulomb hole for electron 1 moves with it. In contrast, in the mean-field theory, electron 1 has no knowledge of the instantaneous position of 2, only its average value, and thus motions are uncorrelated, and there is no depletion in $P(r_1, r_2)$ near $r_1 = r_2$.

The effects of neglecting electron correlation in Hartree-Fock are spectacularly illustrated when one attempts to compute complete potential curves for diatomic molecules using SCF. Figure 1 shows potential curves for H_2 from both a very accurate calculation and from Hartree-Fock. It is seen that the spin-restricted Hartree-Fock (RHF) approximation breaks down as dissociation is reached, predicting energies which are much too high, and a potential curve characteristic of the interaction of ions rather than neutral atoms. The RHF wavefunction for the $X^1\Sigma_g^+$ ground state of H_2 takes the form

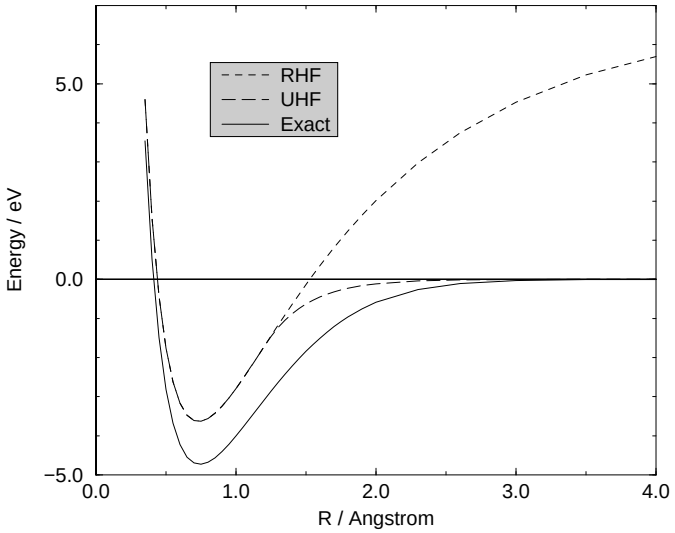
$$\Psi_{\mathrm{X}} = \hat{\mathcal{A}}\sigma_g^{\alpha}(1)\sigma_g^{\beta}(2) \tag{2}$$

where $\hat{\mathcal{A}}$ is the antisymmetrizing operator, α and β are the usual one-electron spin functions, and the bonding orbital $\sigma_g = Z_{\sigma_g}(\chi_{\mathrm{A}} + \chi_{\mathrm{B}})$, with χ_{A} an s-like orbital centred on atom A, and Z_{σ_g} a normalization constant. As the atoms become infinitely separated, $\chi_{\mathrm{A}} \sim 1s_{\mathrm{A}}$, $Z_{\sigma_g} \sim \frac{1}{\sqrt{2}}$ and thus

$$\Psi_{\mathrm{X}} \sim \frac{1}{2}\hat{\mathcal{A}}\left(1s_{\mathrm{A}}^{\alpha}1s_{\mathrm{B}}^{\beta} + 1s_{\mathrm{B}}^{\alpha}1s_{\mathrm{B}}^{\beta} + 1s_{\mathrm{A}}^{\alpha}1s_{\mathrm{A}}^{\beta} + 1s_{\mathrm{B}}^{\alpha}1s_{\mathrm{B}}^{\beta}\right) \tag{3}$$

The first two terms are direct products of neutral ^2S hydrogen atom wavefunctions on the two atoms A and B, as desired. However, the last two terms describe a spurious $H^+ \ldots H^-$ pair. The overall energy of this unphysical wavefunction exceeds the energy of two hydrogen atoms by half the difference of the ionization energy and electron affinity of H (i.e., 6.4 eV), and at long range the potential energy curve has an unphysical ionic R^{-1} behaviour.

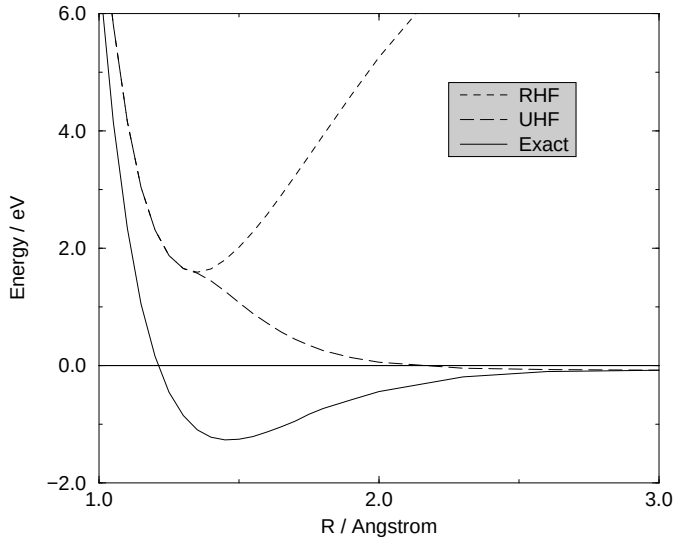
Figure 1. Potential Energy Curves for H_2



The failure of RHF for this example can be easily understood in terms of electron correlation. At long internuclear separations, if one electron is located near atom A, the other will on physical grounds be found close to atom B. This correlation is reflected in the exact wavefunction, which is asymptotically the product of hydrogenic orbitals on the two nuclei. In contrast, within the Hartree-Fock framework, each electron is made to experience only the average effect of the other. Since in RHF, the two electrons are constrained to be in the same spatial orbital, this σ_g orbital will be symmetrical between the atoms, and thus each electron has equal probability of being on A or B, irrespective of the position of the other electron. The possibility of both electrons being on the same atom is not excluded, as reflected in the ionic terms in the RHF wavefunction (3).

In the case of H_2, and in fact for a number of other dissociating molecules, Hartree-Fock theory can give correct behaviour provided the restriction to identical spatial orbitals for α and β spin is relaxed. The Unrestricted HF (UHF) wavefunction for H_2 is identical to RHF at short bond lengths, but when the two atoms are separated, it becomes variationally advantageous for the α and β spin orbitals to localize on different hydrogen atoms. In this way, a correct asymptotic energy is obtained, as seen in Figure 1. However, the wavefunction can never be identical to the exact wavefunction. Asymptotically, the UHF wavefunction is either $\hat{\mathcal{A}}1s_{\mathrm{A}}^{\alpha}1s_{\mathrm{B}}^{\beta}$ or $\hat{\mathcal{A}}1s_{\mathrm{B}}^{\alpha}1s_{\mathrm{A}}^{\beta}$, whereas the true wavefunction is the sum of these two degenerate determinants. Although the energy is unaffected, the UHF wavefunction is not an eigenfunction of the spin-squared operator $\hat{S}^2$, being an unphysical mixture of singlet and triplet states. This spin contamination is displeasing, and

Figure 2. Potential Energy Curves for F_2



can have serious undesirable effects. In the case of the H_2 UHF potential curve, at the point where UHF and RHF diverge, the curve is discontinuous in its second derivative. For more advanced correlation methods which build on UHF, spin contamination has a disastrous effect[1,2]. In the case of F_2 (Figure 2), UHF does not repair the inability of RHF to give an energy at equilibrium geometry which is lower than at dissociation, and as a consequence the UHF potential curve is purely repulsive. For all these reasons, the use of UHF is becoming increasingly rare.

1.2 Long-range correlation — Molecular Dissociation

In order to understand a theory which goes beyond the inability of RHF to describe dissociation, we examine first of all an excited $^1\Sigma_g^+$ state of H_2 for which the RHF wavefunction takes the form

$$\Psi_{\mathrm{E}} = \hat{\mathcal{A}}\sigma_u^{\alpha}(1)\sigma_u^{\beta}(2) \tag{4}$$

and where we now have two electrons in the antibonding orbital $\sigma_u = Z_{\sigma_u}(\chi_{\mathrm{A}} - \chi_{\mathrm{B}})$. Asymptotically, this becomes

$$\Psi_{\mathrm{E}} \sim \tfrac{1}{2}\hat{\mathcal{A}}\left(1s_{\mathrm{A}}^{\alpha}1s_{\mathrm{B}}^{\beta} + 1s_{\mathrm{B}}^{\alpha}1s_{\mathrm{A}}^{\beta} - 1s_{\mathrm{A}}^{\alpha}1s_{\mathrm{A}}^{\beta} - 1s_{\mathrm{B}}^{\alpha}1s_{\mathrm{B}}^{\beta}\right) \tag{5}$$

This wavefunction also contains an unphysical mixture of covalent and ionic terms. However, we observe that it is possible to construct purely ionic or purely covalent wavefunctions by taking a linear combination of Ψ_{X} and Ψ_{E}. In $\Psi_{\mathrm{X}} - \Psi_{\mathrm{E}} = \sigma_g^2 - \sigma_u^2$, the ionic terms cancel exactly, and the correct asymptotic wavefunction is obtained.

This is an example of *configuration interaction* (CI), whereby the wavefunction is considered as being a mixture of several Slater determinants. For H_2 at general internuclear separations, the form of the CI wavefunction is

$$\Psi = c_{\mathrm{X}}\Psi_{\mathrm{X}} + c_{\mathrm{E}}\Psi_{\mathrm{E}} \tag{6}$$

and the coefficients specifying this linear combination must be allowed to vary, since it is known that near equilibrium, the RHF wavefunction is already a good approximation. Thus the best wavefunction near equilibrium will have $c_{\mathrm{X}} \simeq 1$ and c_{E} small, in contrast to their asymptotic values of $\frac{1}{\sqrt{2}}$ and $-\frac{1}{\sqrt{2}}$.

In general, in the standard CI method, the *variational principle* is used to determine the CI coefficients. For any approximate wavefunction, the Rayleigh quotient

$$\overline{E} = \frac{\langle\Psi|\hat{H}|\Psi\rangle}{\langle\Psi|\Psi\rangle} \tag{7}$$

is an upper bound to the exact ground-state energy E, i.e., $\overline{E} \geq E$. Variational methods proceed by assuming that the best wavefunction will be the one which gives the lowest, i.e. minimum, $\overline{E}$. In the specific case of a linear expansion, as in CI, i.e.,

$$\Psi = \sum_I c_I \Phi_I \tag{8}$$

minimising $\overline{E}$ is equivalent to finding the lowest eigensolution of the *hamiltonian matrix* $\mathbf{H}$, whose elements are the integrals

$$H_{IJ} = \langle\Phi_I|\hat{H}|\Phi_J\rangle \ , \tag{9}$$

i.e. one needs to solve

$$\mathbf{Hc} = \overline{E}\mathbf{c} \tag{10}$$

with the minimum Rayleigh quotient $\overline{E}$ appearing as the eigenvalue. The linear ansatz allows also the calculation of approximations to excited states, through the Hylleraas-Undheim-MacDonald theorem, which states that the n-th eigenvalue is an upper bound to the exact energy of the $(n-1)$-th excited state. Finding the lowest few eigensolutions of a symmetric matrix is a well-studied problem; for the diagonally-dominant hamiltonian matrices invariably arising in molecular CI, algorithms exist[3] which will converge in around ten iterations, each of which requires the evaluation of the action of the hamiltonian matrix on some trial vector, i.e.,

$$v_I = \sum_J H_{IJ} c_J \ . \tag{11}$$

This feature allows the solution of CI problems of very large dimensions; because $\mathbf{H}$ is often extremely sparse, forming $\mathbf{H}\cdot\mathbf{c}$ is much easier than forming the matrix itself, and the limiting factor is the availability of memory to store $\mathbf{c}$ and $\mathbf{v}$. Calculations with more than 10^9 configurations have been carried out in this way.

Although consideration of electron correlation is clearly vital for the proper description of molecules closed to dissociation, it also has important implications in situations where Hartree-Fock is a reasonable approximation. Since the hamiltonian operator contains r_{ij}^{-1}, the inverse distance between two electrons, the nature of the electronic wavefunction in regions close to $r_{ij} = 0$ will have a strong effect on the energy.

We will consider initially the helium atom, for which the hamiltonian is

$$\hat{H} = -\tfrac{1}{2}\nabla_1^2 - \tfrac{1}{2}\nabla_2^2 - \frac{2}{r_1} - \frac{2}{r_2} + \frac{1}{r_{12}}\,. \tag{12}$$

The electronic wavefunction will satisfy Schrödinger's equation

$$\hat{H}\Psi(\mathbf{r}_1, \mathbf{r}_2) = E\Psi(\mathbf{r}_1, \mathbf{r}_2) \tag{13}$$

at all points in six-dimensional space. We note that close to $r_{12} = 0$ there is a paradox; the left hand side of (13) apparently becomes infinite, because of the $1/r_{12}$ Coulomb singularity, whereas E is constant, and so the right hand side is well behaved. The local energy $\hat{H}\Psi/\Psi$ cannot have singularities since it is constant, and the inescapable conclusion is that there must be an additional singularity in the left hand side of (13) which exactly cancels $1/r_{12}$ close to $r_{12} = 0$. Since the electrons are not necessarily close to a nucleus, the only candidate for this cancelling term is the kinetic energy. It is convenient to transform to centre-of-mass and relative coordinates,

$$\mathbf{R} = \tfrac{1}{2}\left(\mathbf{r}_2 + \mathbf{r}_1\right)\,; \qquad \mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1\,, \tag{14}$$

in which the hamiltonian becomes

$$\hat{H} = -\tfrac{1}{4}\nabla_{\mathbf{R}}^2 - \frac{2}{r_1} - \frac{2}{r_2} - \nabla_{\mathbf{r}}^2 + \frac{1}{r}\,. \tag{15}$$

If we expand the two-electron wavefunction in a Taylor series in r about $r = 0$, on the (correct for the singlet state) assumption that angular terms in $\mathbf{r}$ can be ignored at low order,

$$\Psi = a_0 + a_1 r + a_2 r^2 + \ldots \tag{16}$$

then the Schrödinger equation expands as

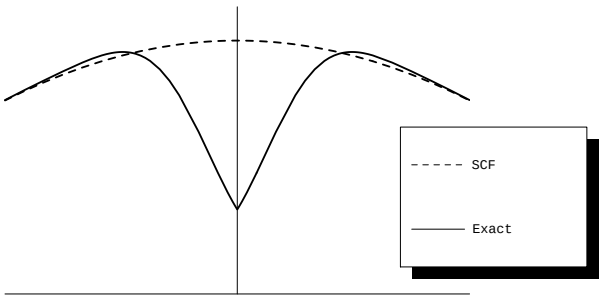
$$0 = r^{-1}\left(a_0 - 2a_1\right) + r^0\left(a_1 - 6a_2 - 4R^{-1} - E\right) + r^1\left(\ldots\right. \tag{17}$$

The r^{-1} singularity is removed if $a_1 = \frac{1}{2}a_0$, or

$$\left.\frac{\partial\Psi}{\partial r}\right|_{r=0} = \left.\frac{\Psi}{2}\right|_{r=0}\,. \tag{18}$$

This is the well-known cusp condition[4,5,6,7], which shows that in whatever direction one moves from $r = 0$, the wavefunction increases linearly. The exact wavefunction must have the shape depicted in Figure 3, showing the existence of a *Coulomb Hole* around the point of coalescence. In Figure 3, the wavefunctions are plotted against $z = z_2 - z_1$, with the two electrons having identical x, y coordinates.

Figure 3. The interelectronic cusp



The Hartree-Fock wavefunction is

$$\Psi_{\rm RHF} = \hat{\mathcal{A}} 1s^{\alpha} 1s^{\beta} = 1s(r_1)1s(r_2)\,\tfrac{1}{\sqrt{2}}(\alpha(1)\beta(2) - \beta(1)\alpha(2)) \tag{19}$$

which has no special behaviour near coalescence; in fact it is easy to show that $\partial\Psi_{\rm RHF}/\partial r = 0$ at $r = 0$. Thus the RHF wavefunction must have the shape shown in Figure 3; clearly, it overestimates the probability of finding the two electrons close together, and this in turn implies an overestimate of the electron repulsion energy. This is consistent with the variational principle, which requires the RHF energy to be higher than the exact energy. We define the correlation energy to be

$$\mathcal{E} = E^{\rm RHF} - E^{\rm exact} \tag{20}$$

where $E^{\rm exact}$ is the lowest exact eigenvalue of Schrödinger's equation. For He, $\mathcal{E} \simeq 0.042\,\text{hartree} = 1.1\,\text{eV}$.

The above analysis for the helium ground state, consisting of two electrons with opposing spin, needs to be modified when spins are instead aligned. A triplet spin wavefunction, e.g., $\alpha(1)\alpha(2)$ is symmetric with respect to electron label exchange, and so, by the Pauli principle, the spatial wavefunction must be antisymmetric. This has the consequences that, in a picture like Figure 3, the triplet wavefunction must pass through the origin, and has dipole rather than monopole $\mathbf{r}$ angular variation. There is a corresponding cusp condition specifying $\partial^2\Psi/\partial r^2$ in terms of $\partial\Psi/\partial r$ at the coalescence point[5], but the important thing is that in the energetically important region, the electrons are already kept apart by the Pauli principle, even in Hartree-Fock, and the effects of electron correlation neglect are fairly minor. Electron correlation effects are most important for electrons with opposing spins.

A further observation for polyelectronic systems is that the biggest contributions will come from pairs of electrons which occupy the same regions of physical space. If orbitals are well localized, there will be a large contribution to the correlation energy from each doubly occupied orbital, with smaller additions from pairs consisting of two different orbitals. This leads to a rough rule of thumb, that each doubly occupied orbital contributes approximately 1 eV to the total electron correlation energy.

In atomic and molecular systems, an alternative and equivalent way of visualising two-electron correlations relative to the nuclear positions is possible. If one electron is far from the nucleus of an atom, then the second electron will prefer to

be closer to the nucleus than its Hartree-Fock average; this is termed *radial* correlation. If a first electron is, say, to the right of a nucleus, then another electron will tend to visit regions of space to the left of that nucleus more than predicted by HF; this is termed *angular* correlation.

These short-range correlation effects arising from the Coulomb hole can be represented using CI wavefunctions just as with the long-range correlations discussed above. The simplest such wavefunction representing the angular correlation in the helium atom would have the form

$$\Psi = \hat{\mathcal{A}}\left(1s^{\alpha}(1)1s^{\beta}(2) + \lambda\left(2p_x^{\alpha}(1)2p_x^{\beta}(2) + 2p_y^{\alpha}(1)2p_y^{\beta}(2) + 2p_z^{\alpha}(1)2p_z^{\beta}(2)\right)\right) \quad (21)$$

It is straightforward to show that such an ansatz introduces explicit r_{12} dependence into the wavefunction. This demonstrates that CI does support correlated wavefunctions. However, unfortunately, the r_{12} dependence introduced is entirely in terms of r_{12}^2; there are no linear terms. A CI wavefunction can *never* satisfy the cusp condition (18), since its gradient will always be zero at coalescence; however, given sufficient terms, the linear combination of functions of r_{12}^2 will give a reasonable representation of the shape of the Coulomb hole. Because the expansion functions are not ideally suited to the problem, the convergence of the CI expansion is unfortunately slow, and this is discussed further below.

Historically, even some of the earliest molecular electronic structure calculations[8,9] used 2-electron basis functions of a type better adapted to the problem than orbital products (i.e., CI). Inclusion of linear terms in r_{12} is an efficient way to obtain an accurate wavefunction with a small number of functions, and probably it will remain the approach of choice when very high accuracy is needed, particularly for atoms. However, despite successful research activity in this area[10,11] this approach has not yet emerged as the best method generally applicable to molecules; CI expansions remain computationally preferable. The reason for this preference is that, although very large numbers of basis functions might be required, the hamiltonian integrals which have to be computed for CI are much simpler than for explicitly correlated wavefunctions. The explicit r_{12} terms introduce 3- and 4-electron integrals[12,13] which are potentially very numerous. In contrast, CI needs only the two-electron integrals required in an SCF calculation. Although the 3- and 4-electron integrals can be reasonably approximated[11], explicitly correlated wavefunctions still remain a specialist rather than general-purpose tool.

1.4 Second Quantization

The adoption of the CI (or other related) approach to electron correlation implies that we deal with wavefunctions which are represented as vectors in a linear space of Slater determinants; this space is in turn a subspace of N-fold products of orbitals. For the moment, we will assume that we generate all of the N-electron basis functions that we can after appropriate symmetry adaptation (electron antisymmetry, point group, etc.). Therefore the N-electron basis set is determined entirely by a choice of 1-electron basis. Before considering what this choice should be for optimum accuracy, we consider the analysis and manipulation of N-electron functions of this orbital-product type. We note initially that the orbital basis will contain at least the SCF occupied orbitals, denoted $\{\phi_i\}$, but in order that further

configurations be generated, it must be augmented by *virtual* or *external* orbitals, $\{\phi_a\}$. Both the occupied and virtual orbitals can be considered as linear combinations of an underlying chosen fixed basis $\{\chi_\alpha\}$, which will usually be atom-centred functions, exactly as in basis-set SCF calculations. The functions ϕ_p and χ_α depend only on the spatial coordinate $\mathbf{r}$; where spin-orbitals are required, they will be denoted by $\psi_p(\mathbf{x})$ and can be constructed as a product of a spatial orbital $\phi_p(\mathbf{r})$ and a spin function α or β.

Consider a complete (infinite) one particle basis set $\{\phi_p(\mathbf{r}), p = 1, 2, \dots\}$; any function of the position $\mathbf{r}$ can be represented as a linear combination of the spatial orbitals

$$f(\mathbf{r}) = \sum_p x_p \phi_p(\mathbf{r}) \; . \tag{22}$$

For a system of N electrons, a complete spatial basis can then be generated by taking all possible products $\phi_{p_1}(\mathbf{r}_1)\phi_{p_2}(\mathbf{r}_2)\dots\phi_{p_N}(\mathbf{r}_N)$, i.e., any N particle spatial function may be expanded as

$$F(\mathbf{r}_1, \mathbf{r}_2, \dots \mathbf{r}_N) = \sum_{p_1 p_2 \dots p_N} X_{p_1 p_2 \dots p_N} \phi_{p_1}(\mathbf{r}_1)\phi_{p_2}(\mathbf{r}_2)\dots\phi_{p_N}(\mathbf{r}_N) \; . \tag{23}$$

This fact is not much use for practical calculations, since we cannot use an infinite set of functions, but if we consider now the case of a finite one particle basis $\{\phi_p, p = 1, 2, \dots, m\}$, then we see the concept of the corresponding complete N particle space, composed of all possible products of orbitals. A variational calculation in such a basis will yield the lowest possible energy eigenvalue for the given one particle basis set, and such a calculation is termed Full or Complete configuration interaction (FCI). It is, however, easily appreciated that the number of possible orbital products m^N (m one electron α and β spin orbitals, N electrons) can become exceedingly large.

We introduce the useful concept of *second quantization* by defining the *orbital excitation operator* as (assuming orthogonal orbitals)

$$\hat{E}_{pq} = \sum_{i=1}^{N} |\phi_p(i)\rangle\langle\phi_q(i)| \; . \tag{24}$$

The Dirac bracket notation means that whenever the brackets become closed, $\langle f(i)|g(i)\rangle$, integration over the coordinates of electron i is performed on the functions within the bracket, $\int d\tau_i f^*(i)g(i)$. If $\hat{E}_{pq}$ is made to act on any N electron function which is a product of orbitals, or a linear combination of such products, the effect is for each occurrence of ϕ_q to generate a function which is identical, but with ϕ_q replaced by ϕ_p. Thus if ϕ_q does not appear, $\hat{E}_{pq}$ annihilates the function.

$\hat{E}_{pq}$ is a spatial orbital excitation operator; it acts on space coordinates and does not affect spin. In fact, it can be decomposed into a sum of operators which excite α and β spin orbitals separately, $\hat{E}_{pq} = \hat{e}^{\alpha}_{pq} + \hat{e}^{\beta}_{pq} = \hat{\eta}^{\alpha\dagger}_p \hat{\eta}^{\alpha}_q + \hat{\eta}^{\beta\dagger}_p \hat{\eta}^{\beta}_q$, where $\hat{\eta}^{\alpha}_q$ destroys α spin orbital ψ_q and $\hat{\eta}^{\alpha\dagger}_p$ creates α spin orbital ψ_p. The idea of second quantization is that the orbitals themselves now become quantum mechanical operators. Thus a Slater determinant can be viewed as arising from successive applications of creation

operators on the empty (vacuum) state,

$$\ldots \eta_r^\dagger \eta_q^\dagger \eta_p^\dagger \Psi_{\text{vacuum}} = \hat{\mathcal{A}}(\psi_p \psi_q \psi_r \ldots) \,. \tag{25}$$

The analysis that follows continues to use pure spatial orbitals ϕ_p; however, exactly analogous results are obtained by using explicit spin-orbitals ψ_p and spin-orbital excitation operators $\hat{e}_{pq}$. Further details of the properties of the second quantization can be found in the literature [14].

As well as the single orbital excitation operators $\hat{E}_{pq}$, it is possible to define multiple excitation operators:

$$\hat{E}_{pq,rs} = \sum_{i \neq j}^{N} |\phi_p(i)\rangle\langle\phi_q(i)| \; |\phi_r(j)\rangle\langle\phi_s(j)| \equiv \hat{E}_{rs,pq} \tag{26}$$

$$\hat{E}_{pq,rs,tu} = \sum_{i \neq j \neq k}^{N} |\phi_p(i)\rangle\langle\phi_q(i)| \; |\phi_r(j)\rangle\langle\phi_s(j)| \; |\phi_t(k)\rangle\langle\phi_u(k)| \tag{27}$$

etc.

These can all be formulated as combinations of the single excitations:

$$\hat{E}_{pq,rs} = \sum_{i,j}^{N} |\phi_p(i)\rangle\langle\phi_q(i)| \; |\phi_r(j)\rangle\langle\phi_s(j)| - \sum_{i}^{N} |\phi_p(i)\rangle\langle\phi_q(i)|\phi_r(i)\rangle\langle\phi_s(i)| \tag{28}$$

$$= \hat{E}_{pq}\hat{E}_{rs} - \delta_{qr}\hat{E}_{ps} \tag{29}$$

Similar consideration of the identical operator $\hat{E}_{rs,pq}$ yields the commutation relation for the single excitations:

$$[\hat{E}_{pq}, \hat{E}_{rs}] = \hat{E}_{pq}\hat{E}_{rs} - \hat{E}_{rs}\hat{E}_{pq} = \delta_{qr}\hat{E}_{ps} - \delta_{ps}\hat{E}_{rq} \,. \tag{30}$$

Given that any wavefunction Ψ we construct is ultimately composed as a linear combination in the space of orbital products, then the following completeness identity is true for all $i = 1, 2, \ldots, N$

$$\left(\sum_{p}^{m} |\phi_p(i)\rangle\langle\phi_p(i)| \right) \; |\Psi\rangle = |\Psi\rangle \,. \tag{31}$$

Now we insert this identity into the electronic hamiltonian operator

$$\hat{H} = Z + \sum_{i}^{N} \hat{h}(i) + \sum_{i>j}^{N} r_{ij}^{-1} \,, \tag{32}$$

where Z is the nuclear repulsion energy, r_{ij} are the separations of the electrons, and $\hat{h}(i)$ is the single particle hamiltonian for each electron, incorporating its kinetic energy and the field of all the nuclei. This has the effect of replacing $\hat{H}$ by the effective *model* or *second quantized hamiltonian* $\hat{H}_M$, with the understanding that

the only thing we will ever do with $\hat{H}_M$ is to take matrix elements between functions in the orbital product space:

$$\begin{aligned}\hat{H}_M = Z &+ \sum_{i}^{N}\sum_{pq}^{m} |\phi_p(i)\rangle \langle\phi_p(i)|\hat{h}(i)|\phi_q(i)\rangle \langle\phi_q(i)| \\ &+ \sum_{i>j}^{N}\sum_{pqrs}^{m} |\phi_p(i)\rangle|\phi_r(j)\rangle \langle\phi_p(i)|\langle\phi_r(j)|r_{ij}^{-1}|\phi_q(i)\rangle|\phi_s(j)\rangle \langle\phi_q(i)|\langle\phi_s(j)| \qquad (33)\end{aligned}$$

$$= Z + \sum_{pq} h_{pq}\hat{E}_{pq} + \tfrac{1}{2}\sum_{pqrs}(pq|rs)\hat{E}_{pq,rs} \ , \qquad (34)$$

where we introduce the one and two electron *hamiltonian integrals*

$$h_{pq} = \langle\phi_p|\hat{h}|\phi_q\rangle = \int d\mathbf{r}_1 \, \phi_p^*(1)\hat{h}(1)\phi_q(1) \qquad (35)$$

$$\begin{aligned}(pq|rs) &= \langle\phi_p(1)|\langle\phi_r(2)|r_{12}^{-1}|\phi_q(1)\rangle|\phi_s(2)\rangle \\ &= \int d\mathbf{r}_1 \int d\mathbf{r}_2 \, \phi_p^*(1)\phi_r^*(2)r_{12}^{-1}\phi_q(1)\phi_s(2) \ . \qquad (36)\end{aligned}$$

For matrix elements between the N electron basis functions we then have

$$\begin{aligned}\langle\Phi_I|\hat{H}|\Phi_J\rangle &= \langle\Phi_I|\hat{H}_M|\Phi_J\rangle \\ &= Z\langle\Phi_I|\Phi_J\rangle + \sum_{pq} h_{pq}\langle\Phi_I|\hat{E}_{pq}|\Phi_J\rangle + \tfrac{1}{2}\sum_{pqrs}(pq|rs)\langle\Phi_I|\hat{E}_{pq,rs}|\Phi_J\rangle \ . \qquad (37)\end{aligned}$$

In this way, we separate *integrals* $h_{pq}, (pq|rs)$ and *coupling coefficients* $d_{pq}^{IJ} = \langle\Phi_I|\hat{E}_{pq}|\Phi_J\rangle$, $D_{pqrs}^{IJ} = \langle\Phi_I|\hat{E}_{pq,rs}|\Phi_J\rangle$. The coupling coefficients depend only on the algebraic structure of the N electron functions, and not on such factors as molecular geometry, external fields, etc.

We illustrate the use of the second-quantized formalism by considering CI wavefunctions for two electrons. Unnormalized spin-adapted basis functions can be constructed as

$$\Phi_{\pm}^{pq} = \tfrac{1}{2}\left(\hat{\mathcal{A}}(\phi_p^\alpha\phi_q^\beta) \pm \hat{\mathcal{A}}(\phi_q^\alpha\phi_p^\beta)\right) \ , \qquad (38)$$

with the upper (+) sign for spin $S = 0$ (singlet) and the lower (−) for $S = 1$ (triplet). The total wavefunction can then be expanded in this basis as

$$\begin{aligned}\Psi &= \sum_{p\geq q} C_{pq}(1 \pm \delta_{pq})\Phi_{\pm}^{pq} \\ &= \sum_{pq} C_{pq}\Phi_{\pm}^{pq} \ , \qquad (39)\end{aligned}$$

The orbital excitation operator $\hat{E}_{rs}$ when acting on $\Phi_{\pm}^{pq}$ will completely annihilate the function if s is not equal to at least one of p, q; otherwise, each occurence of ϕ_s is replaced by ϕ_r. Thus

$$\hat{E}_{rs}\Phi_{\pm}^{pq} = (1 \pm \tau_{pq})\delta_{sq}\Phi_{\pm}^{pr} \qquad (40)$$

and then

$$\hat{E}_{rs,tu}\Phi_{\pm}^{pq} = (1 \pm \tau_{pq})\delta_{sp}\delta_{uq}\Phi_{\pm}^{rt} \ , \tag{41}$$

where τ_{pq} has the effect of swapping the labels p, q in whatever follows it. Then the action of the hamiltonian operator is

$$\hat{H}\Phi_{\pm}^{pq} = (1 \pm \tau_{pq}) \left(\sum_r h_{rq}\Phi_{\pm}^{pr} + \tfrac{1}{2}\sum_{rs}(rp|qs)\Phi_{\pm}^{rs} \right) \ , \tag{42}$$

i.e.,

$$\begin{aligned} \hat{H}\Psi &= \sum_{pq} C_{pq}\hat{H}\Phi_{\pm}^{pq} \\ &= \sum_{rs} \Phi_{\pm}^{rs} \left(K(\mathbf{C})_{rs} + 2(\mathbf{hC})_{rs} \right) \ . \end{aligned} \tag{43}$$

Here, we have defined a generalized exchange matrix $\mathbf{K}(\mathbf{C})$, which for any given coefficient matrix $\mathbf{C}$ is

$$K(\mathbf{C})_{rs} = \sum_{pq} C_{pq}(rp|qs) \ . \tag{44}$$

1.5 Orbital basis sets

Calculations with complete (infinite) orbital basis sets are impossible; therefore, one immediately wants to know how to choose optimally a finite basis set such that the CI wavefunction is as close to the exact wavefunction as possible for a given number of orbitals. Insight into this problem can be gained from the two-electron example developed above. Consider the one-electron *density matrix* generated by the wavefunction, defined as

$$d_{pq} = \langle \Psi | \hat{E}_{pq} | \Psi \rangle \tag{45}$$

For the two-electron example, it is straightforward to show using (40) that

$$d_{pq} = 2\sum_s C_{sp}C_{sq} \ , \tag{46}$$

or $\mathbf{d} = 2\mathbf{C}^\dagger\mathbf{C}$.

Suppose that we now consider truncating the basis set by deleting the last (m-th) orbital to leave $m-1$ remaining functions. The overlap between the new and old wavefunctions is

$$\begin{aligned} \langle \Psi_{\rm New} | \Psi_{\rm Old} \rangle &= \langle \Psi_{\rm Old} | \Psi_{\rm Old} \rangle - 2\sum_{pqr} C_{pq}C_{rm}\langle \Phi_{pq} | \Phi_{rm} \rangle + \sum_{pq} C_{pq}C_{mm}\langle \Phi_{pq} | \phi_{mm} \rangle \\ &= 1 - 2(\mathbf{C}^\dagger\mathbf{C})_{mm} + C_{mm}^2 \end{aligned} \tag{47}$$

Ignoring the last (C_{mm}^2) term, which can be shown to be of lesser importance, we deduce that the amount that the overlap differs from unity is d_{mm}. Consider making linear transformations amongst the underlying orbitals. Of all the possible transformations, the one which minimises d_{mm} is that which brings $\mathbf{d}$ to diagonal form, with d_{mm} being the smallest eigenvalue. Such orbitals are known as *natural*

orbitals (NOs), and are of great utility in interpreting correlated many-electron wavefunctions. The trace of the density matrix is equal to the number of electrons, leading to an interpretation of the eigenvalues as occupation numbers.

In the above example, therefore, if natural orbitals are chosen, the effects of deleting the last (m-th) orbital are minimized. In other words, the CI wavefunction in $m-1$ orbitals is as good as it can be. We have thus shown that of all the possible choices of orbitals, natural orbitals offer the most compact or efficient basis set, for a two-electron system. For many-electron systems, the situation is, of course, more complicated. One can still define natural orbitals as density matrix eigenvalues, but their relationship with the wavefunction is not so transparent. For the special case of CI wavefunctions that contain up to double excitations from the Hartree-Fock determinant, then one can also construct *pair natural orbitals* (PNOs) for each pair of occupied orbitals that are excited; these PNOs do have similar properties to the two-electron NOs, and typically show a similar convergence of eigenvalues towards zero. The true NOs, however, are an average of the various PNOs, and the convergence of their spectrum and their usefulness in evaluating the correlating effect of basis functions is usually less advantageous.

In contrast to Hartree-Fock, where reasonably good wavefunctions can be obtained using a double-zeta plus polarization (DZP) basis set allowing for simple contraction and deformation of atomic orbitals, a much larger basis set is required for recovering a large fraction of the correlation energy; i.e., the sequence of NO occupation numbers is found to be rather slowly convergent. It is then not a trivial problem to decide straightaway what basis functions $\{\chi_\alpha\}$ should be used for optimum recovery of electron correlation effects. The idea of using natural orbitals to obtain basis sets is taken to the extreme in the atomic natural orbital (ANO) basis scheme[15]. Here, the basis functions are (approximate) *atomic* natural orbitals, obtained from a CI calculation on each of the molecule's constituent atoms. The idea is that the ANOs, which are near–optimum correlating functions for the atomic problem, will be good functions for describing molecular electron correlation. Within each of the atomic symmetries $(s, p, d, \dots)$, each contracted basis function is a linear combination of all the primitive gaussian functions; thus each primitive function enters in to all contractions (*general contraction*). Within the ANO scheme, there also arises the concept of sequences of basis sets, in which each basis set is derived from the previous one by the addition of the next most important atomic natural orbital. This allows for the systematic improvement of basis sets and consequent elimination of possible spurious errors arising from unbalanced choices of basis functions. For example, for most first row atoms, examination of the ANO occupation numbers identifies $[3s2p1d]$, $[4s3p2d1f]$ and $[5s4p3d2f1g]$ as good choices of contracted basis sets, whilst a set such as $[5s3p2d2f2g]$ is unbalanced, and would be inefficient in recovering electron correlation effects.

For certain applications, selection of a small or medium–sized ANO set will not necessarily result in a good basis set, and can lead to spurious results. An example is the calculation of atomic or molecular electrical polarizabilities. Here, it is vital to include diffuse basis functions, particularly of d type in the usual case that the highest atomic shell is of p type. Such basis functions do not appear in the set which is optimum for the correlation problem, and so such functions must

be included additionally, or the basis set redesigned somewhat. This case occurs to a milder degree in all molecules, where the atomic functions are polarized by their neighbours; even for SCF calculations, polarization functions are required to cover this effect, and the optimum gaussian exponents are not necessarily related to those best for correlation. Another type of calculation which presents problems for ANO sets is that where several different atomic states are involved; the classic case is in transition metal chemistry, where $d^n s^2$, $d^{n+1} s^1$ and d^{n+2} atomic states often all make significant contributions to the molecular situation. ANO bases based on each state are drastically different, particularly for the d orbitals, which are much more diffuse in d^{n+2} than in d^n; so the use of an ANO set derived from one particular atomic state can introduce an unwanted bias towards that state. A partial solution is to select functions which are eigenfunctions of the sum of the density matrices for each state[16,17,18], although caution is still needed. For general applications, a good compromise is found in the "correlation consistent" basis sets[19], which are similar to ANO sets, except that the most diffuse s and p functions are left uncontracted, and the polarization functions are simple uncontracted gaussians designed to cover both the polarization and correlation requirements. In fact, the advantage in using ANOs for the polarization functions is not that great, and the correlation consistent basis sets are usually more compact than standard ANOs for a given level of accuracy. Just as with ANOs, a systematic sequence of basis sets is defined, with members conventionally denoted cc-pVDZ, cc-pVTZ, cc-pVQZ, cc-pV5Z, etc., which for 1st row atoms comprise $3s2p1d$, $4s3p2d1f$, $5s4p3d2f1g$, $6s5p4d3f2g1h\ldots$.

1.6 Dynamical vs. Non-Dynamical Correlation

The correlation energy arising from overestimation of short-range electron repulsions in Hartree-Fock wavefunctions is usually referred to as dynamical correlation. Dynamical correlation is always reduced when a normal chemical bond (i.e., doubly occupied orbital) is broken. It is the neglect of dynamical correlation which causes the RHF equilibrium energy of F_2 to be higher than twice the RHF energy of a fluorine atom, since in F_2 there are 9 pairs of electrons, but in each F there are only 4. The effect is so pronounced for F because the molecular orbitals are considerably smaller than their atomic parents, and crowding the electrons together means there is more correlation energy. Where dynamical correlation effects are important, Hartree-Fock will therefore generally overestimate bond lengths and underestimate binding. An extreme example is that of rare-gas dimers, which are unbound at the Hartree-Fock level, but in reality are held together by dispersion, which is a manifestation of dynamic correlation.

That part of the correlation energy arising from long-range correlation effects, such as observed on molecular dissociation, is often referred to as non-dynamical (or static) correlation. Static correlation effects mean that (spin-restricted) Hartree-Fock tends to artificially overbind molecules underestimating bond lengths and overestimating vibrational frequencies. Thus the effects of dynamic and non-dynamic correlation are very often in opposition, and the partial cancellation of correlation errors enhances the value of SCF; it is often observed that, for example, use of methods which represent properly the non-dynamical correlation effects leads to

much worse agreement of computed properties with experiment than RHF.

The division between dynamical and non-dynamical correlation is difficult to define in most cases. For example, when thinking about electron correlation in a bond in a molecule, the radial and angular short-range concepts are somewhat blurred with the ideas of long-range dissociation-enabling correlation. One useful visualization is that the non-dynamical correlation is that which is recovered with the minimum CI expansion describing properly all correlation effects; in contrast, convergence of the dynamical correlation energy with increasing size of CI expansion is very slow.

When non-dynamical correlation is weak, Hartree-Fock theory already provides a qualitatively correct description of the wavefunction. Under such circumstances, which, fortunately, apply for the majority of molecules in their ground state near equilibrium geometry, one may use *single-reference* methods for representing the dynamical correlation effect. These methods build on the SCF reference determinant, typically using perturbative arguments to define classes of configurations or excitations deemed to be of most importance in constructing an approximate correlated wavefunction. For most excited states, for molecules that are close to dissociation, and for situations in which there is near electronic degeneracy, Hartree-Fock is a poor approximation. Static correlation effects often mean that there is no single Slater determinant that dominates the wavefunction, and perturbative or other approaches that assume a good single-reference starting point are doomed to failure. Under such circumstances, a viable way forward is to first deal with the static correlation problem using a CI expansion that covers all the important effects. One may then go further using this many-determinant reference as a starting point for further recovery of the dynamic correlation. Such approaches are termed *multi-reference* methods.

2 Closed-Shell single-reference methods

In this section we will discuss the most important electron correlation methods based on closed-shell Hartree-Fock reference functions. This includes Møller-Plesset perturbation theory, singles and doubles configuration interaction (CISD), and non-variational variants like the coupled-electron pair approximation (CEPA), as well as coupled cluster methods with single and double excitations (CCSD). The effect of triple excitations can be accounted for by perturbation theory, leading to CCSD(T).

From a computational point of view, it is important to minimize the logic in the code, and to formulate the theory in terms of matrix and vector operations. The most efficient operations one can perform on any kind of current hardware are matrix multiplications. This applies both to vector computers as well as to RISC workstations or even PCs. The reason for this is that on most machines the bottleneck is not the floating point operation itself, but getting the data from the memory, in particular if the quantities involved do not fit into the fast cache. By an appropriate unrolling of the three loops in a matrix multiplication one can achieve that each data element obtained from memory can be used in several floating point operations, and this this way often about 80% of the theoretical peak performance can be achieved.

For the formulation of the theory in terms of matrix multiplications it is essential to use unnormalized or even non-orthogonal configuration state functions. We start with a general discussion of the configuration spaces which are common to all methods discussed in the subsequent sections.

2.1 The first-order interacting space

According to second-order perturbation theory, the most important contributions to the correlation energy arise from configurations Φ_I which have non-zero matrix elements $\langle\Phi_I|\hat{H}|\Phi^{SCF}\rangle$, i.e., which span the *first-order interacting space* of Φ^{SCF}. In the following, the SCF wavefunction will be denoted $|0\rangle \equiv \Phi^{SCF}$. According to the Slater-Condon rules only Slater determinants can contribute which differ by at most two spin-orbitals from the Hartree-Fock determinant. The spin adapted singly and doubly excited configurations are conveniently generated by applying the excitation operators $\hat{E}_{ai}$ to the reference function

$$\Phi_i^a = \hat{E}_{ai}|0\rangle \;, \tag{48}$$

$$\Phi_{ij}^{ab} = \hat{E}_{ai}\hat{E}_{bj}|0\rangle \;, \tag{49}$$

where i, j refer to occupied orbitals in $|0\rangle$, and a, b to virtual orbitals (unoccupied in $|0\rangle$). If $|0\rangle$ is an optimized closed-shell Hartree-Fock wavefunction, the matrix elements $\langle\Phi_i^a|\hat{H}|0\rangle = 2f_{ai}$ vanish for all single replacements Φ_i^a, since the optimized orbitals satisfy the conditions $f_{ai} = 0$ (Brillouin theorem). Therefore, the first-order wavefunction is a linear combination of all doubly excited configurations Φ_{ij}^{ab}

$$\Psi^{(1)} = \frac{1}{2}\sum_{ij}\sum_{ab} T_{ab}^{ij}\Phi_{ij}^{ab} \;, \tag{50}$$

where T_{ab}^{ij} are the amplitudes. Note that the operators $\hat{E}_{ai}$ and $\hat{E}_{bj}$ commute, and therefore

$$\Phi_{ij}^{ab} = \Phi_{ji}^{ba} \;, \tag{51}$$

i.e., the configuration set used in the expansion of $\Psi^{(1)}$ is redundant. In the formulation of correlation theories it will be convenient to use this redundant set, but we must account for this by the restriction

$$T_{ab}^{ij} = T_{ba}^{ji} \;. \tag{52}$$

We will consider T_{ab}^{ij} as matrices with elements ab. Different matrices are labeled by the superscripts ij:

$$[\mathbf{T}^{ij}]_{ab} = T_{ab}^{ij} \;, \qquad \mathbf{T}^{ij} \;=\; \mathbf{T}^{ji\dagger} \;. \tag{53}$$

The matrix elements for $i > j$, all a, b and $i = j$, $a \geq b$ form the non-redundant set of amplitudes.

The definition of the doubly excited configurations in eq. (49) is most simple, but has the disadvantage that the resulting functions are non-orthogonal. Using the commutation relations (30) and the fact that zero results if an external annihilator acts on the reference function $|0\rangle$ one obtains

$$\langle\Phi_{ij}^{ab}|\Phi_{kl}^{cd}\rangle = \delta_{ac}\delta_{bd}\langle 0|\hat{E}_{ik,jl}|0\rangle + \delta_{ad}\delta_{bc}\langle 0|\hat{E}_{il,jk}|0\rangle \;, \tag{54}$$

where $\langle 0|\hat{E}_{ik,jl}|0\rangle$ are the elements of the second-order reduced density matrix of the reference function. For closed-shell Hartree-Fock reference functions one obtains explicitly

$$\begin{aligned}\langle 0|\hat{E}_{ik,jl}|0\rangle &= 4\delta_{ik}\delta_{jl} - 2\delta_{il}\delta_{jk} \ , \\ \langle \Phi_{ij}^{ab}|\Phi_{kl}^{cd}\rangle &= \delta_{ac}\delta_{bd}(4\delta_{ik}\delta_{jl} - 2\delta_{il}\delta_{jk}) + \delta_{ad}\delta_{bc}(4\delta_{il}\delta_{jk} - 2\delta_{ik}\delta_{jl}) \ . \end{aligned} \tag{55}$$

Straightforward use of these non-orthogonal configurations is in principle possible, but leads to some complications. There are two ways for simplification: in the first case a set of orthogonal configuration state functions is defined as

$$\Phi_{ijp}^{ab} = \frac{1}{2}\left(\Phi_{ij}^{ab} + p\Phi_{ij}^{ba}\right) \qquad \text{for } p = \pm 1,\ i \geq j,\ a \geq b \ , \tag{56}$$

where $p = 1$ corresponds to singlet coupling of the two external electrons, and $p = -1$ to triplet coupling. Note that these functions are not normalized; for a closed-shell reference function we have

$$\langle \Phi_{ijp}^{ab}|\Phi_{klq}^{cd}\rangle = (2-p)\delta_{pq}(\delta_{ac}\delta_{bd} + p\delta_{ad}\delta_{bc})(\delta_{ik}\delta_{jl} + p\delta_{il}\delta_{jk}) \ , \tag{57}$$

and thus the normalization factors are

$$\langle \Phi_{ijp}^{ab}|\Phi_{ijq}^{ab}\rangle = (2-p)\delta_{pq}(1 + p\delta_{ab})(1 + p\delta_{ij}) \ . \tag{58}$$

As will become clear later, for an efficient formulation of all electron correlation methods it is essential not to normalize the configurations. This was first realized in the theory of self-consistent electron pairs (SCEP) by Meyer[20], who showed that by using unnormalized configurations all terms involving the virtual orbital labels $a, b, \ldots$ can be formulated in a computationally convenient matrix form without any logic. Most importantly, this concerns the factor $(1+p\delta_{ab})$, which implies a different normalization for *diagonal* configurations ($a = b$) than for non diagonal ones ($a \neq b$). We note that in the original SCEP theory of Meyer[20] the configurations were normalized by the factors $[(2-p)(1+p\delta_{ij})]^{-1/2}$, but this leads to some unnecessary factors in the resulting equations. A similar definition is possible for multireference wavefunctions and will be used in section 5.

For single-reference methods it turns out that even simpler equations can be obtained by directly using the configurations (49) together with a set of *contravariant* configurations[21,22]

$$\tilde{\Phi}_{ij}^{ab} = \frac{1}{6}(2\Phi_{ij}^{ab} + \Phi_{ji}^{ab}) \tag{59}$$

which have the properties

$$\langle \tilde{\Phi}_{ij}^{ab}|\Phi_{kl}^{cd}\rangle = \delta_{ac}\delta_{bd}\delta_{ik}\delta_{jl} + \delta_{ad}\delta_{bc}\delta_{il}\delta_{jk} \ , \tag{60}$$

$$\langle \tilde{\Phi}_{ij}^{ab}|\Psi^{(1)}\rangle = T_{ab}^{ij} \ , \tag{61}$$

$$\langle \tilde{\Phi}_{ij}^{ab}|\hat{H}|\Psi^{(0)}\rangle = (ai|bj) \ . \tag{62}$$

The last expression is obtained by inserting the hamiltonian in second quantization (cf. eq. (34))

$$\langle \tilde{\Phi}_{ij}^{ab}|\hat{H}|\Psi^{(0)}\rangle = \frac{1}{2}\sum_{rstu}\langle \tilde{\Phi}_{ij}^{ab}|\hat{E}_{rs,tu}|\Psi^{(0)}\rangle(rs|tu) \ , \tag{63}$$

and realizing that the indices r, t must be external and match a, b, while s, u must be internal and match i, j according to eq. (60)

$$\begin{aligned}\langle\tilde{\Phi}_{ij}^{ab}|\hat{H}|\Psi^{(0)}\rangle &= \frac{1}{2}\sum_{kl}\sum_{cd}\langle\tilde{\Phi}_{ij}^{ab}|\hat{E}_{ck}\hat{E}_{dl}|\Psi^{(0)}\rangle(ck|dl) \\ &= \frac{1}{2}\sum_{kl}\sum_{cd}\langle\tilde{\Phi}_{ij}^{ab}|\Phi_{kl}^{cd}\rangle(ck|dl) \;=\; (ai|bj)\;. \end{aligned} \tag{64}$$

We can now express $\Psi^{(1)}$ either in the original basis or in the basis of contravariant functions

$$\Psi^{(1)} = \frac{1}{2}\sum_{ij}\sum_{ab} T_{ab}^{ij}\Phi_{ij}^{ab} \;=\; \sum_{ij}\sum_{ab}\tilde{T}_{ab}^{ij}\tilde{\Phi}_{ij}^{ab}\;, \tag{65}$$

which leads to

$$\tilde{T}_{ab}^{ij} = 2T_{ab}^{ij} - T_{ab}^{ji} \qquad \text{or} \qquad \tilde{\mathbf{T}}^{ij} \;=\; 2\mathbf{T}^{ij} - \mathbf{T}^{ji}\;. \tag{66}$$

The factor $\frac{1}{2}$ has been omitted in the second sum for convenience in later expressions.

For the singles we can define the contravariant space analogously, but in this case only the normalization of Φ_i^a and $\tilde{\Phi}_i^a$ differs

$$\tilde{\Phi}_i^a = \frac{1}{2}\Phi_i^a\;, \tag{67}$$

$$\tilde{t}_a^i = 2t_a^i\;. \tag{68}$$

2.2 Matrix notation

We have seen above that the amplitudes T_{ab}^{ij} for a given correlated orbital pair (ij) can be considered as a matrix $\mathbf{T}^{ij}$, and the amplitudes t_a^i of the single excitations as vectors $\mathbf{t}^i$. Unless otherwise noted, here and in the following i, j, k, l refer to occupied orbitals, a, b, c, d to virtual orbitals (unoccupied in the reference function), and p, q, r, s to any orbitals. In open-shell and MCSCF methods, t, u, v, w will denote open-shell (active) orbitals.

Similarly, it is convenient to order the two-electron integrals over two occupied and two virtual orbitals into matrices. In this case there are two types, namely Coulomb and exchange matrices

$$J_{ab}^{ij} = (ab|ij)\;, \tag{69}$$

$$K_{ab}^{ij} = (ai|bj)\;. \tag{70}$$

The labels ij refer to different matrices, and ab to their elements. Often it will be possible to write equations in matrix form, involving matrix multiplications and additions, and then bold face letters will be used for matrices, e.g., $\mathbf{J}^{ij}$ and $\mathbf{K}^{ij}$. For convenience in later expressions, we also define

$$L_{ab}^{ij} = 2K_{ab}^{ij} - K_{ba}^{ij}\;, \tag{71}$$

and the closed shell Fock matrix

$$f_{rs} = h_{rs} + \sum_i \left[2J_{rs}^{ii} - K_{rs}^{ii}\right]\;. \tag{72}$$

In the subsequent sections, the matrix **f** will only refer to the external part, i.e, the elements f_{ab}.

2.3 Second-order Møller-Plesset perturbation theory

The simplest electron correlation method to treat electron correlation is Møller-Plesset perturbation theory, which is a special variant of Rayleigh-Schrödinger perturbation theory, with the zeroth-order hamiltonian

$$\hat{H}^{(0)} = \sum_{i=1}^{N_{el}} \hat{f}(i) = \sum_{rs} \hat{E}_{rs} f_{rs} \ , \tag{73}$$

and with

$$\hat{H}^{(1)} = \hat{H} - \hat{H}^{(0)} \ , \tag{74}$$

where $\hat{f}(i)$ is the closed-shell Fock operator for electron i. For optimized orbitals the matrix elements f_{ai} vanish (Brillouin conditions), and it is then easily shown that the Hartree-Fock wavefunction $\Psi^{(0)} = \Phi^{\mathrm{SCF}}$ is an eigenfunction of $\hat{H}^{(0)}$, i.e.,

$$\hat{H}^{(0)}\Psi^{(0)} = \hat{E}^{(0)}\Psi^{(0)} \ , \tag{75}$$

$$\hat{E}^{(0)} = 2\sum_{i=1}^{m_{occ}} f_{ii} \ , \tag{76}$$

$$\hat{E}^{(0)} + \hat{E}^{(1)} = \langle\Psi^{(0)}|\hat{H}|\Psi^{(0)}\rangle \ = \ E^{\mathrm{SCF}} \ , \tag{77}$$

where E^{SCF} is the Hartree-Fock energy expectation value.

The first-order wavefunction is expanded according to eq. (50), and the amplitudes T_{ab}^{ij} are obtained by solving the first-order perturbation equations

$$\langle\tilde{\Phi}_{ij}^{ab}|\hat{H}^{(0)} - \hat{E}^{(0)}|\Psi^{(1)}\rangle + \langle\tilde{\Phi}_{ij}^{ab}|\hat{H}|\Psi^{(0)}\rangle = 0 \tag{78}$$

for all $i \geq j, ab$. Inserting eq. (50) and evaluating the matrix elements yields the linear equations

$$R_{ab}^{ij} = K_{ab}^{ij} + \sum_{c}\left(f_{ac}T_{cb}^{ij} + T_{ac}^{ij}f_{cb}\right) - \sum_{k}\left(f_{ik}T_{ab}^{kj} + T_{ab}^{ik}f_{kj}\right) \ = \ 0 \ . \tag{79}$$

For the case that *canonical* Hartree-Fock orbitals are used which obey

$$f_{ij} = \epsilon_i\delta_{ij} \ , \tag{80}$$

$$f_{ab} = \epsilon_a\delta_{ab} \ , \tag{81}$$

one obtains

$$R_{ij}^{ab} = K_{ab}^{ij} + (\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j)T_{ab}^{ij} \tag{82}$$

$$T_{ab}^{ij} = -K_{ab}^{ij}/(\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j) \ , \tag{83}$$

which is, of course, the well known MP2 expression. Using eqs. (61) and (62) the second-order energy takes the form

$$\begin{aligned}\hat{E}^{(2)} &= \langle\Psi^{(0)}|\hat{H}|\Psi^{(1)}\rangle \\ &= \sum_{ij}\sum_{ab}\langle\Psi^{(0)}|\hat{H}|\tilde{\Phi}_{ij}^{ab}\rangle\tilde{T}_{ab}^{ij} \\ &= \sum_{ij}\langle\mathbf{K}^{ij}\tilde{\mathbf{T}}^{ji}\rangle \;=\; \sum_{ij}\langle\mathbf{K}^{ij}(2\mathbf{T}^{ji}-\mathbf{T}^{ij})\rangle\;, \end{aligned} \tag{84}$$

where

$$\langle\mathbf{K}^{ij}\tilde{\mathbf{T}}^{ji}\rangle = \sum_{ab}K_{ab}^{ij}\tilde{T}_{ba}^{ji} \;=\; \sum_{ab}K_{ab}^{ij}\tilde{T}_{ab}^{ij} \tag{85}$$

$$\tag{86}$$

denotes the trace of the matrix product in the brackets.

From the above equations it is obvious that evaluating the second-order energy is trivial once the exchange integrals $K_{ab}^{ij} = (ai|bj)$ are available. These integrals are in the MO basis, and must therefore be generated from the 2-electron integrals in the AO basis by a four-index transformation

$$(ai|bj) = \sum_{\mu\nu\rho\sigma}X_{\mu a}X_{\nu b}X_{\rho i}X_{\sigma j}(\mu\rho|\nu\sigma)\;. \tag{87}$$

This transformation is most efficiently done in four steps, each being a matrix multiplication, i.e.

$$(\mu\rho|\nu j) = \sum_{\sigma}(\mu\rho|\nu\sigma)X_{\sigma j}\;, \tag{88}$$

$$(\mu i|\nu j) = \sum_{\rho}(\mu\rho|\nu j)X_{\rho i}\;, \tag{89}$$

$$(\mu i|bj) = \sum_{\nu}(\mu j|\nu i)X_{\nu b}\;, \tag{90}$$

$$(ai|bj) = \sum_{\mu}(\mu i|bj)X_{\mu a}\;. \tag{91}$$

Since the number of occupied orbitals i, j is usually much smaller than the number of basis functions, the number of transformed integrals becomes smaller in each step, and therefore the first quarter transformation step is most expensive. It requires about $\frac{1}{2}m_{val}m^4$ operations, where m is the number of basis functions and m_{val} the number of correlated orbitals. Since both m_{val} and m increase linearly with system size $\mathcal{N}$, the computational effort scales with $\mathcal{O}(\mathcal{N}^5)$. For large systems not only the computation time but also the storage of the two-electron integrals and intermediate quantities is a severe bottleneck. Chapter 6 discusses *integral-direct* transformations, in which the integrals $(\mu\rho|\nu\sigma)$ are computed on the fly whenever needed, without being ever stored on disk.

An alternative way to compute the second-order energy is to start from the *Hylleraas functional*

$$\begin{aligned} E_2 &= 2\langle\Psi^{(1)}|\hat{H}|\Psi^{(0)}\rangle + \langle\Psi^{(1)}|\hat{H}^{(0)} - \hat{E}^{(0)}|\Psi^{(1)}\rangle \\ &= 2\sum_{ij}\left[\langle\mathbf{K}^{ij}\tilde{\mathbf{T}}^{ji}\rangle + \langle\mathbf{T}^{ij}\mathbf{f}\tilde{\mathbf{T}}^{ji}\rangle - f_{ij}\sum_k\langle\mathbf{T}^{ik}\tilde{\mathbf{T}}^{kj}\rangle\right] \\ &= \sum_{ij}\left[\langle(\mathbf{K}^{ij} + \mathbf{R}^{ij})\tilde{\mathbf{T}}^{ji}\rangle\right] . \end{aligned} \tag{92}$$

Minimizing this functional with respect to the $\tilde{T}_{ab}^{ij}$ yields

$$\frac{\partial E_2}{\partial\tilde{T}_{ab}^{ij}} = 2R_{ab}^{ij} , \tag{93}$$

with the $\mathbf{V}^{ij}$ defined in eq. (79). Thus, the Hylleraas functional is stationary with respect to small variations of the T_{ab}^{ij} if the first-order perturbation equations are fulfilled, i.e. $R_{ab}^{ij} = 0$. For the corresponding amplitudes we have $E_2 = \hat{E}^{(2)}$. It is straightforward to show that in general $E_2 \geq \hat{E}^{(2)}$ for any set of trial function $\Psi^{(1)}$. The stationary property is very convenient for deriving the MP2 gradient expression and in the context of local electron correlation methods to be discussed later.

Even though we will not discuss applications of the methods in this article, it should be noted that the applicability of MP2 is restricted to cases with a sufficient large HOMO-LUMO gap. If this is not the case, the energy denominators in eq. (83) become small and the perturbation expansion diverges.

2.4 Singles and doubles configuration interaction

In singles and doubles configuration interaction (CISD) the expansion coefficients are determined variationally. Consequently, the resulting energy is an upper bound to the exact energy, but it is not size extensive or size consistent, i.e., it does not scale correctly with the number of electrons or the number of independent subsystems. Therefore, CISD usually yields poor results, and it is not recommended to be used. However, much better results can be obtained by some simple modifications of the variational conditions, leading to the coupled electron pair approximation (CEPA)[23,24] or the coupled pair functional (CPF)[25], which are approximately size consistent and yield much better results at the same computational cost as CISD.

The first matrix formulation of CISD is due to Meyer and known as SCEP theory[20] (cf. section 2.1). This method was formulated originally in the AO basis, but here we will continue to work in a basis of orthogonal MOs, which is somewhat simpler. However, we will come back to the AO formulation when discussing local electron correlation theories.

The CISD wavefunction is expanded in terms of the same configurations as used in the MP2 wavefunction, but also includes single excitations

$$\Psi^{\mathrm{CISD}} = \Phi^{\mathrm{SCF}} + \sum_{ia} t_a^i\Phi_i^a + \frac{1}{2}\sum_{ij}\sum_{ab} T_{ab}^{ij}\Phi_{ij}^{ab} . \tag{94}$$

The coefficients t_a^i, T_{ab}^{ij} are optimized variationally by minimizing the Rayleigh quotient

$$E^{\mathrm{CISD}} = \frac{\langle \Psi^{\mathrm{CISD}} | \hat{H} | \Psi^{\mathrm{CISD}} \rangle}{\langle \Psi^{\mathrm{CISD}} | \Psi^{\mathrm{CISD}} \rangle} \,. \tag{95}$$

Using eqs. (61) and (65) one finds for the norm

$$\begin{aligned} N \;=\; \langle \Psi^{\mathrm{CISD}} | \Psi^{\mathrm{CISD}} \rangle &= 1 + \sum_{ai} \tilde{t}_a^i t_a^i + \sum_{ij} \sum_{ab} \tilde{T}_{ab}^{ij} T_{ab}^{ij} \\ &= 1 + \sum_i \langle \tilde{\mathbf{t}}^{i\dagger} \mathbf{t}^i \rangle + \sum_{i \geq j} (2 - \delta_{ij}) \langle \tilde{\mathbf{T}}^{ij} \mathbf{T}^{ji} \rangle \,. \end{aligned} \tag{96}$$

Differentiating the expectation value with respect to the $\tilde{T}_{ab}^{ij}$ yields the eigenvalue equations

$$\begin{aligned} r_a^i &= \langle \tilde{\Phi}_i^a | \hat{H} - E^{\mathrm{CISD}} | \Psi^{\mathrm{CISD}} \rangle \;=\; 0 \,, \\ R_{ab}^{ij} &= \langle \tilde{\Phi}_{ij}^{ab} | \hat{H} - E^{\mathrm{CISD}} | \Psi^{\mathrm{CISD}} \rangle \;=\; 0 \,. \end{aligned} \tag{97}$$

These equations can be solved iteratively (*direct CI*). In each iteration one has to compute the residuals

$$r_a^i = v_a^i - \mathcal{E}^{\mathrm{CISD}} t_a^i \,, \tag{98}$$

$$R_{ab}^{ij} = V_{ab}^{ij} - \mathcal{E}^{\mathrm{CISD}} T_{ab}^{ij} \tag{99}$$

where

$$v_a^i = \langle \tilde{\Phi}_i^a | \hat{H} - E^{\mathrm{SCF}} | \Psi^{\mathrm{CISD}} \rangle \tag{100}$$

$$V_{ab}^{ij} = \langle \tilde{\Phi}_{ij}^{ab} | \hat{H} - E^{\mathrm{SCF}} | \Psi^{\mathrm{CISD}} \rangle \,, \tag{101}$$

and $\mathcal{E}^{\mathrm{CISD}} = E^{\mathrm{CISD}} - E^{\mathrm{SCF}}$ is the correlation energy

$$\mathcal{E}^{\mathrm{CISD}} = \frac{1}{N} \left[\sum_i (f_a^i + v_a^i) \tilde{t}_a^i + \sum_{ij} \sum_{ab} (K_{ab}^{ij} + V_{ab}^{ij}) \tilde{T}_{ab}^{ij} \right] \,. \tag{102}$$

The residuals are used to obtain an update of the CI-coefficients by simple perturbation theory:

$$\Delta t_a^i = \frac{-r_a^i}{\langle \tilde{\Phi}_i^a | \hat{H} - E^{\mathrm{CISD}} | \Phi_i^a \rangle} \,, \qquad \Delta T_{ab}^{ij} \;=\; \frac{-R_{ab}^{ij}}{\langle \tilde{\Phi}_{ij}^{ab} | \hat{H} - E^{\mathrm{CISD}} | \Phi_{ij}^{ab} \rangle} \,. \tag{103}$$

This procedure relies on the fact that the hamiltonian in the configuration basis is diagonal dominant. Convergence can be improved and guaranteed by the Davidson procedure[26].

For the sake of simplicity, we will restrict the following discussion to double excitations (CID); the inclusion of single excitations is quite straightforward and does not lead to any principle difficulties. In the CID case the matrices $\mathbf{V}^{ij}$ take the explicit form

$$V_{ab}^{ij} = K_{ab}^{ij} + K(\mathbf{T}^{ij})_{ab} + \sum_{kl} K_{kl}^{ij} T_{ab}^{kl} + G_{ab}^{ij} + G_{ba}^{ji} \tag{104}$$

with the auxiliary matrices

$$\mathbf{G}^{ij} = \mathbf{T}^{ij}\mathbf{f} - \sum_k \left[\mathbf{T}^{ik} f_{kj} + \mathbf{T}^{ik}\mathbf{J}^{kj} + (\mathbf{T}^{ik}\mathbf{J}^{kj})^\dagger - \tilde{\mathbf{T}}^{ik}\mathbf{K}^{kj}\right] . \tag{105}$$

The matrices $\mathbf{G}^{ij}$ account for the contributions of the two-electron integrals over two external and two occupied orbitals, i.e., all matrices occuring in eq. (105) are defined in the space of external orbitals only. The evaluation of all $\mathbf{G}^{ij}$ requires $2m_{val}^3$ matrix multiplications. Since each matrix multiplication involves $2m_{ext}^3$ floating point operations, the total cost scales with the sixth power of the molecular size. Note the exceedingly simple matrix form of these equations, which do not involve any complicated logic. This is solely due to the fact that unormalized and non-orthogonal configurations are used, as outlined in section 2.1. In contrast, in the early direct CISD method of Roos and Siegbahn[27], which employed orthonormalized configuration state functions, about 140 different types of matrix elements had to be distinguished.

The so called *external exchange operators* $\mathbf{K}(\mathbf{T}^{ij})$ in the second term of (104) account for all contributions of integrals over four external orbitals

$$K(\mathbf{T}^{ij})_{ab} = \sum_{cd} T_{cd}^{ij}(ac|db) . \tag{106}$$

There terms require about $m_{val}^2 m_{ext}^4$ floating point operations, and for large basis sets and not too many correlated orbitals m_{val} their evaluation dominates the total computational cost. As written in eq. (106) one would need a full integral transformation for generating the integrals $(ac|dc)$. This would not only be rather expensive ($\mathcal{O}(\mathcal{N}^5)$ operations), but also double the disk space. The transformation can be avoided by expanding the virtual MOs in the integral, yielding

$$\begin{aligned} K(\mathbf{T}^{ij})_{ab} &= \sum_{\mu\nu} X_{\mu a} X_{\nu b} \sum_{\rho\sigma} \left[\sum_{cd} X_{\rho c} T_{cd}^{ij} X_{\sigma d}\right] (\mu\rho|\sigma\nu) \\ &= \sum_{\mu\nu} X_{\mu a} X_{\nu b} \sum_{\rho\sigma} T_{\rho\sigma}^{ij} (\mu\rho|\sigma\nu) \\ &= \left[\mathbf{X}^\dagger \mathbf{K}(\mathbf{T}^{ij})_{\mathrm{AO}} \mathbf{X}\right]_{ab} . \end{aligned} \tag{107}$$

The quantities $\mathbf{T}_{\mathrm{AO}}^{ij} = \mathbf{X}\mathbf{T}_{\mathrm{MO}}^{ij}\mathbf{X}^\dagger$ are the amplitudes in the AO basis. These are precomputed and then contracted with the two-electron integrals $(\mu\rho|\sigma\nu)$, which very much resembles the calculation of the exchange terms in the Fock matrix. The resulting operators in the AO basis $\mathbf{K}(\mathbf{T}^{ij})_{\mu\nu}$ are finally backtransformed into the MO basis by the two matrix multiplications in the last line. Similar operators are also needed in coupled cluster theory (cf. section 2.5) and multirefence configuration interaction (cf. section 5).

The third-order energy in Møller-Plesset perturbation energy is obtained as

$$E^{(3)} = \sum_{ij}\sum_{ab} (K_{ab}^{ij} + V_{ab}^{ij})\tilde{T}_{ab}^{ij} , \tag{108}$$

where the V_{ab}^{ij} and $\tilde{T}_{ab}^{ij}$ are computed from the MP2 amplitudes. Note that this energy expression is similar to the expectation value, eq. (102), but without the

normalization factor. In contrast to the CID energy $E^{(3)}$ is size consistent, but not an upper bound to the exact energy.

Finally, we note that the CEPA equations[23,24] can be obtained from the CISD equations by replacing in the residual the correlation energy by individual pair energies, e.g., CEPA-2

$$R_{ab}^{ab} = V_{ab}^{ij} - \epsilon_{ij} T_{ab}^{ij} \; , \tag{109}$$

with

$$\epsilon_{ij} = (2 - \delta_{ij}) \sum_{ab} K_{ab}^{ij} \tilde{T}_{ab}^{ij} \; . \tag{110}$$

Other CEPA variants use slightly different expressions for the residual. The CEPA correlation energy is the sum of all pair energies

$$\mathcal{E}^{\rm CEPA} = \sum_{i \geq j} \epsilon_{ij} \; . \tag{111}$$

Obviously, the computational effort per iteration is virtually the same as for CISD, but the results are much better (almost as good as for CCSD(T) if singles are included).

2.5 Singles and doubles coupled-cluster

The main disadvantage of the variational configuration interaction method is the fact that it is not size consistent. This can easily be understood by considering two independent subsystems, e.g., two water molecules. The correct wavefunction for the total system AB should then be the (antisymmetrized) product of the wavefunctions of the two molecules A and B. If each of these wavefunctions contains double excitations from the SCF determinant, the total system will contain quadruple excitations, e.g.,

$$\Psi(A) = \Phi^{\rm SCF}(A) + \Psi^c(A) \; = \; [1 + \frac{1}{2} \sum_{ij}^{(A)} \sum_{ab}^{(A)} T_{ij}^{ab} \hat{E}_{ai} \hat{E}_{bj}] \Phi^{\rm SCF}(A)$$

$$\Psi(B) = \Phi^{\rm SCF}(B) + \Psi^c(B) \; = \; [1 + \frac{1}{2} \sum_{kl}^{(B)} \sum_{cd}^{(B)} T_{kl}^{cd} \hat{E}_{ck} \hat{E}_{dl}] \Phi^{\rm SCF}(B)$$

$$\begin{aligned} \Psi(AB) = \Phi^{\rm SCF}(AB) &+ \hat{\mathcal{A}}[\Phi^{\rm SCF}(A)\Psi^c(B) + \Phi^{\rm SCF}(B)\Psi^c(A)] \\ &+ \frac{1}{4} \sum_{ij}^{(A)} \sum_{ab}^{(A)} \sum_{kl}^{(B)} \sum_{cd}^{(B)} T_{ij}^{ab} T_{kl}^{cd} \hat{E}_{ai} \hat{E}_{bj} \hat{E}_{ck} \hat{E}_{dl} \Phi^{\rm SCF}(AB) \end{aligned} \tag{112}$$

where $\hat{\mathcal{A}}$ is the antisymmetrizer. It is seen that that the coefficients of the quadruple excitations $\Phi_{ijkl}^{abcd} = \hat{E}_{ai} \hat{E}_{bj} \hat{E}_{ck} \hat{E}_{dl} \Phi^{\rm SCF}(AB)$ are simple products $T_{ij}^{ab} T_{kl}^{cd}$ of the coefficients of the subsystems. However, these terms are not included in the CISD wavefunction for the dimer, and therefore the total CISD energy is not equal to the sum of the monomer energies.

In coupled-cluster theory[28,29,30] the wavefunction is generated by an exponential excitation operator

$$\Psi^{CC} = \exp(\hat{T})\Phi^{\mathrm{SCF}} \ , \tag{113}$$

where the exponential is defined by the Taylor expansion

$$\exp(\hat{T}) = 1 + \hat{T} + \frac{1}{2!}\hat{T}\hat{T} + \frac{1}{3!}\hat{T}\hat{T}\hat{T} + \dots \ . \tag{114}$$

The excitation operator $\hat{T}$ may be decomposed into single, double, and possibly higher excitation operators

$$\hat{T} = \hat{T}_1 + \hat{T}_2 + \dots \tag{115}$$

with

$$\hat{T}_1 = \sum_{ai} t_{ai}\hat{E}_{ai} \ , \tag{116}$$

$$\hat{T}_2 = \frac{1}{2}\sum_{ij}\sum_{ab} T^{ij}_{ab}\hat{E}_{ai}\hat{E}_{bj} \ , \tag{117}$$

$$\tag{118}$$

etc. Truncating the expansion after $\hat{T}_2$ yields the CCSD theory[31,21,32,22].

For two independent subsystems we can decompose $\hat{T}$ into a sum of two operators each acting only on one subsystem

$$\begin{aligned}\Psi(AB) = \exp(\hat{T}_A + \hat{T}_B)\Phi^{\mathrm{SCF}}(AB) \ &= \ \hat{\mathcal{A}}\left[\exp(\hat{T}_A)\Phi^{\mathrm{SCF}}(A)\exp(\hat{T}_B)\Phi^{\mathrm{SCF}}(B)\right] \\ &= \hat{\mathcal{A}}\left[\Psi(A)\Psi(B)\right] \ .\end{aligned} \tag{119}$$

Thus, the coupled-cluster wavefunction is size consistent as required. It implicitly contains triple, quadruple, and higher excitations, but the coefficients of these are all products of the single and double excitation amplitudes t^i_a and T^{ij}_{ab}.

Unfortunately, it is not possible to determine these amplitues variationally, since like the full CI expansion (113) includes up to N-fold excitations, which makes the evaluation of an expectation value too expensive. However, one can obtain a non-linear system of equations for the amplitudes by projecting the Schrödinger equation from the left with the contravariant configurations $\tilde{\Phi}^a_i$ and $\tilde{\Phi}^{ab}_{ij}$ as defined in section 2.1. An additional equation for the correlation energy is obtained by projecting with the reference function. This yields

$$\mathcal{E}^{\mathrm{CCSD}} = \langle 0|\hat{H}(1 + \hat{T}_1 + \hat{T}_2 + \frac{1}{2}\hat{T}_1^2)|0\rangle \tag{120}$$

$$r^i_a = \langle\ \Phi^a_i|(\hat{H} - E^{\mathrm{CCSD}})(1 + \hat{T}_1 + \hat{T}_2 + \frac{1}{2}\hat{T}_1^2 + \hat{T}_1\hat{T}_2 + \frac{1}{3!}\hat{T}_1^3)|0\rangle \ = \ 0 \tag{121}$$

$$\begin{aligned}R^{ij}_{ab} = \langle\tilde{\Phi}^{ab}_{ij}|(\hat{H} - E^{\mathrm{CCSD}})(1 + \hat{T}_1 + \hat{T}_2 + \frac{1}{2}\hat{T}_1^2 + \hat{T}_1\hat{T}_2 + \frac{1}{3!}\hat{T}_1^3 \\ + \frac{1}{2}\hat{T}_1^2\hat{T}_2 \quad + \frac{1}{2}\hat{T}_2^2 + \frac{1}{4!}\hat{T}_1^4)|0\rangle \ = \ 0 \qquad (i \geq j, \text{ all } a, b) \ .\end{aligned} \tag{122}$$

The expansions on the right-hand side terminate after the quadruple excitations since the hamiltonian can couple only configurations that differ by at most two excitations. The number of equations corresponds exactly to the number of amplitudes. Even though these equations look quite complicated, it turns out that their solution is not much more difficult than of the CISD equations. It can be shown that in the coupled-cluster case the contributions of the energy in the residual equations cancel out, as required for a size-consistent theory.

In order to exemplify the structure of the resulting equations, we will omit the single excitation operator $\hat{T}_1$ and consider only the coupled-cluster doubles (CCD) case. The full CCSD equations in a similar matrix formulation can be found in Ref. [22]. The explicit expressions for the CCD residual matrices $\mathbf{R}^{ij}$ are

$$\mathbf{R}^{ij} = \mathbf{K}^{ij} + \mathbf{K}(\mathbf{T}^{ij}) \; + \; \sum_{kl} \alpha_{ij,kl} \mathbf{T}^{kl} + \mathbf{G}^{ij} + \mathbf{G}^{ji} \; , \tag{123}$$

with

$$\mathbf{G}^{ij} = \mathbf{T}^{ij}\mathbf{X} - \sum_k \left[\beta_{ik}\mathbf{T}^{kj} - \tilde{\mathbf{T}}^{ik}\mathbf{Y}^{kj} + \frac{1}{2}\mathbf{T}^{ki}\mathbf{Z}^{kj} + (\mathbf{T}^{ki}\mathbf{Z}^{kj})^\dagger \right] \; . \tag{124}$$

The form of these equations is exactly the same as for the CID, discussed in the previous section, but there are now intermediate quantities which depend linearly on the amplitudes. In detail, the integrals K_{kl}^{ij} in the CID equations are replaced by $\alpha_{ij,kl}$, f_{ik} by β_{ik}, $\mathbf{f}$ by $\mathbf{X}$, $\mathbf{K}^{kj}$ by $\mathbf{Y}^{kj}$, and $\mathbf{J}^{kj}$ by $\mathbf{Z}^{kj}$. The explicit form of these quantites is

$$\alpha_{ij,kl} = K_{ij}^{kl} + \mathrm{tr}\left(\mathbf{T}^{ij}\mathbf{K}^{lk}\right) \; , \tag{125}$$

$$\beta_{ik} = f_{ik} + \sum_l \mathrm{tr}\left(\mathbf{T}^{il}\mathbf{L}^{lk}\right) \; , \tag{126}$$

$$\mathbf{X} \; = \mathbf{f} - \sum_{kl} \mathbf{L}^{kl}\mathbf{T}^{lk} \tag{127}$$

$$\mathbf{Y}^{kj} = \mathbf{K}^{kj} - \frac{1}{2}\mathbf{J}^{kj} + \frac{1}{4}\sum_l \mathbf{L}^{kl}\tilde{\mathbf{T}}^{lj} \; , \tag{128}$$

$$\mathbf{Z}^{kj} = \mathbf{J}^{kj} - \frac{1}{2}\sum_l \mathbf{K}^{lk}\mathbf{T}^{jl} \; . \tag{129}$$

The computational effort of the CCD differs from CID basically by the additional $2m^3$ matrix multiplications in eqs. (128) and (129), which doubles the time for evaluating the matrices $\mathbf{G}^{ij}$. However, the same external exchange operators $\mathbf{K}(\mathbf{T}^{ij})$ are needed, and therefore the difference in total time is less significant.

If singles are included, there are additional terms in the intermediates, but these require only minor computational effort. The products of singles arising from the $\hat{T}_1^2$, $\hat{T}_1^3$, and $\hat{T}_1^4$ terms in eqs. (121) and (122) can all be accounted for by defining modified amplitude matrices

$$\mathbf{C}^{ij} = \mathbf{T}^{ij} + \mathbf{t}^i\mathbf{t}^{j\,\dagger} \; , \qquad \bar{\mathbf{C}^{ij}} \; = \; \frac{1}{2}\mathbf{T}^{ij} + \mathbf{t}^i\mathbf{t}^{j\,\dagger} \; , \tag{130}$$

and then all intermediates depend only linearly on either $\mathbf{T}^{ij}$, $\mathbf{C}^{ij}$, or $\bar{\mathbf{C}}^{ij}$. The most notable difference between CISD and CCSD is that in the latter case one needs additional contractions of singles amplitudes with 3-external integrals

$$\mathbf{J}(\mathbf{E}^{ij})_{ab} = \sum_c (ab|ci)\ t_c^j\ , \tag{131}$$

$$\mathbf{K}(\mathbf{E}^{ij})_{ab} = \sum_c (ai|bc)\ t_c^j\ . \tag{132}$$

As the external exchange operators, these terms can be evaluated in two different ways. Either the 3-external integrals $(ab|ci)$ are explicitly generated, which requires a more expensive integral transformation (note, however, that the effort for the first quarter transformation is the same). Alternatively, the storage of these integrals can be avoided by computing these terms directly from the integrals in the AO basis. First, the singles amplitudes are transformed into the AO basis

$$t_\sigma^i = \sum_c X_{\sigma c}\ t_c^i\ , \tag{133}$$

then the operators are computed in the AO basis

$$\mathbf{J}(\mathbf{E}^{ij})_{\mu\nu} = \sum_\rho X_{\rho i} \sum_\sigma t_\sigma^j\ (\mu\nu|\rho\sigma)\ , \tag{134}$$

$$\mathbf{K}(\mathbf{E}^{ij})_{\mu\nu} = \sum_\rho X_{\rho i} \sum_\sigma t_\sigma^j\ (\mu\rho|\sigma\nu)\ , \tag{135}$$

and finally they are back transformed into the MO basis

$$\mathbf{J}(\mathbf{E}^{ij})_{\mathrm{MO}} = \mathbf{X}^\dagger \mathbf{J}(\mathbf{E}^{ij})_{\mathrm{AO}} \mathbf{X}\ , \tag{136}$$

$$\mathbf{K}(\mathbf{E}^{ij})_{\mathrm{MO}} = \mathbf{X}^\dagger \mathbf{K}(\mathbf{E}^{ij})_{\mathrm{AO}} \mathbf{X}\ . \tag{137}$$

This procedure, which is similar to the computation of the operators $\mathbf{J}^{kl}$ and $\mathbf{K}^{kl}$, requires about $\frac{3}{4}m^4 m_{\mathrm{occ}} + 4m^3 m_{\mathrm{occ}}^2$ additions and multiplications (m basis functions, m_{occ} correlated orbitals) rather than $\frac{3}{2}m^3 m_{\mathrm{occ}}^2$ operations if the same quantities are computed from the fully transformed two-electron integrals (the full integral transformation scales as m^5). The additional effort is, however, quite insignificant as compared to the $\frac{1}{2}m^4 m_{\mathrm{occ}}^2$ operations needed to evaluate the operators $\mathbf{K}(\mathbf{T}^{ij})$ and will therefore not introduce a bottleneck. Nevertheless, it should be noted that the three-external integrals $(ab|ci)$ are also needed for evaluating the perturbative correction for triple excitations, and then it is of course advantageous to use them also for the CCSD.

Finally, we note that the QCISD (quadratic configuration interaction) equations[33] are obtained by omitting all $\hat{T}_1^2$, $\hat{T}_1^3$, $\hat{T}_1^4$ terms and the $\hat{T}_1\hat{T}_2$ term in equation 122. The residuals then include only part of the singles terms present in the CCSD. Most notably, the operators $\mathbf{J}(\mathbf{E}^{ij})$ and $\mathbf{K}(\mathbf{E}^{ij})$ are not needed in QCISD; as in the case of CISD all contributions of three-external integrals can be absorbed into the external exchange operators by computing these with modified coefficient matrices[22]. Another variant is the *Brueckner* coupled-cluster doubles

(BCCD) theory[34,35,36,37,38,39,22]. In this case the orbitals are modified in each iteration so that at convergence all singles amplitudes vanish. This can be achieved by aborbing after each update the singles into the orbitals

$$\phi_i \leftarrow \phi_i + \sum_a t_a^i \phi_a \tag{138}$$

with subsequent symmetrical reorthonormalization of the new occupied orbitals. Furthermore, the virtual orbitals have to be Schmidt-orthogonalized to the occupied space. Then the integral transformation must be repeated, since the $\mathbf{J}^{kl}$ and $\mathbf{K}^{kl}$ change. The Brueckner theory has some theoretical advantages. In particular, the resulting wavefunction is less sensitive to symmetry breaking problems than the CCSD wavefunction on the basis of canonical Hartree-Fock orbitals.

2.6 Computational aspects

As already pointed out, the matrix formulation with a minimum amount of logic is one of the prerequisites for an efficient CISD or CCSD program. Often this can be exploited to the best possible extent by using highly optimzed routines for matrix multiplication (e.g, `dgemm`), which are available in BLAS (basic linear algebra subroutines) libraries on many platforms. These routines also allow to transpose one or both of the two matrices to be multiplied on the fly, without the need to precompute and store the transposed matrix. This is often useful, since the amplitudes $\mathbf{T}^{ij}$ are stored only for $i \geq j$, and $\mathbf{T}^{ji}$ is the transpose of $\mathbf{T}^{ij}$. The same holds for the operators $\mathbf{K}^{kl} = \mathbf{K}^{lk\dagger}$.

It is equally important to think carefully about memory and I/O usage. The number of amplitudes $\mathbf{T}^{ij}$, as well as the number of transformed integrals $\mathbf{J}^{kl}$, $\mathbf{K}^{kl}$ scale with the fourth power of the molecular size, and in large calculations it will often not be possible to keep all these quantities simultaneously in high speed memory. One can then use *paging algorithms*, which read blocks of data from disk as required. The algorithm should therefore be optimized so that for a given amount of available memory the I/O is minimized.

As a first example consider the evaluation of the matrices $\mathbf{G}^{ij}$ in the CID case. The $\mathbf{G}^{ij}$ do not need to be stored but their contribution can be immediately added to the residuals $\mathbf{R}^{ij}$. If the outer two loops run over j and k, one $\mathbf{J}^{kj}$ and one $\mathbf{K}^{kj}$ at a time need to be in memory and have to be read just once for a given kj. The simplest algorithm would then assume that all $\mathbf{R}^{ij}$ and $\mathbf{T}^{ik}$ can be kept in memory. Should this not be possible, one could split them into batches. For instance, if k is the outermost loop, one could read in this loop all $\mathbf{T}^{ik}$ for a fixed k; if still not all $\mathbf{R}^{ij}$ fit into memory, one could treat the largest possible subsets of them together. In this case, one would have to read the $\mathbf{J}^{kj}$, $\mathbf{K}^{kj}$, and $\mathbf{T}^{ik}$ for each batch of $\mathbf{R}^{ij}$. Reading all the $\mathbf{J}^{kj}$ and $\mathbf{K}^{kj}$ for each batch of $\mathbf{R}^{ij}$ could be avoided if each batch would comprise only a subset of j.

The situation is more complicated in the coupled cluster case, since then one has to evaluate the intermediates $\mathbf{Y}^{kj}$ and $\mathbf{Z}^{kj}$ instead of simply reading the $\mathbf{J}^{kj}$ and $\mathbf{K}^{kj}$. This requires all operators $\mathbf{K}^{kl}$ for a fixed k and all $\mathbf{T}^{lj}$ for fixed j. Thus, the simplest algorithm requires to keep all $\mathbf{R}^{ij}$ and $\mathbf{T}^{ij}$ together with all $\mathbf{K}^{kj}$ for a fixed k in memory. A simple paging over the $\mathbf{R}^{ij}$ and/or $\mathbf{T}^{ij}$ as in the CI case

is not possible, since this would involve repeated calculation of the intermediate quantities. It would be possible, however, first to evaluate the the $\mathbf{Y}^{kj}$ and $\mathbf{Z}^{kj}$, using a similar paging algorithm as in the CI case, and store these on disk. The $\mathbf{R}^{ij}$ are then computed in a second stage, exactly as in the CI case, but instead of the $\mathbf{J}^{kj}$ and $\mathbf{K}^{kj}$ one would read $\mathbf{Y}^{kj}$ and $\mathbf{Z}^{kj}$.

The computation time and memory requirements can be much reduced if molecular symmetry is exploited, which is easy as long as only one-dimensional irreducible representations are present, i.e. D_{2h} and subgroups. If symmetry adapted molecular orbitals are used, all matrices are blocked. The block structure of a given matrix T_{ab}^{ij} is determined by the product symmetry of the orbitals i and j, which must be the same as the product symmetry of a and b. The same holds for the $\mathbf{R}^{ij}$, $\mathbf{J}^{ij}$, and $\mathbf{K}^{ij}$. Of course, only the non-zero blocks are stored, and since each symmetry block can have a different dimension, the matrices are stored in one-dimensional arrays; block dimensions and offsets are precomputed and kept in memory. It is then convenient to have a set of subroutines for operations like matrix multiplications, matrix traces, outer products etc., which handle all the symmetry blocking internally. Thus, the rest of the program requires only a minimum amount of the symmetry information, and stays most readable and easy to debug.

2.7 Triple excitations

The accuracy of coupled cluster calculations with single and double excitations (CCSD) can be significantly improved by subsequently computing the effects of higher order excitations through Rayleigh-Schrödinger perturbation theory (RSPT) based on the Fock (Møller-Plesset) hamiltonian and the computed CCSD amplitudes of single and double excitations[40,33,41]. The most important such correction is that which is linear in triple excitations, since its inclusion gives an energy expression which is consistent with the exact solution of Schrödinger's equation up to fourth order[41,42,43,44]. The most widely used ansatz of this type, usually denoted CCSD(T)[41], is also consistent with many of the fifth order terms, and includes much of the sixth and higher order energies as well[45,46], provided that the reference wavefunction is a true variational solution of the Hartree-Fock equations. This analysis takes into account the fact that terms such as T_1T_2 present in the CCSD expansion already partially includes the effects of triple excitations.

The evaluation of the triples (T) correction requires terms like

$$W_{abc}^{ijk} = \sum_d (bd|ck)T_{ad}^{ij} - \sum_m (mj|ck)T_{ab}^{im} + \text{ permutations.} \tag{139}$$

The first terms scales with $m_{val}^3 m_{ext}^4$, the second with $m_{val}^4 m_{ext}^3$, where m_{val} and m_{ext} are the number of correlated and virtual (external) orbitals, respectively. Thus, the computational cost increases with $\mathcal{O}(\mathcal{N}^7)$, where $\mathcal{N}$ is a measure of the molecular size. In most cases the calculation of the triples correction is therefore much more expensive than the CCSD calculation itself, and the applicability is limited to quite small molecules. The elapsed time (not the cost!) can be reduced by parallelization of the code, but is should be noted that this does not substantially increase the molecular size that can be handled. Doubling the molecular size increases the time by a factor of 128, and therefore even the largest parallel computers

Table 1. CPU times[a] of coupled cluster calculations for glycine peptides[b]

Program	$(Gly)_1$	$(Gly)_2$	$(Gly)_3$
Basis functions	95	166	237
Transformation[c]	10	180	1471
CCSD (11 iterations)	312	7453	62741
Triples (T) correction	520	21081	220486

a) In seconds on Sun Enterprise 3500, Ultrasparc 336 MHZ processor
b) Using C_s symmetry
c) Partial transformation to generate two-external integrals $\mathbf{J}^{kl}$, $\mathbf{K}^{kl}$ and the three-external integrals $(ab|ci)$.

do not help much further. The dramatic increase of CPU time with molecular size is demonstrated in Table 1 for some glycine peptides, $(Gly)_n \equiv HO[C(O)CH_2NH]_nH$, using the correlation consistent double zeta basis set (cc-pVDZ) of Dunning[19]. The increase of the CPU times is close to the expected theoretical factors. It is easily estimated that the evaluation of the triples correction for the next larger peptide $(Gly)_4$ would already take about three weeks of CPU time. Another bottleneck of the triples calculation is the storage of the integrals $(ab|ci)$ over three external and one occupied orbitals, which must be stored on disk. Since these integrals have less permutational symmetry than the integrals in the AO basis, and the molecular orbitals are more diffuse than the basis functions, the number of significant integrals may even be larger than the number of AO integrals.

The cc-pVDZ basis set used in these calculations is too small for obtaining reliable results. Table 2 shows the dependence of the CPU times on the basis set for closed-shell coupled-cluster calculations on another molecule, p-dimethylbenzene C_8H_{10}, performed in C_s symmetry on a medium workstation. It is seen that increasing the basis set by about a factor of 1.6 increases the CPU times by a factor of 8-12, as expected from the quartic dependence. The larger calculation does not even include f-functions on the carbon atoms, as would be required for accurate results. The computation time is strongly dominated by the triples correction, while the differences of the various methods are quite small. Clearly, the treatment of molecules of this size is about the maximum what can be done in a reasonable time, which demonstrates the limitations of the conventional coupled cluster methods. Even the fastest current workstations or supercomputers are only about a factor of 3-4 faster, and do not much extend the range of applicability. The strong dependence of the computer time on the molecular size can be dramatically reduced using local correlation methods, as will be discussed in section 4. In particular, as will be demonstrated in section 4.3, the evaluation of an approximate local triples corrections no longer dominates the calculation, but takes only a small amount of the total time.

Table 2. CPU times[a] of coupled cluster calculations for C_8H_{10} with different basis sets

Program	cc-pVDZ[b]	cc-pVTZ(d/p)[c]
Transformation[d]	35	318
CCSD/iteration	374	2313
QCISD/iteration	360	2180
BCCD/iteration	399	2520
Transformation[e]	119	1443
Triples (T) correction	9059	122515

a) In seconds on HP J282, PA8000/180MHZ processor
b) 162 basis functions ($114a', 48a''$)
c) 274 basis functions ($188a', 86a''$)
d) Partial transformation to generate the two-external integrals $\mathbf{J}^{kl}$, $\mathbf{K}^{kl}$
e) Partial transformation to generate two-external integrals $\mathbf{J}^{kl}$, $\mathbf{K}^{kl}$ and the three-external integrals $(ab|ci)$

3 Open-shell single-reference methods

The coupled-cluster treatment of open-shell systems is more complicated that the closed shell case since additional types of orbitals and excitations occur. First of all, it is possible to use either a spin-unrestricted (UHF) or a spin-restricted (RHF) Hartree-Fock wavefunction as a reference. In the UHF case the α and β spin orbitals are optimized independently, which leads to a wavefunction that is not an eigenfunction of the total spin operator $\hat{S}^2$. It is well known that the problems associated with the spin-contamination of the UHF wavefunction can become magnified when electron correlation effects are introduced[1], in particular in second-order perturbation theory (UMP2). It is therefore more desirable to use RHF orbitals.

The second difficulty is the definition of the excitation operators used in coupled-cluster treatments. It turns out that a fully spin-adapted treatment based on an RHF reference function and the spin-free excitation operators $\hat{E}_{rs}$ is very complicated. Is is much easier to use spin-orbital excitation operators $\hat{e}_{ai}$, which replace a spin-orbital ψ_i by another spin orbital ψ_a with the same spin. However, then the correlated wavefunction is not spin-adapted, even if an RHF reference functions is used. This problem already arises in the linear configuration interaction theory if the first order interacting space, spanned by the functions $\hat{e}_{ai}\hat{e}_{bj}|\Psi_0\rangle$, is used as a basis; this is due to the fact that for high-spin open shell cases this space does not include all possible Slater determinants of given M_S which arise from a particular occupancy of spatial orbitals. For instance, in a three electron case with reference function $|\phi_1^\alpha\phi_1^\beta\phi_2^\alpha|$, the determinant $|\phi_a^\alpha\phi_b^\alpha\phi_2^\beta|$ is a triple excitation and not included in the first order interacting space. This function would be necessary, however, to generate one of the two possible doublet spin eigenfunctions together with the determinants $|\phi_a^\beta\phi_b^\alpha\phi_2^\alpha|$ and $|\phi_a^\alpha\phi_b^\beta\phi_2^\alpha|$. A quartet spin contamination arises if the

latter two Slater determinants have coefficients of different magnitude. Thus, the RHF-UCISD and RHF-UCCSD theories based on spin-orbital single and double excitations are not spin adapted.

As will be shown in Section 3.2, the spin contamination in the linear UCISD wavefunction can be quite easily removed by applying appropriate projection operators to the UCISD residual vector. The same projection can be used to remove the spin contamination from the linear terms of the CCSD wavefunction. But even then, the presence of higher powers of $\hat{T}$ in the CCSD can introduce a spin contamination in a non–trivial way. Fortunately, this effect is usually very small. The partial spin adpation (PSA-CCSD) of only the linear terms has a number of advantages: the number of independent parameters (amplitudes) is minimized and corresponds exactly to the first-order interacting space; also spin contamination effects are minimized, though not entirely removed. In an optimum implementation, the computational cost of the PSA-CCSD should be approximately the same as for a closed shell calculation with the same number of correlated orbitals.

3.1 Spin-unrestricted coupled-cluster theory (UCCSD)

We will first consider the spin unrestricted coupled cluster (UCCSD) for the case that the reference function is a high-spin RHF Slater determinant with m_{closed} doubly occupied and m_{open} singly occupied orbitals; high spin means that all open-shell electrons have α spin. The UCCSD wavefunction is obtained using the following cluster operator $\hat{T} = \hat{T}_1 + \hat{T}_2$ in the exponential ansatz (113)

$$\begin{aligned}\hat{T} = &\sum_{ia} (\tilde{t}^i_a \hat{e}^\alpha_{ai} + \bar{t}^i_a \hat{e}^\beta_{ai}) + \sum_{it} \bar{t}^i_t \hat{e}^\beta_{ti} + \sum_{ta} \tilde{t}^t_a \hat{e}^\alpha_{at} + \sum_{ij}\sum_{ab} (\tilde{T}^{ij}_{ab} \hat{e}^\alpha_{ai} \hat{e}^\alpha_{bj} + \bar{T}^{ij}_{ab} \hat{e}^\beta_{ai} \hat{e}^\beta_{bj}) \\ &+ \sum_{ij}\sum_{ab} T^{ij}_{ab} \hat{e}^\alpha_{ai} \hat{e}^\beta_{bj} + \sum_{ij}\sum_{at} T^{ij}_{at} \hat{e}^\alpha_{ai} \hat{e}^\beta_{tj} + \sum_{ij}\sum_{tu} \bar{T}^{tu}_{ab} \hat{e}^\beta_{ti} \hat{e}^\beta_{uj} \\ &+ \sum_{tj}\sum_{ab} T^{tj}_{ab} \hat{e}^\alpha_{at} \hat{e}^\beta_{bj} + \sum_{tj}\sum_{au} T^{tj}_{au} \hat{e}^\alpha_{at} \hat{e}^\beta_{uj} + \sum_{tu}\sum_{ab} \tilde{T}^{tu}_{ab} \hat{e}^\alpha_{at} \hat{e}^\alpha_{bu} \ , \end{aligned} \tag{140}$$

where $\hat{e}^\sigma_{ai} = \hat{\eta}^{\sigma\dagger}_a \hat{\eta}^\sigma_i$ are the usual spin-orbital excitation operators. If applied to a Slater determinant, $\hat{e}^\sigma_{ai}$ replaces spin orbital ϕ^σ_i by ϕ^σ_a; $\sigma = \{\alpha, \beta\}$ denotes the spin. Here and in the following, the indices i, j refer to closed-shell orbitals, t, u to open-shell orbitals, and a, b to virtual orbitals. For each orbital pair (ij), there are three sets of amplitudes, namely those for pure α or β-spin excitations $\tilde{T}^{ij}_{ab}$ and $\bar{T}^{ij}_{ab}$, respectively, and those for mixed α, β excitations T^{ij}_{ab}. In total, there are about three times as many amplitudes as in the closed-shell case. The corresponding cluster amplitudes are obtained by solving a non-linear set of equations obtained by projecting the Schrödinger equation on the left with $\Psi^{\rm RHF} \equiv |0\rangle$, $\hat{e}^\sigma_{ai}|0\rangle$, $\hat{e}^\sigma_{ai}\hat{e}^{\sigma'}_{bj}|0\rangle$ etc., as in eqs. (120) - (122). The resulting explicit equations can be found in Ref. [47]. They have a very similar matrix structure as the closed shell equations discussed in the previous sections and will not be further discussed here. It should be noted, however, that there are three times as many equations as in the closed shell case, and the total computational effort is about three times larger.

3.2 Partially spin-resticted coupled-cluster theory (RCCSD)

In fully spin coupled theory[48], it is recognized that the hamiltonian operator is spin free, and therefore the excitation operators used in the previous section may be replaced by the smaller set $\hat{E}_{ai}, \hat{E}_{at}, \hat{E}_{ti}$ and their products, where again $t, u, \ldots$ are used to denote orbitals lying in the singly occupied space, while $i, j, \ldots$ denote true closed shell orbitals, and $a, b, \ldots$ external orbitals. A simpler theory[47,49,50,51], including some but not all of the spin coupling, may be obtained by using the operators $\hat{E}_{ai}, \hat{e}^{\alpha}_{at}, \hat{e}^{\beta}_{ti}$ and their products; because the orbitals ϕ^{α}_{t} are occupied, and ϕ^{β}_{t} are unoccupied in Ψ_0, the wave function is then be spin adapted for a CISD configuration expansion, which is linear in these operators. In the non-linear CCSD case products of these operators can still give a spin-contaminated contribution to the wave function. This ansatz is denoted "partially spin adapted" CCSD (PSA-CCSD). It has the advantage that the complications occuring through the spin adaption are minimized, while most of the spin-contamination is removed.

A slight complication arises for the so called *semi-internal* configurations generated by the operators $\hat{e}^{\alpha}_{at}\hat{e}^{\beta}_{ti}$, which have the same orbital occupancy as the single excitations $\hat{E}_{ai}$ but a different spin contribution. It is easily seen that $\hat{e}^{\alpha}_{at}\hat{e}^{\beta}_{ti}|0\rangle$ is not a spin eigenfunction; a correct spin eigenfunction is generated by the operator $\hat{e}^{\alpha}_{at}\hat{e}^{\beta}_{ti} - \frac{1}{2}\hat{e}^{\alpha}_{ai} + \frac{1}{2}\hat{e}^{\beta}_{ai}$. In fact, analysis of the action of the hamiltonian operator on the RHF reference function shows that this operator together with $\hat{E}_{ai}$ generates the two possible spin eigenfunctions that contribute to the first-order interacting space. The cluster operator can now be written as

$$\begin{aligned}\hat{T} = &\sum_{ia} (\tilde{t}^{i}_{a}\hat{e}^{\alpha}_{ai} + \bar{t}^{i}_{a}\hat{e}^{\beta}_{ai}) + \sum_{it} \bar{t}^{i}_{t}\hat{e}^{\beta}_{ti} + \sum_{ta} \tilde{t}^{t}_{a}\hat{e}^{\alpha}_{at} \\ &+ \sum_{ij}\sum_{ab} T^{ij}_{ab}\hat{E}_{ai}\hat{E}_{bj} + \sum_{ij}\sum_{at} T^{ij}_{at}\hat{E}_{ai}\hat{e}^{\beta}_{tj} + \sum_{tj}\sum_{ab} T^{tj}_{ab}\hat{e}^{\alpha}_{at}\hat{E}_{bj} \\ &+ \sum_{tj}\sum_{au} T^{tj}_{au}\hat{e}^{\alpha}_{at}\hat{e}^{\beta}_{uj} + \sum_{tu}\sum_{ab} T^{tu}_{ab}\hat{e}^{\alpha}_{at}\hat{e}^{\alpha}_{bu} \,, \end{aligned} \tag{141}$$

with the restrictions

$$\tilde{t}^{i}_{a} = t^{i}_{a} - \frac{1}{2}\sum_{t} T^{ti}_{at} \,, \tag{142}$$

$$\bar{t}^{i}_{a} = t^{i}_{a} + \frac{1}{2}\sum_{t} T^{ti}_{at} \,, \tag{143}$$

which account for the fact that there are only two independent spin eigenfunctions for the orbital configurations $\ldots\phi_i\phi_t\phi_a$, as discussed above. Equating the operator $\hat{T}$ with the spin-unrestricted operator in eq. (140) yields the following relations between the amplitudes

$$\tilde{T}^{tu}_{ab} = T^{tu}_{ab} \,, \tag{144}$$

$$\bar{T}^{ij}_{tu} = T^{ij}_{tu} \,, \tag{145}$$

$$\tilde{T}^{pj}_{ab} = T^{pj}_{ab} - T^{pj}_{ba} \,, \tag{146}$$

$$\bar{T}^{ij}_{ar} = T^{ij}_{ar} - T^{ji}_{ar} \,, \tag{147}$$

where p, q refer to all occupied orbitals (closed + open), and rs to all openshell + virtual orbitals. Setting further $t_t^i = \frac{1}{2}\bar{t}_t^i$ and $t_a^t = \frac{1}{2}\tilde{t}_a^t$ we obtain a unique set of amplitudes T_{rs}^{pq} and t_r^p to be solved for. The number of independent parameters is then exactly the same as in a fully spin adapted formulation and about three times smaller than in the spin-unrestricted case. The corresponding minimal set of coupled equations can be obtained by projecting the Schrödinger equation onto the set of functions generated the individual excitation operators in the cluster operator to the reference function. Since the configuration generated in this way are non-orthonormal, simpler equations can again be derived by projecting the Schrödinger equation with the equivalent set of contravariant configurations. For details refer to Ref. [47].

The simplest possibility to solve the PSA-CCSD equations is to compute the UCCSD residuals, and then to form appropriate linear combinations of the different spin components to generate the spin-restricted residuals as needed for updating the amplitudes. Finally, the UCCSD amplitudes can be generated from the PSA-CCSD ones using eqs. (144–147). Of course, this procedure does not save any computer time relative to the UCCSD, but it requires only a minor modification of an existing UCCSD program to perform the spin projection.

4 Linear scaling local correlation methods

As pointed out in the previous sections, the computational cost of conventional electron correlation methods like MP2 or CCSD(T) increases dramatically with the size of the system. The steep scaling mainly originates from the delocalized character of the canonical MO basis. This leads to a quadratic increase of the number of amplitudes used for correlating a given electron pair, and a quartic increase of the total number of parameters. The increase of the CPU time with molecular is even steeper, being $\mathcal{O}(\mathcal{N}^7)$ for the best method of choice, which is usually CCSD(T).

From a physical point of view, however, there should be no need to correlate all electrons in an extended molecular system: dynamic electron correlation in non-metallic systems is a short-range effect with an asymptotic distance dependence of $\propto r^{-6}$ (dispersion energy), and thus the high-order dependence of the computational cost with the number of electrons of the system is just an artifact of the canonical orthogonal basis, in which the diverse correlation methods have traditionally been formulated. One natural way to circumvent this problem is to to use *local* orbitals to span the occupied and virtual spaces. Such *local correlation methods* have been proposed by several authors. Some recent papers which also summarize previous work can be found Refs. [52,53,54,55,56].

Particularly successful has been the local correlation method originally proposed by Pulay[57], which was first implemented by Saebø and Pulay for Møller-Plesset perturbation theory up to fourth order (LMP2 - LMP4(SDQ) without triple excitations) and the coupled-electron pair approximation (CEPA)[58,59]. Later it was generalized to full local CCSD by Hampel and Werner [53]. While in the early work of Saebø and Pulay[58,59] it could already be shown that only 1-2% of the correlation energy (relative to a conventional calculation with the same basis set) is lost

by the local approximation, it was not yet possible at that time to demonstrate that the scaling of the computational cost can actually be reduced, and that larger systems than with conventional methods can be treated. Significant progress in this direction was only made during the last few years when the local correlation methods were combined with newly developed integral-direct techniques[60], which fully expoit the possibilities for integral screening. Within such a framework, it has been possible to develop $\mathcal{O}(\mathcal{N})$ algorithms (asymptotic linear scaling of all computational ressources, i.e. CPU time, memory and disk space with molecular size) for local MP2 [54], local CCSD [61] and even for local connected triples correction (T) [62].

In the local correlation methods the occupied space is usually spanned by *localized molecular orbitals* (LMOs), which are obtained from the occupied canonical orbitals of a preceeding SCF calculation by virtue of a unitary localization procedure [63,64,65], which maintains the orthogonality the occupied SCF orbitals[a]

$$|\phi_k^{\rm LOC}\rangle = \sum_i |\phi_i^{\rm CAN}\rangle W_{ik} \qquad \text{with} \qquad \mathbf{W}\mathbf{W}^\dagger = \mathbf{1}\ . \tag{148}$$

The corresponding MO coefficient matrices are related similarly

$$\mathbf{L} = \mathbf{X}_{\rm OCC}\mathbf{W}\ . \tag{149}$$

(If core orbitals are not correlated, the localization should be restriced to the subspace of correlated valence orbitals.) The idea of Pulay was to abandon the orthogonality of the virtual orbitals, and to use a basis of functions which resemble the atomic orbitals (AOs) as much as possible. Obviously, the AOs are optimally localized, but since they are not orthogonal on the occupied orbitals one cannot use them straightaway. The strong orthogonality between the occupied and virtual spaces must be retained, since otherwise excitations would violate the Pauli exclusion principle and the theory would become very complicated. The orthogonality to the occupied space can be enforced by applying a projection operator $(1 - \sum_i |\phi_i\rangle\langle\phi_i|)$ to the AOs, yielding projected atomic orbitals (PAOs)

$$\begin{aligned} |\tilde{\chi}_r\rangle &= (1 - \sum_{i=1}^{m_{occ}} |\phi_i\rangle\langle\phi_i|)|\chi_r\rangle \\ &= \sum_\mu |\chi_\mu\rangle P_{\mu r} \end{aligned} \tag{150}$$

with

$$\mathbf{P} = \mathbf{1} - \mathbf{L}\mathbf{L}^\dagger\mathbf{S} \ = \ \mathbf{1} - \mathbf{X}_{\rm OCC}\mathbf{X}_{\rm OCC}^\dagger\mathbf{S} \ = \ \mathbf{X}_{\rm VIRT}\mathbf{X}_{\rm VIRT}^\dagger\mathbf{S}. \tag{151}$$

Here, $\mathbf{X}_{\rm OCC}$ and $\mathbf{X}_{\rm VIRT}$ denote the rectangular submatrices of the MO coefficient matrix $\mathbf{X}$ for the occupied and virtual (external) canonical orbitals, respectively, i.e.,

$$(\mathbf{X}_{\rm VIRT}\mathbf{X}_{\rm VIRT}^\dagger)_{\mu\nu} = \sum_a X_{\mu a}X_{\nu a}\ , \tag{152}$$

[a] Recently, it has also been proposed to use non-orthogonal basis functions to span the occupied space[66,67], but the computational efficiency of this approach has not yet been proven.

and the last equality in Eq. (151) follows from the orthonormality condition

$$(\mathbf{X}_{\mathrm{OCC}}\mathbf{X}_{\mathrm{OCC}}^{\dagger} + \mathbf{X}_{\mathrm{VIRT}}\mathbf{X}_{\mathrm{VIRT}}^{\dagger})\mathbf{S} = \mathbf{1} \; . \tag{153}$$

The PAOs are orthogonal to all occupied orbitals

$$\langle \tilde{\chi}_r | \phi_i^{\mathrm{LOC}} \rangle = (\mathbf{P}^{\dagger}\mathbf{S}\mathbf{L})_{ri} \; = \; 0 \tag{154}$$

but non-orthogonal among themselves

$$\langle \tilde{\chi}_r | \tilde{\chi}_s \rangle = (\mathbf{P}^{\dagger}\mathbf{S}\mathbf{P})_{rs} \; = \; (\mathbf{S}\mathbf{X}_{\mathrm{VIRT}}\mathbf{X}_{\mathrm{VIRT}}^{\dagger}\mathbf{S})_{rs} \; . \tag{155}$$

For non-metallic systems the PAOs are intrinsically localized, though less well than the unprojected AOs. Due to the projection the full set of PAOs is linearly dependent, but these linear dependencies can be removed at a later stage.

After having introduced local functions to span both the occupied and the virtual spaces, it is possible to *truncate* the expansion of the wavefunction in a physically reasonable way. First, one assigns to each localized orbital ϕ_i^{LOC} an orbital domain $[i]$ which contains all AOs needed to approximate the orbital ϕ_i^{LOC} with a prescribed accuracy. In practice, always all AOs at a given atom are treated together, and as many atoms are added as required. The order in which atoms are added is determined by gross atomic Mulliken charges. The corresponding orbital domain in the virtual space is spanned by the PAOs generated by applying the projector to the selected AOs. The PAOs in domain $[i]$ are then all spatially close to the localized orbital ϕ_i^{LOC}. This selection procedure can be performed fully automatically as described in Ref. [68].

The first approximation to the correlated wavefunction is now that single excitations from orbital ϕ_i^{LOC} are resticted to PAOs in the domain $[i]$, while double excitations from a pair of occupied LMOs i and j are restricted to a subset $[ij]$ of PAOs. The *pair domain* $[ij]$ is simply the *union* of the two orbital domains $[i]$ and $[j]$. The immediate consequence of these truncations is that for a given pair ij the number of amplitudes T_{rs}^{ij}, $rs \in [ij]$ no longer increases quadratically with increasing molecular size, but instead becomes *independent* of molecular size.

The second approximation is to introduce a *hierarchical treatment* of different pairs based on the interorbital distance R_{ij} between two LMOs i and j. R_{ij} is defined as the shortest distance between any centre included in the orbital domain $[i]$ and any centre in the domain $[j]$. We distinguish *strong*, *weak*, *distant*, and *very distant* pairs. The strong pairs have at least one atom in common and usually account for about 95% of the correlation energy. These pairs are treated at highest level, e.g., CCSD. Weak pairs are those for which the minimum distance is smaller than typcially 8 bohr. These pairs can be treated at lower level, e.g., MP2. Distant pairs ($8 \leq R_{ij} \leq 15$ bohr) are also treated by MP2, but the required two-electron integrals can be approximated by a multipole expansion[69], which reduces the cost for the integral transformation (see section 6.3). Finally, the very distant pairs ($R_{ij} > 15$ bohr) contribute to the correlation energy only by a few micro hartree and can therefore be neglected. The important point to notice is now that the number of strong, weak, and distant pairs all scale linearly with size. Only the number of very distant pairs, which are neglected, scales quadratically. This is demonstrated in Fig. 4 for linear chains of glycine peptides, $(\mathrm{Gly})_n \equiv \mathrm{HO[C(O)CH_2NH]}_n\mathrm{H}$, Thus, the

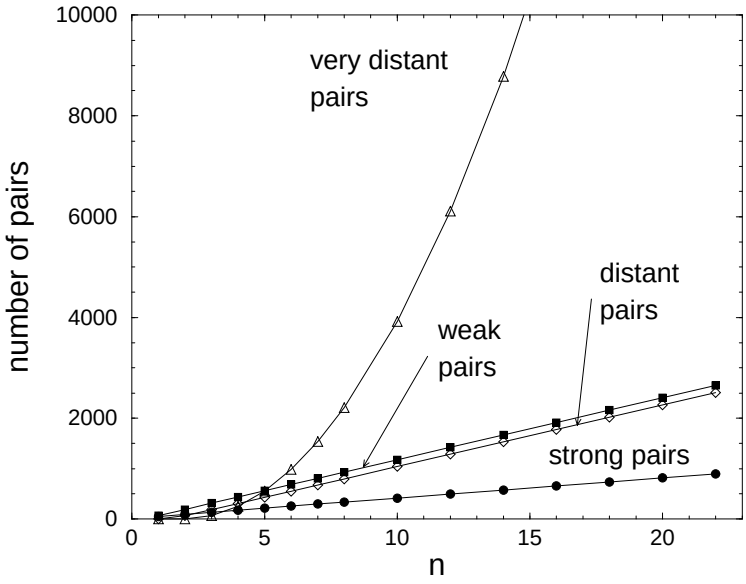


Figure 4. Number of pairs for a chain of Glycine peptides $(\mathrm{Gly})_n$ as function of the chain length.

total number of amplitudes on which the wavefunction depends scales only linearly with molecular size. This forms the basis for the development of electron correlation methods with linear cost scaling. Furthermore, the number of strong pairs remains quite modest, which is very important for an efficient CCSD algorithm (cf. section 4.2).

4.1 Local MP2

In the local LMO/PAO basis, the first-order wave function takes the form

$$|\Psi^{(1)}\rangle = \frac{1}{2} \sum_{ij \in P} \sum_{rs \in [ij]} T_{rs}^{ij} |\Phi_{ij}^{rs}\rangle \qquad \text{with } T_{rs}^{ij} = T_{sr}^{ji} \,, \tag{156}$$

where P represents the truncated pair list and it is implicitly assumed that the pair domains $[ij]$ are defined as described above. The configurations $|\Phi_{ij}^{rs}\rangle$ are defined as in eq. (49), but now the virtual labels r, s refer to the non-orthogonal PAOs. Note that the commutation relations of the excitation operators involving non-orthogonal orbitals are different and depend on overlap matrix elements.

In order to derive the LMP2 equations in the non-orthogonal basis of PAOs we first consider the transformation properties of the operators and amplitudes. The projected orbitals can be expressed in the basis of virtual orbitals as

$$\mathbf{P} = \mathbf{X}_{\mathrm{VIRT}} [\mathbf{X}_{\mathrm{VIRT}}^{\dagger} \mathbf{S}] \;=\; \mathbf{X}_{\mathrm{VIRT}} \mathbf{V} \,, \tag{157}$$

and therefore the MP2 residual given in eq. (79) for a basis of orthogonal MOs can be transformed to the PAO basis as

$$\mathbf{R}^{ij}_{\mathrm{PAO}} = \mathbf{V}^\dagger \mathbf{R}^{ij}_{\mathrm{MO}} \mathbf{V} \; . \tag{158}$$

The Fock and exchange matrices transform similarly. The transformation properties of the amplitude matrices can be obtained by expanding the projected orbitals in the pair correlation functions Ψ_{ij} into the MO basis

$$\Psi_{ij} = \sum_{rs\in[ij]} T^{ij}_{rs} |\ldots\tilde{\chi}_r\tilde{\chi}_s\ldots| \; = \; \sum_{ab} \left(\sum_{rs\in[ij]} V_{ar} T^{ij}_{rs} V_{bs} \right) |\ldots\phi_a\phi_b\ldots| \tag{159}$$

which yields the relation

$$\mathbf{T}^{ij}_{\mathrm{MO}} = \mathbf{V}\mathbf{T}^{ij}_{\mathrm{PAO}}\mathbf{V}^\dagger \; . \tag{160}$$

Inserting this into eq. (79) yields

$$\begin{aligned} \mathbf{R}^{ij}_{\mathrm{PAO}} = \mathbf{K}^{ij}_{\mathrm{PAO}} &+ \mathbf{f}_{\mathrm{PAO}}\mathbf{T}^{ij}_{\mathrm{PAO}}\mathbf{S}_{\mathrm{PAO}} + \mathbf{S}_{\mathrm{PAO}}\mathbf{T}^{ij}_{\mathrm{PAO}}\mathbf{f}_{\mathrm{PAO}} \\ &- \sum_k \mathbf{S}_{\mathrm{PAO}} \left[f_{ik}\mathbf{T}^{kj}_{\mathrm{PAO}} + f_{kj}\mathbf{T}^{ik}_{\mathrm{PAO}} \right] \mathbf{S}_{\mathrm{PAO}} = 0 \; , \end{aligned} \tag{161}$$

where $\mathbf{S}_{\mathrm{PAO}} = \mathbf{P}^\dagger\mathbf{S}_{\mathrm{AO}}\mathbf{P} = \mathbf{V}^\dagger\mathbf{V}$ is the overlap matrix of the projected orbitals, cf. eq. (155). In the local basis the occupied-occupied and virtual-virtual blocks of the Fock matrix are not diagonal, and therefore the linear equations (161) have to be solved iteratively for the amplitudes $\mathbf{T}^{ij}_{\mathrm{PAO}}$. Restricting the excitations to domains $[ij]$ of PAOs means that only the elements T^{ij}_{rs} with $r, s \in [ij]$ are nonzero, and only the corresponding elements of the residual, $R^{ij}_{rs},\ r, s \in [ij]$ must vanish at convergence. For a given set of amplitudes, the Hylleraas functional (eq. 92)

$$E_2 = \sum_{ij\in P} \sum_{rs\in[ij]} (2T^{ij}_{rs} - T^{ij}_{sr})(K^{ij}_{rs} + R^{ij}_{rs}) \tag{162}$$

can be computed. At convergence, $R^{ij}_{rs} = 0$ for $r, s \in [ij]$, and then $E_2 = E^{(2)}$.

Since the projected orbitals are not orthogonal and may even be linearly dependent, straightforward application of an update formular as eq. (103) will lead to slow or no convergence. In order to perform the amplitude update it is therefore necessary to transform the residuals to a pseudo-canonical basis, which diagonalizes the Fock operator in the subspace of the domain $[ij]$, i.e.

$$f^{ij}_{rs}X^{ij}_{rs} = S^{ij}_{rs}X^{ij}_{sa}\epsilon^{ij}_a \qquad \text{for} \qquad r, s \in [ij] \; , \tag{163}$$

$$R^{ij}_{ab} = \sum_{rs\in[ij]} X^{ij}_{sa} R^{ij}_{rs} X^{ij}_{sb} \; . \tag{164}$$

The update is then computed in this orthogonal basis and finally backtransformed to the projected basis

$$\Delta T^{ij}_{ab} = -R^{ij}_{ab}/(\epsilon^{ij}_a + \epsilon^{ij}_b - f_{ii} - f_{jj}) \; , \tag{165}$$

$$\Delta T^{ij}_{rs} = \sum_{ab} X^{ij}_{ra}\Delta T^{ij}_{ab} X^{ij}_{sb} \; . \tag{166}$$

Note that the square transformation matrix $\mathbf{X}^{ij}$ is different for each electron pair. The dimension of this matrix corresponds to the number of projected orbitals in domain $[ij]$ and is therefore independent of the molecular size. If the overlap matrix $S^{ij}_{rs},\ r,s \in [ij]$ has small or zero eigenvalues, i.e, if the functions in the domain are linearly dependent, the corresponding eigenvectors of $\mathbf{S}^{ij}$ are projected out[53]. Convergence of this scheme is reached quickly; usually 5-7 iterations are sufficient to converge the energy to better than 0.1 μH using no further convergence acceleration[70].

In order to compute the residuals, only the small subset of exchange integrals

$$K^{ij}_{rs} = (ri|sj) = \sum_{\nu\mu} P_{\mu r} P_{\nu s} \left[\sum_{\rho\sigma} L_{\rho i} L_{\sigma j} (\mu\rho|\nu\sigma) \right] \qquad r,s \in [ij] \qquad (167)$$

is needed, where all r,s are close either to i or j. This makes it possible to devise an integral-direct transformation scheme which scales only linearly with molecular size[54]. Taking further into account that for a given pair (ij) the number of terms k in the summation of eq. (161) becomes asymptotically independent of the molecular size (provided very distant pairs are neglected), it follows that the computational effort to solve the linear equations scales inearly with molecular size as well[54]. Thus, the overall cost to transform the integrals, to solve the linear equations (161), and to compute the second order energy depends linearly on the molecular size. This has made it possible to perform LMP2 calculations with about 2000 basis functions and 500 correlated electrons without using molecular symmetry. Since also the memory demands are small and scale linearly with molecular size, such calculations can even be performed on low-cost personal computers.

Finally we note that analytical energy gradients for LMP2 have been developed[71]. It has been shown that the local ansatz largely eliminates basis set superposition errors (BSSE), and it is therefore possible to optimize BSSE-free equilibrium structures of molecular clusters[72,73]. Recently, also the theory for computing NMR chemical shifts using the LMP2 method has been derived and first promising results have been obtained[74].

4.2 Local CCSD

The LCCSD equations can be obtained exactly in the same way as indicated above for the LMP2 case, namely by transforming the residuals from the MO to the PAO basis. The resulting equations differ formally from the canonical ones only by the occurrence of additional matrix multiplications with the overlap matrix. The full formalism has been presented in Ref. [53] and will therefore not be repeated here.

As already pointed out before, it is usually sufficient to treat pairs with interorbital distances $R_{ij} \leq 1$ bohr (strong pairs) at the CCSD level. Exceptions are cases where it is of importance to treat longe-range interactions accurately at high level, for instance for computing intermolecular interactions. In the following discussion we will assume, however, that this is not the case, and that the number of strong pairs included in the CCSD treatment is relatively small and scales linearly with molecular size, as shown in Fig. 4.

For the LMP2 case it is immediately obvious that the number of transformed exchange integrals $K_{rs}^{ij} = (ri|js)$ that need to be computed and stored depends only linearly on the molecular size. This follows from the fact that there is a one-to-one correspondence between these integrals and the corresponding amplitudes T_{rs}^{ij}. In the coupled cluster case however, the situation is more complicated, since integrals like the above also couple different electron pairs in the CCSD formalism. Furthermore, as already discussed in section 6.6, there are additional contributions of Coulomb integrals $J_{rs}^{ij} = (ij|rs)$, as well as of integrals $(ir|st)$ and $(rs|tu)$ with three and four external indices, respectively. Closer inspection of the problem reveals, however, that also in the coupled cluster case the number of transformed integrals scales only linearly with molecular size. The same is true for the number of floating point operations needed to compute the residuals.

In order to illustrate the main ideas we will a consider the contribution of the $\mathbf{Y}^{jk}$ intermediates to the LCCD residual, cf. eqs. (124) and (128),

$$\mathbf{G}^{ij} = \ldots + \sum_k \mathbf{S}\tilde{\mathbf{T}}^{ik} \left(\mathbf{K}^{kj} - \frac{1}{2}\mathbf{J}^{kj} \right) + \frac{1}{4} \sum_{kl} \mathbf{S}\tilde{\mathbf{T}}^{ik}\mathbf{L}^{kl}\tilde{\mathbf{T}}^{lj}\mathbf{S} + \ldots \,. \tag{168}$$

Here, all matrices are assumed to be in the PAO basis. Now, since (ik) and (lj) both are strong pairs, there is only a constant number of LMOs k and l interacting with given i and j, respectively. Furthermore, since also (ij) is a strong pair, it follows that for a fixed (ij) the total number of operators contributing to each $\mathbf{G}^{ij}$ is asymptotically constant and independent of the molecular size. Thus, the total number of integral matrices $\mathbf{J}^{kl}$ and $\mathbf{K}^{kl}$ needed in eq. 168 scales linearly; the same holds for the number of matrix multiplications. Furthermore, the LMOs k and l of the surviving operators have to be close, which is important to achieve linear scaling in the integral transformation needed to compute the $\mathbf{J}^{kl}$ and $\mathbf{K}^{kl}$. Note that fewer $\mathbf{J}^{kl}$ than $\mathbf{K}^{kl}$ are needed, since the $\mathbf{J}^{kl}$ only occur in the linear terms. Thus, separate operator lists for the $\mathbf{J}^{kl}$ and $\mathbf{K}^{kl}$ have to be maintained. In contrast to the canonical case, the evaluation of the residuals is driven by individual $\mathbf{J}^{kl}$ and $\mathbf{K}^{kl}$, and the $\mathbf{Y}^{jk}$ and $\mathbf{Z}^{jk}$ intermediates are never explicitly computed.

The PAO range r, s of a particular operator K_{rs}^{kl} is also independent of molecular size: since i must be close to k, and l close to j, all the r, s occuring in the matrix multiplications of eq. 168 must be within a limited distance to k, l. This leads to a different *operator domain* for each surviving operator. Again, the operator domains for the $\mathbf{J}^{kl}$ are smaller than for the $\mathbf{K}^{kl}$. Since the number of Coulomb and exchange matrices scales linearly with molecular size, and the number of elements per matrix is independent of size, it is evident that the overall number of transformed integrals scales linearly with molecular size.

So far, no approximations were involved by introducing the sparse operator lists and operator domains. However, there are a few terms like

$$\mathbf{G}^{ij} = \ldots - \mathbf{S}\mathbf{T}^{ij} \sum_{kl} \mathbf{L}^{kl}\mathbf{T}^{lk}\mathbf{S} \tag{169}$$

with no coupling between ij and kl via pair amplitudes, and for those terms additional approximations have to be introduced to achieve linear scaling. Fortunately, the integrals involved in these contractions diminish quickly with increasing

distance between the pairs (ij) and (kl), and it is well justified to neglect couplings between remote pairs. For a detailed discussion of these approximations we refer to Ref. [61].

Another important feature of LCCSD is the fact that the number of 3-external and 4-external integrals $(ir|st)$ and $(rs|tu)$ also scales linearly with molecular size, and in fact remains rather modest. For the 3-external integrals this follows from the fact that the r, s in the operators

$$J(\mathbf{E}^{kj})_{rs} = \sum_t (rs|tk)t_t^j \ , \qquad K(\mathbf{E}^{kj})_{rs} = \sum_t (rk|ts)t_t^j \tag{170}$$

are restricted to the J-operator domain $[kj]$, while t in the sum is restricted to the pair domain $[jj]$, which is identical to the orbital domain $[j]$. For the 4-external integrals the PAO indices simply all belong to the same pair domain $[ij]$, since there is a one-to-one correspondence between the residual $(\mathbf{R}^{ij})_{rs}$ and the external exchange operators [53,61]

$$K(\mathbf{T}^{ij})_{rs} = \sum_{tu} T_{tu}^{ij}(rt|us) \ . \tag{171}$$

Thus the number of 4-external integrals per pair is a constant. Fig. 5 shows the number 3-external and 4-external integrals in the local basis as a function of the length n of a linear polyglycine peptide chain $(\mathrm{Gly})_n$ in a cc-pVDZ basis. Even for a molecule as large as $(\mathrm{Gly})_{20}$ with about 1500 basis functions and almost 500 correlated electrons, the disk storage requirement to hold the 3-external integrals is less than 1.5 GByte (compared to more than 3000 GByte in the canonical case). A similar amount is required for the 4-external integrals. Disk storage of the 3-external and 4-external integrals is very appealing, since then the computational cost per iteration is minimized. It can be estimated that forming the contractions of the 3-external and 4-external integrals with the amplitudes would take virtually no time (e.g., less than 50 sec for $(\mathrm{Gly})_{20}$). However, the transformation for the 4-external integrals is quite complicated and has not been implemented so far. Alternatively, the contribution of these integrals can be accounted for by computing for each strong pair an external exchange operator, as defined in eq. (107). In an integral direct scheme, as will be discussed in section 6.5, it is then also possible to achieve linear cost scaling.

4.3 Local connected triples corretion

The ultimate bottleneck for accurate conventional coupled cluster calculations is the connected triples correction, as outlined in section 2.7. If canonical orbitals are used, the Fock matrix is diagonal, and the perturbative energy correction can be obtained directly without storing the triples amplitudes. In the local case this is no longer the case, and in principle an iterative scheme is required, as described above for the local MP2. One might therefore think that the evaluation of a local triples correction for large molecules is impossible, since the storage requirements for all triples amplitudes would scale as $\mathcal{O}(\mathcal{N}^6)$. However, as for the doubles,

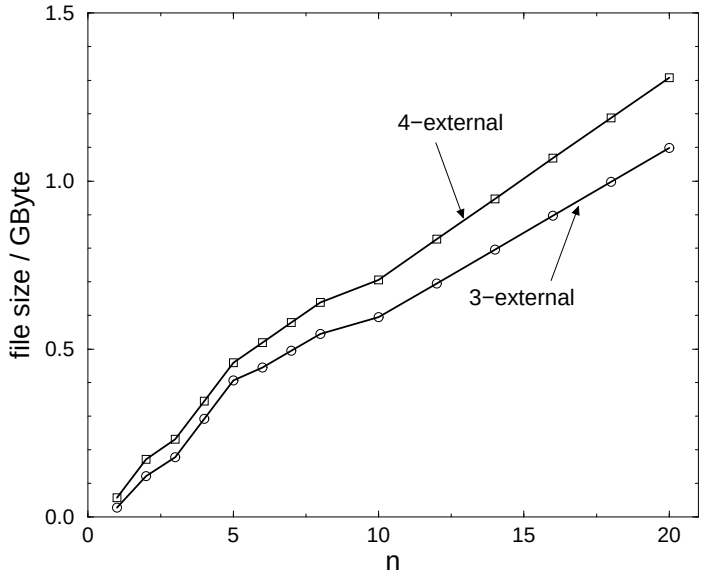


Figure 5. Number of transformed integrals for local CCSD calculations for glycine peptides $(\mathrm{Gly})_n$ as function of the chain length.

triple domains can be introduced, and the correlation of distant electrons can be neglected.

The theory has been outlined in Ref. [62]. Similarly to the LMP2 case the triples amplitudes are obtained by solving a system of linear equations

$$Q_{rst}^{ijk} + W_{rst}^{ijk} = 0 \tag{172}$$

with

$$\begin{aligned} Q_{rst}^{ijk} = &\sum_v \{ \sum_{r's'} f_{tv} T_{r's'v}^{ijk} S_{rr'} S_{ss'} + \text{ permutations} \} \\ &- \sum_m \{ \sum_{r's't'} f_{km} T_{r's't'}^{ijm} S_{rr'} S_{ss'} S_{tt'} + \text{ permutations} \} \end{aligned} \tag{173}$$

and

$$\begin{aligned} W_{rst}^{ijk} = &\sum_v \{ \sum_{r'} (vs|tk) T_{r'v}^{ij} S_{rr'} + \text{ permutations} \} \\ &- \sum_m \{ \sum_{r's'} (mj|kt) T_{r's'}^{im} S_{rr'} S_{ss'} + \text{ permutations} \}. \end{aligned} \tag{174}$$

These equations have to be solved iteratively, and therefore all triples amplitudes T_{rst}^{ijk} must be stored on disk. This seems devastating at a first glance, but by virtue of the local approximations the number of amplitudes can be drastically reduced:

Firstly, a sparse triples list (ijk) of *strong triples* is constructed by restricting the related pairs ij, ik and jk to *strong pairs*. The number of strong triples then scales linearly with molecular size. Secondly, the excitations are restricted to *triple domains* $[ijk]$, constructed as the *union* of the three *strong pair domains*, i.e., $[ijk] = [ij] \cup [ik] \cup [jk]$. Since the sizes of the individual pair domains are independent of the molecular size, the size of the triples domain $[ijk]$ is also independent of the molecular size, yielding overall an asymptotically linear scaling of the number of triples amplitudes.

Another important implication of the constant size of the triple domains is that the number of required 3-external integrals occurring in eq. (174) scales only *linearly* with molecular size. In practice, for each orbital l a *united triple domain* $UT(l)$ is defined as the union of all triple domains $[ijk]$ comprising a common LMO index l, i.e.,

$$UT(l) = \cup[ijk], \quad \text{for} \quad (i = l) \vee (j = l) \vee (k = l), \tag{175}$$

and all 3-external integrals $(vs|tl)$ with $v, s, t \in UT(l)$ are generated using and integral-direct transformation module. Obviously, the size of $UT(l)$ is independent of the molecular size, and the CPU time as well as memory and disk requirements of the transformation scale asymptotically linear with molecular size. In fact, the set of 3-external integrals needed for the triples correction remains pretty small [62], and usually it is a subset of the 3-external integral set required in the preceeding coupled cluster calculation (cf. Fig. 5 in section 4.2).

A linear scaling algorithm for local triples has been implemented in MOLPRO 2000. So far, inter-triples couplings via the occupied-occupied off-diagonal Fock matrix elements are neglected (couplings via the virtual-virtual block and the overlap matrices are included though). This yields about 95% of the local triples correction and has the advantage that the iterative solution of eq. (172) can be avoided. As in the canonical case, the correlation contribution of each individual triple can be computed separately. First test results[62] presented in Table 3 are very promising, showing already for medium sized molecules speedups by factors 500-1000 compared to the conventional (T) calculation presented earlier in Table 1 (note that the calculations in Table 1 used molecular symmetry, while the current calculations were done with no symmetry). In these calculations about 85% of the canonical triples correction was recovered[62]. The savings quickly increase with increasing molecular size. In sharp contrast to the conventional case, the time to compute the local triples corrections is very small as compared to time for the preceding integral transformation and LCCSD calculation. Considering the efficiency of the new triples kernel, it seems even possible to go beyond the CCSD(T) model, i.e. to include the triples into the CC iterations, even for large chemical systems.

Finally, it should be emphasized that the triples amplitudes *can* be stored and an iterative full local triples algorithm is presently under development.

Table 3. CPU, disk and memory requirements for computing the (T) correction. All calculations were performed with a development version of MOLPRO 2000[75]. No molecular symmetry was used.

Molecule	bf	Memory/MW	Disk[a]/MW	CPU/sec[b]
$(Gly)_1$	95	2.57	7.46	187.3
$(Gly)_2$	166	6.38	25.51	757.8
$(Gly)_3$	236	8.82	39.98	934.6
$(Gly)_4$	308	13.28	61.06	1296.6
$(Gly)_6$	450	18.95	94.17	1852.5

a) Disk space for storage of 3-external integrals necessary for (T) only.
b) HP J282 PA8000/180MHz.

5 Multireference electron correlation methods

5.1 Configuration Interaction: general aspects

For a given orbital basis set, Schrödinger's equation as expressed using the second-quantized hamiltonian $\hat{H}$ (equation (34)) is solved by finding eigenvectors and eigenvalues of the hamiltonian matrix in the complete basis of N-electron orbital products. This *full CI* problem is of extremely large dimension for even a small number of electrons with a modest orbital basis size, and is usually intractable. However, it is important to consider it for two reasons: first of all, where the full CI problem can be solved, it provides very important benchmark data against which approximate methods can be evaluated; secondly, the techniques and algorithms applicable to the full CI problem serve as appropriate building blocks for the sometimes more complicated approximate methods.

Although the full configuration space for N electrons in m spatial orbitals consists formally of the complete set of $(2m)^N$ spin-orbital products $\psi_{i_1}(\mathbf{x_1})\psi_{i_2}(\mathbf{x_2})\ldots\psi_{i_N}(\mathbf{x_N})$, the space can be reduced substantially through symmetry considerations:

- *Spatial (point group) symmetry.* $\hat{H}$ is invariant to geometrical transformations whose only effect is to interchange identical nuclei. The action of the symmetry operators on the wavefunction is defined through

$$\hat{T}\Psi(q) = \Psi(\hat{T}^{-1}q) \tag{176}$$

  where q represents the coordinates of the particles. In electronic structure calculations, the use of abelian point group symmetry is straightforward; provided each orbital is a basis for an irreducible representation, then so is every orbital product. All orbital products not of the required symmetry can then be simply discarded from the basis. For non–abelian point groups, orbital products are in general of mixed symmetry, and it is therefore usual to exploit the symmetry of only the highest abelian subgroup.

- *Permutational symmetry.* $\hat{H}$ is totally symmetric in the labels of the electrons, and so is invariant under the operation $\hat{I}_{ij}$ which interchanges the labels of

electrons i, j, i.e., $[\hat{H}, \hat{I}_{ij}] = 0$. At the simplest level, $\hat{I}_{ij}^2 = 1$, and so there are $\frac{1}{2}N(N-1)$ two dimensional symmetry groups $\{1, \hat{I}_{ij}\}$. Symmetry adapted wavefunctions will satisfy $\hat{I}_{ij}\Psi = \pm\Psi$, the different signs corresponding to boson and fermion states. We are interested only in fermion solutions, and so it is vital to use this symmetry to exclude unwanted boson and non-physical states. In further detail, there is actually a total of $N!$ *permutations* of the electron labels, which can be build as products of $\hat{I}_{ij}$ operators. As with point groups, we define the action of a permutation operator on the wavefunction through equation (176). The permutations form a group isomorphic with the *Symmetric Group* S_N, and to use permutational symmetry to the full, we must consider all of these $N!$ operators which commute with $\hat{H}$. Since the electronic wavefunction is antisymmetric with respect to all the $\hat{I}_{ij}$, it must form a basis for the one dimensional totally antisymmetric representation of S_N; the representation matrix elements $\Gamma(\hat{P})$ are equal to the *parity* ϵ_P of the permutation $\hat{P}$, which is ± 1 according to whether $\hat{P}$ is made up from an even or odd number of interchanges $\hat{I}_{ij}$. To enforce the symmetry, we apply a multiple of the Wigner projection operator for this representation, the *antisymmetrizer*

$$\hat{\mathcal{A}} = \frac{1}{\sqrt{N!}} \sum_{P}^{N!} \epsilon_P \hat{P} \,. \tag{177}$$

When applied to a simple product of orbitals, $\hat{\mathcal{A}}$ yields the corresponding Slater determinant

$$\begin{aligned} \hat{\mathcal{A}}\,\psi_1(1)\psi_2(2)\ldots\psi_N(N) &= \frac{1}{\sqrt{N!}} \sum_{P}^{N!} \epsilon_P \hat{P}\,\psi_1(1)\psi_2(2)\ldots\psi_N(N) \\ &= \frac{1}{\sqrt{N!}} \begin{vmatrix} \psi_1(1) & \psi_2(1) & \ldots & \psi_N(1) \\ \psi_1(2) & \psi_2(2) & \ldots & \psi_N(2) \\ \vdots & \vdots & & \vdots \\ \psi_1(N) & \psi_2(N) & \ldots & \psi_N(N) \end{vmatrix} . \end{aligned} \tag{178}$$

Note that, apart from a possible phase factor, exactly the same determinant would arise if $\hat{\mathcal{A}}$ were applied to a string of the same orbitals, but in a different order, e.g., $\psi_2(1)\psi_3(2)\psi_1(3)\ldots$. Therefore we can symmetry reduce the full set of m^N orbital products to a much smaller basis of $\binom{m}{N}$ Slater determinants obtained by acting with the antisymmetrizer on each of the $\binom{m}{N}$ unique orbital products. The valid unique orbital products can be determined by assuming an ordering for the orbitals; each of the m orbitals ψ_i is assigned a sequence number i, $i = 1, 2, \ldots, m$, and only orbital products $\psi_{i_1}\psi_{i_2}\ldots\psi_{i_N}$ for which $i_1 < i_2 < \ldots < i_N$ are included.

- *Spin symmetry.* The electron spin operators are defined through

$$\hat{S}^2 = \hat{\mathbf{S}} \cdot \hat{\mathbf{S}}; \;\; \hat{\mathbf{S}} = \sum_i^N \hat{\mathbf{s}}(i)$$
$$\hat{s}_x\alpha = \frac{1}{2}\beta; \;\; \hat{s}_y\alpha = -\frac{i}{2}\beta; \;\; \hat{s}_z\alpha = \frac{1}{2}\alpha;$$
$$\hat{s}_x\beta = \frac{1}{2}\alpha; \quad \hat{s}_y\beta = \frac{i}{2}\alpha; \quad \hat{s}_z\beta = -\frac{1}{2}\beta, \tag{179}$$

where α and β are the one electron spin eigenfunctions. The non–relativistic hamiltonian contains no spin operators, and so $[\hat{H}, \hat{S}_z] = [\hat{H}, \hat{S}^2] = 0$. It is not possible to use simple group theory to exploit these symmetries, since the operators $\hat{S}_z$, $\hat{S}^2$ do not form a closed finite group. But we can use other considerations to force the N electron basis set, and hence the wavefunction, to be eigenfunctions of $\hat{S}_z$ and/or $\hat{S}^2$.

In the case of $\hat{S}_z$, the approach which is usually used is to use a basis of $2m$ orbitals, made up of m spatial orbitals $\phi_i, i = 1, 2, \ldots m$, each multiplied by a spin function α or β. Then any orbital product, or Slater determinant, is automatically an eigenfunction of $\hat{S}_z$ according to (179), with eigenvalue $\frac{1}{2}(N_\alpha - N_\beta)$, where N_α is the number of α–spin orbitals ϕ_i^α in the function, and $N_\beta = N - N_\alpha$. Thus the basis is already adapted to $\hat{S}_z$ symmetry, and we may discard all those N electron functions with the wrong $\hat{S}_z$ eigenfunction. This reduces the size of the Slater determinant basis from $\binom{2m}{N}$ to

$$M_D = \binom{m}{N_\alpha}\binom{m}{N_\beta}, \tag{180}$$

since for each of the $\binom{m}{N_\alpha}$ possible arrangements of the α spin orbitals there are $\binom{m}{N_\beta}$ choices for the β–spin orbitals.

For $\hat{S}^2$, the situation is not so simple. Orbital products or Slater determinants are not in general eigenfunctions of $\hat{S}^2$; for example, following (179), $\hat{S}^2\phi_1^\alpha(1)\phi_2^\beta(2) = \phi_1^\alpha(1)\phi_2^\beta(2) + \phi_1^\beta(1)\phi_2^\alpha(2)$. If the symmetry is to be exploited, Slater determinants must be linearly combined into functions which are eigenfunctions of $\hat{S}^2$. Such functions are often termed *Configuration State Functions* (CSFs). As a simple example, for two electrons in two orbitals with $N_\alpha = N_\beta = 1$, the normalized Slater determinants are

$$\hat{\mathcal{A}}\phi_1^\alpha\phi_1^\beta, \qquad \hat{\mathcal{A}}\phi_1^\alpha\phi_2^\beta, \qquad \hat{\mathcal{A}}\phi_2^\alpha\phi_1^\beta, \qquad \hat{\mathcal{A}}\phi_2^\alpha\phi_2^\beta;$$

the normalized CSFs with $S = 0$ are

$$\hat{\mathcal{A}}\phi_1^\alpha\phi_1^\beta, \qquad \hat{\mathcal{A}}\phi_2^\alpha\phi_2^\beta, \qquad (1/\sqrt{2})(\hat{\mathcal{A}}\phi_1^\alpha\phi_2^\beta + \hat{\mathcal{A}}\phi_2^\alpha\phi_1^\beta),$$

and the CSF with $S = 1$ (i.e., the eigenvalue of $\hat{S}^2$ is $S(S+1) = 2$) is

$$(1/\sqrt{2})(\hat{\mathcal{A}}\phi_1^\alpha\phi_2^\beta - \hat{\mathcal{A}}\phi_2^\alpha\phi_1^\beta).$$

Generally, the set of Slater determinants exactly spans the sets of CSFs with spin quantum numbers $S = \frac{1}{2}(N_\alpha - N_\beta), \frac{1}{2}(N_\alpha - N_\beta) + 1, \ldots, \frac{1}{2}N$. Ignoring

any point group symmetry, the number of CSFs with spin quantum number S is given by the Weyl formula [76]

$$M_C = \frac{2S+1}{m+1}\binom{m+1}{\frac{1}{2}N-S}\binom{m+1}{m-\frac{1}{2}N-S}, \tag{181}$$

for the case that $S = \frac{1}{2}(N_\alpha - N_\beta)$. So, for example, for $S = 0$ and large m, the number of CSFs is less than the number of Slater determinants by a factor of about $\frac{1}{2}N + 1$. The advantage of reducing the basis in this way has to be offset against the increased complexity of the functions which must be dealt with; in practice both Slater determinants and CSFs are commonly used, and we discuss the practicalities of matrix element evaluation with each below.

- *Orbital rotation symmetry.* If we have *all* (unique) orbital products possible for N electrons in m orbitals, then the basis is invariant to rotations (or in fact any non–singular linear transformation) of the orbitals amongst themselves. These rotations form a continuous group $U(m)$, the unitary group (or $GL(m)$, the general linear group), and the theory of such groups is exploited to advantage, for example, in the Graphical Unitary Group Approach (GUGA) [77] for configuration interaction.

In order to perform a variational configuration interaction calculation in either the full or a truncated configuration space, it is necessary to find an eigenvector of the matrix $\mathbf{H}$ of the hamiltonian operator $\hat{H}$ in the appropriate configuration space. Direct construction and diagonalization of $\mathbf{H}$ is usually out of the question since it is typically of dimension 10^3–10^7; but algorithms to find a few eigenvectors for such matrices exist[3,78], and rely on the construction, for a few ($\sim$ 10–20) given trial vectors $\mathbf{c}$, of the action of $\mathbf{H}$ on $\mathbf{c}$,

$$\mathbf{v} = \mathbf{H}\mathbf{c} \, . \tag{182}$$

Other ab initio approaches which are not simple matrix eigenproblems can also proceed through (182). Therefore it is vital to have an efficient scheme for constructing (182) from the hamiltonian integrals $h_{pq}, (pq|rs)$. Following (37), this means we must be able to compute rapidly the set of one and two particle coupling coefficients d^{IJ}_{pq}, D^{IJ}_{pqrs}.

In many circumstances, the most efficient schemes for building (182) require computation only of the one particle coefficients d^{IJ}_{pq}, without explicit construction of the two body terms D^{IJ}_{pqrs}. This is achieved through a formal insertion of the resolution of the identity as a sum over the complete space of orbital products,

$$\begin{aligned} D^{IJ}_{pqrs} &= \langle \Phi_I | \hat{E}_{pq}\hat{E}_{rs} - \delta_{qr}\hat{E}_{ps} | \Phi_J \rangle \\ &= \sum_K \langle \Phi_I | \hat{E}_{pq} | \Phi_K \rangle \langle \Phi_K | \hat{E}_{rs} | \Phi_J \rangle \; - \; \delta_{qr} \langle \Phi_I \hat{E}_{ps} | \Phi_J \rangle \\ &= \sum_K d^{IK}_{pq} d^{KJ}_{rs} \; - \; \delta_{qr} d^{IJ}_{ps} \, . \end{aligned} \tag{183}$$

Note that $\hat{E}_{pq}$ commutes with electron label permutations and spin operators; therefore the set of intermediate states $\{\Phi_K\}$ can be reduced to the full set of Slater

determinants or CSFs as convenient; but the same is not true for point group operations, and $\{\Phi_K\}$ must therefore extend over all spatial symmetries. The algorithm for building (182) then proceeds as [79]

$$\begin{aligned}
&\text{DO } K = 1, M \\
&\quad \text{DO } p, q = 1, m \text{ such that } d_{pq}^{KJ} \neq 0 \\
&\qquad F_{pq}^{K} = F_{pq}^{K} + d_{pq}^{KJ}\, c_J \\
&\quad \text{END DO} \\
&\text{END DO}
\end{aligned} \tag{184}$$

$$\begin{aligned}
&\text{DO } r \geq s \\
&\quad \text{DO } p \geq q \\
&\qquad \text{DO } K = 1, M \\
&\qquad\quad E_{rs}^{K} = E_{rs}^{K} + F_{pq}^{K}\,(pq|rs) \\
&\qquad \text{END DO} \\
&\quad \text{END DO} \\
&\text{END DO}
\end{aligned} \tag{185}$$

$$\begin{aligned}
&\text{DO } K = 1, M \\
&\quad \text{DO } p, q = 1, m \text{ such that } d_{pq}^{IK} \neq 0 \\
&\qquad v_I = v_I + E_{pq}^{K}\, d_{pq}^{IK} \\
&\quad \text{END DO} \\
&\text{END DO}
\end{aligned} \tag{186}$$

The one electron part and second term of (183) are easily dealt with in an additional stage, or may be included in (184–186) by modifying the two electron integrals. The advantage of using this scheme is that, for sufficiently large cases, the computation time is dominated by (185), requiring approximately $\frac{1}{2}Mm^4$ floating point operations, and this step is a large dimension matrix multiplication capable of driving most computer hardware at optimal speeds. In what follows, therefore, we are concerned principally with the evaluation, rapidly and in the correct order for assembly of (184–186), of the non–zero d_{pq}^{IJ}, without the need to consider the more complicated structure, and much larger number, of D_{pqrs}^{IJ} coefficients. In some circumstances, simple Slater determinants offer the most efficient route to calculating (182), whilst elsewhere the greater compactness of the CSF basis is important. Therefore we develop techniques for evaluating d_{pq}^{IJ} in both types of basis set.

5.2 Matrix elements between Slater Determinants

Any Slater determinant can be written in the form

$$\Phi_{I,J} = \hat{\mathcal{A}}\left({}^{\alpha}\Phi_I\, {}^{\beta}\Phi_J\right) \tag{187}$$

where ${}^{\alpha}\Phi_I(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{N_\alpha})$ is a *string* (product) of occupied α–spin orbitals

$$ {}^{\alpha}\Phi_I = \phi^{\alpha}_{I_1}(1)\phi^{\alpha}_{I_2}(2)\ldots\phi^{\alpha}_{I_N}(N) \; , \tag{188} $$

which is completely specified by the ordered list of sequence numbers of occupied orbitals, $\{I_1 < I_2 < \ldots < I_N\}$. Similarly, ${}^{\beta}\Phi_J(\mathbf{r}_{N_\alpha+1}, \mathbf{r}_{N_\alpha+2}, \ldots, \mathbf{r}_N)$ is a string of occupied β–spin orbitals. For the case of a complete basis of determinants, this is a particularly helpful classification, since a wavefunction Ψ is then specified by a fully populated rectangular matrix of coefficients $\mathbf{C}$,

$$ \Psi = \sum_{IJ} C_{IJ}\Phi_{I,J} \; , \tag{189} $$

and this simple rectangular addressing structure makes for a particularly efficient computer implementation. For certain special types of incomplete CI expansion, it is possible to obtain similar structures[80], but it is the case of full CI (FCI) for which the determinant basis has found particularly useful application.

For the evaluation of coupling coefficients, we can exploit the fact that the orbital excitation operator partitions as

$$ \hat{E}_{pq} = \hat{e}^{\alpha}_{pq} + \hat{e}^{\beta}_{pq} \; , \tag{190} $$

where $\hat{e}^{\alpha}_{pq}$, $\hat{e}^{\beta}_{pq}$ excite only α, β spin orbitals respectively; thus the effect of $\hat{E}_{pq}$ on any determinant is to produce at most two new determinants:

$$ \hat{E}_{pq}\hat{\mathcal{A}}\left({}^{\alpha}\Phi_I\,{}^{\beta}\Phi_J\right) = \hat{\mathcal{A}}\left(\left(\hat{e}^{\alpha}_{pq}\,{}^{\alpha}\Phi_I\right){}^{\beta}\Phi_J\right) + \hat{\mathcal{A}}\left({}^{\alpha}\Phi_I\left(\hat{e}^{\beta}_{pq}\,{}^{\beta}\Phi_J\right)\right) \; . \tag{191} $$

Note that the excitation $\hat{e}^{\alpha}_{pq}\,{}^{\alpha}\Phi_I$ is completely independent of ${}^{\beta}\Phi_J$, and so once a particular α–spin excitation has been characterized, one can use the information found for all β strings, obtaining

$$ \langle\Phi_{I,J}|\hat{E}_{pq}|\Phi_{K,L}\rangle = \langle{}^{\alpha}\Phi_I|\hat{\mathcal{A}}\hat{e}^{\alpha}_{pq}\hat{\mathcal{A}}|{}^{\alpha}\Phi_K\rangle\,\delta_{JL} \; . \tag{192} $$

For this to be non zero, ${}^{\alpha}\Phi_I$ must be identical to ${}^{\alpha}\Phi_K$ apart from the replacement of ϕ^{α}_q by ϕ^{α}_p. Suppose that in Φ_I, ϕ_p appears as a function of electron i, and in Φ_K, ϕ_q is correspondingly in position j, i.e.,

$$ {}^{\alpha}\Phi_I = \phi_{I_1}(1)\phi_{I_2}(2)\ldots\phi_{I_{i-1}}(i-1)\phi_p(i)\phi_{I_{i+1}}(i+1)\ldots\phi_{I_j}(j)\ldots \tag{193} $$

and

$$ {}^{\alpha}\Phi_K = \phi_{I_1}(1)\phi_{I_2}(2)\ldots\phi_{I_{i-1}}(i-1)\phi_{I_{i+1}}(i)\phi_{I_{i+2}}(i+1)\ldots\phi_q(j)\ldots \tag{194} $$

Then

$$ \hat{E}_{pq}\,{}^{\alpha}\Phi_K = \phi_{I_1}(1)\phi_{I_2}(2)\ldots\phi_{I_{i-1}}(i-1)\phi_{I_{i+1}}(i)\phi_{I_{i+2}}(i+1)\ldots\phi_p(j)\ldots \tag{195} $$

This is not the same as the string ${}^{\alpha}\Phi_I$, but is related to it by a permutation of the electron labels, known as the *line–up permutation* $\hat{L}$, which in this case is the *cyclic permutation* $\hat{C}(i,j)$, defined through

$$ \hat{C}(i,j)\phi_1(i)\phi_2(i+1)\ldots\phi_{j-i+1}(j) = \phi_{j-i+1}(i)\phi_1(i+1)\ldots\phi_{j-i}(j) \; ; \tag{196} $$

Thus $\hat{L}\hat{e}^{\alpha}_{pq}\,{}^{\alpha}\Phi_K = \hat{C}(i,j)\hat{e}^{\alpha}_{pq}\,{}^{\alpha}\Phi_K = {}^{\alpha}\Phi_I$. For any permutation $\hat{P}$, the following is true:

$$ \hat{P}\hat{\mathcal{A}} = \hat{\mathcal{A}}\hat{P} = \epsilon_P\hat{\mathcal{A}} \; , \tag{197} $$

and so the matrix element (192) is

$$\begin{aligned}\langle {}^{\alpha}\Phi_I|\hat{\mathcal{A}}\hat{e}^{\alpha}_{tu}\hat{\mathcal{A}}|{}^{\alpha}\Phi_K\rangle &= \langle {}^{\alpha}\Phi_I|\hat{\mathcal{A}}\hat{L}^{-1}\hat{\mathcal{A}}|{}^{\alpha}\Phi_I\rangle \\ &= \epsilon_L\sqrt{N!}\,\langle {}^{\alpha}\Phi_I|\hat{\mathcal{A}}|{}^{\alpha}\Phi_I\rangle \\ &= \epsilon_L\ , \end{aligned} \tag{198}$$

since $\hat{\mathcal{A}}^2 = \sqrt{N!}\hat{\mathcal{A}}$, and the only non–zero contribution to $\langle {}^{\alpha}\Phi_I|\hat{\mathcal{A}}|{}^{\alpha}\Phi_I\rangle$ comes from the identity permutation. Therefore all coupling coefficients are 0 or ± 1, and the sign is determined by the parity of the line–up permutation $\hat{L}$. Hence the construction of $\mathbf{F}$ in (184) proceeds as

```
DO ^αΦ_K
    DO p,q = 1,m such that ^αΦ_I = ±Ê^α_pq ^αΦ_K exists
        Determine parity ε_L of line–up permutation L̂
        DO ^βΦ_J
            F(K,J,pq) ← ε_L C(I,J)
        END DO
    END DO
END DO
```
(199)

The innermost loop over ${}^{\beta}\Phi_J$ contains no logic or even multiplication and vectorizes perfectly on all pipeline computers. A similar loop structure is required for the contributions from $\hat{e}^{\beta}_{pq}$, and the logic of (186) can be treated in a similar fashion. Because the number of α, β strings is rather small ($\sqrt{M_D}$), all the necessary single excitation information can be computed once and held in high speed storage. The result is a perfectly vectorized, disk free algorithm[81,82], where for reasonably sized problems at least, there is practically no overhead above the cost of the matrix multiplication (185).

There have been a number of algorithmic developments which have further enhanced the efficiency and applicability of the determinant FCI method. Olsen *et al.* [80] showed how it was possible to reduce the operation count to be proportional to N^2m^2 rather than m^4, with, however, some degradation of the vector performance; their method is particularly useful when the ration m/N is relatively large. Zarrabian *et al.* [83] have used an alternative resolution of the identity to (183), with an intermediate summation over $N-2$ electron (rather than N electron) Slater determinants. Again, when m/N is large, there are many fewer of these, allowing for considerable enhancement in efficiency.

5.3 Matrix elements between Configuration State Functions

In order to build a basis of spin–adapted CSFs, we begin by finding explicit *spin functions* Θ, which are not dependent on space coordinates, and which satisfy $\hat{S}^2\Theta = S(S+1)\Theta$. Having done this, we then attempt to build fully symmetry adapted space–spin functions. For a single electron, there are two possible spin functions $\theta(s)$, where s represents the spin coordinate, namely the usual α and β. For N electrons, the complete space of spin functions is then spanned exactly by the

N electron *primitive spin functions*, written as $[\theta_{i_1}\theta_{i_2}\ldots\theta_{i_N}]$ where the function θ_i of the spin coordinates of each electron in turn may be α or β. There are a total of 2^N such functions, and they are eigenfunctions of $\hat{S}_z$, the eigenvalue M_S being $\frac{1}{2}(N_\alpha - N_\beta)$ where N_α is the number of times α appears in the function, and $N_\beta = N - N_\alpha$; it is then convenient to group them together in sets of those functions sharing the same M_S, the number in each set being $\binom{N}{\frac{1}{2}N+M_S}$.

The primitive spin functions are not in general eigenfunctions of $\hat{S}^2$, and so we seek linear combinations Θ_μ which will be spin eigenfunctions. This is achieved most simply by repeated application of standard angular momentum coupling theory[84,85]. If we have two independent physical systems in each of which we have sets of angular momentum eigenfunctions, $\{|J_1M_1\rangle\}$ and $\{|J_2M_2\rangle\}$, then the members of the set of all products of such wavefunctions are not in general eigenfunctions of the total angular momentum for the combined system. But for a given J_1, J_2 and feasible final quantum numbers J, M, it is possible to find exactly one composite eigenfunction

$$|JM\rangle = \sum_{M_1 M_2} \langle J_1 J_2 M_1 M_2 | JM\rangle |J_1 M_1\rangle |J_2 M_2\rangle \tag{200}$$

where the number $\langle J_1 J_2 M_1 M_2 | JM\rangle$ is a standard *Clebsch–Gordon coefficient*. Note that all the different M_1 and M_2 components appear in the sum, but only a single J_1 and J_2 value is involved. For N electron spin functions, this suggests a recursive scheme whereby N electron functions are made from such a composite of an $N-1$ electron system with a further single electron. The $N-1$ electron functions arise in the same way from $N-2$ electron spin eigenfunctions, the chain being repeated down to a single particle. For each coupling, the value of J_2 is $\frac{1}{2}$, and so the sum over M_2 extends over two possible values, $\pm\frac{1}{2}$, i.e. a contribution involving α for the last electron and a contribution with β. In this *genealogical* construction, each N electron function is fully described by its parentage — the history of the coupling scheme — which can be visualized as a path on the *branching diagram* shown in Figure 6. Because in the angular momentum coupling one need sum only over the M and not the S quantum numbers, there are in general many independent functions having the same S, M_S, but different ancestry, and we label the functions as $\Theta^N_{S,M,\mu}$ where μ is an index which distinguishes functions with different parentage. The number f^N_S of such functions is indicated at each node on the branching diagram, and one can show inductively that

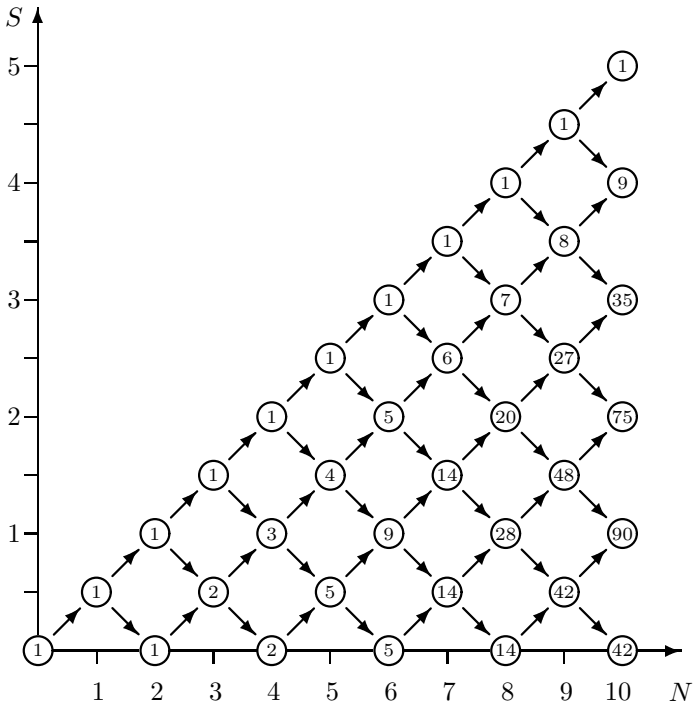
$$f^N_S = \left(\binom{N}{\frac{1}{2}N-S} - \binom{N}{\frac{1}{2}N-S-1}\right) . \tag{201}$$

It follows, again inductively, that

$$\sum_S (2S+1)\, f^N_S = 2^N . \tag{202}$$

It is straightforward to show [86] that the genealogical functions are orthonormal. For each path on the diagram, there are $(2S+1)$ functions, corresponding to the possible different M_S values, and so $\sum_S (2S+1)\, f^N_S$ represents the total number of

Figure 6. The branching diagram



independent N electron branching diagram functions; this is the same as the number of primitive spin functions, and so we have a complete set of spin functions.

Because eventually we need to consider the effect of the antisymmetrizing operator $\hat{\mathcal{A}}$, it is important to develop the permutation properties of the spin functions. Since $\hat{S}^2$ is totally symmetric in the particle labels, it commutes with any permutation, $\hat{P}\hat{S}^2 = \hat{S}^2\hat{P}$. Then it follows that, since $\hat{S}^2\Theta^N_{S,M,\mu} = S(S+1)\Theta^N_{S,M,\mu}$,

$$\hat{S}^2\left(\hat{P}\Theta^N_{S,M,\mu}\right) = \hat{P}\hat{S}^2\Theta^N_{S,M,\mu} \tag{203}$$

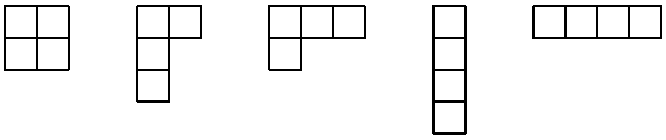
$$= S(S+1)\left(\hat{P}\Theta^N_{S,M,\mu}\right) \ , \tag{204}$$

i.e. $\hat{P}\Theta^N_{S,M,\mu}$ is a spin eigenfunction with quantum numbers S, M. Since $\{\Theta^N_{S,M,\lambda}, \lambda = 1, 2, \ldots, f^N_S\}$ is a complete set of such functions, then $\hat{P}\Theta^N_{S,M,\mu}$ must be a linear combination of these:

$$\hat{P}\Theta^N_{S,M,\mu} = \sum_\lambda \Theta^N_{S,M,\lambda} U_{\lambda\mu}(\hat{P}) \ , \tag{205}$$

i.e., $\{\Theta^N_{S,M,\lambda}, \lambda = 1, 2, \ldots, f^N_S\}$ is a basis for a representation of the symmetric group S_N. The representation is actually isomorphic with particular cases of *Young's Orthogonal Representation*, which is generated (also genealogically) using ideas from

the theory of S_N. Young's orthogonal representation is often depicted graphically. A given representation is drawn as a *Young diagram*, consisting of N adjoining square boxes with rows numbered numerically downwards, and columns rightwards; there may not be more rows in column i than in column $i-1$, nor columns in row j than in row $j-1$. For example, in S_4, the possible Young diagrams are



(206)

For the case of the spin–$\frac{1}{2}$ particles which are our exclusive concern, then only those representations whose Young diagram has at most two rows are relevant, and they correspond to spin quantum numbers S equal to half the difference between the number of boxes in the two rows. Thus for S_4,  ,  ,  represent, respectively, the sets of spin functions with $S = 0,\ 1,\ 2$.

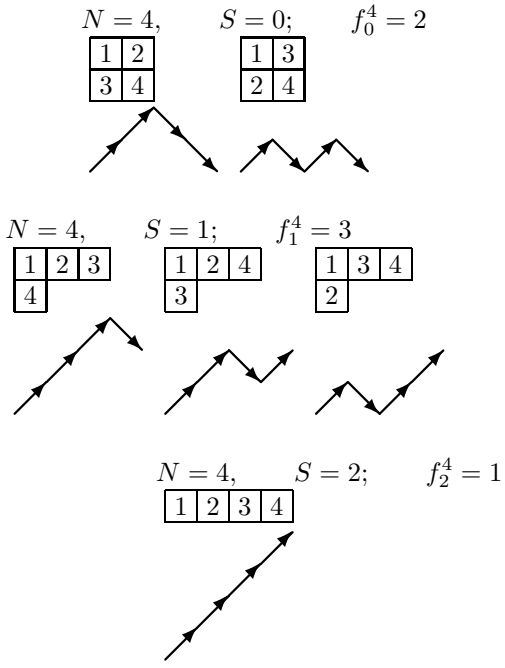
Within each representation, a given basis function is depicted as a *Young tableau*, which is an arrangement of the numbers $1, 2, \ldots, N$ in the Young Diagram, such that numbers always increase along all rows and down all columns. For the two–row Young frames which we consider, the number of such tableaux (i.e., the dimension of the representation) is exactly f_S^N, and in fact there is a one–to–one correspondence between the branching diagram functions and the tableaux; when a particle number appears in the first row, its spin is coupled up, and for those in the second row, the spin is coupled down. For the case of four electrons, the complete set of Young tableaux and corresponding branching diagram functions are shown in Figure 7. The representation matrices $\mathbf{U}(\hat{P})$ constitute all the information which we require for developing properties of the branching diagram functions; for example, the branching diagram functions themselves can be generated from a primitive spin function by use of a suitable projection operator. Formulae for the $U_{\lambda\mu}(\hat{P})$ for any permutation $\hat{P}$ are straightforward to derive from simple rules given in terms of the Young tableaux [86], or, equivalently, from consideration of the Clebsch–Gordon coefficients [86].

Having obtained the representation matrices, we are now in a position to use them in constructing a basis of space and spin functions which are spin eigenfunctions and satisfy the Pauli principle. We write members of this basis as

$$\Phi_{A\lambda} = \hat{\mathcal{A}} \left(\Phi_A \, \Theta^N_{S,M,\lambda} \right) \tag{207}$$

where the spatial function Φ_A is usually an ordered product of spatial orbitals, and $\Theta^N_{S,M,\lambda}$ is a branching diagram function. Note that the antisymmetrizer involves a sum over all permutations $\hat{P}$, and each $\hat{P}$ permutes both the space and the spin

Figure 7. Branching diagram symbols and Young tableaux for 4 electrons



coordinate labels. Inserting the definition of the antisymmetrizer,

$$\begin{aligned}\Phi_{A\lambda} &= \sqrt{\frac{1}{N!}} \sum_P \epsilon_P \left(\hat{P}_{\rm space}\Phi_A\right)\left(\hat{P}_{\rm spin}\Theta_\lambda\right) \\ &= \sqrt{\frac{1}{N!}} \sum_P \epsilon_P \left(\hat{P}_{\rm space}\Phi_A\right) \sum_\mu^f U_{\mu\lambda}(\hat{P})\Theta_\mu \qquad (208) \\ &= \sqrt{\frac{1}{f}} \sum_\mu^f \Theta_\mu \Phi_{A\mu\lambda} \ , \qquad (209)\end{aligned}$$

where we define a set of spatial functions

$$\Phi_{A\mu\lambda} = \sqrt{\frac{f}{N!}} \sum_P \epsilon_P U_{\mu\lambda}(\hat{P})\hat{P}\Phi_A \ . \qquad (210)$$

This has the appearance of a projection operator on Φ_A for a representation with matrices $V_{\mu\lambda}(\hat{P}) = \epsilon_P \ U_{\mu\lambda}(\hat{P})$. This is the *conjugate* representation to that supported by the spin functions, and appears in the Young theory as the reversal of the roles of rows and columns, e.g.,  (spin) $\rightarrow$  (space). Note that all

$\Theta_\mu\,, \mu = 1, 2, \ldots, f$ are involved in each of the space–spin functions $\Phi_{A\lambda}$.

For the the coupling coefficients

$$d_{pq}^{A\lambda,B\mu} = \langle \Phi_{A\lambda} | \hat{E}_{pq} | \Phi_{B\mu} \rangle \;, \tag{211}$$

as with determinants, a non–zero contribution will arise only if Φ_A and Φ_B differ by the orbital excitation $\phi_q \rightarrow \phi_p$. Ignoring any complications which arise from doubly occupied orbitals, we must again have $\Phi_A = \hat{L}\hat{E}_{pq}\Phi_B$, where $\hat{L}$ is the appropriate line–up permutation. Inserting (208) into (211) we obtain

$$\begin{aligned} d_{pq}^{A\lambda,B\mu} &= \frac{1}{N!} \sum_{PQ} \epsilon_P \epsilon_Q \langle \hat{P}\Phi_A | \hat{Q}\hat{L}^{-1}\Phi_A \rangle \sum_{\rho\sigma} U_{\rho\lambda}(\hat{P}) U_{\sigma\mu}(\hat{Q}) \langle \Theta_\rho | \Theta_\sigma \rangle \\ &= \frac{1}{N!} \sum_{PQ} \epsilon_P \epsilon_Q \langle \hat{P}\Phi_A | \hat{Q}\hat{L}^{-1}\Phi_A \rangle \sum_{\rho} U_{\rho\lambda}(\hat{P}) U_{\rho\mu}(\hat{Q}) \\ &\qquad\qquad \text{since the spin functions are orthogonal} \\ &= \frac{1}{N!} \sum_{PQ} \epsilon_P \epsilon_Q \langle \hat{P}\Phi_A | \hat{Q}\hat{L}^{-1}\Phi_A \rangle U_{\lambda\mu}(\hat{P}^{-1}\hat{Q}) \;, \end{aligned} \tag{212}$$

using the representation property of $\mathbf{U}(\hat{P})$. Orbital orthogonality then gives the requirement that $\hat{P} = \hat{Q}\hat{L}^{-1}$, and so

$$\begin{aligned} d_{tu}^{A\lambda,B\mu} &= \frac{1}{N!} \sum_{Q} \epsilon_L \epsilon_Q^2 U_{\lambda\mu}(\hat{L}\hat{Q}^{-1}\hat{Q}) \\ &= \epsilon_L U_{\lambda\mu}(\hat{L}) \;. \end{aligned} \tag{213}$$

Thus knowledge of the line–up permutation and the representation matrix elements is sufficient to generate any desired one–particle coupling coefficient.

The above is based on the assumption that ϕ_p and ϕ_q are singly occupied in Φ_A, Φ_B respectively. When one or both orbitals are doubly occupied, further considerations are necessary. Firstly, many of the spin functions give rise to vanishing $\Phi_{A\lambda}$ because of the operation of the Pauli principle acting through the antisymmetrizer. If the orbitals are ordered such that the doubly occupied appear first in their respective pairs, then only those spin functions which couple each pair to singlet are allowed. This of course gives a drastic reduction in the number of possible spin functions, since it is now f_S^N with N referring to the number of singly occupied orbitals only. Following this, there are slight complications to the above scheme for the coupling coefficients; there appear four distinct cases depending on the excited orbital occupancies, of which (213) is one.

How are the relevant representation matrices obtained? Equation (213) shows that one needs all of the representation matrices for all possible cyclic permutations. These matrices can be generated by writing the cycle as a sequence of elementary transpositions,

$$\hat{C}(i,j) = \hat{C}(j-1,j)\hat{C}(j-2,j-1)\ldots\hat{C}(i+1,i+2)\hat{C}(i,i+1) \;; \tag{214}$$

the representation matrices for these transpositions are very sparse, and can be obtained from the shapes of the Young tableaux[86]. The matrix for the cycle is then obtained by matrix multiplication. Unfortunately, this algorithm is too slow for

practical use, and it is much better to precompute and store all of the necessary matrices. The number of matrices that must be stored can be reduced considerably by using a resolution of the identity analogous to that used to factorize two-body matrix elements into sums of products of one-body elements (equation (183)). If we introduce a (fictitious) additional orbital ϕ_α which is defined to occur lexically always after any other orbital, the following identity holds.

$$\hat{E}_{pq} = \hat{E}_{pa}\hat{E}_{aq} \tag{215}$$

This allows one to make use of just those cycles involving the last electron, since the orbital ϕ_a will always be occupied by only this electron in the ordered orbital product string. This is the basis of an efficient algorithm for matrix element evaluation that is fast enough for general use in full and other CI computations[87].

5.4 Molecular Dissociation and the MCSCF method

As discussed in section 1.6, in many situations electron correlation effects are purely of the 'dynamic' type, in the sense that Hartree-Fock is a good zero-order approximation, and under such circumstances, single-reference methods provide an efficient and accurate way to getting correlation energies and correlated wavefunctions. However, wherever bonds are being broken, and for many excited states, the Hartree-Fock determinant does not dominate the wavefunction, and may sometimes be just one of a number of important electronic configurations. If this is the case, single-reference methods, which often depend formally on perturbation arguments for their validity, are inappropriate, and one must seek from the outset to have a first description of the system that is better than Hartree-Fock. Only then can one go on to attempt to recover the remaining dynamic correlation effects.

As in H_2, we can build a general qualitatively correct wavefunction by selecting a number of configurations which are meant to describe all possible dissociation pathways, etc., and then writing the wavefunction as a linear CI expansion

$$\Psi = \sum_I^M c_I \Phi_I \ . \tag{216}$$

The energy is then minimized with respect to not only the c_I (as in the CI method), but also to changes in the common set of orbitals ϕ_t which are used to construct the Φ_I. This orbital optimization is analogous to what is done in the SCF method, hence the name *multiconfiguration self consistent field* (MCSCF), which is given to this approach. Provided all the necessary configurations are included in the set Φ_I, then the method should give a qualitatively correct description of the electronic structure.

Nearly all molecules dissociate to valence states of their constituent atoms, in which only the valence orbitals (e.g., $2s, 2p$ in carbon) are occupied. So ignoring the complications which might occur for Rydberg molecular states, a good description can be obtained by including Φ_I which have only valence orbitals of the molecule occupied. This has important computational consequences, and we distinguish in a calculation the relatively small number of *internal* (or valence) orbitals $\phi_t, \phi_u, \phi_v, \ldots$ from the usually much larger number of *external* orbitals $\phi_a, \phi_b, \ldots$,

which are unoccupied in all configurations, and so actually are not part of the wavefunction. We continue to use the notation $\phi_p, \phi_q, \phi_r, \ldots$ to denote general molecular orbitals from any set. The internal and external orbitals take the roles of the occupied and virtual orbitals in an SCF calculation; as the calculation proceeds, the internal and external orbitals are mixed amongst each other until the optimum internal orbitals are found. Taking these ideas to the extreme suggests the use of a CI expansion consisting of all possible configurations in the valence space, i.e., a FCI type of wavefunction. This approach [88,89,90] is often termed *complete active space SCF* (CASSCF) and has the feature that it is to some extent a 'black box'; the sometimes rather difficult problem of selecting suitable configurations Φ_I is replaced by the simpler identification of important orbitals. If the active orbital space coincides with the true valence space, then correct dissociation at all limits is automatically guaranteed, although there may be many configurations included which are completely unimportant. As a simple example, consider the ground state of N_2. The quartet spin N atom ground state is described by the configuration $2p_x^\alpha 2p_y^\alpha 2p_z^\alpha$. On bringing two N atoms together, one can make 20 CSFs with the correct spin (singlet) and space (A_g in D_{2h}) symmetries, of which one is dominant near equilibrium bond length, but all of which are important at dissociation. The CASSCF wavefunction, a FCI expansion of 6 electrons in 6 orbitals, contains 32 CSFs. Although the ansatz may be wasteful in this way, we note that a complete CI expansion enables the use of special efficient techniques[91], so a CASSCF calculation may actually be easier than a smaller more general MCSCF calculation with the same internal orbital space.

5.5 Determination of MCSCF wavefunctions

We have considered earlier how the matrix elements $H_{IJ} = \langle \Phi_I | \hat{H} | \Phi_J \rangle$ are obtained in terms of one and two electron integrals h_{tu}, $(tu|vw)$ and coupling coefficients d_{tu}^{IJ}, D_{tuvw}^{IJ}:

$$\langle \Phi_I | \hat{H} | \Phi_J \rangle = \sum_{tu} d_{tu}^{IJ} h_{tu} + \tfrac{1}{2} \sum_{tuvw} D_{tuvw}^{IJ} (tu|vw) \; . \tag{217}$$

Thus the expression for the energy is

$$\begin{aligned} E &= \langle \sum_I c_I \Phi_I | \hat{H} | \sum_J c_J \Phi_J \rangle \\ &= \sum_{tu} \sum_{IJ} c_I c_J d_{tu}^{IJ} h_{tu} + \tfrac{1}{2} \sum_{tuvw} \sum_{IJ} c_I c_J D_{tuvw}^{IJ} (tu|vw) \\ &= \sum_{tu} d_{tu} h_{tu} + \tfrac{1}{2} \sum_{tuvw} D_{tuvw} (tu|vw) \; , \end{aligned} \tag{218}$$

where we see the introduction of the one and two electron *density matrices* d_{tu}, D_{tuvw}, which in this context can be viewed as expectation values of the coupling coefficients. This energy expression is the quantity which must be made stationary with respect to changes in the CI coefficients c_I and the orbitals ϕ_t, subject to the

constraints

$$\sum_I c_I^2 = 1 \qquad \text{(normalization)} \tag{219}$$

$$\langle \phi_t | \phi_u \rangle = \delta_{tu} \qquad \text{(orbital orthogonality)} \, . \tag{220}$$

For the CI coefficients, introducing a Lagrange multiplier $\mathcal{E}$ for the first constraint, and setting the differential with respect to c_I to zero, gives the stationary conditions

$$\sum_J \langle \Phi_I | \hat{H} | \Phi_J \rangle c_J - \mathcal{E} c_I = 0 \, , \tag{221}$$

i.e., the usual matrix eigenvalue equations obtained in regular CI theory. For the orbitals, the most straightforward approach is to parametrize orthogonal rotations $\mathbf{U}$ amongst the orbitals ($\phi_t \leftarrow \sum_p \phi_p u_{pt}$) by means of the matrix elements R_{tu} of an antisymmetric matrix. Any orthogonal matrix may be represented as

$$\mathbf{U} = \exp(\mathbf{R}) \qquad \text{where } \mathbf{R}^\dagger = -\mathbf{R} \, . \tag{222}$$

The advantage of this formulation is that the $\frac{1}{2}m(m+1)$ orthogonality constraints are automatically satisfied, leaving $\frac{1}{2}m(m-1)$ free parameters which are contained in the lower triangle of $\mathbf{R}$. There is then no need for Lagrange multipliers, and numerical methods for unconstrained optimization may be used.

To derive the variational conditions for orbital rotations, we note that the orbitals vary on $\mathbf{R}$ through (222) as

$$\left. \frac{\partial \phi_p}{\partial R_{rs}} \right|_{\mathbf{R}=\mathbf{0}} = \delta_{sp} \phi_r - \delta_{rp} \phi_s \, , \tag{223}$$

and that the integrals h_{tu}, $(tu|vw)$ given by (35), (36) are quadratic and quartic, respectively, in the orbitals. Then we obtain

$$\left. \frac{\partial}{\partial R_{rs}} h_{tu} \right|_{\mathbf{R}=\mathbf{0}} = (1 - \tau_{rs})(1 + \tau_{tu}) \delta_{st} h_{ru} \tag{224}$$

$$\left. \frac{\partial}{\partial R_{rs}} (tu|vw) \right|_{\mathbf{R}=\mathbf{0}} = (1 - \tau_{rs})(1 + \tau_{tu})(1 + \tau_{tu,vw}) \delta_{st} (ru|vw) \, , \tag{225}$$

where the operator τ_{ij} permutes the labels i, j in what follows it. Thus the derivative of the energy, which is zero for the converged wavefunction, is given by

$$0 = \frac{\partial E}{\partial R_{rs}} = 2(1 - \tau_{rs}) F_{rs} \, , \tag{226}$$

with

$$F_{rs} = \sum_u d_{su} h_{ru} + \sum_{uvw} D_{suvw} (ru|vw) \, . \tag{227}$$

Equations (221) and (226) must be solved to obtain the MCSCF wavefunction. Note that for some orbital rotations R_{rs}, the variational condition (226) is always obeyed automatically; for example, if both r, s are external, then the density matrix elements are all zero. The same can occur in a more subtle way for certain internal–internal orbital rotations, e.g., for a CASSCF, all internal–internal rotations show this behaviour. When an R_{rs} behaves like this it is known as a redundant variable,

and is best removed from the optimization altogether[92]. Note also that (226) is highly non–linear, in contrast to the linear eigenvalue problem which appears in the CI method; E is 4^{th} order in the orbitals, and infinite order in $\mathbf{R}$, since the orbitals are in fact periodic functions because of the orthogonality constraint.

In order to solve numerically the variational equations (221) and (226), the standard approach is to use some kind of quasi-Newton approach[93,94] that utilizes the gradients of the energy expression to construct a Taylor series for the energy in powers of the parameters that express changes in the wavefunction. Truncation of this power series gives an approximate energy expression that is accurate for small displacements, and which is easier to minimize than the full energy expression. For a given approximate solution, we construct the gradient vector

$$g_\lambda = \left(\frac{\partial E}{\partial p_\lambda} \right)_{\mathbf{p}=\mathbf{0}} \tag{228}$$

and hessian matrix

$$h_{\lambda\mu} = \left(\frac{\partial^2 E}{\partial p_\lambda \partial p_\mu} \right)_{\mathbf{p}=\mathbf{0}} \tag{229}$$

where the set of parameters $\{p_\lambda\}$ contains the changes in CI coefficients $\{\Delta c_I\}$ and the non-redundant orbital change generators $\{R_{rs}\}$. The approximate energy expression

$$E_2(\mathbf{p}) = E_2(\mathbf{0}) + \sum_\lambda g_\lambda p_\lambda + \tfrac{1}{2} \sum_{\lambda\mu} h_{\lambda\mu} p_\lambda p_\mu \tag{230}$$

is then minimized by solving the linear equations

$$0 = g_\lambda + \sum_\mu h_{\lambda\mu} p_\mu \tag{231}$$

The solution $\mathbf{p}$ defines a step that is applied to the wavefunction to improve it. Thus the overall procedure is iterative, each iteration consisting of the construction of the energy, gradient and hessian, followed by solution of the linear Newton-Raphson equations. The Newton-Raphson equations can be very large in dimension, particularly for a large CASSCF full CI expansion; therefore, usually, they have to be solved iteratively as well, using relaxation or expansion vector techniques[95] similar to the Davidson diagonalization algorithm[3]. These iterations are usually referred to as *microiterations* to distinguish them from the enclosing *macroiterations* in each of which a new expansion point is defined.

The generic Newton-Raphson algorithm suffers in this context from two distinct problems associated with robustness and efficiency. First of all, the second-order expansion (230) is valid only for small displacements $\mathbf{q}$, and it is often the case that the predicted step length is outside the 'trust region' of the truncated Taylor series. Modifications that restrict the step length[96], or recast the linear equation system as an eigenvalue problem such that the step length is automatically restricted (*augmented hessian* method[97]) are helpful in improving global convergence. Secondly, however, even with such methods, as many as 20 macroiterations may be required, and each macroiteration is expensive. For each new set of orbitals, in order to construct the gradient and hessian, a subset of the molecular-orbital electron-repulsion

integrals must be constructed, specifically those with up to two external indices $(\mathbf{J}^{tu}, \mathbf{K}^{tu})$, by a computationally demanding transformation of the atomic-orbital integrals, which themselves have to be read from disk or computed on the fly. It is therefore highly desirable to reduce the number of macroiterations. Both problems are solved by adopting an ansatz[98,91] in which the microiterations involve optimization of an approximate energy functional that is second order in the orbital changes themselves, $\Delta\mathbf{T} = \mathbf{U} - \mathbf{1}$, rather than in the generators $\mathbf{R}$. This energy functional is periodic in the orbitals, just like the true energy, and its use gives an algorithm that is much more robust; in fact, in almost all cases, quadratic convergence is seen from the outset, and typically only three macroiterations are needed. Of course, there is additional complication in that the microiterations are solving non-linear rather than linear equations, but these can be effectively addressed using convergence accelerators such as DIIS[99].

5.6 Multireference Perturbation Theory

In order to go beyond a qualitatively correct MCSCF wavefunction Ψ^{REF} and recover as much of the correlation energy as possible, as in the single-reference case, we begin by writing the exact wavefunction in a perturbation series

$$\Psi_{\mathrm{Exact}} = \Psi^{\mathrm{REF}} + \lambda\Psi^{(1)} + \lambda^2\Psi^{(2)} + \dots \,, \tag{232}$$

where λ is an ordering parameter which will eventually be set to 1. Suppose that we can find an operator $\hat{H}^{(0)}$ such that $\hat{H}^{(0)}\Psi^{\mathrm{REF}} = E^{(0)}\Psi^{\mathrm{REF}}$. In the particular case where Ψ^{REF} is the solution of the SCF equations, an appropriate $\hat{H}^{(0)}$ is the many-electron Fock operator,

$$\hat{H}^{(0)} = \sum_{i}^{N} \hat{f}(i) = \sum_{tu}^{m} f_{tu}\hat{E}_{tu} \tag{233}$$

where $\hat{f}$ is the orbital Fock operator; in other cases it may or may not be possible to find a suitable operator, but the arguments we develop still hold. If we write $\hat{H} = \hat{H}^{(0)} + \lambda\hat{H}^{(1)}$, and separate terms of different order in λ in the Schrödinger equation, at first order we obtain

$$\left(\hat{H}^{(0)} - E^{(0)}\right)\Psi^{(1)} + \hat{H}^{(1)}\Psi^{\mathrm{REF}} - \langle\Psi^{\mathrm{REF}}|\hat{H}^{(1)}|\Psi^{\mathrm{REF}}\rangle\Psi^{\mathrm{REF}} = 0 \,. \tag{234}$$

We expand $\Psi^{(1)}$, the first order correction to the wavefunction, and also $\hat{H}\Psi^{\mathrm{REF}}$, the action of the full hamiltonian on the approximate wavefunction, as linear combinations of N-electron configurations in the full space,

$$\begin{aligned} \Psi^{(1)} &= \sum_{I} \Phi_I c_I^{(1)} \\ \hat{H}\Psi^{\mathrm{REF}} &= \sum_{I} \Phi_I h_I \,, \end{aligned} \tag{235}$$

and assume (although again this is not critical) that $\hat{H}^{(0)}\Phi_I = \mathcal{E}_I\Phi_I$. This will be true for the Fock $\hat{H}^{(0)}$ ($\mathcal{E}_I$ is then the sum of the Fock eigenvalues for the orbitals

occupied in Φ_I), and approximately true for others. The first order equation then becomes

$$\sum_I c_I^{(1)} \Phi_I \left(\mathcal{E}_I - E^{(0)} \right) = -\sum_I h_I \Phi_I + \langle \Psi^{\mathrm{REF}} | \hat{H} | \Psi^{\mathrm{REF}} \rangle \Psi^{\mathrm{REF}} \ . \tag{236}$$

This tells us that the basis functions which are required for $\Psi^{(1)}$ are exactly those which appear in the action of $\hat{H}$ on Ψ^{REF}. This set of functions is the *first order interacting space*. Recall that the hamiltonian consists of single and double excitation operators; this means that in turn the first order space consists of all those configurations which are at most doubly excited with respect to the reference function Ψ^{REF}. In the language of second quantization, the first-order space consists of all the non-null configurations $\{\hat{E}_{tu,vw}\Psi^{\mathrm{REF}}\}$.

These arguments can be generalized to higher orders of perturbation theory; at second order, configurations related to the first-order wavefunction by up to double excitations will be introduced, and so the second-order interacting space consists of configurations which are singly, doubly, triply and quadruply excited relative to Ψ^{REF}.

One route to carry these ideas forward is to simply apply regular Rayleigh-Schrödinger perturbation theory to obtain the perturbation series for the energy. With the choice of Fock $\hat{H}^{(0)}$, this is the single-reference Møller-Plesset theory (MP)[100,101] or Many-Body Perturbation Theory (MBPT)[102]. For multiconfigurational Ψ^{REF}, the choice of zero order hamiltonian is not so obviously unique, but a number of different variants have been very successfully used[103,104,105,106,107]. These are generally non-diagonal in the configuration basis, and so solution of the first-order equations must be carried out iteratively; in contrast, for a Hartree-Fock reference with canonical molecular orbitals, each Slater determinant is an eigenfunction of $\hat{H}^{(0)}$, and so the first-order equations have an explicit analytic solution.

Multireference perturbation theory at second order (MRPT2 or CASPT2) is now well established as a robust and reliable technique particularly, for example, in the computation of electronic excitation energies[106], and is computationally feasible in almost all cases where the underlying MCSCF or CASSCF calculation is possible. Third-order perturbation theory[103,108] can also be carried out for smaller systems, and the results show significant differences from second order, indicating the need for caution in the use of CASPT2.

5.7 Multireference Configuration Interaction

Although perturbation theory may be a dangerous tool to rely on, the interacting space hierarchy concept provides useful insight on how to design other methods. If we consider doing a variational CI calculation, we now know that, even though FCI may be impossible, we expect to obtain most of the correlation energy using a basis consisting of the first-order interacting space. In the case of an RHF reference wavefunction Ψ^{REF} this is the singles and doubles (CISD) method, with the basis consisting of all Slater determinants which are related to Ψ^{REF} by a single or double spin-orbital excitation. Strictly speaking, for RHF Ψ^{REF}, singles do not formally enter until second order perturbation theory, but in practice their effect can be quite

significant, and there are fewer of them than doubles, and so they are invariably included as well.

The same kind of approach can be taken for an MCSCF $\Psi^{\rm REF}$. The first-order space is certainly spanned by a wavefunction of the form

$$\Psi = \sum_I c_I \Phi_I + \sum_{Sa} c_a^S \Phi_S^a + \sum_{Pab} C_{ab}^P \Phi_P^{ab} \ , \tag{237}$$

where the three types of configuration Φ_I, Φ_S^a, Φ_P^{ab} contain respectively 0, 1, 2 occupied external orbitals, and the set of configurations is the union of the sets of CSFs obtained by making all possible single and double excitations on each reference configuration in turn. For the case that $\Psi^{\rm REF}$ consists of a single closed shell configuration, (237) is the single-reference CISD wavefunction; when $\Psi^{\rm REF}$ contains more than one configuration, variational treatment of (237) is usually referred to as *multireference CI* (MRCI)[109,110,111,112].

Since there are usually many more external orbitals than internal orbitals, the doubly external configurations Φ_P^{ab} are expected to be by far the most numerous, just as in the single-reference case, and we focus attention on these in considering what work has to be done in evaluating hamiltonian interactions. In the general multi-reference case, it is not possible to arrive at explicit matrix-oriented expressions for the hamiltonian matrix elements. However, some simplification beyond the general CI matrix element strategy presented in section 5.3 is certainly possible; just as in the single-reference case, there is special structure associated with the pairs of external orbitals ϕ_a, ϕ_b. In the formation of CSFs Φ_P^{ab}, it is advantageous to take the occupied orbital string which is inserted into equation (207) such that the orbitals ϕ_a and ϕ_b appear as functions of the coordinates of electrons 1 and 2 respectively; this means that the function is pure singlet or triplet coupled in the two external orbitals, exactly as in the single-reference case, and allows for some simplification in matrix element evaluation. The structure of the wavefunction in the external orbitals is then no more complicated than in the single-reference problem, and so closed formulae for those parts involving external orbitals are obtainable; for example, the contribution from all external integrals has exactly the same form as in single-reference SDCI, and can be obtained efficiently by computing the external exchange matrices for each pair P. However, for the internal orbitals, the CSFs are completely general in character, and ultimately one must compute one and two particle coupling coefficients using the general techniques of section 5.3. For example, that part of the hamiltonian containing the Coulomb integrals, $\sum_{tuab} J_{ab}^{tu} \hat{E}_{ab} \hat{E}_{tu}$, gives rise to matrix elements

$$\langle \Phi_P^{ab} | \hat{H} | \Phi_Q^{cd} \rangle = \frac{1}{2} \delta_{pq} \sum_{mn} \alpha_{mn}(P,Q) \, (1 + p\tau_{ab})(1 + q\tau_{cd}) \, \delta_{bd} \, J_{ac}^{tu} \ , \tag{238}$$

where $p = \pm 1$ according to whether Φ_P^{ab} is singlet or triplet coupled in the external space. $\alpha_{tu}(P,Q)$ is simply a one particle coupling coefficient for the operator $\hat{E}_{tu}$ between the functions Φ_P^{ab} and Φ_Q^{ab},

$$\alpha_{tu}(P,Q) = \langle \Phi_P^{ab} | \hat{E}_{tu} | \Phi_Q^{cd} \rangle \ . \tag{239}$$

Although coupling coefficient evaluation is required, all the coupling coefficients are completely independent of the external orbital labels; thus many hamiltonian matrix elements share the same coupling coefficients in a regular manner. Discovery of this property [113,114] first opened the way for large scale MRCI calculations. Although the coupling coefficient evaluation problem is dramatically reduced by exploiting these special properties, the MRCI method is still severely restricted by computational difficulties. For even quite modest numbers of reference configurations, the number of pair functions Φ_P^{ab} can be rather large; this means that the dimension of the hamiltonian matrix can easily exceed the length of vector which can be stored on the computer, and, more importantly, the number of matrix elements which must be evaluated becomes completely unmanageable. Nevertheless, benchmark calculations, in which MRCI results are compared with those from full CI in the same basis, indicate that MRCI is the ab initio method of choice for all circumstances in which single determinant descriptions do not work, and that very high accuracy may be obtained[115,116].

An alternative formulation which avoids the rapid increase in basis size with the number of reference configurations is possible [113]. Instead of selecting singly and doubly excited CSFs from each reference configuration, we can construct configurations by applying excitation operators to the reference wavefunction as a single entity:

$$\Psi = \sum_{tuvw} C^{tuvw} \hat{E}_{tu,vw} \Psi^{\rm REF} + \sum_{tuva} C_a^{tuv} \hat{E}_{at,uv} \Psi^{\rm REF} + \sum_p \sum_{ab} \sum_{t \geq u} C_{ab}^{tup} \, \tfrac{1}{2} \left(\hat{E}_{at,bu} + p \hat{E}_{au,bt} \right) \Psi^{\rm REF} \; . \tag{240}$$

This is the *internally contracted* MRCI (ICMRCI)[113,117,118] wavefunction, and it is obvious that the number of configurations is now independent of the number of reference functions, depending only on the numbers of internal and external orbitals. In this way, the size of CI expansion is reduced typically by one or two orders of magnitude; the configuration set, however, still spans the first order interacting space, and although CMRCI can be considered as only an approximation to MRCI, benchmark calculations show that in most cases the extra error introduced by the contraction is several times smaller than the error of MRCI relative to full CI[118]. The price that is paid is that the configurations are now much more complicated, being in fact linear contractions of CSFs according to the values of the reference coefficients. This means that coupling coefficient evaluation is now a formidable problem; the simple CSF coupling coefficients are replaced by reduced density matrices of high order. For example, for the Coulomb integrals considered previously, the coupling coefficients are

$$\alpha_{tu}(vwp, xyq) = \delta_{pq}(1 + p\tau_{xy}) \, \langle \Psi^{\rm REF} | \hat{E}_{vx,wy,tu} | \Psi^{\rm REF} \rangle \; . \tag{241}$$

This third-order density matrix is evaluated using the general resolution-of-identity techniques used in the full CI problem, i.e.,.

$$\langle \Psi^{\rm REF} | \hat{E}_{vx,wy,tu} | \Psi^{\rm REF} \rangle = \sum_K \langle \Psi^{\rm REF} | \hat{E}_{vx,wy} | \Phi_K \rangle \langle \Phi_K | \hat{E}_{tu} | \Psi^{\rm REF} \rangle + \text{ lower order terms} \tag{242}$$

where the $\{\Phi_K\}$ are appropriate CSFs. For a given bra (vw) and ket (xy), all the matrix elements $\langle\Psi^{\rm REF}|\hat{E}_{vx,wy}|\Phi_K\rangle$ are found by successively applying the operators $\hat{E}_{av}, \hat{E}_{xa}, \hat{E}_{aw}, \hat{E}_{ya}$ (ϕ_a is a 'fictitious' unoccupied orbital) to $\Psi^{\rm REF}$. For processing a given Coulomb matrix $\mathbf{J}^{tu}$, these matrix elements are combined with precomputed $\langle\Phi_K|\hat{E}_{tu}|\Psi^{\rm REF}\rangle$.

An additional complication in ICMRCI is that the configurations are non–orthogonal in a non–trivial way, and their orthogonalization can be a computational bottleneck[117]. For this reason, the standard approach to ICMRCI is a hybrid that combines the best features of uncontracted and contracted wavefunctions [118]; contraction is carried out only where it is easiest, and of most benefit, namely for the doubly external configurations, and the all-internal and singly-externals are left uncontracted.

An unfortunate feature of an MRCI calculation is that, just as in the single-reference CISD case, the energy is not an *extensive* function of the number of electrons as it should be. This undesirable feature of any truncated variational CI calculation can to some extent be avoided in MRCI by error cancellation across a potential energy surface; provided, for example, dissociation asymptotes are computed as supermolecules rather than by adding fragment energies, reasonable results can be obtained for dissociation energies. It is also true that the size-consistency errors for MRCI are usually much less than for single-reference CISD, since MRCI already contains some of the important quadruple configurations. However, the effects can never be completely avoided.

One way to view the lack of size-consistency in variational CI is by considering the Rayleigh quotient correlation energy functional itself,

$$\mathcal{E} = \frac{\langle\Psi|\hat{H} - E^{\rm REF}|\Psi\rangle}{\langle\Psi|\Psi\rangle}\,. \tag{243}$$

Suppose Ψ is, for example, restricted to contain double excitation configurations only, and that the coefficient of the reference wavefunction is kept fixed (intermediate normalization, $\langle\Psi|\Psi^{\rm REF}\rangle = 1$). Then the numerator of this expression can be shown to grow linearly with system size $\mathcal{N}$; however, the denominator also grows, but as $1 + \lambda\mathcal{N}$, where λ is a constant. This spoils the proper linear scaling of the correlation energy. In the absence so far of problem-free multireference coupled-cluster approaches, this analysis gives rise to a number of approximate ways to correct for the effects of lack of extensivity. The simplest, the Davidson or '+Q' correction [119,26], involves a straightforward rescaling of the correlation energy by $\langle\Psi|\Psi\rangle$, i.e. replacing the denominator of (243) by 1 once the wavefunction has been determined. More explicitly,

$$\mathcal{E}^{\rm CI+Q} = \frac{1 - c_0^2}{c_0^2}\mathcal{E}^{\rm CI}\,, \tag{244}$$

where c_0^2 is the weight of the reference wavefunction $\Psi^{\rm REF}$ in the final normalized CI wavefunction. Alternative approaches (ACPF[120], AQCC[121]) introduce at the outset a denominator in the energy functional that does not increase with system size. This modified approximate functional is then minimized to determine the wavefunction and energy.

6 Integral-direct methods

Since the first formulation of the LCAO finite basis scheme for molecular Hartree-Fock calculations, computer implementations of this method have traditionally been organised as a two-step process. In the first step all the two-electron repulsion integrals (ERIs) over four contracted Gaussian basis functions are calculated and stored externally on disk, while the second step comprises the iterative solution of the Hartree-Fock Roothaan equations, where in each iteration the integrals from the first step are retrieved from disk and contracted with the present density matrix to form a new Fock matrix. This subdivision of the computational process into the two steps was motivated by the relatively high CPU cost necessary to generate the ERIs using rather complicated analytical recurrence relations, which was clearly dominating a Hartree-Fock calculation. For post Hartree-Fock calculations, which are traditionally formulated using the canonical SCF orbitals from a preceding Hartree-Fock calculation as a basis, an integral transformation of the AO ERIs generated in the first step to the canonical MO basis is required prior to the actual correlated calculation. The computational complexity of such an integral transformation scales with $\mathcal{O}(\mathcal{N}^5)$, where $\mathcal{N}$ is a measure of the molecular size or the number of correlated electrons. It also is quite memory and disk intensive. The amount of disk space required to hold the AO (and MO) ERIs scales as $\mathcal{O}(\mathcal{N}^4)$.

The last several decades have witnessed continuous rapid advances in computer technology, and in fact the progress in CPU technology has been much faster than the development of I/O facilities. Furthermore, much effort has been invested in improving integration techniques. Hence, with the conventional two step procedure one now faces the dilemma of being able to compute large numbers of integrals rapidly, but spending a relatively large amount of time and resources in their storage and retrieval. In fact, the size of chemical systems one can handle today with the conventional method described above is primarily limited by the disk space required to store the AO ERIs, rather than the CPU time required to compute these. Integral-direct methods offer a solution to this problem. The philosophy is to eliminate the $\mathcal{O}(\mathcal{N}^4)$ bottleneck of AO ERI storage altogether by recomputing the ERIs on the fly whenever needed, thus trading disk space and I/O load at the expense of additional CPU time. Integral-direct methods were first used in Hartree-Fock (SCF) theory almost two decades ago ("direct SCF" approach by Almlöf *et al.* [122]), and it constituted a break of a paradigm at that time. These days, direct SCF programs are part of virtually all ab initio program packages used by the community. Since the pioneering direct SCF work integral-direct methods have been extended to electron correlation methods like multiconfigurational SCF [123,124,60], many-body perturbation theory [MBPT(2)] [125,126,127,60], MBPT(2) gradients [128] and coupled cluster methods [129,130,60]. In contrast to the SCF method, where the ERIs over atomic orbitals (AOs) (i.e., the basis functions) are immediately contracted to the Fock matrix in AO basis, and only AO integrals are needed, correlation methods including MCSCF require an AO to MO integral transformation, as discussed above. Hence an intermediate four-indexed quantity (rather than the two-indexed Fock matrix in direct SCF procedures) arises and has to be dealt with. A full 4-index transformation, carried out as four quarter transformations

has a flop count that scales as $\mathcal{O}(m^5)$ with the number of basis functions m, and has $\mathcal{O}(m^4)$ storage requirements. At a first sight the storage requirements for such an integral transformation seem to rule out any integral-direct implementation of a correlated method, since no savings to the conventional method seem to be possible. Fortunately enough, however, most correlation methods can be reformulated in terms of AO ERIs and a reasonably small subset of MO integrals [20]. Such MO integral subsets typically have two indices restricted to the *occupied* orbital space of dimension m_{occ}, which is usually much smaller than m. For example, the computation of the MBPT(2) energy requires only the exchange integrals $(ia|jb)$, while for direct MCSCF and all other correlation methods the Coulomb $(ij|pq)$ and exchange $(ip|jq)$ MO integrals are needed. The disk space necessary to hold such a subset of MO integrals then is $\mathcal{O}(m_{\mathrm{occ}}^2 m^2)$, i.e. for a ratio $m/m_{\mathrm{occ}} \approx 10$ this means savings of a factor of 100 and larger in the storage requirements, compared to the conventional method. In the work by Schütz *et al.* [60] it was demonstrated that for integral-direct implementations of most electron correlation methods (MP2-4(SDQ), CCSD, QCISD, BCCD, MCSCF, MRPT2/3, MRCI) only three integral-direct kernel procedures are necessary. The only exception are methods involving triply or higher excited configurations. Apart from the trivial Fock matrix construction routine these involve a generalized partial integral transformation and a module for the construction of *external exchange operators* which corresponds basically to a two-index contraction of AO ERIs with the doubles amplitude matrices, backtransformed to AO basis, as explained in section 2.4.

Integral-direct methods are especially powerful in the context of local correlation methods [57,58,59,53,54]. Here, additional savings are possible by describing occupied and virtual correlation spaces in terms of localized MOs and projected (non-orthogonal) AOs, respectively, which in turn allows to exploit the short range character of dynamic correlation (asymptotic distance dependence is $\propto r^{-6}$ in insulators). In such a scheme, a hierarchical treatment of different electron pairs is possible, depending on relative distance of the corresponding LMOs. Furthermore, the virtual space spanned by the non-orthogonal projected AOs can be partitioned into domains (cf. section 4). As a result of this, only very small subsets of (transformed) integrals are required even for methods including triply excited configurations, and the number of these integrals scales linearly with the molecular size. This, in turn, opens the path for $\mathcal{O}(\mathcal{N})$ electron correlation methods and hence the treatment of very large molecular systems at a level of very high accuracy.

6.1 The direct SCF method

In the most naive implementation, writing a computer code for a direct SCF scheme comprises little more than just replacing the reading of one- and two-electron integrals in the SCF algorithm by their repeated calculation. However, in order to get an efficient program, it is clear that such a change in the paradigm calls for major restructuring of the code. Since the computation of the two-electron integrals is rather expensive, a direct algorithm should be *integral driven*, i.e. integral evaluation concerns should dictate the order of events. Once an integral has been computed, it should be used to the maximum extent possible, as long as no external

storage is invoked.

Two-electron repulsion integrals (ERIs) are integrals of the following form (assuming real basis functions)

$$(\mu\rho|\nu\sigma) = \int\int \chi_\mu(1)\chi_\rho(1) r_{12}^{-1} \chi_\nu(2)\chi_\sigma(2) dr_1 dr_2, \tag{245}$$

where $\chi_\mu, \chi_\rho, \chi_\nu, \chi_\sigma$ denote contracted Cartesian Gaussians,

$$\chi_\mu = \sum_\alpha c_{\alpha\mu} \bar{\chi}_\alpha(r) = \sum_\alpha c_{\alpha\mu} (\bar{\chi}_\alpha^x(x) \bar{\chi}_\alpha^y(y) \bar{\chi}_\alpha^z(z)), \tag{246}$$

with

$$\bar{\chi}_\alpha^x(x) = (x - x_\alpha)^{k_\alpha} \exp[-a_\alpha (x - x_\alpha)^2], \tag{247}$$

and $\bar{\chi}_\alpha^x(x) \ldots$ symbolize Cartesian components of primitive Gaussians, centred at origins $\mathbf{r}_\alpha = (x_\alpha, y_\alpha, z_\alpha)$. Usually, these centres are taken to be the atoms, but sometimes basis functions are also positioned between atoms. One of the most important reason to choose Gaussians as basis functions is the separability into products of Cartesian components, as indicated in eq. (246). Another equally important reason for the efficacy of a Gaussian basis set is the fact that a two-centre product of Gaussians can be expressed as a short expansion of one-centre Gaussians – *the Gaussian Product Theorem, (GPT)*

$$\begin{aligned} \bar{\chi}_\alpha^x(x) \bar{\chi}_\beta^x(x) &= \sum_{i=0}^{k_\alpha + k_\beta} C_i^{k_\alpha + k_\beta} \phi_{Pi}(x), \qquad \text{with} \\ x_P &= \frac{a_\alpha x_\alpha + a_\beta x_\beta}{a_P}, \\ a_P &= a_\alpha + a_\beta, \\ \phi_{Pi}(x) &= x^i e^{a_P (x - x_P)^2} \ . \end{aligned} \tag{248}$$

For the case of two s-type Gaussians ($k_\alpha = k_\beta = 0$) the single expansion coefficient is

$$C_0^0 = \exp[-(a_\alpha a_\beta / a_P)(\mathbf{r}_\alpha - \mathbf{r}_\beta)^2]. \tag{249}$$

In a geometrical interpretation, the GPT states that the *product* of two Gaussian functions (with arbitrary polynomial factors) can be expressed as a finite sum of new Gaussians, all centred at a single point P, which is located on the line connecting the two original centres $\mathbf{r}_\alpha$ and $\mathbf{r}_\beta$.

The ERIs as given in eq. (245) can be evaluated analytically using various methods. At the heart of all these methods lies the GPT and some recurrence relations to shift angular momenta from one function to the other. Here, we will not go into the details; for a recent review we refer to Ref. [131].

From eq. (245) it is immediately evident that the ERIs obey the permutational symmetry relations

$$\begin{aligned} (\mu\rho|\nu\sigma) = (\rho\mu|\nu\sigma) = (\mu\rho|\sigma\nu) = (\rho\mu|\sigma\nu) \\ = (\nu\sigma|\mu\rho) = (\sigma\nu|\mu\rho) = (\nu\sigma|\rho\mu) = (\sigma\nu|\rho\mu) \ . \end{aligned} \tag{250}$$

By exploiting this permutational symmetry the number of integrals that need to be evaluated can be reduced by about a factor of eight. In modern quantum chemical codes the ERIs are usually evaluated over *shell quadruplet batches*. A shell typically comprises all contracted functions of a given centre and given angular momentum. For example, an *s*-shell of a 3s2p1d basis set comprises three functions, a *p*-shell six, and a *d-shell* 5 functions. In order to exploit an integral shell quadruplet batch to its maximum extent, i.e. to make use of the permutational symmetry mentioned above, the code should drive *triangularly* over the shell quadruplets. In the following we will use M, R, N, S as symbols for shells of basis functions, i.e., $\mu \in M, \rho \in R$, etc. A direct Fock builder performs a two-index contraction of each integral batch `(MR|NS)` with the related piece of the density matrix. If it runs over the *minimal integral list* (i.e. exploits the full permutational symmetry of the ERIs), each integral batch contributes to the Fock matrix via two Coulomb and four exchange components, as indicated in the pseudocode below.

```
DO M=1,NShell
  DO R=1,M
    DO N=1,M
      DO S=1,N | R (for N=M)
        compute integral shell quadruplet block (MR|NS)
        compute Coulomb component of Fock matrix:
          f(M,R)=f(M,R)+4*(MR|NS)*d(N,S)
          f(N,S)=f(N,S)+4*(MR|NS)*d(M,R)
        compute exchange component of Fock matrix:
          f(R,N)=f(R,N)-(MR|NS)*d(M,S)
          f(R,S)=f(R,S)-(MR|NS)*d(M,N)
          f(M,N)=f(M,N)-(MR|NS)*d(R,S)
          f(M,S)=f(M,S)-(MR|NS)*d(R,N)
      END DO
    END DO
  END DO
END DO
```

6.2 Integral prescreening

Obviously, the ERI supermatrix is a four-indexed quantity. Therefore, the computational effort to evaluate the ERIs scales nominally as $\mathcal{N}^4$, where $\mathcal{N}$ is a measure for the size of the chemical system (e.g. the number of basis functions for a given basis set). For instance, for a system with 100-200 atoms, involving about 2000 basis functions or more, the ERI supermatrix would comprise $10^{12} - 10^{13}$ integrals. It is clear that even though the algorithms for ERI evaluation have been drastically improved over the last two decades, no code can deal with all these integrals in a routine calculation.

In the integral-direct approach the storage bottleneck is removed by reevaluating ERIs on the fly whenever needed. One is then in the situation that the integral evaluation is the bottleneck. The solution to the problem is not only to generate

the ERIs more efficiently, but to search for algorithms that can avoid the calculation of negligible integrals altogether. Fortunately, the ERI supermatrix is very sparse for extended chemical systems. Consider for a moment an ERI $(\mu\rho|\nu\sigma)$, as given in eq. (245). Since both μ and ρ are Gaussian functions and involve the same electron coordinate r_1, it is immediately clear from eqs. (248) and (249) that the integrand decreases exponentially with the distance between the centres $\mathbf{r}_\alpha - \mathbf{r}_\beta$. The same holds for ν and σ. In fact, also the value of the ERI drops exponentially with the distance between μ and ρ or ν and σ. Unfortunately, the two Gaussian pairs $(\mu\rho)$ and $(\nu\sigma)$ are coupled by the Coulomb interaction $1/r_{12}$, which is long range. Hence, the ERI still might be significant even if $(\mu\rho)$ is far away from $(\nu\sigma)$. Therefore, the number of non-vanishing ERIs scales asymptotically with $\mathcal{N}^2$ rather than with $\mathcal{N}^4$. In a direct SCF scheme the ERIs are reevaluated in each iteration and immediately contracted over two indices with the corresponding density matrix elements. Now, for an extended (but non-periodical) chemical system, the density itself is also sparse (i.e. $D(M,N)$ becomes small if M is distant from N), provided that the HOMO-LUMO gap is large enough (which is usually the case for non-metallic systems). Furthermore, the exchange components of the Fock matrix requires contractions of the ERIs where the first index involves one function of the first Gaussian pair $(\mu\rho)$, while the second index corresponds to one function of the second pair $(\nu\sigma)$. Hence, by virtue of the sparsity of the density matrix, the number of ERIs with non vanishing contributions to the Fock exchange component scales asymptotically linear (i.e. as $\mathcal{O}(\mathcal{N})$) with molecular size. Unfortunately, this is not true for the Coulomb component, where the density connects just functions within each pair. Thus, a straightforward scheme would lead to $\mathcal{O}(\mathcal{N}^2)$ scaling. However, since Coulomb repulsion is a relatively simple (i.e. classical) form of interaction, one can employ multipole expansions[132,133,134,135] for the long range interactions, for which linear scaling with molecular size can be achieved. If then the evaluation of the Coulomb and exchange contributions to the Fock matrix is done separately, an overall linear scaling of the Fock matrix construction in integral-direct SCF calculations can be achieved.[136].

A prerequisite for approaching quadratic or even linear scaling in a direct SCF scheme is a method to estimate the integral values as accurately as possible without actually computing them. This estimate must not be done for each integral or each integral batch individually, since then the test would scale itself with $\mathcal{N}^4$ and become the bottleneck. A strict upper bound for the ERI $(\mu\rho|\nu\sigma)$ can be obtained from the Schwartz inequality [137]

$$|(\mu\rho|\nu\sigma)| \leq Q_{\mu\rho}Q_{\nu\sigma}, \qquad \text{with} \qquad Q_{\mu\rho} = \sqrt{(\mu\rho|\mu\rho)}\,. \tag{251}$$

The $Q_{\mu\rho}$ necessary to compute the Schwartz estimates for the ERIs are just two indexed quantities, and can easily be precomputed outside the the nested loop over shell quadruplet batches. The number of non-negligible such integrals scales linearly with molecular size, and it is possible to evaluate them in a way that the overhead with quadratic scaling is very small. Furthermore, since the ERI prescreening takes place at the level of shell batches, only the maximum values of $Q_{\mu\rho}$ over the respective shells, i.e. the

$$Q_{MR} = \underset{\mu\in M,\rho\in R}{\text{Max}}\, Q_{\mu\rho} \tag{252}$$

are required. The four nested shell loops can now be replaced by two loops over the pairs (MR) and (NS) with non-negligible Q_{MR} and Q_{NS}, respectively, and within these loops the product $Q_{MR}Q_{NS}$ can be tested against a threshold. Formally, this prescreening procedure scales quadratically with molecular size, but the prefactor is very small. A more powerful prescreening scheme has also to take the density matrix into account. As we have seen above, each ERI contributes with two Coulomb and four exchange components to the Fock matrix, and therefore the following test is required

$$\begin{aligned} &Q_{MR}Q_{NS}d_{\max} \geq \tau, \quad \text{with} \\ &\qquad d_{\max} = \max(4|d_{MR}|, 4|d_{NS}|, |d_{MN}|, |d_{MS}|, |d_{RN}|, |d_{RS}|) \; . \end{aligned} \tag{253}$$

If the exchange component of the Fock matrix is constructed separately, eq. (253) reduces to

$$Q_{MR}Q_{NS}d_{\max} \geq \tau, \quad \text{with} \quad d_{\max} = \max(|d_{MN}|, |d_{MS}|, |d_{RN}|, |d_{RS}|), \tag{254}$$

leading to an overall linear scaling of shell quadruplets that survive the test, and consequently the number of ERIs that have to be computed.

The efficiency of this prescreening scheme can be enhanced in several ways. First, since ERIs are evaluated batchwise over whole shells, it might be desirable to split off diffuse functions (small exponents) from tight functions (large exponents), and to treat diffuse functions in separate shells. Even though this will increase the total number of shell quadruplets, the actual number of integrals to be computed can be reduced. Second, the effectivity of the prescreening schemes in eqs. (253) and (254) can be enhanced further by constructing *incremental* Fock matrix updates in each new iteration, rather than the total Fock matrix. Consider the the Fock matrices of two consecutive iterations $m-1$ and m:

$$\begin{aligned} f_{\mu\rho}^{(m-1)} &= h_{\mu\rho} + \sum_{\nu\sigma} d_{\nu\sigma}^{(m-1)} \{2(\mu\rho|\nu\sigma) - (\mu\nu|\rho\sigma)\}, \\ f_{\mu\rho}^{(m)} &= h_{\mu\rho} + \sum_{\nu\sigma} d_{\nu\sigma}^{(m)} \{2(\mu\rho|\nu\sigma) - (\mu\nu|\rho\sigma)\} \; . \end{aligned} \tag{255}$$

Obviously, the m^{th} Fock matrix can also be computed via the recurrence relation

$$f_{\mu\rho}^{(m)} = f_{\mu\rho}^{(m-1)} + \sum_{\nu\sigma} \{d_{\nu\sigma}^{(m)} - d_{\nu\sigma}^{(m-1)}\} \{2(\mu\rho|\nu\sigma) - (\mu\nu|\rho\sigma)\},$$

i.e. by generating an incremental two-electron repulsion matrix, obtained by contracting the ERIs with an difference density matrix $\Delta\mathbf{d}^{(m)} = \mathbf{d}^{(m)} - \mathbf{d}^{(m-1)}$. Towards convergence, $\Delta\mathbf{d}^{(m)}$ will become very sparse, and thus the prescreening be more and more effective. The advantages of this recursive construction of the Fock matrix can be further enhanced by the 'minimized density difference' approach [137], where rather than simple density differences a *linear combination* of a history of densities (and Fock matrices) is used, which minimizes the density residual. On should note at this point, however, that the prescreening thresholds may have to be tightened towards convergence in order to avoid numerical noise and thus a deterioration of the convergence behaviour of the SCF. Changing the thresholds on the other hand implies the calculation of a full Fock matrix, i.e., a *restart* of the

density difference procedure. Moreover, the DIIS (direct inversion of the iterative subspace [99]) convergence accelerator has to be restarted as well.

The philosophy of the direct SCF approach was based on the observation that the efficiency of integral processing had outgrown the storage and I/O capacities on modern computer systems. Evidently though, after eliminating the storage and I/O bottleneck at the cost of additional CPU time, the evaluation of the ERIs again becomes the bottleneck in large direct SCF calculations, despite of all the ERI prescreening discussed above. Much work has therefore been dedicated to improve the efficiency of ERI evaluation and Fock matrix construction. Some of these ideas can be summarized as *early contraction* schemes, where the Fock matrix is built directly from the two-centre integrals in the Gaussian Product basis (cf. GPT, eq. (248)), avoiding the handling of explicit four-centre ERIs over primitive or contracted basis functions as much as possible. Other ideas go into the direction of (approximately) reexpanding a product of basis functions in a new auxiliary basis (approximate three-centre expansions[138]). The approximate three-centre expansions appear in a different context (RI-DFT, RI-MP2) in other lectures of this winter school. A discussion of these methods is beyond the scope of this brief overview. Excellent overviews of these methods can be found in Refs. [139,140].

6.3 Integral-direct MP2

As shown in section 2.3, the MP2 contribution to the correlation energy for a closed shell system can be written in spin-free formalism as

$$E^{(2)} = \sum_{i,j,a,b} \frac{(ia|jb)[2(ia|jb) - (ib|ja)]}{\epsilon_i + \epsilon_j - \epsilon_a - \epsilon_b}, \tag{256}$$

where ϵ_i, ϵ_j, ϵ_a, ϵ_b are the corresponding eigenvalues of the Fock matrix. The MO exchange integrals $(bj|ia)$ are computed from the AO integrals (ERIs) through a four-index transformation as shown in eq. (91). In the following, we will denote the four quarter transformation steps by Q1, Q2, Q3 and Q4, respectively. The nominal operation count (without any prescreening) of the Q1 step scales with $\mathcal{O}(m_{\text{occ}}m^4)$, while the others scale with $\mathcal{O}(m_{\text{occ}}^2m^3)$, i.e. the cost of all steps increases with $\mathcal{O}(\mathcal{N}^5)$. For applications on large molecules it is therefore essential to reduce this steep scaling by prescreening techniques, similar to the direct SCF case.

The memory requirements of the four individual transformation steps can be minimized by performing these over *fixed shells*. This seems to be quite natural, since the ERIs are generated anyway as individual batches over shell quadruplets. In a straightforward scheme of that type the storage requirements to hold an individual AO ERI batch then are $\mathcal{O}(s^4)$ (s denotes an average shell size, which is independent of the molecular size), $\mathcal{O}(m_{\text{occ}}ms^2)$ for the ERIs after the Q1 and Q2 steps, and $\mathcal{O}(m_{\text{occ}}^2m^2)$ after the Q3 and Q4 steps, respectively. Apparently, while the computational burden is largest for the initial transformation step, the memory requirements are highest for the final step. In the canonical MP2 case the MO integrals are immediately consumed and accumulated to the MP2 correlation energy, according to eq. (256). A straightforward way to reduce the memory requirements of the critical Q3 and Q4 steps then is to *segment* the first MO index i into indi-

vidual chunks (as large chunks as possible, given by the available memory) and to *multipass* over the AO integral list for each chunk individually [126,127]. This reduces the memory requirements from $\mathcal{O}(m_{\mathrm{occ}}^2 m^2)$ to $\mathcal{O}(I m_{\mathrm{occ}} m^2)$ (I denotes the chunk size) at the cost of repeated ERI evaluations. In order for this algorithm to work, one of the ERI permutational symmetries (i.e. the $(\mu\rho) \leftrightarrow (\nu\sigma)$ symmetry) must be abandoned, thus one integral pass involves twice as many ERIs as the minimal list. The algorithm is free of any I/O operations and can be considered as *fully direct*. Yet the disadvantages are obvious: repeated ERI evaluation might become quite costly, and the number of passes increases quartically with increasing system size and constant memory. A more efficient, *semi direct* algorithm generates in a first step the whole set of *half transformed* integrals $(\mu j|i\nu)$. The transformation of the remaining two indices μ, ν to the virtual basis takes place after an intermediate *bucket sort*, which rearranges the ERIs to *integral matrices* $K_{\mu\nu}^{ij}$, and transforms individual $\mathbf{K}^{ij}$ matrices one after the other. If the permutational symmetry of the slow pair $(\mu\rho)$ (i.e. $\mu \leftrightarrow \rho$) is abandoned, the maximum memory requirements are solely $\mathcal{O}(s m_{\mathrm{occ}} m^2)$. Such an algorithm is outlined in pseudocode below (algorithm A)

```
DO M=1,NShell
  DO R=1,NShell
    DO N=1,M
      DO S=1,N | R (for M=N)
        Compute integral block (MR|NS)
        Q1 step over shell block:
        (MR|Nj) = (MR|Nj) + (MR|NS) * X(S,j)
        (MR|Sj) = (MR|Sj) + (MR|NS) * X(N,j)
      END DO
    END DO
    (Mi|Nj) = (Mi|Nj) + (MR|Nj) * X(R,i)
  END DO
  write (Mi|Nj) to disk
END DO
perform bucket sort/(Mi|Nj)=(Mi|Nj)+(Nj|Mi)
```

Note, that in order to keep the $(\mu\rho) \leftrightarrow (\nu\sigma)$ permutational symmetry the triangularity in the operator indices i, j is lost. The final operator matrices $\mathbf{K}^{ij} (i \geq j)$ are formed by adding up the partial results $\mathcal{K}_{\mu\nu}^{ij} + \mathcal{K}_{\nu\mu}^{ji} (i \geq j)$, which is performed during the bucket sort, as indicated above.
By virtue of an elaborate paging algorithm, it is even possible to maintain also the $\mu \leftrightarrow \rho$ permutational symmetry (algorithm B), i.e.

```
    R_End=0
    R_Pass=0
1   R_Start=R_End+1
    R_End=MIN(NShell,R_End+R_Batch)
    R_Pass=R_Pass+1
```

```
if(R_Pass.gt.1) Read (Ri|Nj) for shells R_Start to R_End
DO M=R_Start,NShell
  IF(R_Pass.gt.1.and.M.gt.R_End) Read (Mi|Nj) for shell M
  DO R=R_Start,MIN(R_End,M)
    DO N=1,M
      DO S=1,N | R (for M=N)
        Compute integral block (MR|NS)
        Q1 step over shell block:
        (MR|Nj) = (MR|Nj) + (MR|NS) * X(S,j)
        (MR|Sj) = (MR|Sj) + (MR|NS) * X(N,j)
      END DO
    END DO
    (Mi|Nj) = (Mi|Nj) + (MR|Nj) * X(R,i)
    (Ri|Nj) = (Ri|Nj) + (MR|Nj) * X(M,i)
  END DO
  IF(M.GT.R_End) Write (Mi|Nj) for shell M
END DO
Write (Ri|Nj) for shells R_Start to R_End
If(R_End.LT.NShell) goto 1
perform bucket sort/(Mi|Nj)=(Mi|Nj)+(Nj|Mi)
```

This means that the full permutational symmetry of the AO ERIs is exploited. This algorithm is very efficient for molecular systems of intermediate size. However, for large systems and limited memory, the paging overhead might become too excessive (even though no multipassing whatsoever over the integral list is involved, as in the fully direct scheme), and algorithm A becomes more efficient.

The Q1 and Q2 transformation steps require matrix multiplications, in which at least one of the matrix dimensions corresponds to the shell size. For small shells the vector lengths are too short for a good performance to be achieved. Therefore, it is advantageous to merge adjacent `R` and `S` shells until an upper limit of 32-64 basis functions is reached. Significant speedups (factors of 4-6) were observed, if such *shell merging* was invoked [60].

For applications on larger molecules, integral prescreening is of utmost importance. In order to assess the values of the AO ERIs, the Schwartz inequality (eq. 251) is again employed. Furthermore a test density $\mathbf{D}^{\max}$ is constructed from the MO coefficient matrix $\mathbf{C}$ as

$$D^{\max}_{\rho\sigma} = \underset{ij}{\mathrm{Max}}\, C_{\sigma i} C_{\rho j} \tag{257}$$

The prescreening criterions for the direct transformation at the level of shell quadrulets then are

$$Q_{MR} Q_{NS} D^{\max}_{RS} \geq \tau_1 \tag{258}$$

before integral evaluation, and

$$\underset{\mu\in M,\rho\in R,\nu\in N,\sigma\in S}{\mathrm{Max}} (\mu\rho|\nu\sigma) D^{\max}_{RS} \geq \tau_2 \tag{259}$$

before the Q1 step, respectively. Such a prescreening leads to a reduction of the computational cost of the dominant Q1 step from $\mathcal{O}(\mathcal{N}^5)$ to $\mathcal{O}(\mathcal{N}^3)$ [60]. The overall scaling however deteriorates again for larger molecules due to the subsequent transformations steps, which, because of the delocalized character of canonical orbitals, scale worse than $\mathcal{O}(\mathcal{N}^3)$. In particular the Q4 step (i.e. the transformation of the $\mathbf{K}^{ij}_{\mu\nu}$ to the canonical virtuals) would still scale as $\mathcal{O}(\mathcal{N}^5)$, although with a small prefactor, but nevertheless will ultimately constitute the bottleneck of the calculation. The remedy to this problem are *local correlation methods*, discussed in section 4. In combination with local correlation methods integral-direct MP2 algorithms with linear cost scaling have been implemented, which enable calculations of molecules with more than 2000 basis functions and 500 correlated electrons [54].

6.4 Integral-direct MCSCF

In MCSCF calculations the orbitals are optimized simultaneously with the CI coefficients. Thus, an integral transformation is required in each iteration, which constitutes one of the major bottlenecks in conventional MCSCF calculations. In a direct scheme, this bottleneck is even much more severe, since each direct transformation also involves recomputation of all AO ERIs. It is therefore of utmost importance that the MCSCF converges in as few iterations as possible.

MCSCF orbital optimization methods can be classified as first-order or second-order methods. In the former only the first derivatives of the energy with respect to the variational parameters are computed exactly, and updates of the parameters are obtained using some approximation of the Hessian (e.g. a BFGS update scheme). In first-order methods the coupling of the orbitals and CI-coefficients is neglected. One particular advantage of first-order methods is that only a very compact set of transformed integrals is required, i.e. an integral distribution of the form $(pj|kl)$ with only a single external index. In fact, j, k, l here run just over active orbitals, while the inactive orbitals (doubly occupied in all CSFs) can be accounted for by a single Fock matrix [141,142]. Thus, any storage bottleneck connected to the integral transformation is avoided. An integral-direct first-order MCSCF method has been described by Frisch *et al.*[124].

In second-order methods, also the second energy derivatives are computed exactly, yielding quadratic convergence near the final solution. Naturally, first-order methods require less effort per iteration, but are often slowly convergent and appear to be only useful for the optimization of CASSCF wavefunctions[141]. In this case convergence is facilitated by the fact that orbital rotations among active orbitals are redundant. Even with second-order methods convergence is often difficult to achieve for general MCSCF wavefunctions[142]. The radius of convergence and the speed of convergence can be substantially increased by taking into account certain higher-order terms, as first proposed by Werner and Meyer[143,144] and further refined by Werner and Knowles[98,91]. Using the latter method (in the following denoted WMK), convergence can often be achieved in only 2-3 iterations, in particular for CASSCF wavefunctions. Almost cubic convergence behaviour is observed near the solution. In the light of the discussion above, the WMK method is particulary useful in an integral-direct context, while the advantage of the simple and efficient

transformation of first-order methods is spoilt by its slow convergence behaviour. The integral sets required by the WMK method are identical to those used by ordinary second-order methods: in additon to the exchange integrals $(ip|jq)$ also the Coulomb integrals $(pq|ij)$ are necessary. Furthermore, the very same integral sets, generated in the last iteration, can be reused in a subsequent CASPT2 or MRCI calculation. The additional Coulomb integral set can be produced simultaneously with the exchange integrals by modifying the above MP2 transformation algorithm A in the following way (algorithm C):

```
DO M=1,NShell
  DO R=1,NShell
    DO N=1,NShell
      DO S=1,R
        Compute integral block (MR|NS)
        Q1 step over shell block:
        (MR|Nj) = (MR|Nj) + (MR|NS) * X1(S,j)
        (MR|Sj) = (MR|Sj) + (MR|NS) * X1(N,j)
      END DO
    END DO
    Q2 (J) step:
    (MR|ij) = (MR|Nj) * X2(N,i)    (summed over N)
    write (MR|ij) to disk
    Q2 (K) step:
    (Mi|Nj) = (Mi|Nj) + (MR|Nj) * X2(R,i)
  END DO
  write (Mi|Nj) to disk
END DO
perform bucket sort
```

Note, that compared to algorithm A the permutational symmetry between the pairs $(\mu\rho) \leftrightarrow (\nu\sigma)$ is lost, thus the AO integral list in algorithm C is four times as long as the minimal list. As in the MP2 case (algorithm B), the permutational symmetry $(\mu\rho)$ can be maintained by using an analogous paging algorithm, which might be advantageous for intermediate cases.

6.5 Integral-direct multireference correlation methods

The internally contracted MRCI and MRPT methods as discussed in section 5 can be formulated in terms of matrix operations[142] involving the same Coulomb and exchange matrices $\mathbf{J}^{ij}$ and $\mathbf{K}^{ij}$ as needed in the preceeding MCSCF. In the MRCI and MRPT3 all contributions of 4-external integrals $(ab|cd)$ can be taken into account by computing for each pair P an *external exchange operator* (EEO), as defined in eq. (106)[117,118,108]. These operators can be computed directly from the two-electron integrals in the AO basis by first transforming the amplitude matrices into the AO basis and finally transforming these back into the MO basis (cf. eqs (107). For an integral-direct implementation the internally contracted MRCI

scheme is particularly useful, since the number of pairs and thus external exchange operators that need to be computed is minimized and does not depend on the number of reference configurations. In uncontracted MRCI methods the number of pairs P for which the EEOs $\mathbf{K}(\mathbf{T}^P)$ must be computed is excessively larger than in the internally contracted case. This does not only lead to higher computational cost, but also to a storage bottleneck in the direct evaluation of these operators

The direct construction of the EEOs from the minimal AO integral list is accomplished by contracting two indices of the AO ERI $(\mu\rho|\sigma\nu)$ with the two AO indices of the backtransformed amplitudes 107, in all possible ways, which result in exchange type contributions, and can be regarded as a 'Fock build' (excluding Coulomb contributions) of n_P Fock matrices simultaneously (n_P denotes the number of pairs P). A shell driven out-of-core algorithm for such a construction of the EEOs, as implemented in MOLPRO [60], is given in pseudocode below (module `DKEXT`).

```
    R_End=0
    R_Pass=0
1   R_Start=R_End+1
    R_End=MIN(NShell,R_End+R_Batch)
    Read amplitudes for shells R_Start to R_End
    R_Pass=R_Pass+1
    IF(R_Pass.gt.1) Read operators for shells R_Start to R_End
    DO M=R_Start,NShell
      If(M.GT.R_End) then
        Read amplitudes for shell M
        If(R_Pass.gt.1)  Read operators for shell M
      End If
      DO R=R_Start,MIN(R_End,M)
        DO N=1,M
          S_End=N
          If(N.EQ.M) S_End=R
          DO S=1,S_End
            Compute integral block (MR|NS)
            Compute contributions to operators
          END DO
        END DO
      END DO
      IF(M.GT.R_End) Write operators for shell M
    END DO
    Write operators for shells R_Start to R_End
    If(R_End.LT.NShell) goto 1
```

The algorithm employs a paging algorithm, which is quite similar to that used in the direct transformation scheme discussed in section 6.3. The amplitudes and EEOs are presorted according to shell blocks $T^{MR}_{P,\mu\rho}$ with M running slowest, and stored on disk. In this way it is possible to read/write them for a given shell M

and for all P and R.

All contributions arising from integrals over one occupied and three external orbitals $(ia|bc)$ can be taken into account by an additional set of EEOs $\mathbf{K}(\mathbf{D}^P)$, where $\mathbf{D}^P$ are modified coefficient matrices[117,118,22], which differ from the $\mathbf{T}^P$ by the addition of internal-external blocks arising from contributions of single excitations. In single-reference methods (CISD, MP4(SDQ), QCSID, CCSD) as well as for evaluating the MRPT3 energy it is sufficient to compute only the latter set of operators. In MRCI calculations, this would in principle be possible as well, but since then complicated correction terms are necessary[118] it is easier to compute the operators $\mathbf{K}(\mathbf{T}^P)$ and $\mathbf{K}(\mathbf{D}^P)$ separately. Of course, the two sets can be computed together in a single integral pass.

Since the EEOs depend explicitly on the amplitudes that must be computed in each iteration. The computational complexity of EEO formation is nominally a task $\mathcal{O}(m_{\rm occ}^2 m^4) = \mathcal{O}(\mathcal{N}^6)$. In an integral-direct context this can be reduced to $\approx \mathcal{O}(\mathcal{N}^4)$ by virtue of integral prescreening [60]. In order to get efficient prescreening, it is important to include the amplitudes into the prescreening scheme. Nevertheless, in integral-direct calculations with large basis sets, the EEO construction often dominates the computational effort.

6.6 Integral-direct coupled cluster methods

The first integral-direct CCSD method was developed by Koch and coworkers[129,145]. In this method the transformed integrals are never stored on disk. Instead, "distributions" of AO integrals $(\mu\rho|\nu\sigma)$ are generated for fixed μ, all ρ, $\nu \geq \sigma$. One such distribution at a time is kept in memory and consumed immediately to compute all contributions to the CCSD residual (fully direct CCSD). This method, although very efficient on vector computers due to long vector lenghths, suffers from some severe bottlenecks (most importantly, the m^3 memory requirements of the integral distributions, mentioned above), which limit the application range for larger systems. An alternative method has been proposed by Schütz, Werner and Lindh [60], which differs from the above method by the fact, that the partially transformed integrals are stored on disk ($3/2 m_{\rm occ}^2 m^2$ words are required). Considering that the doubles amplitudes as the variational parameters of the iterative CCSD procedure and the residuals have to be stored on disk anyway in several instances (due to DIIS convergence acceleration), with a required diskspace of $n_{\rm DIIS} m_{\rm occ}^2 m^2$, this certainly does not constitute a further bottleneck, and seems to be a reasonable strategy. The immediate advantage is that the remaining program remains entirely unchanged, and that the same integral-direct modules as for the MCSCF and MRCI programs can be used. Furthermore, in such a scheme the maximum memory requirements can be reduced to $\mathcal{O}(m_{\rm occ} m^2)$, and to $\mathcal{O}(\mathcal{N})$ for local CCSD (cf. section 4.2).

The MP3, MP4(SDQ), QCISD and CCSD methods, which all are related, require the same internal operators $\mathbf{J}^{ij}$, $\mathbf{K}^{ij}$, and the EEOs $\mathbf{K}(\mathbf{D}^{ij})$ as introduced for the MRCI case in the previous section. A further complication arises in the CCSD method [22], where the additional operators $\mathbf{J}(\mathbf{E}^{ij})$ and $\mathbf{K}(\mathbf{E}^{ij})$ (cf. eqs. (132)) are needed. As discussed in section 2.5, these operators can be obtained by a generalized integral transformation (cf. eqs. (133)-(137)). This transformation can be

performed using the same integral-direct module as employed for generating the $\mathbf{J}^{ij}$ and $\mathbf{K}^{ij}$ matrices, but since they depend on the singles amplitudes they must be performed in each iteration. An important point to notice is that the latter operators are only needed for CCSD, but not for the QCISD (quadratic configuration interaction) method[33]. While the computational effort for these two methods is not too much different in conventional calculations[22], in the integral-direct case the full CCSD takes significantly more time, due to this additional transformation which must be performed in each iteration. For most applications, QCISD and CCSD results are very similar, and QCISD may often be more cost effective for integral-direct calculations of large molecules, even though from a theoretical point of view CCSD is more satisfactory. If the 3-external integrals are available though, as is usually the case for local CCSD calculations, then the construction of the $\mathbf{J}(\mathbf{E}^{ij})$ and $\mathbf{K}(\mathbf{E}^{ij})$ operators takes little time, hence there is little reason to use the QCISD model in that case.

Acknowledgments

Financial support from the EC as part of the TMR network "Potential Energy Surfaces for Spectroscopy and Dynamics", contract No. FMRX-CT96-088 (DG 12 – BIUO) and as part of the RTN network "Theoretical Studies of Electronic and Dynamical Processes in Molecules and Clusters (THEONET II)", contract No. RTN1-1999-00121 is gratefully acknowledged. Much of the research described in these notes has been supported by DFG, EPSRC, Fonds der Chemischen Industrie and BASF AG.

References

1. R. H. Nobes, J. A. Pople, L. Radom, N. C. Handy, and P. J. Knowles, Chem. Phys. Letters **138**, 481 (1987).
2. P. J. Knowles and N. C. Handy, J. Phys. Chem. **92**, 3097 (1988).
3. E. R. Davidson, J. Comput. Phys. **17**, 87 (1975).
4. T. Kato, Commun. Pure Appl. Math. **10**, 151 (1957).
5. R. T. Pack and W. Byers Brown, J. Chem. Phys. **45**, 556 (1966).
6. W. A. Bingel, Z. Naturforsch. Teil A **18**, 1249 (1963).
7. V. A. Rassolov and D. M. Chipman, J. Chem. Phys. **104**, 9908 (1996).
8. E. A. Hylleraas, Z. Phys. A **54**, 347 (1929).
9. H. M. James and A. S. Coolidge, J. Chem. Phys. **1**, 825 (1933).
10. W. Kutzelnigg, Theor. Chim. Acta **68**, 445 (1985).
11. W. Kutzelnigg and W. Klopper, J. Chem. Phys. **94**, 1985 (1991).
12. D. C. Clary and N. C. Handy, Phys. Rev. **A14**, 1607 (1976).
13. D. C. Clary, Mol. Phys. **34**, 793 (1977).
14. P. Jørgensen and J. Simons, *Second Quantization–Based Methods in Quantum Chemistry* (Academic Press, New York, 1981).
15. J. Almlöf and P. R. Taylor, J. Chem. Phys. **86**, 4070 (1987).
16. J. Almlöf and P. R. Taylor, J. Chem. Phys. **92**, 551 (1990).
17. P.-O. Widmark, P.-Å. Malmqvist, and B. O. Roos, Theor. Chim. Acta **77**,

291 (1990).
18. J. Almlöf and P. R. Taylor, Adv. Quant. Chem. **22**, 301 (1991).
19. T. H. Dunning Jr., J. Chem. Phys. **90**, 1007 (1989).
20. W. Meyer, J. Chem. Phys. **64**, 2901 (1976).
21. P. Pulay, S. Saebø, and W. Meyer, J. Chem. Phys. **81**, 1901 (1984).
22. C. Hampel, K. A. Peterson, and H.-J. Werner, Chem. Phys. Lett. **190**, 1 (1992).
23. W. Meyer, Int. J. Quantum Chem. Symp. **5**, 341 (1971).
24. W. Meyer, J. Chem. Phys. **58**, 1017 (1973).
25. R. Ahlrichs, P. Scharf, and C. Ehrhardt, J. Chem. Phys. **82**, 890 (1985).
26. S. R. Langhoff and E. R. Davidson, Int. J. Quant. Chem. **8**, 61 (1974).
27. B. O. Roos and P. E. M. Siegbahn, in *Methods of Electronic Structure Theory*, edited by H. F. Schaefer III (Plenum, New York, 1977).
28. J. Čížek, J. Chem. Phys. **45**, 4256 (1966).
29. J. Čížek, Adv. Chem. Phys. **14**, 35 (1969).
30. J. Čížek and J. Paldus, Int. J. Quantum Chem. **5**, 359 (1971).
31. G. D. Purvis and R. J. Bartlett, J. Chem. Phys. **76**, 1910 (1982).
32. G. E. Scuseria, C. L. Janssen, and H. F. Schaefer III, J. Chem. Phys. **89**, 7382 (1988).
33. J. A. Pople, M. Head-Gordon, and K. Raghavachari, J. Chem. Phys. **87**, 5968 (1987).
34. J. Čížek and J. Paldus, Phys. Scripta **21**, 251 (1980).
35. R. J. Bartlett and G. D. Purvis, Phys. Scripta **21**, 255 (1980).
36. R. A. Chiles and C. E. Dykstra, J. Chem. Phys. **74**, 4544 (1981).
37. G. Scuseria and H. F. Schaefer III, Chem. Phys. Lett. **142**, 354 (1987).
38. N. C. Handy, J. A. Pople, M. Head-Gordon, K. Raghavachari, and G. W. Trucks, Chem. Phys. Lett. **164**, 185 (1989).
39. K. Raghavachari, J. A. Pople, E. S. Replogle, M. Head-Gordon, and N. C. Handy, Chem. Phys. Lett. **167**, 115 (1990).
40. M. Urban, J. Noga, S. J. Cole, and R. J. Bartlett, J. Chem. Phys. **83**, 4041 (1985).
41. K. Raghavachari, G. W. Trucks, J. A. Pople, and M. Head-Gordon, Chem. Phys. Letters **157**, 479 (1989).
42. S. A. Kucharski and R. J. Bartlett, Adv. Quantum Chem. **18**, 281 (1986).
43. S. A. Kucharski, J. Noga, and R. J. Bartlett, J. Chem. Phys. **90**, 7282 (1989).
44. K. Raghavachari, J. A. Pople, E. S. Replogle, and M. Head-Gordon, J. Phys. Chem. **94**, 5579 (1990).
45. Z. He and D. Cremer, Theor. Chim. Acta **85**, 305 (1993).
46. M. J. O. Deegan and P. J. Knowles, Chem. Phys. Letters **227**, 321 (1994).
47. P. J. Knowles, C. Hampel, and H.-J. Werner, J. Chem. Phys. **99**, 5219 (1993).
48. C. Janssen and H. F. Schaefer III, Theor. Chim. Acta **79**, 1 (1991).
49. P. J. Knowles, C. Hampel, and H.-J. Werner, J. Chem. Phys. **111**, 0000 (2000).
50. P. Neogrády, M. Urban, and I. Hubač, J. Chem. Phys. **100**, 3706 (1994).
51. P. G. Szalay and J. Gauss, J. Chem. Phys. **107**, 9028 (1997).
52. S. Saebø and P. Pulay, Annu. Rev. Phys. Chem. **44**, 213 (1993).

53. C. Hampel and H.-J. Werner, J. Chem. Phys. **104**, 6286 (1996).
54. M. Schütz, G. Hetzer, and H.-J. Werner, J. Chem. Phys. **111**, 5691 (1999).
55. R. A. Friesner, R. B. Murphy, M. D. Beachy, M. N. Ringnalda, W. T. Pollard, B. D. Dunietz, and Y. Cao, J. Phys. Chem. A **103**, 1913 (1999).
56. G. Reynolds, T. J. Martinez, and E. A. Carter, J. Chem. Phys. **105**, 6455 (1996).
57. P. Pulay, Chem. Phys. Letters **100**, 151 (1983).
58. P. Pulay and S. Saebø, Theor. Chim. Acta **69**, 357 (1986).
59. S. Saebø and P. Pulay, J. Chem. Phys. **86**, 914 (1987).
60. M. Schütz, R. Lindh, and H.-J. Werner, Mol. Phys. **96**, 719 (1999).
61. M. Schütz and H.-J. Werner, manuscript in preparation.
62. M. Schütz and H.-J. Werner, Chem. Phys. Lett., in press.
63. S. F. Boys, in *Quantum Theory of Atoms, Molecules, and the Solid State*, edited by P. O. Löwdin, page 253 (Academic, New York, 1966).
64. C. Edmiston and K. Ruedenberg, J. Chem. Phys. **43**, S97 (1965).
65. J. Pipek and P. G. Mezey, J. Chem. Phys. **90**, 4916 (1989).
66. M. Head-Gordon, P. E. Maslen, and C. A. White, J. Chem. Phys. **108**, 616 (1998).
67. G. E. Scuseria and P. Y. Ayala, J. Chem. Phys. **111**, 8330 (1999).
68. J. W. Boughton and P. Pulay, J. Comput. Chem. **14**, 736 (1993).
69. G. Hetzer, P. Pulay, and H.-J. Werner, Chem. Phys. Lett. **290**, 143 (1998).
70. G. Rauhut, P. Pulay, and H.-J. Werner, J. Comput. Chem. **19**, 1241 (1998).
71. A. ElAzhary, G. Rauhut, P. Pulay, and H.-J. Werner, J. Chem. Phys. **108**, 5185 (1998).
72. M. Schütz, G. Rauhut, and H.-J. Werner, J. Phys. Chem. A **102**, 5997 (1998).
73. N. Runeberg, M. Schütz, and H.-J. Werner, J. Chem. Phys. **110**, 7210 (1999).
74. J. Gauss and H.-J. Werner, Mol. Phys., in press.
75. MOLPRO is a package of *ab initio* programs written by H.-J. Werner and P. J. Knowles, with contributions from J. Almlöf, R. D. Amos, A. Berning, P. Celani, D. L. Cooper, M. J. O. Deegan, A. J. Dobbyn, F. Eckert, S. T. Elbert, C. Hampel, G. Hetzer, T. Korona, R. Lindh, A. W. Lloyd, W. Meyer, M. E. Mura, A. Nicklass, K. Peterson, R. Pitzer, P. Pulay, G. Rauhut, M. Schütz, H. Stoll, A. J. Stone, P. R. Taylor, and T. Thorsteinsson.
76. J. Paldus, J. Chem. Phys. **61**, 5321 (1974).
77. J. Hinze, editor, *The Unitary Group* (Springer–Verlag, Berlin, 1979).
78. E. R. Davidson, in *Methods in Computational Molecular Physics*, edited by G. H. F. Diercksen and S. Wilson (Reidel, Dordrecht, 1983).
79. P. E. M. Siegbahn, Chem. Phys. Letters **109**, 417 (1984).
80. J. Olsen, B. O. Roos, P. Jørgensen, and H. J. A. Jensen, J. Chem. Phys. **89**, 2185 (1988).
81. P. J. Knowles and N. C. Handy, Chem. Phys. Letters **111**, 315 (1984).
82. P. J. Knowles and N. C. Handy, Comput. Phys. Commun. **54**, 75 (1989).
83. S. Zarrabian, C. R. Sarma, and J. Paldus, Chem. Phys. Letters **155**, 183 (1989).
84. D. M. Brink and G. R. Satchler, *Angular Momentum* (Clarendon, Oxford, 2nd edition, 1968).

85. R. N. Zare, *Angular Momentum* (Wiley, New York, 1988).
86. R. Pauncz, *Spin Eigenfunctions* (Plenum, New York, 1979).
87. P. J. Knowles and H.-J. Werner, Chem. Phys. Lett. **145**, 514 (1988).
88. K. Ruedenberg, L. M. Cheung, and S. T. Elbert, Int. J. Quantum Chem. **16**, 1069 (1979).
89. B. O. Roos, P. Taylor, and P. E. M. Siegbahn, Chem. Phys. **48**, 157 (1980).
90. P. E. M. Siegbahn, J. Almlöf, A. Heiberg, and B. O. Roos, J. Chem. Phys. **74**, 2384 (1981).
91. P. J. Knowles and H.-J. Werner, Chem. Phys. Letters **115**, 259 (1985).
92. M. R. Hoffmann, D. J. Fox, J. F. Gaw, Y. Osamura, Y. Yamaguchi, R. S. Grev, G. Fitzgerald, H. F. Schaefer III, P. J. Knowles, and N. C. Handy, J. Chem. Phys. **80**, 2660 (1984).
93. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in Fortran 77: The Art of Scientific Computing* (Cambridge University Press, 2nd edition, 1992).
94. P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization* (Academic Press, 1981).
95. J. A. Pople, R. Krishnan, H. B. Schlegel, and J. S. Binkley, Int. J. Quant. Chem. **S13**, 225 (1979).
96. J. Olsen, D. L. Yeager, and P. Jørgensen, Adv. Chem. Phys. **54**, 1 (1983).
97. D. Yarkony, Chem. Phys. Letters **77**, 634 (1981).
98. H.-J. Werner and P. J. Knowles, J. Chem. Phys. **82**, 5053 (1985).
99. P. Császár and P. Pulay, J. Mol. Struc. **114**, 31 (1984).
100. C. Møller and M. S. Plesset, Phys. Rev. **46**, 618 (1934).
101. J. A. Pople, R. Krishnan, H. B. Schlegel, and J. S. Binkley, Int. J. Quant. Chem. **14**, 545 (1978).
102. R. J. Bartlett and D. M. Silver, J. Chem. Phys. **62**, 3258 (1975).
103. R. B. Murphy and R. P. Messmer, Chem. Phys. Letters **183**, 443 (1991).
104. K. Andersson, P.-Å. Malmqvist, and B. O. Roos, J. Chem. Phys. **96**, 1218 (1992).
105. K. Hirao, Chem. Phys. Letters **196**, 397 (1992).
106. B. O. Roos, K. Andersson, M. P. Fulscher, P. A. Malmqvist, L. Serranoandres, K. Pierloot, and M. Merchan, Adv. Chem. Phys. **93**, 219 (1996).
107. P. Celani and H.-J. Werner, J. Chem. Phys., in press.
108. H.-J. Werner, Mol. Phys. **89**, 645 (1996).
109. R. J. Buenker and S. D. Peyerimhoff, Theor. Chim. Acta **35**, 33 (1974).
110. P. E. M. Siegbahn, Int. J. Quantum Chem. **18**, 1229 (1980).
111. J. Lischka, R. Shepard, F. B. Brown, and I. Shavitt, Int. J. Quantum Chem. Symp. **15**, 91 (1981).
112. V. R. Saunders and J. H. van Lenthe, Mol. Phys. **48**, 923 (1983).
113. W. Meyer, in *Methods of Electronic Structure Theory*, edited by H. F. Schaefer III (Plenum, New York, 1977).
114. P. E. M. Siegbahn, J. Chem. Phys. **72**, 1647 (1980).
115. C. W. Bauschlicher, P. R. Taylor, N. C. Handy, and P. J. Knowles, J. Chem. Phys. **85**, 1469 (1986).
116. C. W. Bauschlicher, Jr., S. R. Langhoff, and P. R. Taylor, Adv. Chem. Phys.

77, 103 (1990).
117. H.-J. Werner and E. A. Reinsch, J. Chem. Phys. **76**, 3144 (1982).
118. H.-J. Werner and P. J. Knowles, J. Chem. Phys. **89**, 5803 (1988).
119. E. R. Davidson, in *The world of quantum chemistry*, edited by R. Daudel and B. Pullman (Reidel, Dordrecht, 1974).
120. R. J. Gdanitz and R. Ahlrichs, Chem. Phys. Lett. **143**, 413 (1988).
121. P. G. Szalay and R. J. Bartlett, Chem. Phys. Letters **214**, 481 (1993).
122. J. Almlöf, J. K. Faegri, and K. Korsell, J. Comput. Chem. **3**, 385 (1982).
123. P. Taylor, Int. J. Quantum Chem. **31**, 521 (1987).
124. M. Frisch, I. N. Raganzos, M. A. Robb, and H. B. Schlegel, Chem. Phys. Lett. **189**, 524 (1992).
125. S. Sæbø and J. Almlöf, Chem. Phys. Letters **154**, 83 (1989).
126. M. Head-Gordon, J. Pople, and M. Frisch, Chem. Phys. Letters **153**, 503 (1988).
127. M. Schütz and R. Lindh, Theor. Chim. Acta **95**, 13 (1997).
128. M. Frisch, M. Head-Gordon, and J. Pople, Chem. Phys. Letters **166**, 275 (1990).
129. H. Koch, O. Christiansen, R. Kobayashi, P. Jørgensen, and T. Helgaker, Chem. Phys. Lett. **228**, 233 (1994).
130. W. Klopper and J. Noga, J. Chem. Phys. **103**, 6127 (1995).
131. R. Lindh, in *The Encyclopedia of Computational Chemistry Vol.2*, edited by P. v. R. Schleyer, N. L. Allinger, T. Clark, J. Gasteiger, P. A. Kollman, H. F. S. III, and P. R. Schreiner, page 1337 (John Wiley & Sons: Chichester, 1998).
132. C. A. White, B. G. Johnson, P. M. W. Gill, and M. Head-Gordon, Chem. Phys. Letters **230**, 8 (1994).
133. C. A. White, B. G. Johnson, P. M. W. Gill, and M. Head-Gordon, Chem. Phys. Letters **253**, 268 (1996).
134. J. C. Burant, G. E. Scuseria, and M. J. Frisch, J. Chem. Phys. **105**, 8969 (1996).
135. M. Challacombe, E. Schwegler, and J. Almlöf, J. Chem. Phys. **104**, 4685 (1996).
136. C. Ochsenfeld, C. A. White, and M. Head-Gordon, J. Chem. Phys. **109**, 1663 (1998).
137. M. Häser and R. Ahlrichs, J. Comput. Chem. **10**, 104 (1989).
138. O. Vahtras, J. Almlöf, and M. Feyereisen, Chem. Phys. Letters **213**, 514 (1993).
139. J. Almlöf, in *Modern Electronic Structure Theory*, edited by D. Yarkony, number I in Advanced Series in Physical Chemistry - Vol. 2, page 110 (World Scientific Publishing Co. Pte. Ltd., 1995).
140. J. Almlöf, in *Lecture Notes in Quantum Chemistry, European Summer School in Quantum Chemistry*, edited by B. Roos, number 64 in Lecture Notes in Chemistry, page 1 (Springer-Verlag Berlin Heidelberg, 1994).
141. B. O. Roos, Int. J. Quantum Chem. Symp. **14**, 175 (1980).
142. H.-J. Werner, Adv. Chem. Phys. **69**, 1 (1987).
143. H.-J. Werner and W. Meyer, J. Chem. Phys. **73**, 2342 (1980).
144. H.-J. Werner and W. Meyer, J. Chem. Phys. **74**, 5794 (1981).

145. H. Koch, A. S. de Merás, T. Helgaker, and O. Christiansen, J. Chem. Phys. **104**, 4157 (1996).

R12 METHODS, GAUSSIAN GEMINALS

WIM KLOPPER
Theoretical Chemistry Group
Debye Institute, Utrecht University
P.O. Box 80052
3508 TB Utrecht
The Netherlands
E-mail: w.m.klopper@chem.uu.nl

The basis-set truncation error of standard electron-correlation treatments represents a serious problem in computational chemistry. Wavefunction models that are restricted to products of one-particle basis functions – that is, the orbital approximation – converge very slowly to the true solutions that would be obtained in a complete basis. The convergence of the orbital approximation can be characterized as N^{-1}, where N is the number of functions in the correlation-consistent basis. Extrapolation schemes appear to help to overcome this basis-set convergence problem and even more so do expansions that not only employ one-particle basis functions, but also basis functions that depend on the coordinates of more than one electron. R12 methods and Gaussian geminals employ two-electron basis functions that depend on the interparticle distance r_{12}. In more general explicitly correlated Gaussians (ECG) also functions that depend on the coordinates of more than two electrons occur. The present discussion of the various many-electron basis functions focuses on methods that can be applied to molecular many-electron systems and is less concerned with highly accurate calculations of atomic electronic structure.

1 Introduction

R12 methods, Gaussian geminals, and explicitly correlated Gaussians have been introduced to computational quantum chemistry with the purpose to facilitate calculations of dynamical electron-correlation effects and have in common that they include the interparticle distances r_{ij} into the many-electron wavefunction.

It had been recognized already in the early days of quantum mechanics that the electron-electron distance r_{12} ought to be included into the wavefunction if quantitatively accurate computational results were to be obtained[1–6]. In 1937, Hellmann wrote about the electron-correlation effect[4]:

> *Der Effekt spielt bei allen feineren quantitativen Rechnungen eine große Rolle, ist aber meist schwer rechnerisch zu erfassen, da er die Einführung von r_{12} in die Eigenfunktion erfordert.*

(The correlation effect plays an important rôle in all precise quantitative calculations but is mostly difficult to describe computationally, as it requires the inclusion of r_{12} into the wavefunction.) Indeed, as we will see later in the present lecture notes, the accurate computation of the dynamical electron-correlation effects is very difficult with the standard electron-correlation treatments[7,8] within the orbital approximation – that is, treatments that are based on wavefunction expansions in terms of antisymmetrized orbital products (Slater determinants). These calculations are difficult in the sense that the computed electronic energies and molecular

properties are very dependent on the orbital basis sets used. When larger and larger basis sets are used, the computed data keep changing and converge only very slowly to the limiting value of an infinitely large, complete basis set. In particular, the convergence is so slow that the computational costs of the correlated calculations grow four orders of magnitude faster than the rate with which the basis-set truncation errors decrease. In order to reach the next level of accuracy, which corresponds to a reduction of the truncation error by a factor of ten, calculations are required that consume roughly 10000 times more computing time. We will see that, as a function of the error δ, the required computing time of standard electron-correlation treatments can be expressed as:

$$t_{\mathrm{cpu}}(\delta) = \delta^{-4}. \tag{1}$$

This means in practice that if we want to improve (by one order of magnitude) certain calculations that take something like one minute of computing time on a given computer, we have to perform calculations that run for a whole week on that same computer. We would have to wait almost 200 years to achieve results at the next level of accuracy! Evidently, the scaling of standard *ab initio* electronic-structure calculations in terms of the basis-set truncation error is nothing else but disastreous.

In the present lecture, we will be concerned with computational approaches that address this slow basis-set convergence and the corresponding scaling problems.

2 Errors in electronic-structure calculations

What precisely are the errors we are dealing with when we perform electronic-structure calculations? It is of course important to ask this question and especially to find out about the most significant sources of error in our calculations. If more accurate calculations are required, that is, if more reliable computational data are needed, one can then focus on the most important sources of error and address these.

In *ab initio* electronic-structure calculations, approximate solutions are obtained to the nonrelativistic electronic Schrödinger equation in the framework of the Born-Oppenheimer (BO) approximation. We can thus define the errors of our *ab initio* calculations relative to the true solutions of that equation, and that is what we will do in the following. This implies that we will not be concerned with relativistic or non-BO effects. There are cases, of course, where these effects become important, but we will restrict our discussion to the nonrelativistic BO case. Clearly, it depends on the level of accuracy we are aiming at whether relativistic or non-BO effects should be taken into account. Table 1 shows the order of magnitude of these effects on electronic barriers in the H_2O and SiH_3^- molecules. When aiming at an accuracy of about ± 50 cm^{-1} (± 0.6 kJ mol^{-1}), as in the cited works[9,10], it becomes mandatory to include these effects. Scalar relativistic effects and spin-orbit interactions become certainly very important for molecular systems containing heavy elements. We will not discuss this topic further, although it must be said that the basis-set convergence of relativistic calculations is even slower – in fact, significantly slower[11] – than in the nonrelativistic case, implying that explicitly

Table 1. Barrier to linearity of H_2O [a,b] and barrier to inversion of SiH_3^-.[c,d]

	$\Delta E/\mathrm{cm}^{-1}$	
	H_2O	SiH_3^-
Nonrelativistic BO-value	11086	8314
Relativistic correction	58	50
Diagonal BO-correction	−17	−13
Total	11127	8351

a C_{2v}: $R_{O-H} = 95.885$ pm, $\angle_{H-O-H} = 104.343°$.[9]
b $D_{\infty h}$: $R_{O-H} = 93.411$ pm.[9]
c C_{3v}: $R_{Si-H} = 153.753$ pm, $\angle_{H-Si-H} = 95.196°$.[10]
d D_{3h}: $R_{Si-H} = 147.641$ pm.[10]

correlated methods would be very powerful in particular in relativistic electronic-structure calculations.

In any case we define[12,13] the *apparent error* as the difference between the true solution of the electronic Schrödinger equation (equal to the experimental value of the property of interest if relativistic and non-BO effects can be neglected) and the particular calculation that we carry out. This apparent error can be subdivided into the *basis-set error* and the *n-electron error*, as our calculation will employ a truncated one-electron basis set and a truncated n-electron wavefunction model. Thus, two approximations are introduced, leading to two sources of error to our computed data. These two errors (basis-set error and n-electron error) depend on the basis set used. In the limit of a complete basis, the basis-set error vanishes and only the n-electron error remains. This remaining n-electron error at the limit of a complete basis is denoted as the *intrinsic error* of the respective n-electron wavefunction model.

We will only be concerned with the basis-set error in the present lecture.

3 The basis-set error

3.1 Correlation-consistent basis sets

The development of the correlation-consistent basis sets (cc-pVXZ) by Dunning and co-workers[14−19] has prompted a series a studies of the basis-set dependence of computed molecular properties in terms of these basis sets. The basis sets are characterized by the *cardinal number* X, and basis sets for $X = 2, \ldots, 6$ have been derived, also in the aug-cc-pVXZ[15] and cc-pCVXZ[18] forms. The latter two series of basis sets are cc-pVXZ sets to which diffuse, respectively tight Gaussians have been added. Also aug-cc-pCVXZ basis sets exist as well as doubly augmented sets, and so on.

A typical example of a correlation-consistent basis-set convergence study is shown in Table 2, which shows the all-electron correlation energy of the H_2O molecule at the CCSD(T) level[20,21]. Empirically, as shown in Figure 1, one finds

Table 2. All-electron correlation energies (in E_h) of the H_2O molecule[a].

Basis	N[b]	E_{SCF}	ΔE^c_{MP2}	ΔE^d_{CCSD}	$\Delta E^e_{CCSD(T)}$
Orbital basis					
cc-pCVDZ	28	−76.0272	−0.2413	−0.0104	−0.0033
cc-pCVTZ	71	−76.0574	−0.3175	−0.0067	−0.0082
cc-pCVQZ	144	−76.0649	−0.3426	−0.0039	−0.0095
cc-pCV5Z	255	−76.0671	−0.3516	−0.0018	−0.0100
cc-pCV6Z[f]	412	−76.0673	−0.3563	−0.0005	−0.0102
R12 basis[20] O:15s9p7d5f3g1h/ H:9s7p5d3f1g	320	−76.0674	−0.3615	0.0015	−0.0100

a H_2O geometry: $R_{O-H} = 1.80885\ a_0$, $\angle_{H-O-H} = 104.52°$.[20]

b Number of basis functions.

c Total second-order correlation energy. The MP2-R12 calculations were based on standard approximation B.

d CCSD increment, $E_{CCSD} - E_{MP2}$.

e Triples contribution, $E_{CCSD(T)} - E_{CCSD}$.

f Not available in basis set library[18]. Constructed in analogy to the smaller sets[21].

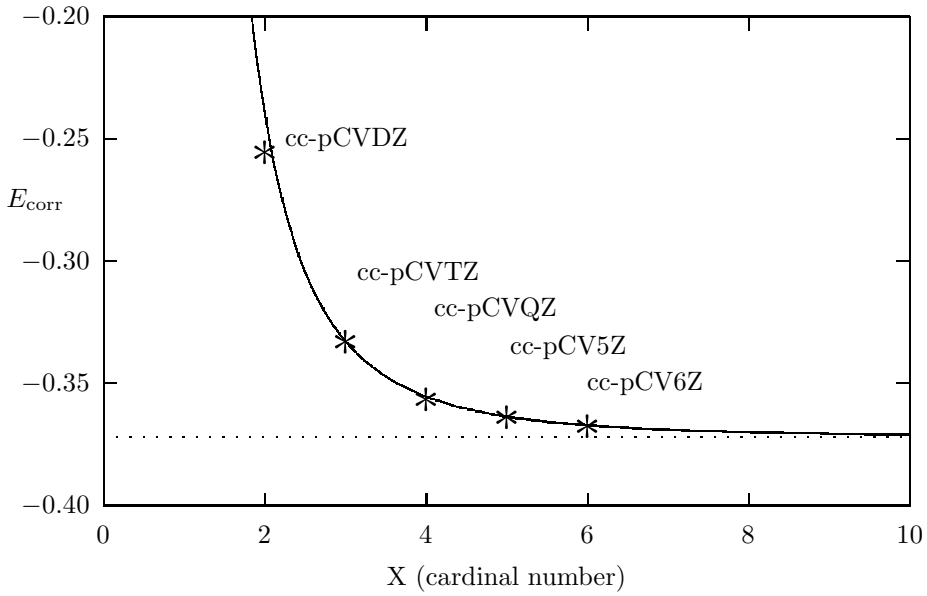


Figure 1. All-electron CCSD(T) correlation energy (in E_h) of the water molecule as a function of the cc-pCVXZ basis sets. Cf. Table 2.

that the cc-pCVXZ results are well represented by the formula:

$$E_{corr}(X) = E_{corr}(\infty) + c\,X^{-3}. \qquad (2)$$

The actual curve in Figure 1 corresponds to $E_{\rm corr}(\infty) = -0.372\ E_{\rm h}$ and $c = 1.07\ E_{\rm h}$. Equation (2) provides an estimate for the basis-set error as a function of X. This basis-set error is equal to $c\,X^{-3}$, and thus, if we would like to reduce the basis-set error of our standard CCSD(T) calculation of H_2O to below 1 m$E_{\rm h}$, we ought to use basis sets with cardinal numbers $X > 10$. Such basis sets do not exist, however. We note in passing that also in the case of the above-mentioned barrier to linearity of the H_2O molecule one finds that a ridiculously large (1276 functions) basis set with $X = 10$ ought to be used to compute the nonrelativistic BO barrier accurate to within 10 cm^{-1}.[9]

Furthermore, we note that the number of basis functions in a cc-pVXZ basis set grows with the third power of X. For first-row atoms, the number is:

$$N = \frac{1}{3}(X+1)(X+\frac{3}{2})(X+2). \tag{3}$$

This cubic dependence implies that we can represent the computed correlation energies obtained with correlation-consistent basis sets alternatively as[22,23]:

$$E_{\rm corr}(N) = E_{\rm corr}(\infty) + c'\,N^{-1} + \mathcal{O}(N^{-2}). \tag{4}$$

It must be said, however, that the latter equation is only correct for the "magic" numbers N that correspond to a correlation-consistent hierarchy of basis sets.

The number of two-electron integrals grows with the fourth power of the number of basis functions, and thus, also the computing time will grow as N^4 when the basis set is enlarged. In the light of this scaling, (1) thus follows from (4). Note that the N^4-scaling refers to an increase of the number of functions per atom. Correlation treatments are being developed that scale linearly with the size of the system[24–26], but from that perspective, the whole molecular system is increased while the number of functions per atom remains constant. Intregral-screening techniques are of little help to the N^4-scaling problem when we enlarge only the basis set per atom.

3.2 Principal expansion

For the ground state of the He atom, it is empirically found that the energy contribution of an individual natural orbital is almost completely independent from its angular momentum and magnetic quantum numbers and only proportional to the inverse sixth power of its principal quantum number n:[27,28]

$$\Delta E_{n\ell m} = -A\left(n - \frac{1}{2}\right)^{-6}. \tag{5}$$

On itself, the observation that the natural orbital increments are proportional to n^{-6} is a strong motivation for the development of hierarchies of basis sets in terms of the principal quantum number n, as is the case for the cc-pVXZ sets. We will denote such an expansion as *principal expansion*. The basis set on a given atom is then defined by some principal quantum number $n_{\rm max} = X$ and all shells of orbitals $\chi_{n\ell m}$ up to that quantum number are included in the principal expansion.

For a calculation that includes only shells with principal quantum number $n \leq X$, the truncation error δE_X is obtained by summing all increments $\Delta E_{n\ell m}$ for

those orbitals not included in the expansion:

$$\delta E_X = \sum_{n=X+1}^{\infty} \sum_{\ell=0}^{n-1} \sum_{m=-\ell}^{\ell} |\Delta E_{n\ell m}|. \tag{6}$$

The summation can be carried out in closed form and expressed in terms of the polygamma function $\Psi(n,x)$, but replacing the summation over n by integration leads immediately to the same leading X^{-3}- and X^{-4}-terms:

$$\begin{aligned}
\delta E_X &= A \sum_{n=X+1}^{\infty} n^2 \left(n - \frac{1}{2}\right)^{-6} \\
&= \frac{A}{6}\Psi(3, X+\frac{1}{2}) - \frac{A}{24}\Psi(4, X+\frac{1}{2}) + \frac{A}{480}\Psi(5, X+\frac{1}{2}) \\
&= \frac{A}{3}X^{-3} + \frac{A}{4}X^{-4} - \frac{7A}{60}X^{-5} + \mathcal{O}(X^{-6}), \qquad (7) \\
\delta E_X &\approx A \int_{X+\frac{1}{2}}^{\infty} n^2 \left(n - \frac{1}{2}\right)^{-6} dn = \frac{A}{3}X^{-3} + \frac{A}{4}X^{-4} + \frac{A}{20}X^{-5}. \qquad (8)
\end{aligned}$$

In a similar fashion, we can compute the truncation error δE_L of a partial-wave expansion that includes all contributions up to $\ell \leq L$:

$$\begin{aligned}
\delta E_L &= \sum_{n=L+2}^{\infty} \sum_{\ell=L+1}^{n-1} \sum_{m=-\ell}^{\ell} |\Delta E_{n\ell m}| \\
&= \frac{A}{6}\Psi(3, L+\frac{3}{2}) - \frac{A}{24}\Psi(4, L+\frac{3}{2}) - \frac{A}{120}(L^2+2L+\frac{3}{4})\Psi(5, L+\frac{3}{2}) \\
&= \frac{2A}{15}(L+1)^{-3} + \frac{A}{4}(L+1)^{-4} + \frac{2A}{15}(L+1)^{-5} + \mathcal{O}\left((L+1)^{-6}\right), \qquad (9) \\
\delta E_L &\approx A \int_{L+\frac{3}{2}}^{\infty} \left[n^2 - (L+1)^2\right] \left(n - \frac{1}{2}\right)^{-6} dn \\
&= \frac{2A}{15}(L+1)^{-3} + \frac{A}{4}(L+1)^{-4} + \frac{A}{20}(L+1)^{-5}. \qquad (10)
\end{aligned}$$

For variational calculations on He, the expression (10) for the basis-set error of the partial-wave expansion has been found empirically[28]. Hill has shown how the coefficients can be derived from the exact He ground-state wavefunction[29] and a very similar expression was derived nearly 40 years ago by Schwartz for the partial-wave expansion in the framework of perturbation theory[30,31]. Kutzelnigg and Morgan have shown that similar rates of convergence apply to many-electron atoms as well[32,33].

It is remarkable that no such analysis exists for the principal expansion, but since the working assumption (5), which is supported by strong numerical evidence, leads to the correct expression for the basis-set error of the partial-wave expansion, it is safe to assume that the basis-set error of the principal expansion is given by (8). This is also supported by the observation that the basis-set error of molecular calculations appears to vanish as X^{-3} with the cardinal number of the correlation-consistent basis sets (cf. Figure 1) and by the fact that extrapolations based on the X^{-3}-form (2) work so well, as we will see in the next section.

Table 3. Statistical measure of errors for all-electron CCSD(T) atomization energies relative to experiment[a]. All values in kJ mol^{-1}.

	Raw cc-pCVXZ results				
	2	3	4	5	6
Δ	−103.07	−34.00	−13.46	−6.61	−4.06
$\Delta_{\rm std}$	37.19	13.58	5.64	3.09	2.14
$\bar{\Delta}_{\rm abs}$	103.07	34.00	13.46	6.61	4.06
$\Delta_{\rm max}$	155.71	51.58	20.15	10.71	7.04
	cc-pCV[(X-1)X]Z extrapolated				
		23	34	45	56
Δ		−14.67	−0.23	0.08	−0.44
$\Delta_{\rm std}$		8.39	2.16	1.12	1.05
$\bar{\Delta}_{\rm abs}$		14.74	1.68	0.90	0.84
$\Delta_{\rm max}$		29.53	4.01	2.45	2.31

a The experimental atomization energies were corrected for (anharmonic) zero-point vibrational energies and (scalar and spin-orbit) relativistic effects.

3.3 Extrapolation schemes

Recently, Bak *et al.*[13,34] applied two-point extrapolations of the form[35]:

$$E_{\rm corr}(X) = E_{\rm corr}(\infty) + cX^{-3} \tag{11}$$

to the CCSD(T)(FULL)/cc-pCVXZ atomization energies of the following 16 closed-shell molecules: CH_2 ($\tilde{a}\ ^1A_1$ state), CH_4, NH_3, H_2O, HF, C_2H_2, C_2H_4, HCN, N_2, CO, H_2CO, HNO, H_2O_2, F_2, CO_2, and H_2. Table 3 shows the mean error ($\bar{\Delta}$), standard deviation ($\Delta_{\rm std}$), mean absolute error ($\bar{\Delta}_{\rm abs}$), and maximum error ($\Delta_{\rm max}$) of the raw CCSD(T)(FULL)/cc-pCVXZ calculations of these molecules in comparison with the two-point extrapolations. The two-point extrapolations are based on calculations with two subsequent basis sets with cardinal numbers $X-1$ and X. For example, the extrapolation denoted as "56" refers to the results that were obtained by inserting the CCSD(T)(FULL)/cc-pCV5Z and CCSD(T)(FULL)/cc-pCV6Z correlation energies into (11) and extracting $E_{\rm corr}(\infty)$ from the two equations with two unknowns. Note that this extrapolation only applies to the electron-correlation contribution to the atomization energy. It was not clear how to extrapolate the Hartree-Fock energy[36] (which perhaps converges exponentially)[37] and thus the Hartree-Fock energy of the larger of the two basis sets was taken.

The extrapolation appears to work! The results displayed in Table 3 show that all errors are significantly reduced by applying the two-point extrapolation. At the raw CCSD(T)(FULL)/cc-pCV6Z level, the mean absolute error is still as large as 4.06 kJ mol^{-1}. We usually refer to an accuracy of about 4 kJ mol^{-1} (or 1 kcal mol^{-1}) as "chemical accuracy". This chemical accuracy is clearly achieved already at the 34-extrapolated level, where the mean absolute error is only 1.68 kJ mol^{-1}. At the 45- and 56- extrapolated levels, the mean absolute errors are 0.90 and 0.84 kJ mol^{-1} and these errors are good estimates of the intrinsic errors of the

Table 4. Application of the two-point (X-1,X)-extrapolation to the MP2(FC) correlation contribution (in kJ mol^{-1}) to the interaction energy of the H_2O dimer in the Halkier geometry[41].

	Raw results[a]	(X-1,X) Extrapolation
aug-cc-pVDZ	−3.35	
aug-cc-pVTZ	−4.78	−5.38
aug-cc-pVQZ	−5.33	−5.74
aug-cc-pV5Z	−5.54	−5.75
Schütz *et al.*[42]	−5.74[b]	
R12 result[43]	−5.78	

a Within the full counterpoise framework.

b Original value transferred to Halkier geometry[41].

CCSD(T)(FULL) atomization energies.

The two-point X^{-3}-extrapolation has also been tested on reaction enthalpies[13,34], weak interactions[38], spectroscopic constants[39], and dipole moments[40], all with encouraging results. As an example, consider the MP2(FC) contribution to the interaction energy of the H_2O dimer in the fixed geometry optimized by Halkier *et al.*[41] (cf. Table 4). Already the extrapolation from the aug-cc-pVDZ and aug-cc-pVTZ data provides a significant improvement over the raw aug-cc-pVTZ results. The extrapolation is comparable to the aug-cc-pVQZ level. The extrapolation from the aug-cc-pVTZ and aug-cc-pVQZ basis sets virtually coincides with very large calculations by Schütz *et al.*[42] (using more than 1000 basis functions up to $\ell = 7$) and R12 calculations[43]. It must be noted, however, that the extrapolations only work well when applied to the counterpoise-corrected raw data. In general, the extrapolations can be expected to work only when the basis-set error is solely due to the slow convergence of the Coulomb cusp – that is, other basis-set errors such as the basis-set superposition error must be taken care of otherwise. The poor description of the electron-electron cusp must be the only remaining basis-set error. Also in case of the extrapolation of the dipole moments, for example, it was important to use aug-cc-pVXZ basis sets, as the extrapolation is unable to account for missing diffuse functions.

An important feature of the two-point X^{-3}-extrapolation is that it can in principle be applied to any point on the molecular potential energy hypersurface. It is possible to define a model chemistry on the basis of calculations with two basis sets in conjunction with the corresponding two-point X^{-3}-extrapolation. One can optimize molecular geometries and compute harmonic vibrational frequencies within the particular model chemistry[44−47]. Of course, one must be able to separate the Hartree-Fock and correlation contributions, because the extrapolation applies to the (dynamical) correlation energy only. This separation is not always clear-cut on every point on the hypersurface, but maybe one can use an appropriate MCSCF energy as a reference.

Furthermore, we note that in the past other extrapolations from correlation-consistent basis sets have also been tried[12,48−52]. These include the exponential

form[48,52]:

$$E(X) = E(\infty) + a\,\exp(-bX), \tag{12}$$

and the functional forms[49−51]:

$$E(X) = E(\infty) + a\left(X + \frac{1}{2}\right)^{-4}, \tag{13}$$

$$E(X) = E(\infty) + b\left(X + \frac{1}{2}\right)^{-4} + c\left(X + \frac{1}{2}\right)^{-6}, \tag{14}$$

$$E(X) = E(\infty) + d\left(X + \frac{1}{2}\right)^{-e}. \tag{15}$$

It is clear, however, that the exponential form overestimates the rate of convergence of the correlation energy obtained with the correlation-consistent basis sets. Moreover, the leading term is clearly X^{-3}.

All of the above extrapolations were mainly concerned with the (correlation) energy. The basis-set convergence of molecular properties might of course be different from the X^{-3}-form for the energy. When we think of the property of interest as an energy-derivative concerning the Hamiltonian:

$$\hat{H}(\lambda) = \hat{H}(0) + \lambda\,\hat{V}, \tag{16}$$

then we can expand the correlation energy as:

$$E_{\mathrm{corr}}(\lambda, X) = \sum_{n=0}^{\infty}\sum_{k=0}^{\infty}\frac{C_n^{(k)}\lambda^k}{k!}X^{-n}. \tag{17}$$

The coefficients $C_n^{(0)}$ govern the convergence of the energy, the $C_n^{(1)}$ govern the convergence of the first-order property, and so on[53].

Let us finally have a brief look at the CBS (complete basis set) extrapolation to the MP2 limit developed by Petersson and co-workers[54−62], which is found in standard and widely distributed quantum chemistry programs. For pairs of occupied α and β spin-orbitals ($\alpha\beta$ pairs) and for $\alpha\alpha$-type second-order pair energies, the CBS extrapolation is based on the expressions:

$${}^{\alpha\beta}e_{ij}^{(2)}(N) = {}^{\alpha\beta}e_{ij}^{(2)}(\infty) + {}^{\alpha\beta}f_{ij}\frac{25}{512}(N + {}^{\alpha\beta}\delta_{ij})^{-1}, \tag{18}$$

$${}^{\alpha\alpha}e_{ij}^{(2)}(N) = {}^{\alpha\alpha}e_{ij}^{(2)}(\infty) + {}^{\alpha\alpha}f_{ij}\frac{25}{512}(N + {}^{\alpha\alpha}\delta_{ij})^{-5/3}, \tag{19}$$

where:

$${}^{\alpha\beta}f_{ij} = |S|_{ij}^2, \qquad {}^{\alpha\alpha}f_{ij} = 2|S|_{ij}^2\left(\frac{1-|S|_{ij}^2}{1+|S|_{ij}^2}\right), \tag{20}$$

and:

$$|S|_{ij} = \int |\varphi_i(\mathbf{r})\varphi_j(\mathbf{r})|d\tau. \tag{21}$$

Equation (18) shows the same N^{-1}-dependence as found in (4), corresponding to the X^{-3}-form. This is the correct basis-set error of the principal expansion of the

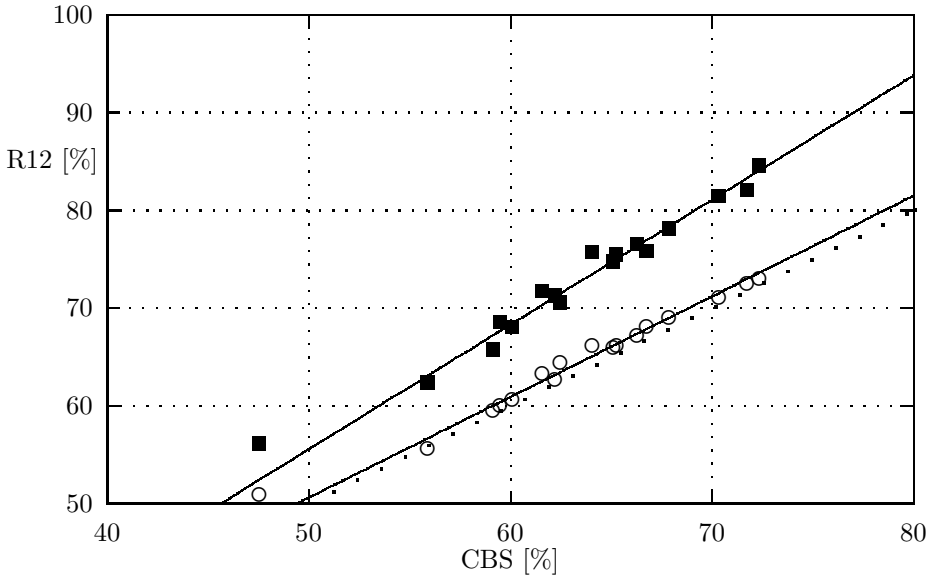


Figure 2. Ratio of CCSD(T) to MP2 basis-set errors, $\delta E_{\mathrm{CCSD(T)}}/\delta E_{\mathrm{MP2}}$, of the APNO basis set. Comparison of the ratio (i.e., interference factor) computed by the CBS-QCI/APNO method with the ratio obtained from R12 calculations. At the R12 level, the CCSD(T) error is $\delta E_{\mathrm{CCSD(T)}}$ = CCSD(T)-R12/B − CCSD(T)/APNO, while the MP2 error is either δE_{MP2} = MP2-R12/A − MP2/APNO (○) or δE_{MP2} = MP2-R12/B − MP2/APNO (■). The dotted line is a straight line with slope one through the origin.

He ground state. For a triplet state of He, however, or in general for triplet pair energies, the basis-set error of the principal expansion is X^{-5} (or $(L+1)^{-5}$).[32] This triplet basis-set error is reflected by the $(N + {}^{\alpha\alpha}\delta_{ij})^{-5/3}$ dependence.

We note that the $\alpha\beta$ and $\alpha\alpha$ pairs of the spin-orbital CBS method do not represent spin-adapted singlet and triplet pairs. Rather, the $\alpha\beta$ pair contributes to both singlet and triplet pairs. The purpose of the prefactors ${}^{\alpha\beta}f_{ij}$ and ${}^{\alpha\alpha}f_{ij}$ is to damp the extrapolated truncation errors for spatially distant pairs of (localized) orbitals.

In (18) and (19), ${}^{\alpha\beta}e_{ij}^{(2)}(\infty)$ and ${}^{\alpha\beta}\delta_{ij}$, which constitute the fitting parameters, are obtained from two-point fits. The first point is chosen as $N = 1$ – that is, as the Hartree–Fock calculation, for which ${}^{\alpha\beta}e_{ij}^{(2)}(1)$=0. A prescribed range of N values are then tried for the second point, each time computing the corresponding pair energy ${}^{\alpha\beta}e_{ij}^{(2)}(N)$. For each N, the two equations are solved for the two unknowns and the most negative ${}^{\alpha\beta}e_{ij}^{(2)}(\infty)$ is taken as the final, extrapolated second-order pair energy. The same procedure is followed for the $\alpha\alpha$ pairs.

To obtain the infinite-order corrections, the second-order corrections are scaled

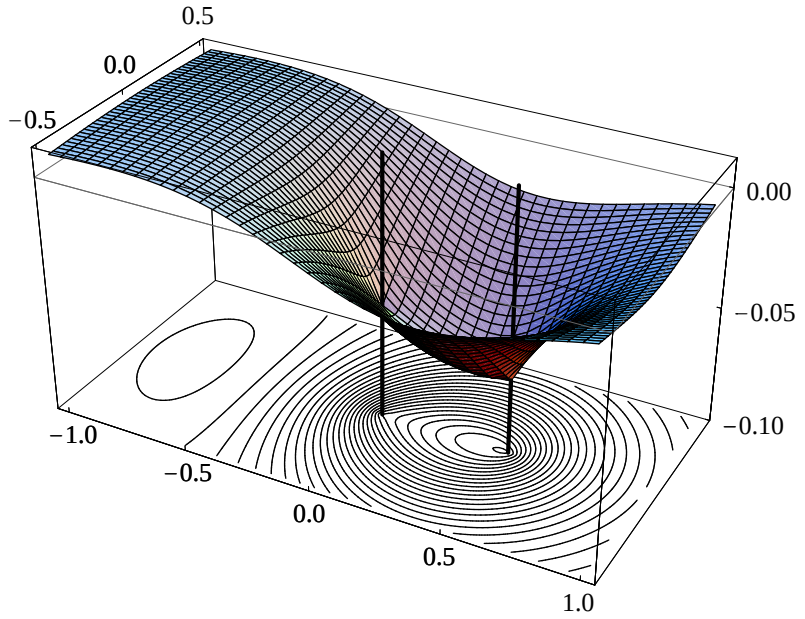


Figure 3. Coulomb hole (difference between the exact and the Hartree-Fock wavefunctions) of the He ground state as a function of the x_2 and y_2 coordinates of the second electron. The nucleus is located in the origin and the first electron is kept fixed at $x_1 = 0.5\ a_0$, $y_1 = z_1 = 0$.

by the interference factors:

$$\delta e_{ij}^{(\infty)} = \left(\sum_{\mu_{ij}=1}^{N_{\mathrm{virt}}+1} c_{\mu_{ij}}^{(1)} \right)^2 \delta e_{ij}^{(2)}, \tag{22}$$

computed from the first-order wavefunction. The scaling with this interference factor is remarkably accurate. For a variety of small closed-shell molecules[63], Figure 2 shows the overall CBS scaling factor – that is, the scaling factor for the sum of all pair energies – in comparison with the CCSD(T) to MP2 ratio as computed by the R12/A and R12/B methods. Hence, the interference factor appears to provide, by simple scaling, a reasonable estimate of the basis-set error at the CCSD(T) level once the corresponding MP2 basis-set error is known or obtained by extrapolation[62].

4 Coulomb hole

An explicit dependence of the wavefunction on r_{12} was already discussed by Slater[1] and Hylleraas[2] in the early days of quantum mechanics[64]. Slater and Hylleraas were concerned with the He atom, where the ground-state wavefunction can be written as a function $\Psi(r_1, r_2, r_{12})$ of the two nucleus-electron distances r_1 and r_2 and the electron-electron distance r_{12}. Then, the corresponding effective Schrödinger

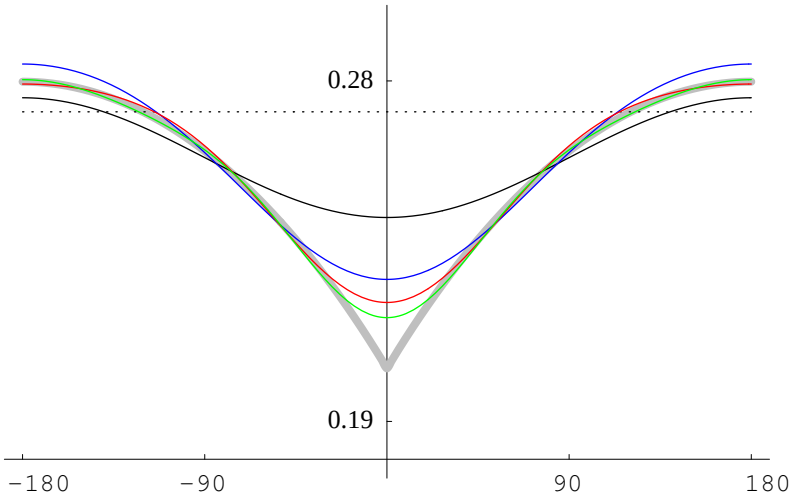


Figure 4. Coulomb hole of the He ground state. Comparison of standard CI wavefunctions obtained with the cc-pVDZ, cc-pVTZ, cc-pVQZ, and cc-pV5Z basis sets (upper curves, from top to bottom) with the exact wavefunction $\Psi(r_1, r_2, r_{12})$ (lower curve) as a function of the angle θ_{12} (deg) between the position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$, with fixed $r_1 = r_2 = 0.5\ a_0$.

equation has the form[2]:

$$H\,\Psi(r_1, r_2, r_{12}) = E\,\Psi(r_1, r_2, r_{12}), \tag{23}$$

with

$$\begin{aligned} H = &-\frac{1}{2}\frac{\partial^2}{\partial r_1^2} - \frac{1}{r_1}\frac{\partial}{\partial r_1} - \frac{Z}{r_1} - \frac{1}{2}\frac{\partial^2}{\partial r_2^2} - \frac{1}{r_2}\frac{\partial}{\partial r_2} - \frac{Z}{r_2} - \\ &- \frac{\partial^2}{\partial r_1 \partial r_{12}}\frac{r_1^2 - r_2^2 + r_{12}^2}{r_1 r_{12}} - \frac{\partial^2}{\partial r_2 \partial r_{12}}\frac{r_2^2 - r_1^2 + r_{12}^2}{r_2 r_{12}} - \\ &- \frac{\partial^2}{\partial r_{12}^2} - \frac{2}{r_{12}}\frac{\partial}{\partial r_{12}} + \frac{1}{r_{12}}, \end{aligned} \tag{24}$$

where Z is the nuclear charge. The exact solution to this equation should contain terms that balance the Coulomb singularities. For example, it should be ensured that:

$$\left\{ H\,\Psi(r_1, r_2, r_{12}) \right\}_{\mathbf{r}_1 = \mathbf{r}_2 = \mathbf{r}_c} = E\,\Psi(r_c, r_c, 0) \tag{25}$$

for any point $\mathbf{r}_c$. This balance is established if:

$$\left\{ \frac{\partial \Psi(r_1, r_2, r_{12})}{\partial r_{12}} \right\}_{\mathbf{r}_1 = \mathbf{r}_2 = \mathbf{r}_c} = \frac{1}{2}\Psi(r_c, r_c, 0), \tag{26}$$

because then the last and second-last terms of (24) cancel. Similar conditions can be derived for the Z/r_1 and Z/r_2 singularities. Equation (26) is known as Kato's

Table 5. Principal expansion of the two-electron Darwin energy (in $\mu E_{\rm h}$) of the He ground state[a,b].

$X=2$	$X=3$	$X=4$	$X=5$	$X=6$	$X=7$	$X=8$
-23.579	-21.564	-20.605	-20.039	-19.664	-19.397	-19.197

a As a function of numerically optimized cc-pVXZ basis sets[53].

b The exact value is $-\frac{\pi}{c^2}\langle\delta(r_{12})\rangle = -17.791\ \mu E_{\rm h}$.

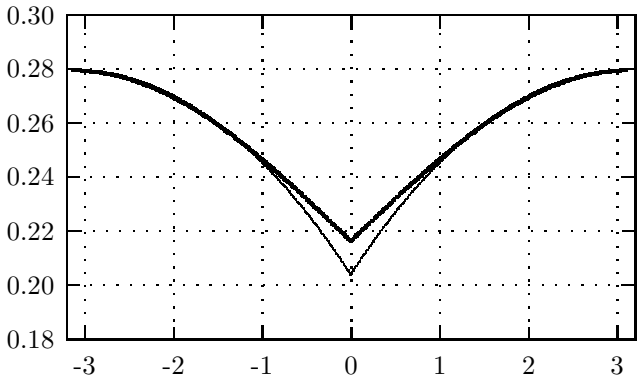


Figure 5. Coulomb hole of the He ground state. Comparison of the three-parameter Hylleraas function (33) (upper curve) with the exact wavefunction $\Psi(r_1, r_2, r_{12})$ (lower curve) as a function of the angle θ_{12} (rad) between the position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$, with fixed $r_1 = r_2 = 0.5\ a_0$.

electron-electron cusp condition[65], which implies that for small r_{12} the (unnormalized) wavefunction behaves as:

$$\Psi(r_1, r_2, r_{12}) = 1 + \frac{1}{2} r_{12} + \mathcal{O}(r_{12}^2). \tag{27}$$

The cusp is easy to see at the bottom of the Coulomb hole in Figure 3.

A CI expansion in orbital products cannot reproduce this electron-electron cusp, as can be seen in Figure 4. At the bottom of the Coulomb hole of standard CI wavefunctions, there is no cusp as the first derivative is zero. Moreover, the convergence to the exact bottom of the Coulomb hole is extremely slow, as illustrated by the convergence of the expectation value $\langle\delta(r_{12})\rangle$, which is nothing but an integral over configuration space of the bottom of the Coulomb hole. This expectation value is related to the relativistic two-electron Darwin energy as:

$$E_{\text{two-electron Darwin}} = -\frac{\pi}{c^2}\langle\delta(r_{12})\rangle, \tag{28}$$

and this two-electron Darwin energy is displayed in Table 5 as a function of the cardinal number of numerically optimized cc-pVXZ basis sets. Interestingly, the two-electron Darwin term appears to converge as X^{-1}.[53] Thus, whereas the Coulomb hole itself converges as X^{-1}, its consequence for the correlation energy is that the latter converges as X^{-3}.

Hylleraas proposed to expand the He ground-state wavefunction as:

$$\Psi_N = \exp(-\zeta s) \sum_{i=1}^{N} c_i s^{l_i} t^{2m_i} u^{n_i}, \tag{29}$$

with:

$$s = r_1 + r_2, \quad t = r_1 - r_2, \quad u = r_{12}. \tag{30}$$

The first function studied by Hylleraas was the 3-term function:

$$\Psi_3 = \exp(-\zeta s)(c_1 + c_2 u + c_3 t^2), \tag{31}$$

and the variationally determined energy for this function was $E = -2.902\,43\ E_\mathrm{h}$ for $\zeta = 1.816$. Hylleraas furthermore reported that he had spent considerable effort to find the next most important terms in the expansion (29). He ended up with the 6-term function:

$$\Psi_6 = \exp(-\zeta s)(c_1 + c_2 u + c_3 t^2 + c_4 s + c_5 s^2 + c_6 u^2), \tag{32}$$

leading to the energy $E = -2.903\,33\ E_\mathrm{h}$ for $\zeta = 1.756$.

It is truly amazing how accurate these compact wavefunctions are. Let us for example consider the 3-term function in more detail:

$$\Psi_3(r_1, r_2, r_{12}) = 1.331\, e^{-1.816(r_1+r_2)} \left[1 + 0.292\, r_{12} + 0.131\, (r_1 - r_2)^2\right]. \tag{33}$$

When we plot this wavefunction as a function of the angle between the position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$, keeping the electrons fixed at a distance of $0.5\ a_0$ from the nucleus, we see that it already closely resembles the exact wavefunction (cf. Figure 5). In this plot, as r_1 and r_2 are constant, the function takes the form:

$$\Psi_3(0.5, 0.5, r_{12}) = 0.2165\, (1 + 0.292\, r_{12}), \tag{34}$$

that is, it has only one term linear in r_{12}. When comparing it with the standard CI wavefunctions in Figure 4, one is tempted to conclude that the 3-term Hylleraas-function is of an accuracy comparable to that of a cc-pVQZ- or cc-pV5Z/CI wavefunction. Indeed, the two-electron Darwin energy computed from the 3-term wavefunction amounts to $-19.427\ \mu E_\mathrm{h}$. This value is practically identical to the CI value ($-19.397\ \mu E_\mathrm{h}$) obtained with the fully optimized cc-pV7Z basis set! The correlation contribution from the 6-term Hylleraas-function is $-18.732\ \mu E_\mathrm{h}$. We can represent the two-electron Darwin energy as:

$$E_{\mathrm{two-electron\ Darwin}}(X)/\mu E_\mathrm{h} = -17.791 - 11.25/X, \tag{35}$$

and thus, this 6-term value corresponds roughly to the $X = 12$ level.

5 Many-electron systems

For two-electron systems, the inclusion of r_{12} into the electronic wavefunction is not too difficult. The situation is different for many-electron systems, however. Hylleraas coordinates have been used in calculations on atoms with up to four electrons[66,67], but it is almost impossible to perform any useful calculations with

these coordinates on molecules with more than two electrons. This is very unfortunate because the Hylleraas expansion is in principle very rapidly convergent.

The very difficult many-electron many-center integrals are the reason why many-electron molecular calculations with Hylleraas coordinates are too complicated to be practical. In the remaining part of the present lecture, we will be concerned with the auxiliary techniques that are utilized to avoid the very difficult many-electron many-center integrals, thereby providing new opportunities for accurate calculations of molecular electronic structure.

6 Second quantization

We use the following notation for one- and many-electron functions:

$$\begin{aligned} \Psi,\ \Psi_N,\ \ldots &\quad n\text{-electron wavefunctions,} \\ \Phi,\ \Phi_k,\ |\Phi\rangle,\ |\Phi_i^a\rangle,\ |\Phi_{ij}^{ab}\rangle\ \ldots &\quad \text{Slater determinants,} \\ \varphi_\kappa,\ \varphi_p\ \ldots &\quad \text{orthonormal spin-orbitals,} \\ \chi_q\ \ldots &\quad \text{atomic basis functions.} \end{aligned}$$

Two-electron Slater determinants are also written as:

$$|ij\rangle = \frac{1}{\sqrt{2}}\big\{\varphi_i(1)\varphi_j(2) - \varphi_i(2)\varphi_j(1)\big\}. \tag{36}$$

We employ a given, finite basis set $\{\varphi_p\}$ of one-electron functions in which the calculations are performed as well as a complete set $\{\varphi_\kappa\}$, which is used in the second-quantization formalism. Orbital indices are used as follows:

$$\begin{aligned} i,\ j,\ k,\ \ldots &\quad \text{occupied spin-orbitals,} \\ a,\ b,\ c,\ \ldots &\quad \text{virtual spin-orbitals within the given basis set,} \\ p,\ q,\ r,\ \ldots &\quad \text{arbitrary spin-orbitals within the given basis set,} \\ \alpha,\ \beta,\ \gamma,\ \ldots &\quad \text{virtual spin-orbitals within a complete basis set,} \\ \kappa,\ \lambda,\ \mu,\ \ldots &\quad \text{arbitrary spin-orbitals that form a complete basis set.} \end{aligned}$$

This convention is illustrated in Figure 6. Antisymmetrized integrals and replacement operator amplitudes are written in tensor form[68−70]:

$$X_p^q = \langle p|X|q\rangle, \quad \bar{Y}_{rs}^{pq} = \langle rs|Y|pq\rangle, \tag{37}$$

assuming the Einstein summation convention over repeated indices. In the second-quantization particle-hole formalism, the closed-shell reference state $|\Phi\rangle$ is taken as the physical vacuum and the Hamiltonian is shifted by the reference energy. With $g = r_{12}^{-1}$ and h the usual one-electron Hamiltonian, the normal-ordered Hamiltonian can be written as:

$$H_N = H - \langle\Phi|H|\Phi\rangle = F_N + W_N, \tag{38}$$

$$F_N = f_\kappa^\lambda \tilde{a}_\lambda^\kappa, \quad f_\kappa^\lambda = h_\kappa^\lambda + \bar{g}_{\kappa i}^{\lambda i}, \quad W_N = \tfrac{1}{4}\bar{g}_{\kappa\lambda}^{\mu\nu}\tilde{a}_{\mu\nu}^{\kappa\lambda}, \tag{39}$$

in terms of the orthonormal one-electron basis $\{\varphi_\kappa\}$. Greek letters emphasize that the second-quantized Hamiltonian is exact only if it is defined in terms of a complete

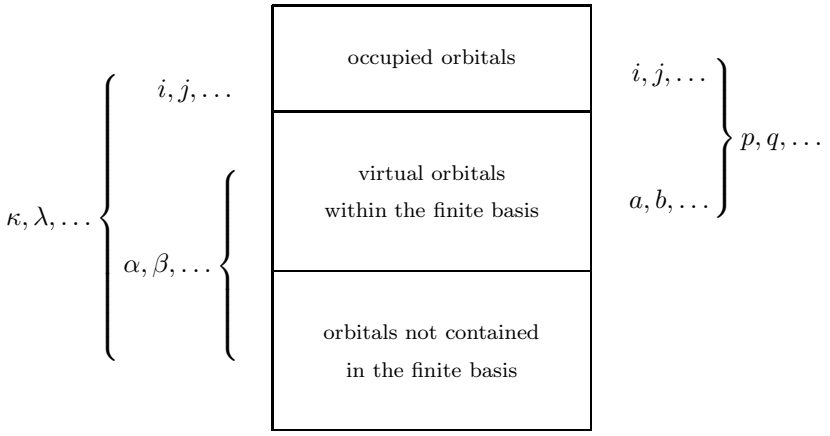


Figure 6. Illustration of the spaces spanned by the one-particle basis functions: $i, j, \cdots$ = occupied spin-orbitals, $a, b, \cdots$ = non-occupied spin-orbitals contained in the finite basis, $\alpha, \beta, \cdots$ = complete set of virtuals, $p, q, \cdots$ = arbitrary spin-orbital contained in the finite basis, $\kappa, \lambda, \cdots$ = complete basis set.

basis set. The $\tilde{a}^{\kappa}_{\lambda}$ and $\tilde{a}^{\kappa\lambda}_{\nu\mu}$ are replacement operators in normal order, defined with respect to $|\Phi>$ according to:

$$\tilde{a}^{q_1 q_2 \ldots q_n}_{p_1 p_2 \ldots p_n} = (\tilde{a}^{p_1 p_2 \ldots p_n}_{q_1 q_2 \ldots q_n})^{\dagger} = \{a^{\dagger}_{q_1} a^{\dagger}_{q_2} \ldots a^{\dagger}_{q_n} a_{p_n} \ldots a_{p_2} a_{p_1}\}, \tag{40}$$

where $a^{\dagger}_{q}$ and a_p are creation and annihilation operators, respectively. The braces in (40) denote the normal-ordering with respect to the physical vacuum, while the counterparts of (40) without a tilde refer to the genuine vacuum. The multiplicative first-quantized operator $r = \sum_{i<j} r_{ij}$ of the interelectronic coordinates can thus be written in the second-quantization form:

$$r = \tfrac{1}{4} \bar{r}^{\mu\nu}_{\kappa\lambda} a^{\kappa\lambda}_{\mu\nu}. \tag{41}$$

If $|\Phi\rangle$ is the Hartree-Fock reference, the f^{λ}_{κ}'s in (39) are elements of the Fock matrix. In terms of canonical orbitals this matrix is diagonal – that is, $f^{\lambda}_{\kappa} = \delta^{\lambda}_{\kappa} \varepsilon_{\lambda}$.

7 Explicitly correlated coupled-cluster doubles model

In single-reference coupled-cluster doubles (CCD) theory, the wavefunction is written in the exponential form:

$$|\Psi\rangle = e^{S} |\Phi\rangle, \tag{42}$$

where $|\Phi\rangle$ is the Hartree-Fock determinant and S the cluster operator, which in this special case is restricted to all double replacements:

$$S \equiv T_2 = \tfrac{1}{4} t^{ij}_{ab} a^{ab}_{ij}. \tag{43}$$

The energy and the amplitudes are determined by:

$$E = \langle \Phi | \tilde{H}_S | \Phi \rangle, \quad \tilde{H}_S = e^{-S} H e^{S}, \tag{44}$$

$$\langle\left(\frac{\partial S\Phi}{\partial t_{ab}^{ij}}\right)|\tilde{H}_S|\Phi\rangle = 0. \tag{45}$$

Jeziorski *et al.*[71] have formulated a first-quantization form of the CCD equations where the pair functions are not expressed in terms of double replacements – as in the orbital approximation – but as expansions in Gaussian geminals. In the original derivation of the theory, they have employed a spin-adapted formulation in terms of singlet and triplet pairs but we will discuss the theory in a spin-orbital formalism.

In the first-quantization formulation, the cluster operator is written as a sum of two-electron operators:

$$T_2 = \sum_{\mu<\nu} t(\mu\nu), \quad t(12) = \sum_{i<j} |\tau_{ij}\rangle\langle ij|. \tag{46}$$

For each pair of occupied spin-orbitals there is one pair function $|\tau_{ij}\rangle$, which is determined by solving the coupled CCD equations for all pairs simultaneously. In conventional CCD theory, that is, in the orbital approximation, each $|\tau_{ij}\rangle$ is expanded as:

$$|\tau_{ij}\rangle = \sum_{a<b} t_{ab}^{ij}\,|ab\rangle, \tag{47}$$

but Szalewicz *et al.*[72–85] expand the pair function in a basis of (Gaussian) geminals:

$$|\tau_{ij}\rangle = \sum_{k_{ij}} c_{k_{ij}}\, Q_{\rm occ}(12)\, \mathcal{A}\big\{\Theta_{k_{ij}}(12)\, G_{k_{ij}}(12)\big\}. \tag{48}$$

The notation with k_{ij} indicates that both the expansion length and the geminals themselves can be chosen individually for each pair of occupied orbitals. $\Theta_{k_{ij}}(12)$ is a proper spin function and the projection operator:

$$Q_{\rm occ}(12) = \big\{1 - P_{\rm occ}(1)\big\}\big\{1 - P_{\rm occ}(2)\big\}, \quad P_{\rm occ} = \sum_i |\varphi_i\rangle\langle\varphi_i| \tag{49}$$

ensures that the $|\tau_{ij}\rangle$'s satisfy the strong-orthogonality condition:

$$|\tau_{ij}\rangle = Q_{\rm occ}(12)\,|\tau_{ij}\rangle. \tag{50}$$

Obviously, the amplitudes are obtained from:

$$\langle\left(\frac{\partial S\Phi}{\partial c_{k_{ij}}}\right)|\tilde{H}_S|\Phi\rangle = 0. \tag{51}$$

A second-quantization formulation can be used if a complete basis set is introduced formally (cf. Section 6). This is convenient for the discussion of the various many-electron integrals that appear when geminals are employed in the framework of CCD theory. In particular, up to four-electron integrals occur in the MP2, MP3, LCCD (linearized coupled-cluster doubles), and FCCD (factorizable coupled-cluster doubles) approaches. The complete CCD equations involve five-electron integrals and the FCCD approximation to CCD consists of ignoring the (nonfactorizable) terms or diagrams that involve these five-electron integrals.

Using the complete basis set notation, the cluster operator can be written as:

$$T_2 = \tfrac{1}{4} c_{k_{ij}}^{ij} \bar{G}_{\alpha\beta}^{k_{ij}} a_{ij}^{\alpha\beta}, \quad c_{k_{ij}}^{ij} \equiv c_{k_{ij}} \tag{52}$$

where:

$$\bar{G}_{\alpha\beta}^{k_{ij}} = \langle \alpha\beta | \mathcal{A}\big\{ \Theta_{k_{ij}}(12)\, G_{k_{ij}}(12) \big\} \rangle. \tag{53}$$

$\bar{G}_{\alpha\beta}^{k_{ij}}$ is the two-electron overlap between two virtual orbitals and a geminal. Note that the projection operator $Q_{\mathrm{occ}}(12)$ is not needed in (53) by virtue of $\langle \alpha\beta |$. The CCD energy and amplitude equations can be written schematically as[86]:

$$E = \tfrac{1}{2} A_{ij}^{k_{ij}} c_{k_{ij}}^{ij}, \tag{54}$$

$$A_{k_{ij}}^{ij} + (L_1)_{k_{ij}}^{ij} + (L_2)_{k_{ij}}^{ij} + Q_{k_{ij}}^{ij} = 0, \tag{55}$$

where A, L, and Q represent the absolute, linear, and quadratic terms, respectively. The absolute term takes the following form:

$$A_{k_{ij}}^{ij} = \langle \left(\frac{\partial S\Phi}{\partial c_{k_{ij}}^{ij}} \right) | W_N | \Phi \rangle = \tfrac{1}{2} \bar{G}_{k_{ij}}^{\alpha\beta} \bar{g}_{\alpha\beta}^{ij} = \langle \mathcal{A}\big\{ \Theta_{k_{ij}}(12)\, G_{k_{ij}}(12) \big\} | Q_{\mathrm{occ}}(12) \frac{1}{r_{12}} | ij \rangle. \tag{56}$$

This term contains three-electron integrals due to the strong-orthogonality projector $Q_{\mathrm{occ}}(12)$, for example:

$$\langle G_{k_{ij}}(12) || \varphi_m(1) \rangle \langle \varphi_m(1) | \frac{1}{r_{12}} | \varphi_i(1) \varphi_j(2) \rangle = \langle G_{k_{ij}}(12) \varphi_m(3) | \frac{1}{r_{32}} | \varphi_m(1) \varphi_j(2) \varphi_i(3) \rangle. \tag{57}$$

Using the two-index antisymmetrizer $\mathcal{A}_{ij} = 1 - (i \leftrightarrow j)$, the first of the two linear terms is:

$$\begin{aligned} (L_1)_{k_{ij}}^{ij} &= \langle \left(\frac{\partial S\Phi}{\partial c_{k_{ij}}^{ij}} \right) | [F_N, S] | \Phi \rangle \\ &= \tfrac{1}{2} \bar{G}_{k_{ij}}^{\alpha\beta} \left(\mathcal{A}_{\alpha\beta} f_\alpha^\gamma \bar{G}_{\gamma\beta}^{l_{ij}} c_{l_{ij}}^{ij} - \mathcal{A}_{ij} f_k^i \bar{G}_{\alpha\beta}^{l_{kj}} c_{l_{kj}}^{kj} \right). \end{aligned} \tag{58}$$

The term involving the f_k^i elements of the Fock matrix is related to the overlap matrix of the geminals and involves three-electron integrals. The other term, however, is more complicated. After realizing that $f_\alpha^\gamma = h_\alpha^\gamma + \bar{g}_{\alpha i}^{\gamma i}$ it becomes clear that it contains four-electron integrals. Note that this term also occurs in MP2 theory. The second linear term is:

$$\begin{aligned} (L_2)_{k_{ij}}^{ij} &= \langle \left(\frac{\partial S\Phi}{\partial c_{k_{ij}}^{ij}} \right) | [W_N, S] | \Phi \rangle \\ &= \tfrac{1}{2} \bar{G}_{k_{ij}}^{\alpha\beta} \left(\tfrac{1}{2} \bar{g}_{\alpha\beta}^{\gamma\delta} \bar{G}_{\gamma\delta}^{l_{ij}} c_{l_{ij}}^{ij} + \tfrac{1}{2} \bar{g}_{kl}^{ij} \bar{G}_{\alpha\beta}^{l_{kl}} c_{l_{kl}}^{kl} + \mathcal{A}_{\alpha\beta} \mathcal{A}_{ij} \bar{g}_{\alpha k}^{i\gamma} \bar{G}_{\beta\gamma}^{l_{jk}} c_{l_{jk}}^{jk} \right). \end{aligned} \tag{59}$$

The three distinct terms (or diagrams) contain four-electron, three-electron, and four-electron integrals, in the order of appearance. Finally, the quadratic term is:

$$\begin{aligned} Q^{ij}_{k_{ij}} &= \tfrac{1}{2}\langle\left(\frac{\partial S\Phi}{\partial c^{ij}_{k_{ij}}}\right)|[[W_N,S],S]|\Phi\rangle \\ &= \tfrac{1}{2}\bar{G}^{\alpha\beta}_{k_{ij}}\Big(\mathcal{A}_{\alpha\beta}\bar{G}^{l_{ik}}_{\alpha\gamma}c^{ik}_{l_{ik}}\bar{G}^{l_{jl}}_{\beta\delta}c^{jl}_{l_{jl}} + \tfrac{1}{2}\mathcal{A}_{\alpha\beta}\bar{G}^{l_{ij}}_{\alpha\delta}c^{ij}_{l_{ij}}\bar{G}^{l_{kl}}_{\beta\gamma}c^{kl}_{l_{kl}} \\ &+ \tfrac{1}{2}\mathcal{A}_{ij}\bar{G}^{l_{il}}_{\alpha\beta}c^{il}_{l_{il}}\bar{G}^{l_{jk}}_{\gamma\delta}c^{jk}_{l_{jk}} + \tfrac{1}{4}\bar{G}^{l_{ij}}_{\gamma\delta}c^{ij}_{l_{ij}}\bar{G}^{l_{kl}}_{\alpha\beta}c^{kl}_{l_{kl}}\Big)\bar{g}^{\gamma\delta}_{kl}. \end{aligned} \tag{60}$$

The last two terms in (60) factorize into products of three-electron integrals, but the first two terms involve five-electron integrals. However, these terms are ignored in the FCCD approximation (the whole quadratic term is, of course, absent in the LCCD approach). For not too large systems, it is believed that this nonfactorizable term is small and can be neglected.

Clearly, the computation of four- and five-electron integrals is a serious obstacle for large-scale applications on molecules of the geminals-based CCD approach.

Szalewicz and co-workers[72−75] have developed alternative approaches that involve only three-electron integrals at the MP2, MP3, LCCD, and FCCD levels (and four-electron integrals at the complete CCD level) by introducing new computational techniques. The techniques have been designated as weak orthogonality (WO), super weak orthogonality (SWO), and SWO plus projection (SWOP), and the techniques are sketched in the following section.

8 Weak orthogonality techniques

In order to illustrate the simplifications introduced by Szalewicz *et al.*[72−75] it is convenient to define an iterative CCD procedure by rewriting (55) as:

$$-(L_1)^{ij}_{k_{ij}} = A^{ij}_{k_{ij}} + (L_2)^{ij}_{k_{ij}} + Q^{ij}_{k_{ij}} \equiv \tfrac{1}{2}\bar{G}^{\alpha\beta}_{k_{ij}}V^{ij}_{\alpha\beta} \tag{61}$$

The meaning of V is easily recognized in (56), (59), and (60). Using canonical Hartree-Fock orbitals, the iterative process takes the form:

$$-\tfrac{1}{2}\bar{G}^{\alpha\beta}_{k_{ij}}\Big(f^{\gamma}_{\alpha}\bar{G}^{l_{ij}}_{\gamma\beta} + f^{\gamma}_{\beta}\bar{G}^{l_{ij}}_{\alpha\gamma} - (\varepsilon_i+\varepsilon_j)\bar{G}^{l_{ij}}_{\alpha\beta}\Big)c^{ij\ [n+1]}_{l_{ij}} = \tfrac{1}{2}\bar{G}^{\alpha\beta}_{k_{ij}}V^{ij}_{\alpha\beta}(\mathbf{c}^{[n]}). \tag{62}$$

The first simplification (WO) is achieved by replacing the l.h.s of (62) by an analogous expression without strong-orthogonality projector, that is[72]:

$$-\tfrac{1}{2}\bar{G}^{\kappa\lambda}_{k_{ij}}\Big(\tilde{f}^{\mu}_{\kappa}\bar{G}^{l_{ij}}_{\mu\lambda} + \tilde{f}^{\mu}_{\lambda}\bar{G}^{l_{ij}}_{\kappa\mu} - (\varepsilon_i+\varepsilon_j)\bar{G}^{l_{ij}}_{\kappa\lambda}\Big)c^{ij\ [n+1]}_{l_{ij}} = \tfrac{1}{2}\bar{G}^{\alpha\beta}_{k_{ij}}V^{ij}_{\alpha\beta}(\mathbf{c}^{[n]}), \tag{63}$$

which eliminates the four-electron integrals in L_1. The operator $\tilde{f}$ is defined as:

$$\tilde{f}(1) = f(1) + \Delta_{ij}P_{\mathrm{occ}}(1), \tag{64}$$

with:

$$\Delta_{ij} = \tfrac{1}{2}(\varepsilon_i+\varepsilon_j) - \varepsilon_{\min} + \eta, \tag{65}$$

where $\varepsilon_{\min}$ is the lowest eigenvalue and η a positive parameter (note that the operator $\tilde{f}(1)$ depends on the electron pair ij). Some improved WO functionals are proposed and described in the more recent literature[76,82]. The second approximation (SWO) is concerned with the r.h.s. of 62, which is replaced by:

$$-\tfrac{1}{2}\bar{G}^{\kappa\lambda}_{k_{ij}}\left(\tilde{f}^{\mu}_{\kappa}\bar{G}^{l_{ij}}_{\mu\lambda}+\tilde{f}^{\mu}_{\lambda}\bar{G}^{l_{ij}}_{\kappa\mu}-(\varepsilon_i+\varepsilon_j)\bar{G}^{l_{ij}}_{\kappa\lambda}\right)c^{ij\ [n+1]}_{l_{ij}}$$
$$=\tfrac{1}{2}\left(\bar{G}^{\kappa\lambda}_{k_{ij}}V^{ij}_{\kappa\lambda}(\mathbf{c}^{[n]})-\bar{G}^{mn}_{k_{ij}}V^{ij}_{mn}(\mathbf{c}^{[n]})\right). \tag{66}$$

The third simplification (SWOP) consists of re-expanding after each CCD iteration the strongly orthogonalized pair functions in the corresponding basis of geminals by solving the set of linear equations[75]:

$$\bar{G}^{\kappa\lambda}_{m_{ij}}\bar{G}^{k_{ij}}_{\kappa\lambda}\,\tilde{c}^{ij\ [n]}_{k_{ij}}=G^{\alpha\beta}_{m_{ij}}\bar{G}^{l_{ij}}_{\alpha\beta}\,c^{ij\ [n]}_{l_{ij}}. \tag{67}$$

The amplitudes $\tilde{\mathbf{c}}^{[n]}$ are inserted into a modified iterative process:

$$-\tfrac{1}{2}\bar{G}^{\kappa\lambda}_{k_{ij}}\left(\tilde{f}^{\mu}_{\kappa}\bar{G}^{l_{ij}}_{\mu\lambda}+\tilde{f}^{\mu}_{\lambda}\bar{G}^{l_{ij}}_{\kappa\mu}-(\varepsilon_i+\varepsilon_j)\bar{G}^{l_{ij}}_{\kappa\lambda}\right)c^{ij\ [n+1]}_{l_{ij}}$$
$$=\tfrac{1}{2}\left(\bar{G}^{\kappa\lambda}_{k_{ij}}\tilde{V}^{ij}_{\kappa\lambda}(\tilde{\mathbf{c}}^{[n]})-\bar{G}^{mn}_{k_{ij}}\tilde{V}^{ij}_{mn}(\tilde{\mathbf{c}}^{[n]})\right). \tag{68}$$

Note that this equation contains $\tilde{V}$, which is analogous to V but with the strong-orthogonality projectors removed. In the second-quantization notation, this means that in $\tilde{V}$ all $\alpha,\beta,\ldots$ appearing in V are replaced by the indices $\kappa,\lambda,\ldots$ of the entire basis. New amplitudes $\mathbf{c}^{[n+1]}$ are obtained and subsequently re-expanded according to (67).

The SWOP-FCCD approach is an approximative CCD method that is drastically less demanding than the corresponding rigorous and complete treatment. To illustrate this, let the one-particle basis set has dimension M and the geminal basis dimension K (i.e., the same dimension for all pairs). By inspection of (60) it becomes clear that the complete CCD approach would require the computation of M^4K^3 five-electron integrals. The FCCD method would require M^4K^2 four-electron integrals in L_1 and L_2 if no weak orthogonality techniques were introduced and only M^2K^2 (and M^4K) three-electron integrals in the SWOP framework. Hence, MP2 calculations based on the WO functional involve $aM^2K^2+bM^4K$ three-electron integrals and so do the LCCD and FCCD methods based on the SWOP technique. A SWOP-FCCD computation is not substantially more time consuming than a third-order calculation.

The most elaborate CCD – and partly CCSD – calculations with Gaussian geminals have been performed on small atomic and diatomic systems such as Be, LiH, and Ne[76]. Unfortunately, the geminals-based coupled-cluster approach has not yet evolved to a widely applicable tool of numerical quantum chemistry, and not many applications are known. Recently, however, second-order results for H_2O have been reported[82].

It is stressed that the WO, SWO, and SWOP techniques are not approximations. These are different methods to enforce the strong orthogonality, capable of giving

the exact results provided that the employed geminal basis set is sufficiently large. If we aim at extremely accurate results, then these techniques are certainly helpful tools, but their rôle is not clear when we plan to use geminals to compute efficiently the bulk of dynamical correlation (of the order of 90%).

For more details on the geminals-based CCSD method the reader is referred to the very recent ans comprehensive overview presented by Bukowski *et al.*[85]

9 R12 methods

The R12 methods have first been proposed by Kutzelnigg in 1985[87]. They have been implemented at various levels of many-body perturbation theory and at the coupled-cluster level[88−93]. A detailed description of the coupled-cluster theory with linear R12 terms has been presented by Noga and Kutzelnigg[94], and we will therefore focus only on the main formulae in the present section. It is also noted that Gdanitz[95,96] has extended the application of R12 methods to the multireference case.

The CCD-R12 ansatz is:

$$T_2 = \tfrac{1}{8} c_{kl}^{ij} \bar{R}_{\alpha\beta}^{kl} a_{ij}^{\alpha\beta} + \tfrac{1}{4} t_{ab}^{ij} a_{ij}^{ab}, \tag{69}$$

with:

$$\bar{R}_{\mu\nu}^{kl} = \langle \mu\nu | Q(12)\, r_{12} \, |kl\rangle \;=\; \bar{r}_{\mu\nu}^{kl} - \delta_\nu^q \bar{r}_{\mu q}^{kl} - \delta_\mu^p \bar{r}_{p\nu}^{kl} + \tfrac{1}{2} \Delta_{\mu\nu}^{pq} \bar{r}_{pq}^{kl}, \tag{70}$$

where:

$$\Delta_{\kappa\lambda}^{\mu\nu} = \delta_\kappa^\mu \delta_\lambda^\nu - \delta_\lambda^\mu \delta_\kappa^\nu \tag{71}$$

and:

$$Q(12) = \{1 - P(1)\}\{1 - P(2)\}, \quad P = \sum_p |\varphi_p\rangle\langle\varphi_p|. \tag{72}$$

The use of the operator $\bar{R}$ implies that the r_{12}-dependent functions $\bar{R}_{\alpha\beta}^{kl} a_{ij}^{\alpha\beta} |\Phi\rangle$ are strongly orthogonal to all Slater determinants built from orbitals contained in the finite basis set. This is very convenient. In a first-quantization form[97], the ansatz (69–70) is:

$$T_2 = \sum_{\mu<\nu} t(\mu\nu), \tag{73}$$

$$t(12) = \sum_{i<j} \sum_{k<l} c_{kl}^{ij}\, Q(12)\, r_{12}\, |kl\rangle\langle ij| + \sum_{i<j} \sum_{a<b} t_{ab}^{ij}\, |ab\rangle\langle ij|. \tag{74}$$

There are two coupled sets of amplitude equations, one for the conventional amplitudes t:

$$A_{ab}^{ij} + (L_1)_{ab}^{ij} + (L_2)_{ab}^{ij} + Q_{ab}^{ij} = 0, \tag{75}$$

and one for the r_{12}-dependent amplitudes c:

$$A_{kl}^{ij} + (L_1)_{kl}^{ij} + (L_2)_{kl}^{ij} + Q_{kl}^{ij} = 0, \tag{76}$$

while the energy is given by:

$$E = \tfrac{1}{4}(A_{ij}^{kl} c_{kl}^{ij} + A_{ij}^{ab} t_{ab}^{ij}). \tag{77}$$

Here and in the following, it is understood that the A, L_1, L_2, and Q terms refer to the amplitude equations determining the r_{12}-dependent amplitudes c if all four sub- and superscripts are indices of occupied orbitals and to the equations determining the conventional amplitudes t otherwise.

The absolute terms take the form:

$$A_{ab}^{ij} = \langle \left(\frac{\partial S\Phi}{\partial t_{ab}^{ij}} \right) | W_N | \Phi \rangle = \bar{g}_{ab}^{ij}, \tag{78}$$

$$A_{kl}^{ij} = \langle \left(\frac{\partial S\Phi}{\partial c_{kl}^{ij}} \right) | W_N | \Phi \rangle = \tfrac{1}{2} \bar{R}_{kl}^{\alpha\beta} \bar{g}_{\alpha\beta}^{ij}. \tag{79}$$

Application of the standard approximation[70] to (79) gives:

$$\tfrac{1}{2} \bar{R}_{kl}^{\alpha\beta} \bar{g}_{\alpha\beta}^{ij} = \tfrac{1}{2} \left(\bar{r}_{kl}^{\kappa\lambda} \bar{g}_{\kappa\lambda}^{ij} - \bar{r}_{kl}^{p\lambda} \bar{g}_{p\lambda}^{ij} - \bar{r}_{kl}^{\kappa q} \bar{g}_{\kappa q}^{ij} + \bar{r}_{kl}^{pq} \bar{g}_{pq}^{ij} \right) \approx \Delta_{kl}^{ij} - \tfrac{1}{2} \bar{r}_{kl}^{pq} \bar{g}_{pq}^{ij}, \tag{80}$$

which only involves two-electron integrals over the operators r_{12} and r_{12}^{-1}.

The L_1 terms are

$$(L_1)_{ab}^{ij} = \langle \left(\frac{\partial S\Phi}{\partial t_{ab}^{ij}} \right) | [F_N, S] | \Phi \rangle = \mathcal{A}_{ab} f_a^c t_{cb}^{ij} - \mathcal{A}_{ij} f_k^i t_{ab}^{kj}, \tag{81}$$

$$\begin{aligned} (L_1)_{kl}^{ij} &= \langle \left(\frac{\partial S\Phi}{\partial c_{kl}^{ij}} \right) | [F_N, S] | \Phi \rangle \\ &= \tfrac{1}{2} \bar{R}_{kl}^{\alpha\beta} \left(\mathcal{A}_{\alpha\beta} f_\alpha^\gamma \bar{R}_{\gamma\beta}^{mn} c_{mn}^{ij} - \mathcal{A}_{ij} f_o^i \bar{R}_{\alpha\beta}^{mn} c_{mn}^{oj} \right). \end{aligned} \tag{82}$$

There is no coupling between the conventional and r_{12}-dependent L_1 terms by virtue of the strongly orthogonal $\bar{R}$. Since MP2-R12 theory is obtained by removing the L_2 and Q terms from (75) and (76), this level of theory is exactly decoupled[98], that is, the MP2-R12 energy is a sum of the conventional MP2 energy and a contribution from the r_{12}-dependent terms. Therefore, this r_{12}-dependent contribution has been denoted in previous work as "basis set incompleteness correction", as it represents a contribution that is computed independently and added to the conventional MP2 energy. The latter is computed as usual in the prescribed basis.

A coupling between the conventional and r_{12}-dependent amplitudes appears for the first time in the L_2 terms:

$$(L_2)_{ab}^{ij} = \tfrac{1}{2} \bar{g}_{kl}^{ij} t_{ab}^{kl} + \tfrac{1}{2} \bar{g}_{ab}^{cd} t_{cd}^{ij} + \mathcal{A}_{ij} \mathcal{A}_{ab} \bar{g}_{ak}^{ic} t_{bc}^{jk} + \tfrac{1}{4} \bar{g}_{ab}^{\gamma\delta} \bar{R}_{\gamma\delta}^{kl} c_{kl}^{ij}. \tag{83}$$

Introducing the amplitude-free intermediate:

$$\bar{V}_{ab}^{kl} = \tfrac{1}{2} \bar{g}_{ab}^{\gamma\delta} \bar{R}_{\gamma\delta}^{kl} = \tfrac{1}{2} \bar{g}_{ab}^{\kappa\lambda} \bar{R}_{\kappa\lambda}^{kl} \approx -\tfrac{1}{2} g_{ab}^{pq} \bar{r}_{pq}^{kl}, \tag{84}$$

this term can be evaluated in a straightforward manner as:

$$(L_2)_{ab}^{ij} = \tfrac{1}{2} \bar{g}_{kl}^{ij} t_{ab}^{kl} + \tfrac{1}{2} \bar{g}_{ab}^{cd} t_{cd}^{ij} + \mathcal{A}_{ij} \mathcal{A}_{ab} \bar{g}_{ic}^{ak} t_{jk}^{bc} + \tfrac{1}{2} \bar{V}_{ab}^{kl} c_{kl}^{ij}. \tag{85}$$

In terms of the effective interactions $\bar{P}$, $\bar{X}$, and $\bar{Z}$, the r_{12}-dependent L_2 amplitude equation can be written as:

$$(L_2)^{ij}_{kl} = \tfrac{1}{2}\Big\{(\bar{V}^\dagger)^{ab}_{kl}t^{ij}_{ab} + \bar{P}^{oo'}_{kl}c^{ij}_{oo'} + \tfrac{1}{2}\bar{X}^{oo'}_{kl}\bar{c}^{mn}_{oo'}\bar{g}^{ij}_{mn} - \mathcal{A}_{ij}\bar{Z}^{oo';i}_{kl;m}c^{mj}_{oo'}\Big\}. \tag{86}$$

The $\bar{X}$ and $\bar{P}$ intermediates are easily computed in the standard approximation:

$$\bar{X}^{ij}_{kl} \approx \overline{(r^2)}^{ij}_{kl} - \tfrac{1}{2}\bar{r}^{pq}_{kl}\bar{r}^{ij}_{pq}, \quad P^{oo'}_{kl} \approx \tfrac{1}{4}\bar{r}^{rs}_{kl}\bar{g}^{pq}_{rs}\bar{r}^{oo'}_{pq} - \bar{r}^{oo'}_{kl}. \tag{87}$$

The effective interaction $\bar{Z}^{oo';i}_{kl;m}$ is, however, more complicated. Its evaluation has been discussed in detail by Noga and Kutzelnigg[94] in the Appendix F of their paper. Intermediates of the type $\bar{Z}^{kl;i}_{kl;m}$ already appeared in the early MP3, CID, and LCCD calculations based on the former, non-orbital-invariant ansatz[99]. In these calculations, the evaluation of the $\bar{Z}$ term was performed according this Appendix F[94]:

$$\begin{aligned}\bar{Z}^{kl;q}_{ij;p} &= \bar{R}^{\alpha\beta}_{ij}\bar{g}^{q\gamma}_{p\beta}\bar{R}^{kl}_{\alpha\gamma} \approx \bar{R}^{\alpha\beta}_{ij}g^{q\gamma}_{p\beta}\bar{R}^{kl}_{\alpha\gamma} = \bar{R}^{\mu\nu}_{ij}g^{q\lambda}_{p\nu}\bar{R}^{kl}_{\mu\lambda}\\ &\approx \bar{R}^{\mu\nu}_{ij}g^{q\lambda}_{p\nu}\bar{r}^{kl}_{\mu\lambda} + \bar{r}^{\mu\nu}_{ij}g^{q\lambda}_{p\nu}\bar{R}^{kl}_{\mu\lambda} - \bar{r}^{\mu\nu}_{ij}g^{q\lambda}_{p\nu}\bar{r}^{kl}_{\mu\lambda} + \bar{r}^{rs}_{ij}g^{qt}_{ps}\bar{r}^{kl}_{rt}\\ &\approx \mathcal{A}_{ij}g^{qr}_{pi}\bar{Y}^{kl}_{rj} + \mathcal{A}_{kl}g^{qk}_{pr}\bar{Y}^{rl}_{ij} + \bar{r}^{rs}_{ij}g^{qt}_{ps}\bar{r}^{kl}_{rt},\end{aligned} \tag{88}$$

where:

$$\bar{Y}^{rs}_{pq} = \bar{X}^{rs}_{pq} - \tfrac{1}{2}\overline{(r^2)}^{rs}_{pq}. \tag{89}$$

The quadratic term to the conventional doubles amplitude equation is:

$$\begin{aligned}Q^{ij}_{ab} = &\Big[\mathcal{A}_{ab}(t^{ik}_{ac}t^{jl}_{bd} + \tfrac{1}{2}t^{ij}_{ad}t^{kl}_{bc}) + \tfrac{1}{2}\mathcal{A}_{ij}t^{il}_{ab}t^{jk}_{cd} + \tfrac{1}{4}t^{ij}_{cd}t^{kl}_{ab}\Big]\bar{g}^{cd}_{kl}\\ &+ \tfrac{1}{2}\mathcal{A}_{ij}t^{il}_{ab}\bar{V}^{mn}_{kl}c^{jk}_{mn} + \tfrac{1}{4}t^{kl}_{ab}\bar{V}^{mn}_{kl}c^{ij}_{mn},\end{aligned} \tag{90}$$

where:

$$\bar{V}^{mn}_{kl} = \tfrac{1}{2}\bar{g}^{\alpha\beta}_{kl}\bar{R}^{mn}_{\alpha\beta} \approx \Delta^{mn}_{kl} - \tfrac{1}{2}\bar{g}^{oo'}_{kl}\bar{r}^{mn}_{oo'}. \tag{91}$$

The quadratic term to the r_{12} amplitude equation is:

$$\begin{aligned}Q^{ij}_{kl} = &\tfrac{1}{8}\bar{R}^{\alpha\beta}_{kl}\mathcal{A}_{\alpha\beta}\Big(c^{in}_{mm'}\bar{R}^{mm'}_{\alpha\gamma}\bar{R}^{oo'}_{\beta\delta}c^{jn'}_{oo'} + \tfrac{1}{2}c^{ij}_{mm'}\bar{R}^{mm'}_{\alpha\delta}\bar{R}^{oo'}_{\beta\gamma}c^{nn'}_{oo'}\Big)\bar{g}^{\gamma\delta}_{nn'}\\ &+ \tfrac{1}{4}\mathcal{A}_{ij}\bar{X}^{oo'}_{kl}c^{im'}_{oo'}\Big(\bar{V}^{nn'}_{mm'}c^{jm}_{nn'} + \bar{g}^{cd}_{mm'}t^{jm}_{cd}\Big)\\ &+ \tfrac{1}{8}\bar{X}^{oo'}_{kl}c^{mm'}_{oo'}\Big(\bar{V}^{nn'}_{mm'}c^{ij}_{nn'} + \bar{g}^{cd}_{mm'}t^{ij}_{cd}\Big),\end{aligned} \tag{92}$$

which by virtue of the standard approximation, that is, by inserting the resolution of the identity, simplifies to:

$$\begin{aligned}Q^{ij}_{kl} \approx &\tfrac{1}{4}\mathcal{A}_{ij}\bar{X}^{oo'}_{kl}c^{im'}_{oo'}\Big(\bar{V}^{nn'}_{mm'}c^{jm}_{nn'} + \bar{g}^{cd}_{mm'}t^{jm}_{cd}\Big)\\ &+ \tfrac{1}{8}\bar{X}^{oo'}_{kl}c^{mm'}_{oo'}\Big(\bar{V}^{nn'}_{mm'}c^{ij}_{nn'} + \bar{g}^{cd}_{mm'}t^{ij}_{cd}\Big).\end{aligned} \tag{93}$$

The terms that vanish due to the standard approximation are the nonfactorizable Q-term diagrams that involve five-electron integrals. Hence, the FCCD-R12 and CCD-R12 methods do not differ in their r_{12} parts but only in the conventional nonfactorizable quadratic terms.

The reader is referred to recent reviews of R12 theory for further details[23,100].

9.1 Expansion of r_{12} in terms of Gaussian geminals

Persson and Taylor[101] have suggested to expand the linear r_{12} terms in a basis of Gaussian geminals. The linear r_{12} terms introduced in the preceding section can subsequently be replaced by this expansion, with the advantage that the many-electron integrals become tractable. Persson and Taylor studied fits of the type:

$$r_{12} \approx \sum_{v=1}^{N} b_v \left[1 - \exp(-\gamma_v r_{12}^2) \right] \equiv s(12), \tag{94}$$

where the notation $s(12)$ has been introduced to highlight the fact that the r.h.s of (94) is some particular multiplicative two-electron operator. Hence, the theory of explicitly correlated wavefunctions with terms linear in r_{12} can be used with no or minor modifications by replacing the ansatz (69) by:

$$T_2 = \tfrac{1}{8} c_{kl}^{ij} \bar{S}_{\alpha\beta}^{kl} a_{ij}^{\alpha\beta} + \tfrac{1}{4} t_{ab}^{ij} a_{ij}^{ab}, \tag{95}$$

where:

$$\bar{S}_{\mu\nu}^{kl} = \langle \mu\nu | Q(12)\, s(12)\, | kl \rangle. \tag{96}$$

The analogy with fitting Slater-type orbitals (STO) as expansions in Gaussian orbitals is obvious. As for the STO-fits, the main objective of the present approach is to simplify (or make possible) the evaluation of many-electron integrals. Persson and Taylor investigated even-tempered sets of Gaussian geminals of the type $\gamma_v = a_N\, 3^{v-1}$ and obtained very satisfying exponentionally weighted linear least-squares fits of r_{12} with six ($a_6 = 0.111111$) or nine terms ($a_9 = 0.037037$) by minimizing:

$$\chi^2 = \int_0^\infty e^{-2x} \left[x - \sum_{v=1}^{N} b_v (1 - e^{-\gamma_v x^2}) \right]^2 dx. \tag{97}$$

(Actually, Persson and Taylor did not integrate from 0 to ∞, but over a preselected range, for example from 0 to 2, 5, 10, or 15 a_0.) The corresponding fits are shown in Figure 7. Clearly, these expansions will approach a constant value for $r_{12} \to \infty$, but this does not matter since the purpose of the terms is to describe the short-range correlation effects for small r_{12}. For large interelectronic distances, a constant or linearly growing r_{12} term is in fact not desirable and in this respect the correlating functions $s(12)$ are not necessarily inferiour to the linear r_{12} terms. Although the expansions do not describe the cusp for $r_{12} = 0$ (which can not be seen in Figure 7 on the present scale), it has been observed that electron correlation effects are very effectively recovered by these functions[101]. The expansion in Gaussian geminals might perhaps not – or not so efficiently – speed up the convergence in the asymptotic region where the objective is the compute 99% of the correlation

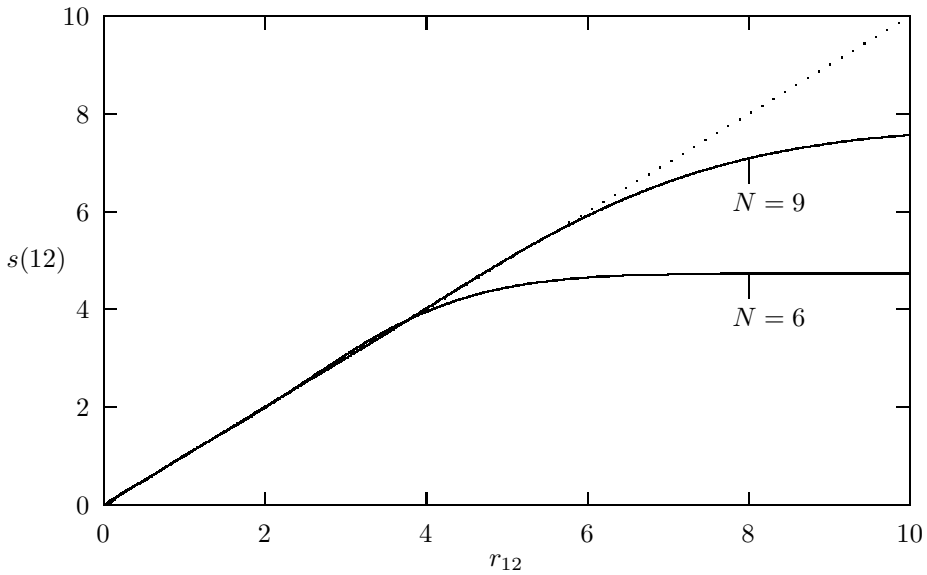


Figure 7. Fits of r_{12} as expansions in terms of six and nine even-tempered Gaussian geminals.

energy or more, but appears to recover a significant part of the correlation energy when used with small or medium-sized basis sets (i.e., basis sets that recover roughly 70–80% of the correlation energy).

Furthermore, it was realized that the even-tempered basis of Gaussian-type geminals could also be applied in the uncontracted form:

$$T_2 = \tfrac{1}{8} \sum_{v=1}^{N} (c_v)_{kl}^{ij} (\bar{S}_v)_{\alpha\beta}^{kl} a_{ij}^{\alpha\beta} + \tfrac{1}{4} t_{ab}^{ij} a_{ij}^{ab}, \tag{98}$$

where:

$$(\bar{S}_v)_{\mu\nu}^{kl} = \langle \mu\nu | Q(12) \, \exp(-\gamma_v r_{12}^2) \, | kl \rangle. \tag{99}$$

Some initial calculations were performed on He, and it was observed that the variationally determined linear parameters for the Gaussian geminals in (98) were quite similar to the coefficients obtained from the fitting procedure. Results for Ne and H_2O obtained from this (uncontracted) ansatz are presented in Table 6. For the practical calculations in Table 6, the sum over ij and kl in (98) was restricted to the diagonal terms where $ij = kl$ while the occupied orbitals in the integral (99) were expressed in terms of a very small AO basis set (e.g., by one single AO of the total set). This was done for technical reasons. The results nevertheless show that augmentation of standard basis sets with a few explicitly correlated Gaussian geminals improves the percentage of the correlation energy recovered by the basis set from about 73–85 % to ca. 92–98 %.

If the approximations of the linear R12 theory would be exploited in the Persson-Taylor geminals approach (which was not done in their work), not much would

Table 6. MP2(FC) correlation energy (in mE_h) for Ne and H_2O obtained by Persson and Taylor[101] from augmentation of standard orbital basis sets with six Gaussian-type geminals.

		Orbital basis		+ 6 geminals	
	Basis[a]	E_{corr}	%[b]	E_{corr}	%[c]
Ne	cc-pVTZ(uc)	−273.1	85.3	−314.0	98.1
H_2O	cc-pVDZ	−219.8	73.1	−277.0	92.1

a Using Cartesian Gaussians.
(uc) denotes a fully uncontracted basis set.
b Percentage of the limiting value of −320.2 mE_h.[20,102]
c Percentage of the limiting value of −300.5 mE_h.[20]

be gained. The many-electron integrals of the linear R12 methods are only of the two-electron type, and very easy to compute. However, it will certainly be interesting to further investigate the expansion of r_{12} in terms of geminals if a method is developed where all (or some important) many-electron integrals are computed rigorously. One advantage of this method is that it does not require the optimization of nonlinear parameters.

10 Explicitly correlated Gaussians

Cencek and Rychlewski[103−107] have developed a variational method where the n-electron wavefunction is expanded in a basis $\{\Phi_k\}$ of the form:

$$\Phi_k(12\ldots n) = \mathcal{A}\Big\{\Theta(\sigma_1\sigma_n\ldots\sigma_n)\mathcal{P}_R\Big[G_k^0(12)\prod_{i>2}^{n} g_k^0(i)\Big]\Big\}, \tag{100}$$

where $\mathcal{A}$ is the usual antisymmetrizer, Θ a proper spin function, and $\mathcal{P}_R$ the symmetry projector onto the irreducible representation R of the molecular point group taking care of the symmetry adaptation. The important point to note is that the n-electron basis functions contain only one geminal:

$$G_k^0(12) \equiv G^0_{\alpha_k,\beta_k,\bar\beta_k,\mathbf{B}_k,\bar{\mathbf{B}}_k}(12) = \exp(-\alpha_k r_{12}^2 - \beta_k r_{1B_k}^2 - \bar\beta_k r_{2\bar B_k}^2), \tag{101}$$

where r_{12} is the interelectronic distance, r_{1B_k} is the distance of electron 1 to center $\mathbf{B}_k$, $r_{2\bar B_k}$ is the distance of electron 2 to center $\bar{\mathbf{B}}_k$, and α_k, β_k, and $\bar\beta_k$ are Gaussian exponents (which to some extent are allowed to be negative as long as the function is square integrable). Thus, one Gaussian geminal contains 9 nonlinear variational parameters (3 exponents and 6 coordinates).

By the choice (100), the complexity of the many-electron integrals for any n-electron system is limited to at most four-electron integrals, which can be evaluated analytically[103].

The nonlinear parameters are considered variational parameters in each individual Φ_k and are optimized on the fly. This means that, for a general n-electron molecular system, an expansion in terms of N basis functions of the type (100) contains $N \times (4n + 1)$ nonlinear parameters.

For the systems studied so far, containing up to four electrons, the Cencek-Rychlewski method has provided variational energies for molecules that are superior to all other variational calculations, including the Kołos-Wolniewicz-type calculations for H_2. This success can be attributed to the rigorous and efficient optimization of the nonlinear parameters. Note that the Gaussian centers are not restricted to the positions of the nuclei, but are completely free to float.

Equation (100) is restricted to Gaussian functions with only one single geminal depending on only one single interparticle distance. In later calculations by the authors of the ECG method, however, explicitly correlated Gaussians have been introduced that depend on the interparticle distances of all electrons in the system. For example, spatial basis functions of the form:

$$\Phi_k(1\ldots4) = \exp\left(-\sum_{i=1}^{4}\alpha_{i,k}|\mathbf{r}_i - \mathbf{C}_{i,k}|^2 - \sum_{i=1}^{3}\sum_{j=i+1}^{4}\beta_{ij,k}r_{ij}^2\right) \tag{102}$$

have been employed in calculations on four-electron systems[106,107].

11 Similarity transformed Hamiltonians

In about 1969 the "transcorrelated method" of Boys and Handy[108−116] appeared to be an interesting alternative to the standard methods of numerical quantum chemistry employing explicitly correlated functions. Boys and Handy proposed to consider the non-Hermitian Hamiltonian:

$$\tilde{H}_C = e^{-C}He^{C}, \quad \text{with} \quad C = \sum_{i<j} f(\mathbf{r}_i, \mathbf{r}_j). \tag{103}$$

This similarity transformed Hamiltonian only contains three-electron operators, since the Hausdorff expansion

$$\tilde{H}_C = H + [H, C] + \frac{1}{2}[[H, C], C] + \ldots \tag{104}$$

breaks off after the double commutator. Introducing the short-hand notation $f_{ij} \equiv f(\mathbf{r}_i, \mathbf{r}_j)$ and assuming that the correlation factor C is symmetrical ($f_{ij} = f_{ji}$), we find that:

$$[H, C] = -\frac{1}{2}\sum_i\sum_{j\neq i}(\Delta_i f_{ij} + 2\nabla_i f_{ij}\cdot\nabla_i), \tag{105}$$

$$\frac{1}{2}[[H, C], C] = -\frac{1}{2}\sum_i\sum_{j\neq i}\sum_{k\neq i}\nabla_i f_{ij}\cdot\nabla_i f_{ik}. \tag{106}$$

The double commutator is a three-electron operator that commutes with C.

As an example of the similarity transformed Hamiltonian, consider the correlating function $C = f(\mathbf{r}_1, \mathbf{r}_2) = \frac{1}{2}r_{12}$ for the He atom. With this C, it follows that

$$[H, C] = -\frac{1}{r_{12}} - \frac{1}{2}\frac{\mathbf{r}_{12}}{r_{12}}\cdot(\nabla_1 - \nabla_2), \quad \frac{1}{2}[[H, C], C] = -\frac{1}{4}, \tag{107}$$

Table 7. He ground state energies from the similarity transformed Hamiltonian.

	$\exp(-\frac{1}{2}r_{12})H\exp(\frac{1}{2}r_{12})$		Configuration Interaction	
Basis	E/E_h	$\delta E^a/\mathrm{m}E_\mathrm{h}$	E/E_h	$\delta E^a/\mathrm{m}E_\mathrm{h}$
11s	−3.010 759 3	−107.035	−2.879 009 2	24.715
11s8p	−2.904 819 1	−1.095	−2.900 486 5	3.238
11s8p7d	−2.904 048 7	−0.324	−2.902 725 3	0.999
11s8p7d6f	−2.903 793 4	−0.069	−2.903 262 1	0.462
11s8p7d6f5g	−2.903 741 1	−0.017	−2.903 427 2	0.297
11s8p7d6f5g4h	−2.903 729 6	−0.005	−2.903 468 2	0.256

a Error with respect to the exact value of $E = -2.903\,724\,377\ E_\mathrm{h}$.[117]

and the transformed Hamiltonian becomes

$$\tilde{H}_C = h_1 + h_2 - \frac{1}{2}\frac{\mathbf{r}_{12}}{r_{12}}\cdot(\nabla_1 - \nabla_2) - \frac{1}{4}. \tag{108}$$

Since the partial-wave expansion of the operator $U_{12} = -\frac{1}{2}\frac{\mathbf{r}_{12}}{r_{12}}\cdot(\nabla_1 - \nabla_2)$ converges rapidly[87], good convergence of the computed energy with respect to the maximum angular momentum quantum number L_max contained in the basis can be expected. Some preliminary results from the ansatz

$$\Psi = \sum_{pq} c_{pq}\Phi_{pq}, \quad \Phi_{pq} = |\varphi_p(\mathbf{r}_1)\varphi_q(\mathbf{r}_2)|, \tag{109}$$

$$\langle\frac{\partial\Psi}{\partial c_{pq}}|\tilde{H}_C - E|\Psi\rangle = 0, \quad E = \langle\Psi|\tilde{H}_C|\Psi\rangle, \tag{110}$$

are presented in Table 7. The orbitals φ_p were expanded in a Gaussian basis of the type 11s8p7d6f5g4h, which was derived from the 11s8p6d5f4g3h basis used in earlier work on the He dimer[20]. (This 11s8p6d5f4g3h set was augmented with sets of d-, f-, g-, and h-type functions with exponents 22.841124, 10.566783, 4.95893, and 2.39464, respectively.) The results of Table 7 look promising, and it seems worthwhile to reinvestigate the Boys-Handy method (or other explicitly correlated similarity transformed effective Hamiltonians) in the framework of modern coupled-cluster theory[118]. Note that the correlation function $\exp(C)$ was only multiplied with one single determinant in the original Boys-Handy method.

12 MP2-limit corrections

MP2 calculations are computationally less demanding than high-order electron-correlation treatments such as MP4(SDTQ) perturbation theory or the CCSD(T) approach. Therefore, it will often be easier to determine the basis-set error at the MP2 level than at some higher level, and we may ask ourselves whether the MP2 basis-set error could be transferred in an easy way to that higher level.

Suppose we have available the correlation energy or correlation contribution to some molecular property at a given method/basis level of theory, which we denote

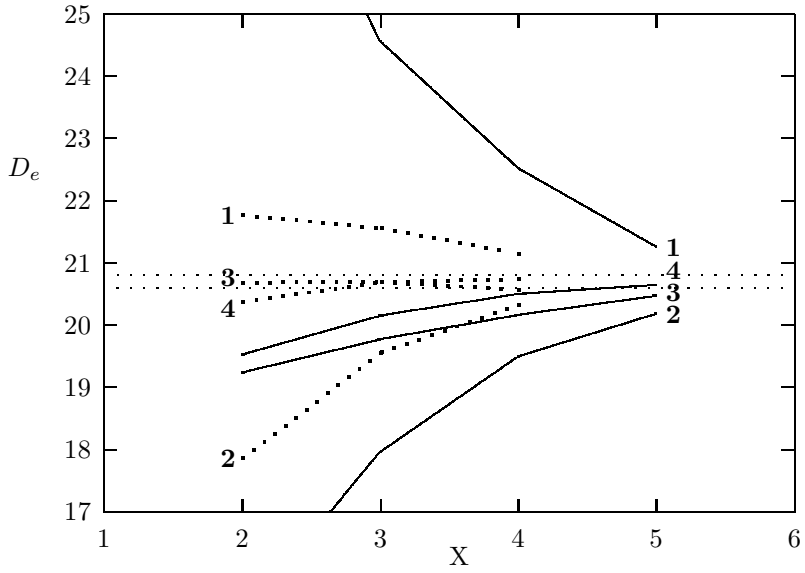


Figure 8. Equilibrium dissociation energy (D_e in kJ mol^{-1}) of the H_2O dimer in the fixed Halkier geometry[41]. The (valence only) unscaled MP2-limit correction is applied to CCSD(T)(FC)/cc-pVXZ (solid lines) and CCSD(T)(FC)/aug-cc-pVXZ (dashed lines) results. Meaning of the curves: **1** = no-CP, no-MP2-limit; **2** = CP, no-MP2-limit; **3** = no-CP, MP2-limit; **4** = CP, MP2-limit.

as Δ(method/basis), and also the corresponding correlation part at the MP2 level in a (nearly) complete basis, denoted as Δ(MP2/∞). Then, there are two evident choices of how to improve the computed Δ(method/basis) data with the known MP2-limits:

$$\Delta(\text{method}/\infty) \approx \Delta(\text{method/basis}) - \Delta(\text{MP2/basis}) + \Delta(\text{MP2}/\infty), \quad (111)$$

$$\Delta(\text{method}/\infty) \approx \Delta(\text{method/basis}) \times \frac{\Delta(\text{MP2}/\infty)}{\Delta(\text{MP2/basis})}, \quad (112)$$

that is, we can add the Δ(MP2/basis) to Δ(MP2/∞) increment to the high-order results or we can scale the high-order results with the Δ(MP2/basis) to Δ(MP2/∞) ratio.

When the finite basis set is enlarged, the increment (111) will eventually vanish and the ratio (112) will converge to unity. Thus, in both cases, it is guaranteed that the improved – that is, the MP2-limit corrected – results will converge to the true limiting values of the high-order approach.

Figure 8 shows the application of (111) with regard to the interaction energy between two H_2O molecules[43]. The correction has been applied to two series of calculations, namely, calculations with the cc-pVXZ sets and with the aug-cc-pVXZ sets, either with (CP) or without (no-CP) counterpoise correction. It is obvious that the convergence to the limiting value (20.7$\pm$0.1 kJ mol^{-1} in this case) is significantly accelerated by the add-on MP2-limit correction (111). The direct addition of the MP2-limit correction has been applied on several occasions, for example in order

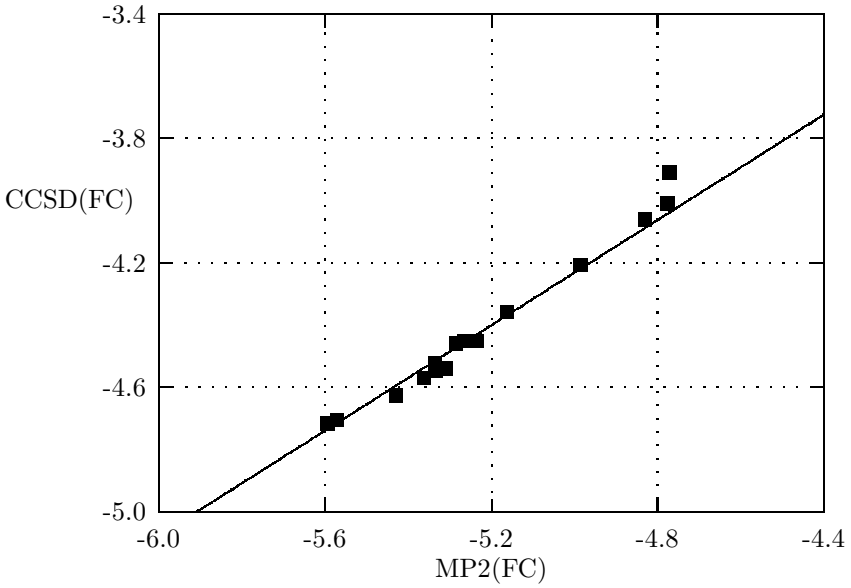


Figure 9. CCSD(FC) correlation contribution versus MP2(FC) correlation contribution to the interaction energy (in kJ mol^{-1}) of the H_2O dimer in the fixed Halkier geometry[41], as obtained from a variety of different basis sets[125].

to obtain accurate results for the ferrocene molecule, the weak benzene$\cdots$neon and benzene$\cdots$argon interactions, and the CO_2 trimer[119−121].

One can argue, however, that the MP2-correlation treatment often overestimates the all-order correlation effects and that, accordingly, the MP2-limit correction for the basis-set error is likely to overestimate the true basis-set error of the high-level method. We have already seen in Section 3.3 that the CBS method utilizes an interference factor to scale down the MP2 basis-set error that was obtained by extrapolation. Also concerning this issue, Van de Bovenkamp and Van Duijneveldt[122] argue that the binding energy of the He dimer as computed by Korona *et al.*[123] (11.06 K) is larger than what is regarded as the limiting value (11.01 K) because Korona *et al.* are believed to have incorrectly combined low-order Gaussian-geminals results with high-order orbital approximations, without taking into account that the low-order Gaussian-geminals corrections give too large contributions.

In view of these difficulties, Martin has proposed to scale the MP2-limit correction Q as follows[124]:

$$\Delta(\text{method}/\infty) \approx \Delta(\text{method/basis}) + Q \times \{\Delta(\text{MP2}/\infty) - \Delta(\text{MP2/basis})\}\,, \quad (113)$$

where Q is computed from the correlation-energy increments from two basis sets at both the MP2 level and the high-level method of interest:

$$Q = \frac{\Delta(\text{method/basis} - \Delta(\text{method/basis}')}{\Delta(\text{MP2/basis} - \Delta(\text{MP2/basis}')}, \quad \text{basis}' < \text{basis}. \quad (114)$$

Indeed, it seems a reasonable assumption that the ratio of these increments is

nearly constant for many basis sets beyond some minimum level of accuracy. This is illustrated in Figure 9 for calculations on the H_2O dimer, where CCSD and MP2 correlation contributions to the interaction energy are compared. These correlation contributions have been obtained with various – but all state-of-the-art – basis sets[125]. A nearly linear dependence is found with Q in (113) being the slope of the straight line in Figure 9. Of course, if the ratio of these increments is indeed constant, we can also use[125]:

$$Q = \frac{\Delta(\text{method/basis})}{\Delta(\text{MP2/basis})}. \tag{115}$$

Concerning the H_2O dimer, we can estimate from the results displayed in Figure 9 that the limiting CCSD(FC) correlation contribution is likely to amount to -4.89 kJ mol^{-1} as the valence-only MP2-limit contribution is -5.78 kJ mol^{-1} (cf. Table 4). Adding the SCF (-14.78 kJ mol^{-1}) and (T) triples corrections (-1.05 kJ mol^{-1}) then yields a total CCSD(T)(FC) interaction enery of -20.72 kJ mol^{-1} for the fixed Halkier geometry. Core-correlation effects add -0.14 kJ mol^{-1} and geometry-relaxation effects add -0.12 kJ mol^{-1}. Hence, the total equilibrium binding energy can safely be said to be established to $D_e = 21.0 \pm 0.1$ kJ mol^{-1}.[125]

13 Computational aspects of R12 methods

13.1 Integral-direct implementation

The explicitly correlated methods that rigorously evaluate the many-electron integrals use Gaussian geminals or explicitly correlated Gaussians. The three- and four-electron integrals appearing in the variational method of Cencek and Rychlewski can be computed in closed form[103], while the five-electron integrals appearing in the geminals-based coupled-cluster method are avoided by means of ignoring the corresponding non-factorizable diagrams (FCCD approximation). On the whole, not much is known about the efficient generation of three- and four-electron integrals over Gaussian geminals, but some aspects are discussed by Persson and Taylor[126].

The many-electron integrals over linear r_{12}-dependent functions are more complicated than the integrals over Gaussian geminals, and unfortunately, optimized strategies or algorithms for their computation have not been developed. However, the R12 methods avoid the corresponding three-, four-, and five-electron integrals by virtue of the standard approximations. What remains in the R12 methods are two-electron integrals, some of them special integrals appearing only in the R12 methods.

The algorithms used for the R12 computations have been tailored towards large-scale applications, that is, calculations on molecules with high-quality AO basis sets.

Since the introduction of the integral-direct implementation of the Hartree-Fock method by Almlöf *et al.*[127] in 1982, the integral-direct (or semi-direct) techniques have been refined further and have also been applied to the CCSD level[128–130]. Calculations with more than 500 basis functions are possible at this level today.

The current CCSDT-R12 computer program also employs integral-direct algorithms. In this section, the computational aspects of the implementation that are specific to the CC-R12 approach will be discussed in detail. The notation of Noga and Kutzelnigg[94] is adopted and it is referred to this paper for definitions of the intermediates, integrals, and so forth.

This section will be concerned with the following three R12 related computational tasks: First, the R12 part of the energy, second, the coupling of the R12 functions to the T_1 and T_2 equations, and third, the equation which determines the amplitudes of the r_{12}-type double excitations.

Furthermore, it will be assumed that quantities of the order of N^2 or possibly n^2N^2, where n is the number of occupied orbitals and N is the total number of orbitals, can be kept in the main memory of the computer and can be computed by integral-direct schemes that are analogous to the integral-direct MP2 algorithm (i.e., by an integral-direct partial four-index transformation).

If triple excitations are involved in the method, not only the n^2N^2 quantities, but also the two-electron integrals of the type g_{iq}^{rs} ($\propto nN^3$) are generated by integral-direct techniques. However, these nN^3 quantities are stored on disk.

The R12 contribution to the total CCSD-R12 correlation energy is represented by the diagram on Figure 2d of the paper by Noga and Kutzelnigg[94]. This diagram is evaluated as:

$$E = \tfrac{1}{4} A_{ij}^{kl} c_{kl}^{ij}, \tag{116}$$

where c_{kl}^{ij} are the amplitudes of the r_{12} double replacements, and:

$$A_{kl}^{ij} \approx \Delta_{kl}^{ij} - \frac{1}{2} \bar{r}_{kl}^{pq} \bar{g}_{pq}^{ij}. \tag{117}$$

Except for the numerical value of the amplitudes, this energy expression is identical with the R12 part of the MP2-R12 energy that has been implemented in the integral-direct `SORE` program. It is obvious that the n^2N^2 quantities $\bar{g}_{kl}^{pq}$ and $\bar{r}_{pq}^{ij}$ can be generated by the integral-direct partial four-index transformation, and that the n^4 intermediates A_{kl}^{ij} can be computed using an in-core algorithm.

Whereas only these n^4 intermediates occur in the MP2-R12 method, the higher-order methods also require (although not for the energy) the n^2N^2 intermediates:

$$\bar{V}_{rs}^{ij} = \frac{1}{2} \bar{g}_{rs}^{\alpha\beta} \bar{R}_{\alpha\beta}^{ij} \approx \Delta_{rs}^{ij} - \frac{1}{2} \bar{g}_{rs}^{pq} \bar{r}_{pq}^{ij}. \tag{118}$$

These intermediates, which are computed only once and kept in main memory, are evaluated as follows: In a first step, the $\bar{r}_{pq}^{ij}$ integrals are generated by the integral-direct partial four-index transformation. In a second step, the product $\bar{g}_{rs}^{pq} \bar{r}_{pq}^{ij}$ is computed in full analogy to the "B-term" of conventional integral-direct CC implementations[128,129]:

$$\Omega_{aibj}^{B} = t_{ij}^{cd} \bar{g}_{cd}^{ab}. \tag{119}$$

It is noticed that the integrals $\bar{r}_{pq}^{ij}$ play the role of the amplitudes t_{ij}^{cd} that occur in this B-term.

The new T_1 and T_2 equations of the CCSD-R12 method are easily obtained by modifying the established T_1 and T_2 equations of the traditional CC method

as well as the well-known intermediates F_{mi} and W_{mnij} (for a definition of these intermediates, see for example Equations (4) and (6) in the paper by Stanton *et al.*[131]. Four minor modifications are required. One must add:

$$-\frac{1}{2}\sum_{kl}\sum_{m} c_{kl}^{im}\bar{V}_{ma}^{kl} \qquad \text{to r.h.s. of } T_1 \text{ eq.,} \tag{120}$$

$$\frac{1}{2}\sum_{kl} c_{kl}^{ij}\left(\bar{V}_{ab}^{kl} - \sum_{m} t_b^m \bar{V}_{am}^{kl} + \sum_{m} t_a^m \bar{V}_{bm}^{kl}\right) \qquad \text{to r.h.s. of } T_2 \text{ eq.,} \tag{121}$$

$$\frac{1}{2}\sum_{kl}\sum_{n} c_{kl}^{in}\bar{V}_{mn}^{kl} \qquad \text{to } F_{mi}, \tag{122}$$

$$\frac{1}{2}\sum_{kl} c_{kl}^{ij}\bar{V}_{mn}^{kl} \qquad \text{to } W_{mnij}. \tag{123}$$

The important observation is that only the n^2N^2 quantities $\bar{V}_{rs}^{ij}$ are needed for these modifications. As discussed before, the $\bar{V}_{rs}^{ij}$ intermediates are computed employing integral-direct schemes and stored for later use in the subsequent CCSD-R12 iterations.

For completeness, it is noted that Equation (120) corresponds to diagram d3.15 of the paper by Noga and Kutzelnigg[94], whereas Equation (121) is the sum of the diagrams d4.1 and d4.2. Diagrams d3.16 and d4.5 result from the modified F_{mi} intermediate of Equation (122), and diagrams d4.3 and d4.4 can be traced back to the new W_{mnij} intermediate of Equation (123).

The equation determining the amplitudes of the r_{12} double excitations is given by Equation (III.4) in Table III of the paper by Noga and Kutzelnigg[94]:

$$\begin{aligned}
-\frac{1}{2}\bar{B}_{kl}^{mn}c_{mn}^{ij} = &-\left(\bar{F}_m^j\tilde{\bar{X}}_{kl}^{im} + \bar{F}_m^i\tilde{\bar{X}}_{kl}^{mj}\right) + (\bar{V}^\dagger)_{kl}^{ij} + \frac{1}{2}\bar{G}_{mn}^{ij}\tilde{\bar{X}}_{kl}^{mn} + \frac{1}{2}\bar{P}_{kl}^{mn}c_{mn}^{ij} \\
&+ \frac{1}{2}(\bar{V}^\dagger)_{kl}^{ab}\tau_{ab}^{ij} \; - \frac{1}{2}\left(\bar{Z}_{kl;o}^{mn;j}c_{mn}^{io} + \bar{Z}_{kl;o}^{mn;i}c_{mn}^{oj}\right) \\
&- \frac{1}{2}\left(\bar{Z}_{kl;o}^{mn;a}c_{mn}^{io}t_a^j + \bar{Z}_{kl;o}^{mn;a}c_{mn}^{oj}t_a^i\right) \; + \frac{1}{2}\bar{Z}_{kl;o}^{mn;a}c_{mn}^{ij}t_a^o.
\end{aligned} \tag{124}$$

Inspection of the auxiliary quantities[94] $\bar{F}$, $\tilde{\bar{X}}$, $\bar{V}$, $\bar{G}$, $\bar{P}$, and $\bar{Z}$ reveals that these intermediates only depend on integrals and/or other auxiliary quantities that carry *at least two* occupied indices. Therefore, the constructon and solution of (124) involves only in-core procedures. Recall that n^2N^2 quantities are kept or handled in main memory. For example, consider:

$$\bar{P}_{kl}^{mn} = \frac{1}{4}\bar{R}_{kl}^{\alpha\beta}\bar{g}_{\alpha\beta}^{\gamma\delta}\bar{R}_{\gamma\delta}^{mn} \approx -\frac{1}{2}\bar{r}_{kl}^{pq}\bar{V}_{pq}^{mn}. \tag{125}$$

The computation of $\bar{P}_{kl}^{mn}$ is straightforward as soon as the integrals $\bar{r}_{kl}^{pq}$ and $\bar{V}_{pq}^{mn}$ are available. The only exception is the computation of:

$$\bar{X}_{rs}^{ij} = \frac{1}{2}\bar{R}_{rs}^{\alpha\beta}\bar{R}_{\alpha\beta}^{ij} \approx (\bar{r^2})_{rs}^{ij} - \frac{1}{2}\bar{r}_{rs}^{pq}\bar{r}_{pq}^{ij}, \tag{126}$$

which requires a B-term-like procedure analogous to Equation (118). It is noted that the integral $(\bar{r^2})_{rs}^{ij}$ factorizes into products of one-electron integrals.

Furthermore, a direct MP2-like partial four-index transformation must be performed for the two-electron integrals $\bar{t}_{pq}^{ij}$ over the operators $[T(1), r_{12}]$ and $[T(2), r_{12}]$. These integrals also occur in the direct MP2-R12 program `SORE`. Notice that the `SORE` program is based on standard approximation A, which does not involve the quantities $\bar{X}_{rs}^{ij}$, whereas these quantities are required by the standard approximation B based MP2-R12/B method.

The integral-direct procedure to generate the R12 specific auxiliary quantities that are required by the CCSD-R12 method can be summarized as follows: In a first pass through the two-electron integral generator, the integral-direct partial four-index transformation is processed to generate the integrals $\bar{t}_{pq}^{ij}$, $\bar{r}_{pq}^{ij}$, and $\bar{g}_{pq}^{ij}$. This step corresponds to the MP2-R12/A method as implemented in the `SORE` program. In a second pass, the integral-direct algorithm for the B-term is performed simultaneously with the two-electron integrals over $1/r_{12}$ and r_{12} to obtain the products $\bar{g}_{pq}^{rs}\bar{r}_{rs}^{ij}$ and $\bar{r}_{pq}^{rs}\bar{r}_{rs}^{ij}$, respectively.

For methods that involve triple excitations, an integral-direct scheme to compute the integrals g_{iq}^{rs} is processed and these integrals are stored on disk along with appropriate labels. It is noted that in the present CC-R12 implementation the energy contributions due to triple excitations depend on the R12 basis functions not explicitly, but only indirectly through the R12 induced changes in the amplitudes of the conventional single and double excitations.

In each CCSD-R12 or CCSDT-1a-R12 iteration, the two-electron integrals over $1/r_{12}$ are recomputed to contribute to the integral-direct part of the conventional CC equations. On the other hand, all of the R12 part of the CC equations is computed beforehand, and the computational cost of this part roughly compares with the computational work of just a few (≈ 6) conventional CC iterations.

Hence, an important computational aspect is the generation of the R12 related integrals.

13.2 Two-electron integrals

The linear R12 methods require the computation of four types of two-electron integrals over Cartesian Gaussian functions:

$$(ab|r_{12}^{-1}|cd) = \int\int \psi_a(1)\psi_c(2)\, r_{12}^{-1}\, \psi_b(1)\psi_d(2)\, d\tau_1 d\tau_2, \tag{127}$$

$$(ab|r_{12}|cd) = \int\int \psi_a(1)\psi_c(2)\, r_{12}\, \psi_b(1)\psi_d(2)\, d\tau_1 d\tau_2, \tag{128}$$

$$(ab|[T_1, r_{12}]|cd) = \int\int \psi_a(1)\psi_c(2)\, [T_1, r_{12}]\, \psi_b(1)\psi_d(2)\, d\tau_1 d\tau_2, \tag{129}$$

$$(ab|[T_2, r_{12}]|cd) = \int\int \psi_a(1)\psi_c(2)\, [T_2, r_{12}]\, \psi_b(1)\psi_d(2)\, d\tau_1 d\tau_2. \tag{130}$$

$T_i = -\frac{1}{2}\Delta_i$ is the kinetic energy operator ($i = 1, 2$) and ψ_a, ψ_b, ... are primitive Cartesian Gaussians with exponents a, b, ...,

$$\psi_a(1) = g_{a,\mathbf{A},\mathbf{l}}(1) = N_A x_{1A}^{l_x} y_{1A}^{l_y} z_{1A}^{l_z} \exp(-a r_{1A}^2). \tag{131}$$

This function is centered at $\mathbf{A}$, N_A is a normalization constant, and $x_A = x - A_x$, etc. It has been shown shown that the integral (129) can be computed very

efficiently from the relationship[132]:

$$(ab|[T_1, r_{12}]|cd) = -\left(\frac{a-b}{a+b}\right)(ab|r_{12}^{-1}|cd) - \nabla_P \cdot \nabla_R\,(ab|r_{12}|cd), \tag{132}$$

where the relative coordinates $\mathbf{P}$ and $\mathbf{R}$ are defined by:

$$\mathbf{P} = \frac{a\mathbf{A} + b\mathbf{B}}{a+b} \quad \text{and} \quad \mathbf{R} = \mathbf{A} - \mathbf{B}. \tag{133}$$

The integral (130) can be computed similarly:

$$(ab|[T_2, r_{12}]|cd) = -\left(\frac{c-d}{c+d}\right)(ab|r_{12}^{-1}|cd) - \nabla_Q \cdot \nabla_S\,(ab|r_{12}|cd), \tag{134}$$

where:

$$\mathbf{Q} = \frac{c\mathbf{C} + d\mathbf{D}}{c+d} \quad \text{and} \quad \mathbf{S} = \mathbf{C} - \mathbf{D}. \tag{135}$$

By expanding the Gaussian charge distributions in Hermite functions:

$$\psi_a\psi_b = \sum_{tuv} E_{tuv}^{ab}\Lambda_{tuv}, \tag{136}$$

$$\Lambda_{tuv}(\mathbf{r}, p, \mathbf{P}) = \left(\frac{\partial}{\partial P_x}\right)^t \left(\frac{\partial}{\partial P_y}\right)^u \left(\frac{\partial}{\partial P_z}\right)^v \exp(-pr_P^2), \tag{137}$$

(with $p = a + b$), the integrals can be obtained from:

$$(ab|r_{12}^{-1}|cd) = \sum_{t'u'v'} E_{t'u'v'}^{cd} \sum_{tuv} E_{tuv}^{ab}(tuv|r_{12}^{-1}|t'u'v'), \tag{138}$$

$$(ab|r_{12}|cd) = \sum_{t'u'v'} E_{t'u'v'}^{cd} \sum_{tuv} E_{tuv}^{ab}(tuv|r_{12}|t'u'v'). \tag{139}$$

The integral over $[T_1, r_{12}]$ can be computed from:

$$\begin{aligned}(ab|[T_1, r_{12}]|cd) = &-\left(\tfrac{a-b}{a+b}\right)\textstyle\sum_{t'u'v'} E_{t'u'v'}^{cd} \sum_{tuv} E_{tuv}^{ab}(tuv|r_{12}^{-1}|t'u'v') \\ &- \textstyle\sum_{t'u'v'} E_{t'u'v'}^{cd} \sum_{tuv} \Big\{ E_{tuv}^{ab;x}(t{+}1, uv|r_{12}|t'u'v') + E_{tuv}^{ab;y}(t, u{+}1, v|r_{12}|t'u'v') \\ &+ E_{tuv}^{ab;z}(tu, v{+}1|r_{12}|t'u'v') \Big\}.\end{aligned} \tag{140}$$

Here it was used that:

$$\frac{\partial}{\partial P_x}(tuv|r_{12}|t'u'v') = (t{+}1, uv|r_{12}|t'u'v'), \tag{141}$$

$$E_{tuv}^{ab;x} = \frac{\partial}{\partial R_x} E_{tuv}^{ab}. \tag{142}$$

The integral $[T_2, r_{12}]$ can be computed analogously. By virtue of the change of variables to relative coordinates, the range of summation over t, u, and v is the same for all types of integrals. But note that the range of intermediate integrals over Hermite functions has been increased by one "quantum" due to the differentiation with respect to P_x (P_y, P_z) or Q_x (Q_y, Q_z). Furthermore, it has been shown that the integrals $(tuv|r_{12}|t'u'v')$ over Hermite functions can be obtained from the

electron-repulsion integrals $(tuv|r_{12}^{-1}|t'u'v')$ with negligible computational cost[132]. As an overall result, the computational effort required for the evaluation of the four types of two-electron integrals occuring in the R12 methods is roughly equal to four times the effort for the computation of the usual electron-repulsion integrals alone.

A recent development has been concerned with the computation of the $[T_1, r_{12}]$ and $[T_2, r_{12}]$ integrals at a later stage of the four-index transformation of the integrals from the atomic orbital (AO) basis into the orthonormal Hartree-Fock molecular orbital (MO) basis[133]: It was suggested to exploit the relation:

$$\begin{aligned}(ab|[T_1, r_{12}]|JQ) = &-\left(\tfrac{a-b}{a+b}\right)\sum_{tuv} E_{tuv}^{ab}(tuv|r_{12}^{-1}|JQ) \\ -\sum_{tuv}&\Big\{E_{tuv}^{ab;x}(t{+}1, uv|r_{12}|JQ) + E_{tuv}^{ab;y}(t, u{+}1, v|r_{12}|JQ) \\ &+E_{tuv}^{ab;z}(tu, v{+}1|r_{12}|JQ)\Big\}, \end{aligned} \tag{143}$$

where J and Q are MOs (J occupied, Q arbitrary), and simularly for the integral $(IP|[T_2, r_{12}]|cd)$. In the proposed algorithm, the Hermite integrals are transformed partially into the MO basis before the final $[T_1, r_{12}]$ and $[T_2, r_{12}]$ integrals are assembled. Since the number of occupied orbitals is relatively small (i.e., much smaller than the size of the AO basis), the new scheme based on (143) seems very attractive. At this point it is noted that the coupled-cluster R12 methods require exactly the same transformed MO integrals of the type $(IP|[T_1, r_{12}]|JQ)$ and $(IP|[T_2, r_{12}]|JQ)$, such that the CC-R12 methods will be improved to the same extent.

The performance of the R12 integral generation and corresponding four-index transformation is illustrated in Table 8. Shown are the timings for MP2-R12/A calculations on ferrocene and the permanganate ion. Remember that these MP2-R12/A calculations implicitly generate the transformed $[T(1), r_{12}]$ and $[T(2), r_{12}]$ integrals required by the CC-R12 methods, in other words, that the extra work in CC-R12 calculations compared with conventional coupled-cluster calculations in a good approximation consists of the computational steps displayed in Table 8. The computation time needed for the evaluation of the two-electron integrals for the ferrocene molecule might seem somewhat high: 1100 minutes. This is due to the fact that four passes through the program were needed as not more than ca. 15 gigabytes of external storage were available for the partially transformed integrals.

In summarizing, it is concluded that interesting applications of the R12 methods are possible today and will be even more so in the future due to the efficient atomic orbital driven direct implementations of the two-electron integrals in the MP2-R12 and CC-R12 computer programs.

14 Numerical examples

Today, explicitly correlated calculations can be performed on transition metal compounds. To illustrate that this type of calculations is not confined to small two- or four-electron systems (ten-electron sytems at the best), Table 9 shows the optimized structures of ferrocene ($Fe(C_5H_5)_2$ or $Fe(Cp)_2$ for short) and the permanganate ion (MnO_4^-) as obtained from MP2-R12/A calculations. For ferrocene, only the distance from Fe to the center of mass of the cyclopentadienyl ring was opti-

Table 8. Performance assessment of the SORE program, obtained from calculations on $FeCp_2$ and MnO_4^- on a NEC SX-3/24R supercomputer.

	$FeCp_2$ [a]		MnO_4^- [b]	
	t_{cpu}/min[c]	GFLOPS[d]	t_{cpu}/min[c]	GFLOPS[d]
Integrals				
Evaluation	1100	0.5	53	0.6
Sorting	75	–	5	–
Transformation				
1st step	48	3.0	6	3.7
2nd step	41	5.1	7	5.5
3rd step	10	1.3	2	1.3
4th step	20	0.3	2	0.3
Miscellaneous[e]	97	≈ 0	10	≈ 0
Total	1391	0.7	85	1.1

a Exploiting D_{5d} symmetry, correlating 66 electrons, and using 864 basis functions. The calculation required 4×12 gigabytes of disk space, the mean vector length was 176 elements, and the degree of vectorization was 96.4%.

b Exploiting T_d symmetry, correlating 40 electrons, and using 520 basis functions. The calculation required 15 gigabytes of disk space, the mean vector length was 192 elements, and the degree of vectorization was 97.6%.

c Central processor unit (cpu) time in minutes.

d 1 GFLOPS = 1×10^9 floating point operations per second.

e Consists mainly of the computation time required for the computation of the MP2-R12/A energy. This part of the calculation has not been vectorized.

Table 9. Fe–Cp and Mn–O bond lengths (in pm) of the ferrocene molecule and the permanganate ion, computed at the basis set limit of second-order perturbation theory[a].

Molecule	Basis	N^b	$n_e{}^c$	SCF	MP2[d]	Experiment
$Fe(Cp)_2$	Fe:16s12p8d6f/ C:8s5p4d3f/H:3s2p	864	66	187.2	146.8	166.1 ± 0.4[e]
			58		148.1	
MnO_4^-	Mn:17s13p10d6f/ O:13s9p5d4f	520	40	154.3	158.1	162.9 ± 0.8[f]

a Obtained from MP2-R12/A (SORE) calculations.

b Number of basis functions.

c Number of electrons correlated.

d K-shells and L-shell of transition metal not correlated.

e Gas-phase bond-distance as reported by Haaland[134].

f Crystal structure of $KMnO_4$ determined by Palenik[135].

mized. The basis set for ferrocene is described elsewhere[120], while the basis set for MnO_4^- was derived from the ANO basis sets of Widmark *et al.*[136] and Pou-Amérigo

Table 10. Bond disruption enthalpy (ΔH°_{298} in kJ mol^{-1}) of ferrocene computed from the heterolytic dissociation: $FeCp_2 \rightarrow Fe^{2+}(^5D) + 2\,Cp^-$.

CCSD(T)	CASPT2	Method/Basis set	N^a	$n_e{}^b$
3 046		CCSD(T)/TZV2P+f[c]	373	66
	3 117	CASPT2/basis 4[c,d]	255	58
+42	−42	MP2-R12/A correction[e]	865	58&66
+8	+38	Other corrections[f]		
3 096	3 114			
−363	−363	$\Delta E(^1I–^5D)$ excitation energy [145]		
2 733	2 751	Theoretical ΔH°_{298}		
2 742 ± 60		Best estimate		
2 658 ± 26		Experimental value[146]		

a Number of contracted basis functions.

b Numer of electrons correlated.

c With respect to the ^1I state of Fe^{2+}.

d Basis 4 = Fe:6s5p4d2f/C:4s3p1d/H:2s.

e Using the basis set Fe:16s12p8d6f/C:13s7p4d3f/H:6s2p.

f Semicore 3s3p correlation, vibrational zero-point energy, translational, rotational, and vibrational thermal energy, structural relaxation, and relativistic corrections.

et al.[137] (The spdf part of the primitive 21s15p10d6f4g set for Mn was contracted to a segmented contraction of the type 17s13p10d6f by contracting the innermost orbitals. Similarly, the primitive 14s9p4d3f set for O was contracted to a segmented contraction of the type 11s7p4d3f and diffuse functions (2s2p1d1f) were added to the latter by geometrical progression.) The results in Table 9 represent the AO basis set limit results for the SCF and MP2 levels of theory and can be used to calibrate standard one-particle basis sets for transition metal compounds at the Hartree-Fock and correlated levels. For example, the present Fe–Cp bond length at the MP2 level (correlating 66 electrons) is ca. 11 pm shorter than the MP2 value of 158 pm obtained by Park and Almlöf[138] – indicating serious basis set deficiencies in their calculations – while the present limiting value of 148.1 pm (correlating 58 electrons) agrees well with the MP2 result of 148.6 pm obtained by Pierloot *et al.*[139]

The benchmark calculations on MnO_4^- were performed to calibrate basis sets for future theoretical studies of the electronic spectrum. Such studies using density functional, symmetry-adapted-cluster CI (SAC-CI), and coupled-cluster theory have been performed by Dickson and Ziegler[140], Nakai *et al.*[141,142], and Nooijen[143], respectively. For both ferrocene and permanganate, it is interesting to investigate how well single-reference based coupled-cluster methods perform despite the well-known Hartree-Fock failure for these transition metal compounds.

Recent complete active space self-consistent-field (CASSCF), complete active space second-order perturbation theory (CASPT2), and coupled-cluster calculations using large Gaussian basis sets were concerned with the molecular structure and binding energy of ferrocene[139,144].

Koch *et al.*[144] determined the equilibrium bond length to $R_{Fe-Cp} = 166.0$ pm,

in full agreement with the experimental value, and reported the total CCSD(T) energy of ferrocene obtained from calculations using a large Gaussian basis set containing 373 contracted basis functions. These authors correlated all 66 valence electrons.

Pierloot *et al.*[139], using an active space consisting of 10 electrons distributed among 10 orbitals and a basis set containing 255 functions, obtained a bond length of $R_{\mathrm{Fe-Cp}} = 164.3$ pm at the counterpoise corrected CASPT2 level correlating 58 electrons (i.e., freezing the 3s/3p semicore orbitals of Fe). They also computed the bond disruption enthalpy from the heterolytic dissociation

$$\mathrm{FeCp_2} \rightarrow \mathrm{Fe^{2+}}(^5\mathrm{D}) + 2\mathrm{Cp^-},$$

which amounted to 2 628 kJ mol^{-1} at the CASPT2 level, in good agreement with the experimental value of $2\,658 \pm 26$ kJ mol^{-1}.[146]

MP2-R12/A calculations were performed to obtain insight into the remaining errors in the CCSD(T) and CASPT2 calculations related to the use of a finite AO basis set[120]. These MP2-R12/A calculations were performed with a large Gaussian basis set of the type Fe:16s12p8d6f/C:13s7p4d3f/H:6s2p containing more than 1000 Cartesian Gaussian functions. Furthermore, core and core-valence correlation effects as well as thermal, vibrational, relativistic, and structural relaxation effects were studied[120].

The results of this study are presented in Table 10. The CCSD(T) and CASPT2 calculations used very different basis sets, and hence, the MP2-R12/A complete basis set corrections differ largely. However, the two corresponding extrapolations of the theoretical bond disruption enthalpy are very similar, and the deduced averaged value amounts to $2\,742 \pm 60$ kJ mol^{-1}. This is the currently most accurate theoretical estimate. The agreement of this estimate with the experimental value is not satisfactory and the latter is challenged by the *ab initio* calculations. To resolve the disagreement in more detail, future studies of the individual heats of formation of ferrocene, $\mathrm{Fe^{2+}}$, and $\mathrm{Cp^-}$ are required. Especially for $\mathrm{Cp^-}$, the experimental value might be uncertain.

15 Concluding remark

In the present lecture, we have seen how the poor description of the Coulomb cusp leads to a very poor basis-set convergence of the orbital approximation to the electron correlation problem. Wavefunctions with explicit depence on the interelectronic coordinates r_{ij} help to overcome the poor convergence. One can add the r_{ij}-dependent terms to the wavefunction or transform the Hamiltonian accordingly.

There are, still, a few more approaches that aim at quantitatively correct computations of electron correlation effects. One could for example employ quantum Monte Carlo methods[147] or integrate the correlation energy functional with the Hartree-Fock density if that functional were known to high accuracy[148]. Another set of techniques, including extrapolation, scaling, and empirical corrections can be found among the Gn ($n = 1, 2, 3$) family of methods[149]. Cf. the paper by Curtiss *et al.*[149] for more references. We have not discussed these alternative methods as the focus of the present lecture was on R12 methods and Gaussian geminals.

Acknowledgments

The author thanks Dr. A. Halkier and Dr. J.H. van Lenthe for helpful discussions. A genereous grant of computing time from the Centro Svizzero di Calcolo Scientifico in Manno, Switzerland, is gratefully acknowledged. The research of the author has been made possible by a fellowship of the Royal Netherlands Academy of Arts and Sciences.

References

1. J.C. Slater, Central fields and Rydberg formulas in wave mechanics, Phys. Rev. **31**, 333-343 (1928).
2. E.A. Hylleraas, Neue Berechnung der Energie des Heliums im Grundzustande, sowie des tiefsten Terms von Ortho-Helium, Z. Phys. **54**, 347-366 (1929).
3. L. Pauling and E.B. Wilson, Jr., *Introduction to Quantum Mechanics*, McGraw-Hill, New York (1935).
4. H. Hellmann, *Einführung in die Quantenchemie*, Deuticke, Leipzig (1937).
5. J.C. Slater, *Quantum Theory of Matter*, McGraw-Hill, New York (1968).
6. H.A. Bethe and E.E. Salpeter, *Quantum Mechanics of One- and Two-Electron Atoms*, Plenum Press, New York (1977).
7. F. Jensen, *Introduction to Computational Chemistry*, Wiley, Chichester (1998).
8. T. Helgaker, P. Jørgensen, and J. Olsen, *Molecular Electronic Structure Theory*, Wiley, New York (2000).
9. G. Tarczay, A. G. Császár, W. Klopper, V. Szalay, W.D. Allen, and H.F. Schaefer III, The barrier to linearity of water, J. Chem. Phys. **110**, 11971-11981 (1999).
10. K. Aarset, A. G. Császár, E.L. Sibert III, W.D. Allen, H.F. Schaefer III, W. Klopper, and J. Noga, Anharmonic force field, vibrational energies, and barrier to inversion of SiH_3^-, *J. Chem. Phys.* (in press).
11. S. Salomonson and P. Öster, Relativistic all-order pair functions from a discretized single-particle Dirac Hamiltonian, Phys. Rev. A **40**, 5548-5558 (1989).
12. D. Feller and K.A. Peterson, An examination of intrinsic errors in electronic structure methods using the Environmental Molecular Sciences Laboratory computational results database and the Gaussian-2 set, J. Chem. Phys. **108**, 154-176 (1998).
13. K.L. Bak, P. Jørgensen, J. Olsen, T. Helgaker, and W. Klopper, The accuracy of atomization energies and reaction enthalpies in standard and extrapolated electronic wavefunction/basis set calculations, *J. Chem. Phys.* (submitted).
14. T.H. Dunning, Jr., Gaussian basis sets for use in correlated molecular calculations, J. Chem. Phys. **90**, 1007-1023 (1989).
15. R.A. Kendall, T.H. Dunning, Jr., and R.J. Harrison, Electron affinities of the first-row atoms revisited. Systematic basis sets and wave functions, J. Chem. Phys. **96**, 6796-6806 (1992).
16. D.E. Woon and T.H. Dunning, Jr., Gaussian basis sets for use in correlated molecular calculations. III. The atoms aluminum through argon, J. Chem. Phys. **98**, 1358-1371 (1993).

17. D.E. Woon and T.H. Dunning, Jr., Gaussian basis sets for use in correlated molecular calculations. IV. Calculation of static electrical response properties, J. Chem. Phys. **100**, 2975-2988 (1994).
18. D.E. Woon and T.H. Dunning, Jr., Gaussian basis sets for use in correlated molecular calculations. V. Core-valence basis sets for boron through neon, J. Chem. Phys. **103**, 4572-4585 (1995).
19. A.K. Wilson, T. van Mourik, and T.H. Dunning, Jr., Gaussian basis sets for use in correlated molecular calculations. VI. Hextuple zeta correlation consistent basis sets for boron through neon, J. Mol. Struct. (Theochem) **388**, 339-349 (1996).
20. W. Klopper, Limiting values for Møller-Plesset second-order correlation energies of polyatomic systems: A benchmark study on Ne, HF, H_2O, N_2, and He...He, J. Chem. Phys. **102**, 6168-6179 (1995).
21. T. Helgaker, W. Klopper, H. Koch, and J. Noga, Basis-set convergence of correlated calculations on water, J. Chem. Phys. **106**, 9639-9646 (1997).
22. W. Klopper, M. Schütz, H.P. Lüthi, and S. Leutwyler, An ab initio derived torsional potential energy surface for $(H_2O)_3$. II. Benchmark studies and interaction energies, J. Chem. Phys. **103**, 1085-1098 (1995).
23. W. Klopper, r_{12}-Dependent wavefunctions. In: P.v.R. Schleyer, N.L. Allinger, T. Clark, J. Gasteiger, P.A. Kollman, H.F. Schaefer III, and P.R. Schreiner, editors, *The Encyclopedia of Computational Chemistry*, Wiley, Chichester (1998), pp. 2351-2375.
24. C. Hampel and H.-J. Werner, Local treatment of electron correlation in coupled cluster theory, J. Chem. Phys. **104**, 6286-6297 (1996).
25. G. Rauhut, P. Pulay, and H.-J. Werner, Integral transformation with low-order scaling for large second-order Møller-Plesset calculations, J. Comput. Chem. **19**, 1241-1254 (1998).
26. M. Schütz, G. Hetzer, and H.-J. Werner, Low-order scaling local electron correlation methods. I. Linear scaling local MP2, J. Chem. Phys. **111**, 5691-5705 (1999).
27. C.F. Bunge, Theor. Chim. Acta **16**, 126 (1970).
28. D.P. Carroll, H.J. Silverstone, and R.M. Metzger, Piecewise polynomial configuration interaction natural orbital study of $1s^2$ helium, J. Chem. Phys. **71**, 4142-4163 (1979).
29. R.N. Hill, Rates of convergence and error estimation formulas for the Rayleigh-Ritz variational method, J. Chem. Phys. **83**, 1173-1196 (1985).
30. C. Schwartz, Importance of angular correlations between atomic electrons, Phys. Rev. **126**, 1015-1019 (1962).
31. C. Schwartz, Estimating convergence rates of variational calculations, Meth. Comput. Phys. **2**, 241-266 (1963).
32. W. Kutzelnigg and J.D. Morgan III, Rates of convergence of the partial wave expansions of atomic correlation energies, J. Chem. Phys. **96**, 4484-4508 (1992).
33. W. Kutzelnigg and J.D. Morgan III, Erratum: Rates of convergence of the partial wave expansions of atomic correlation energies, J. Chem. Phys. **97**, 8821-8821 (1992).

34. K.L. Bak, A. Halkier, P. Jørgensen, J. Olsen, T. Helgaker, and W. Klopper, Chemical accuracy from "Coulomb-hole" extrapolated molecular quantum-mechanical calculations, *J. Chem. Acc.* (submitted).
35. A. Halkier, T. Helgaker, P. Jørgensen, W. Klopper, H. Koch, J. Olsen, and A.K. Wilson, Basis-set convergence in correlated calculations on Ne, N_2, and H_2O, Chem. Phys. Lett. **286**, 243-252 (1998).
36. A. Halkier, T. Helgaker, P. Jørgensen, W. Klopper, and J. Olsen, Basis-set convergence of the energy in molecular Hartree-Fock calculations, Chem. Phys. Lett. **302**, 437-446 (1999).
37. F. Jensen, The basis set convergence of the Hartree-Fock energy for H_2, J. Chem. Phys. **110**, 6601-6605 (1999).
38. A. Halkier, W. Klopper, T. Helgaker, P. Jørgensen, and P.R. Taylor, Basis-set convergence of the interaction energy of hydrogen-bonded complexes, J. Chem. Phys. **111**, 9157-9167 (1999).
39. W. Klopper and T. Helgaker, Extrapolation to the limit of a complete basis set for electronic structure calculations on the N_2 molecule, Theor. Chem. Acc. **99**, 265-271 (1998).
40. A. Halkier, W. Klopper, T. Helgaker, and P. Jørgensen, Basis-set convergence of the molecular electric dipole moment, J. Chem. Phys. **111**, 4424-4430 (1999).
41. A. Halkier, H. Koch, P. Jørgensen, O. Christiansen, I.M. Beck Nielsen, and T. Helgaker, A systematic ab initio study of the water dimer in hierarchies of basis sets and correlation models, Theor. Chem. Acc. **97**, 150-157 (1997).
42. M. Schütz, S. Brdarski, P.-O. Widmark, R. Lindh, and G. Karlström, The water dimer interaction energy: Convergence to the basis set limit at the correlated level, J. Chem. Phys. **107**, 4597-4605 (1997).
43. W. Klopper and H.P. Lüthi, The MP2-limit correction applied to coupled-cluster calculations of the electronic binding energies of the hydrogen fluoride and water dimers, Mol. Phys. **96**, 559-570 (1999).
44. D.G. Truhlar, Basis-set extrapolation, Chem. Phys. Lett. **294**, 45-48 (1998).
45. Y.-Y. Chuang and D.G. Truhlar, Geometry optimization with an infinite basis set, J. Phys. Chem. A **103**, 651-652 (1999).
46. A. Halkier, T. Helgaker, W. Klopper, P. Jørgensen, and A. G. Császár, Comment on "Geometry Optimization with an Infinite Basis Set" [J. Phys. Chem. A **103** (1999) 651] and "Basis-set extrapolation" [Chem. Phys. Lett. **294** (1998) 45], Chem. Phys. Lett. **310**, 385-389 (1999).
47. P.L. Fast, M.L. Sánchez, and D.G. Truhlar, Infinite basis limits in electronic structure theory, J. Chem. Phys. **111**, 2921-2926 (1999).
48. D. Feller, Application of systematic sequences of wave functions to the water dimer, J. Chem. Phys. **96**, 6104-6114 (1992).
49. J.M.L. Martin, Ab initio total atomization energies of small molecules – towards the basis set limit, Chem. Phys. Lett. **259**, 669-678 (1996).
50. J.M.L. Martin and P.R. Taylor, Benchmark quality total atomization energies of small polyatomic molecules J. Chem. Phys. **106**, 8620-8623 (1997).
51. J.M.L. Martin and G. de Oliveira, Towards standard methods for benchmark quality ab initio thermochemistry – W1 and W2 theory, J. Chem. Phys. **111**,

1843-1856 (1999).
52. A.K. Wilson and T.H. Dunning, Jr., Benchmark calculations with correlated molecular wave functions. X. Comparison with "exact" MP2 calculations on Ne, HF, H_2O, and N_2, J. Chem. Phys. **106**, 8718-8726 (1997).
53. A. Halkier, T. Helgaker, W. Klopper, and J. Olsen, Basis-set convergence of the two-electron Darwin term, *Chem. Phys. Lett.* (submitted).
54. G.A. Petersson, A.K. Yee, and A. Bennett, Complete basis set correlation energies. III. The total correlation energy of the neon atom, J. Chem. Phys. **83**, 5105-5128 (1985).
55. G.A. Petersson and M. Braunstein, Complete basis set correlation energies. IV. The total correlation energy of the water molecule, J. Chem. Phys. **83**, 5129-5134 (1985).
56. G.A. Petersson, A. Bennett, T.G. Tensfeldt, M.A. Al-Laham, W.A. Shirley, and J. Mantzaris, A complete basis set model chemistry. I. The total energies of closed-shell atoms and hydrides of the first-row elements, J. Chem. Phys. **89**, 2193-2218 (1988).
57. G.A. Petersson and M.A. Al-Laham, A complete basis set model chemistry. II. Open-shell systems and the total energies of the first-row atoms, J. Chem. Phys. **94**, 6081-6090 (1991).
58. G.A. Petersson, T.G. Tensfeldt, and J.A. Montgomery, Jr., A complete basis set model chemistry. III. The complete basis set-quadratic configuration interaction family of methods, J. Chem. Phys. **94**, 6091-6101 (1991).
59. J.W. Ochterski, G.A. Petersson, and K.B. Wiberg, A comparison of model chemistries, J. Am. Chem. Soc. **117**, 11299-11308 (1995).
60. J.W. Ochterski, G.A. Petersson, and J.A. Montgomery, Jr., A complete basis set model chemistry. V. Extensions to six or more heavy atoms, J. Chem. Phys. **104**, 2598-2619 (1996).
61. G.A. Petersson, D.K. Malick, W.G. Wilson, J.W. Ochterski, J.A. Montgomery, Jr., and M.J. Frisch, Calibration and comparison of the Gaussian-2, complete basis set, and density functional methods for computational thermochemistry, J. Chem. Phys. **109**, 10570-10579 (1998).
62. G.A. Petersson and M.J. Frisch, A journey from generalized valence bond theory to the full CI complete basis set limit, *J. Phys. Chem. A* (in press).
63. W. Klopper, K.L. Bak, P. Jørgensen, J. Olsen, and T. Helgaker, Highly accurate calculations of molecular electronic structure, J. Phys. B **32**, R1-3-R130 (1999).
64. T. Helgaker and W. Klopper, Perspective on "Neue Berechnung der Energie des Heliums im Grundzustande, sowie des tiefsten Terms von Ortho-Helium" by E. A. Hylleraas [Z. Phys. 54, 347 (1929)], *Theor. Chem. Acc.* (in press).
65. T. Kato, On the eigenfunctions of many-particle systems in quantum mechanics, Commun. Pure Appl. Math. **10**, 151-177 (1957).
66. H. Kleindienst, A. Lüchow, and H.-P. Merckens, Accurate upper and lower bounds for some excited S states of the He atom, Chem. Phys. Lett. **218**, 441-444 (1994).
67. G. Büsse and H. Kleindienst, Double-linked Hylleraas configuration-interaction calculation for the nonrelativistic ground-state energy of the Be atom, Phys.

Rev. A **51**, 5019-5020 (1995).
68. W. Kutzelnigg, Quantum chemistry in Fock space. I. The universal wave and energy operators, J. Chem. Phys. **77**, 3081-3097 (1982).
69. W. Kutzelnigg, Quantum chemistry in Fock space. III. Particle-hole formalism, J. Chem. Phys. **80**, 822-830 (1984).
70. W. Kutzelnigg and W. Klopper, Wave functions with terms linear in the interelectronic coordinates to take care of the correlation cusp. I. General theory, J. Chem. Phys. **94**, 1985-2001 (1991).
71. B. Jeziorski, H.J. Monkhorst, K. Szalewicz, and J.G. Zabolitzky, Atomic and molecular correlation energies with explicitly correlated Gaussian geminals. III. Coupled cluster treatment for He, Be, H_2, and LiH, J. Chem. Phys. **81**, 368-388 (1984).
72. K. Szalewicz, B. Jeziorski, and H.J. Monkhorst, A new functional for variational calculations of atomic and molecular second-order correlation energies, Chem. Phys. Lett. **91**, 169-172 (1982).
73. K. Szalewicz, B. Jeziorski, H.J. Monkhorst, and J.G. Zabolitzky, Atomic and molecular correlation energies with explicitly correlated Gaussian geminals. I. Second-order perturbation treatment for He, Be, H_2, and LiH, J. Chem. Phys. **78**, 1420-1430 (1983).
74. K. Szalewicz, B. Jeziorski, H.J. Monkhorst, and J.G. Zabolitzky, Atomic and molecular correlation energies with explicitly correlated Gaussian geminals. II. Perturbation treatment through third-order for He, Be, H_2, and LiH, J. Chem. Phys. **79**, 5543-5552 (1983).
75. K. Szalewicz, J.G. Zabolitzky, B. Jeziorski, and H.J. Monkhorst, Atomic and molecular correlation energies with explicitly correlated Gaussian geminals. IV. A simplified treatment of strong orthogonality in MBPT and coupled cluster calculations, J. Chem. Phys. **81**, 2723-2731 (1984).
76. K.B. Wenzel, J.G. Zabolitzky, K. Szalewicz, B. Jeziorski, and H.J. Monkhorst, Atomic and molecular correlation energies with explicitly correlated Gaussian geminals. V. Cartesian Gaussian geminals and the neon atom, J. Chem. Phys. **85**, 3964-3974 (1986).
77. K.B. Wenzel and J.G. Zabolitzky, Efficient approximations for a rapid optimization of explicitly correlated Gaussian geminals, J. Chim. Phys. **84**, 691-694 (1987).
78. S.A. Alexander, H.J Monkhorst, and K. Szalewicz, Random tempering of Gaussian-type geminals. I. Atomic systems, J. Chem. Phys. **85**, 5821-5825 (1986).
79. S.A. Alexander, H.J. Monkhorst, and K. Szalewicz, Random tempering of Gaussian-type geminals. II. Molecular systems, J. Chem. Phys. **87**, 3976-3980 (1987).
80. S.A. Alexander, H.J. Monkhorst, and K. Szalewicz, Random tempering of Gaussian-type geminals. III. Coupled pair calculations on lithium hydride and beryllium, J. Chem. Phys. **89**, 355-359 (1988).
81. R. Bukowski, B. Jeziorski, and K. Szalewicz, New effective strategy of generating Gaussian-type geminal basis sets for correlation energy calculations, J. Chem. Phys. **100**, 1366-1374 (1994).

82. R. Bukowski, B. Jeziorski, S. Rybak, and K. Szalewicz, Second-order correlation energy for H_2O using explicitly correlated Gaussian geminals, J. Chem. Phys. **102**, 888-897 (1995).
83. R. Bukowski, B. Jeziorski, and K. Szalewicz, Basis set superposition problem in interaction energy calculations with explicitly correlated bases: Saturated second- and third-order energies for He_2, J. Chem. Phys. **104**, 3306-3319 (1996).
84. R. Bukowski, B. Jeziorski, and K. Szalewicz, Analytic first-order properties from explicitly correlated many-body perturbation theory and Gaussian geminal basis, J. Chem. Phys. **108**, 7946-7958 (1998).
85. R. Bukowski, B. Jeziorski, and K. Szalewicz, Gaussian geminals in explicitly correlated coupled cluster theory including single and double excitations, J. Chem. Phys. **110**, 4165-4183 (1999).
86. J. Paldus, Algebraic approach to coupled cluster theory. In: G.L. Malli, editor, *Relativistic and Electron Correlation Effects in Molecules and Solids*, NATO ASI Series B, Vol. **318**, Plenum Press, New York (1994), pp. 207-282.
87. W. Kutzelnigg, r_{12}-Dependent terms in the wave function as closed sums of partial wave amplitudes for large ℓ, Theor. Chim. Acta **68**, 445-469 (1985).
88. W. Klopper and W. Kutzelnigg, Møller-Plesset calculations taking care of the correlation cusp, Chem. Phys. Lett. **134**, 17-22 (1987).
89. V. Termath, W. Klopper, and W. Kutzelnigg, Wave functions with terms linear in the interelectronic coordinates to take care of the correlation cusp. II. Second-order Møller-Plesset (MP2-R12) calculations on closed-shell atoms, J. Chem. Phys. **94**, 2002-2019 (1991).
90. W. Klopper and W. Kutzelnigg, Wave functions with terms linear in the interelectronic coordinates to take care of the correlation cusp. III. Second-order Møller-Plesset (MP2-R12) calculations on molecules of first row atoms, J. Chem. Phys. **94**, 2020-2030 (1991).
91. J. Noga, W. Klopper, and W. Kutzelnigg, CC-R12, a correlation cusp corrected coupled-cluster method with a pilot application to the Be_2 potential curve, Chem. Phys. Lett. **199**, 497-504 (1992).
92. W. Klopper and J. Noga, An explicitly correlated coupled cluster calculation of the helium–helium interatomic potential, J. Chem. Phys. **103**, 6127-6132 (1995).
93. W. Klopper, W. Kutzelnigg, H. Müller, J. Noga, and S. Vogtner, Extremal electron pairs – Application to electron correlation, especially the R12 method, Topics Curr. Chem. **203**, 21-42 (1999).
94. J. Noga and W. Kutzelnigg, Coupled cluster theory that takes care of the correlation cusp by inclusion of linear terms in the interelectronic coordinates, J. Chem. Phys. **101**, 7738-7762 (1994).
95. R.J. Gdanitz, A formulation of multiple-reference CI with terms linear in the interelectronic distances, Chem. Phys. Lett. **210**, 253-260 (1993).
96. R.J. Gdanitz, Accurately solving the electronic Schrödinger equation of atoms and molecules using explicitly correlated (r_{12}-) multireference configuration interaction. III. Electron affinities of first-row atoms, J. Chem. Phys. **110**, 706-710 (1999).

97. W. Klopper, Orbital-invariant formulation of the MP2-R12 method, Chem. Phys. Lett. **186**, 583-585 (1991).
98. W. Klopper and J. Almlöf, Towards the one-particle basis set limit of second-order correlation energies: MP2-R12 calculations on small Be_n and Mg_n clusters (n=1–4), J. Chem. Phys. **99**, 5167-5177 (1993).
99. W. Klopper, R. Röhse, and W. Kutzelnigg, CID and CEPA calculations with linear r_{12} terms, Chem. Phys. Lett. **178**, 455-461 (1991).
100. J. Noga, W. Klopper, and W. Kutzelnigg, CCR12: An explicitly correlated coupled-cluster theory. In: R.J. Bartlett, editor, *Recent Advances in Coupled-Cluster Methods*, *Recent Advances in Computational Chemistry*, Vol. **3**, World Scientific, Singapore (1997), pp. 1-48.
101. B.J. Persson and P.R. Taylor, Accurate quantum-chemical calculations: The use of Gaussian-type geminal functions in the treatment of electron correlation, J. Chem. Phys. **105**, 5915-5926 (1996).
102. J.R. Flores, Computation of the second-order energies of Ne using a finite-element method, Phys. Rev. A **46**, 6063-6066 (1992).
103. W. Cencek and J. Rychlewski, Many-electron explicitly correlated Gaussian functions. I. General theory and test results, J. Chem. Phys. **98**, 1252-1261 (1993).
104. W. Cencek and J. Rychlewski, Many-electron explicitly correlated Gaussian functions. II. Ground state of the helium molecular ion He_2^+, J. Chem. Phys. **102**, 2533-2537 (1995).
105. W. Cencek, J. Komasa, and J. Rychlewski, Benchmark calculations for two-electron systems using explicitly correlated Gaussian functions, Chem. Phys. Lett. **246**, 417-420 (1995).
106. J. Komasa, W. Cencek, and J. Rychlewski, Explicitly correlated Gaussian functions in variational calculations: The ground state of the beryllium atom, Phys. Rev. A **52**, 4500-4507 (1995).
107. J. Komasa and J. Rychlewski, Explicitly correlated Gaussian functions in variational calculations. Microhartree accuracy of the helium dimer energy, Chem. Phys. Lett. **249**, 253-256 (1996).
108. N.C. Handy, Correlated Wavefunctions. In: G.H.F. Diercksen, B.T. Sutcliffe, and A. Veillard, editors, *Computational Techniques in Quantum Chemistry and Molecular Physics*, NATO ASI Series C, Vol. **15**, Reidel, Dordrecht (1975), pp. 425-433.
109. S.F. Boys and N.C. Handy, A condition to remove the indeterminacy in interelectronic correlation functions, Proc. Roy. Soc. (London) A **309**, 209-220 (1969).
110. S.F. Boys and N.C. Handy, The determination of energies and wavefunctions with full electronic correlation, Proc. Roy. Soc. (London) A **310**, 43-61 (1969).
111. S.F. Boys and N.C. Handy, A calculation for the energies and wavefunctions for states of neon with full electronic correlation accuracy, Proc. Roy. Soc. (London) A **310**, 63-78 (1969).
112. S.F. Boys and N.C. Handy, A first solution, for LiH, of a molecular transcorrelated wave equation by means of restricted numerical integration, Proc. Roy. Soc. (London) A **311**, 309-329 (1969).

113. N.C. Handy, Energies and expectation values for Be by the transcorrelated method, J. Chem. Phys. **51**, 3205-3212 (1969).
114. N.C. Handy, On the minimization of the variance of the transcorrelated Hamiltonian, Mol. Phys. **21**, 817-828 (1971).
115. N.C. Handy, The transcorrelated method for accurate correlation energies using gaussian-type functions: Examples on He, H_2, LiH, and H_2O, Mol. Phys. **23**, 1-27 (1972).
116. N.C. Handy, Correlated gaussian wavefunctions, Mol. Phys. **26**, 169-176 (1973).
117. G.W.F. Drake and Z.-C. Yan, Variational eigenvalues for the S states of helium, Chem. Phys. Lett. **229**, 486-490 (1994).
118. M. Nooijen and R.J. Bartlett, Elimination of Coulombic infinities through transformation of the Hamiltonian, J. Chem. Phys. **109**, 8232-8240 (1998).
119. W. Klopper, H.P. Lüthi, T. Brupbacher, and A. Bauder, Ab initio computations close to the one-particle basis set limit on the weakly bound van der Waals complexes benzene-neon and benzene-argon, J. Chem. Phys. **101**, 9747-9754 (1994).
120. W. Klopper and H.P. Lüthi, Towards the accurate computation of properties of transition metal compounds: The binding energy of ferrocene, Chem. Phys. Lett. **262**, 546-552 (1996).
121. S. Tsuzuki, W. Klopper, and H.P. Lüthi, High-level ab initio computations of structures and relative energies of two isomers of the CO_2 trimer, J. Chem. Phys. **111**, 3846-3854 (1999).
122. J. van de Bovenkamp and F.B. van Duijneveldt, MRCI calculations on the helium dimer employing an interaction optimized basis set, J. Chem. Phys. **110**, 11141-11151 (1999).
123. T. Korona, H.L. Williams, R. Bukowski, B. Jeziorski, and K. Szalewicz, Helium dimer potential from symmeytry-adapted perturbation theory calculations using large Gaussian geminal and orbital basis sets, J. Chem. Phys. **106**, 5109-5122 (1997).
124. J.M.L. Martin, Coupling between the convergence behavior of basis set and electron correlation: a quantitative study, Theor. Chem. Acc. **97**, 227-231 (1997).
125. W. Klopper, J.G.C.M. van Duijneveldt-van de Rijdt, and F.B. van Duijneveldt, Computational determination of equilibrium geometry and dissociation energy of the water dimer, *Phys. Chem. Chem. Phys.* (submitted).
126. B.J. Persson and P.R. Taylor, Molecular integrals over Gaussian-type geminal basis functions, Theor. Chem. Acc. **97**, 240-250 (1997).
127. J. Almlöf, K. Fægri, Jr., and K. Korsell, Principles for a direct SCF approach to LCAO-MO ab initio calculations, J. Comput. Chem. **3**, 385-399 (1982).
128. H. Koch, O. Christiansen, R. Kobayashi, P. Jørgensen, and T. Helgaker, A direct atomic orbital driven implementation of the coupled cluster singles and doubles (CCSD) model, Chem. Phys. Lett. **228**, 233-238 (1994).
129. H. Koch, A. Sánches de Merás, T. Helgaker, and O. Christiansen, The integral-direct coupled-cluster singles and doubles model, J. Chem. Phys. **104**, 4157-4165 (1996).

130. M. Schütz, R. Lindh, and H.-J. Werner, Integral-direct electron correlation methods, Mol. Phys. **96**, 719-733 (1999).
131. J.F. Stanton, J. Gauss, J.D. Watts, and R.J. Bartlett, A direct product decomposition approach for symmetry exploitation in many-body methods. I. Energy calculations, J. Chem. Phys. **94**, 4334-4345 (1991).
132. W. Klopper and R. Röhse, Computation of some new two-electron Gaussian integrals, Theor. Chim. Acta **83**, 441-453 (1992).
133. W. Klopper, Integrated integral evaluation, J. Mol. Struct. (Theochem) **388**, 175-185 (1996).
134. A. Haaland, Organometallic compounds studied by gas-phase electron diffraction, Topics Curr. Chem. **53**, 1-23 (1975).
135. G.J. Palenik, Crystal structure of potassium permanganate, Inorg. Chem. **6**, 503-507 (1967).
136. P.-O. Widmark, P.-Å. Malmqvist, and B.O. Roos, Density matrix averaged atomic natural orbital (ANO) basis sets for correlated molecular wave functions, Theor. Chim. Acta **77**, 291-306 (1990).
137. R. Pou-Amérigo, M. Merchán, I. Nebot-Gil, P.-O. Widmark, and B.O. Roos, Density matrix averaged atomic natural orbital (ANO) basis sets for correlated molecular wave functions. III. First row transition metal atoms, Theor. Chim. Acta **92**, 149-181 (1995).
138. C. Park and J. Almlöf, The electronic and molecular structure of ferrocene, J. Chem. Phys. **95**, 1829-1833 (1991).
139. K. Pierloot, B.J. Persson, and B.O. Roos, Theoretical study of the chemical bonding in $Ni(C_2H_4)$ and ferrocene, J. Phys. Chem. **99**, 3465-3472 (1995).
140. R.M. Dickson and T. Ziegler, A density functional study of the electronic spectrum of permanganate, Int. J. Quantum Chem. **58**, 618-687 (1996).
141. H. Nakai and H. Nakatsuji, Mechanism of photochemical reaction of permanganate ion, J. Mol. Struct. (Theochem) **311**, 141-151 (1994).
142. H. Nakai, Y. Ohmori, and H. Nakatsuji, Theoretical study on the ground and excited states of MnO_4^-, J. Chem. Phys. **95**, 8287-8291 (1991).
143. M. Nooijen, Combining coupled cluster and perturbation theory, J. Chem. Phys. **111**, 10815-10826 (1999).
144. H. Koch, P. Jørgensen, and T. Helgaker, The molecular structure of ferrocene, J. Chem. Phys. **104**, 9528-9530 (1996).
145. C. Corliss and J. Sugar, Energy levels of iron, Fe I through Fe XXVI, J. Phys. Chem. Ref. Data **11**, 135-241 (1982).
146. M.F. Ryan, J.R. Eyler, and D.E. Richardson, Adiabatic ionization energies, bond disruption enthalpies, and solvation free energies for gas-phase metallocenes and metallocenium ions, J. Am. Chem. Soc. **114**, 8611-8619 (1992).
147. A. Lüchow, J.B. Anderson, and D. Feller, Improved estimates of the total correlation energy in the ground state of the water molecule, J. Chem. Phys. **106**, 7706-7708 (1997).
148. S. Ivanov and M. Levy, Adiabatic integration formula for the correlation energy functional of the Hartree-Fock density, Theor. Chem. Acc. **103**, 117-123 (1999).
149. L.A. Curtiss, K. Raghavachari, P.C. Redfern, and J.A. Pople, Gaussian-3

theory using scaled energies, J. Chem. Phys. **112**, 1125-1132 (2000).

DIRECT SOLVERS FOR SYMMETRIC EIGENVALUE PROBLEMS

BRUNO LANG
Aachen University of Technology
Computing Center
Seffenter Weg 23, 52074 Aachen, Germany
E-mail: lang@rz.rwth-aachen.de

This article reviews classical and recent direct methods for computing eigenvalues and eigenvectors of symmetric full or banded matrices. The ideas underlying the methods are presented, and the properties of the algorithms with respect to accuracy and performance are discussed. Finally, pointers to relevant software are given.

This article reviews classical, as well as recent state-of-the-art, direct solvers for standard and generalized symmetric eigenvalue problems. In Section 1 we explain what direct solvers for symmetric eigenvalue problems are. Section 2 describes what we may reasonably expect from an eigenvalue solver in terms of accuracy and how algorithms should be structured in order to minimize the computing time, and introduces two basic tools on which most eigensolvers are based, namely similarity transformations and deflation. For accuracy reasons, orthogonal transformations should be used whenever possible. Some simple orthogonal transformations are discussed in Section 3.

Most eigenvalue solvers work in two phases: First the matrix is reduced to tridiagonal form and then the eigenvalue problem for the tridiagonal matrix is solved. Algorithms for these two phases are discussed in Sections 4 and 5, respectively, whereas Section 6 reviews algorithms that do not rely on an initial reduction. A synopsis of the available algorithms is given in Section 7. The methods presented here also carry over to complex Hermitean matrices. For simplicity we will focus on the real case. Finally, Section 8 points to relevant software.

1 Setting the Stage

After introducing some notational conventions, this section recalls the symmetric standard and generalized eigenvalue problems and points out the differences between direct and iterative eigenvalue solvers.

1.1 Some Notation

Throughout this article, matrices are denoted by uppercase letters $\mathsf{A}, \mathsf{B}, \ldots$, and the (i, j) entry of A is referred to as $\mathsf{A}(i, j)$. Analogously, lowercase letters $\mathbf{x}, \mathbf{y}, \ldots$ stand for (column) vectors with entries $\mathbf{x}(i)$, whereas greek letters $\alpha, \beta, \ldots$ denote scalars. Unless explicitly stated otherwise, all matrices are n-by-n and all vectors have length n.

I is the identity matrix (with entries $\mathsf{I}(i, j) = 1$ if $i = j$ and $\mathsf{I}(i, j) = 0$ otherwise), and $\mathsf{0}$ is the matrix with all entries equal to zero. For any matrix $\mathsf{A} \in \mathbb{R}^{m \times n}$, $\mathsf{A}^\top \in \mathbb{R}^{n \times m}$ denotes the *transpose* of A, i.e., $\mathsf{A}^\top(i, j) = \mathsf{A}(j, i)$ for all i, j. A square

matrix A is *symmetric* if $\mathsf{A}^\top = \mathsf{A}$.

1.2 The Symmetric Eigenvalue Problem

The *symmetric (standard) eigenvalue problem* consists of computing all or selected *eigenvalues* and associated *eigenvectors* of a symmetric matrix $\mathsf{A} \in \mathbb{R}^{n\times n}$, that is, scalars λ_i and vectors $\mathbf{q}_i \neq \mathbf{0}$ satisfying

$$\mathsf{A} \cdot \mathbf{q}_i = \mathbf{q}_i \cdot \lambda_i \ . \tag{1}$$

The eigenvalues of A are just the n roots (counting multiplicity) of its *characteristic polynomial* $p(\lambda) = \det(\mathsf{A} - \lambda\mathsf{I})$. (It is tempting to use this property for computing the eigenvalues: First determine the coefficients of the characteristic polynomial and then its roots. However, this method cannot be recommended because it gives highly inaccurate results.)

A pair $(\lambda_i, \mathbf{q}_i)$ satisfying Eq. (1) is called an *eigenpair* of A, and the set of all eigenvalues is called the *spectrum*, $\mathrm{spec}(\mathsf{A})$. For symmetric matrices all eigenvalues are real, and there exists a complete set of n mutually orthogonal, normalized (*orthonormal*, for brevity) real eigenvectors:

$$\mathbf{q}_i^\top \mathbf{q}_j = 0 \quad \text{for } i \neq j \qquad \text{and} \qquad \mathbf{q}_i^\top \mathbf{q}_i = 1 \quad \text{for } i = 1, \ldots, n \ .$$

Together with Eq. (1) this implies that A has an *eigendecomposition*

$$\mathsf{A} = \mathsf{Q}\mathsf{\Lambda}\mathsf{Q}^\top \ , \tag{2}$$

where $\mathsf{Q} = (\, \mathbf{q}_1 \,|\, \ldots \,|\, \mathbf{q}_n \,) \in \mathbb{R}^{n\times n}$ is an *orthogonal* matrix (i.e., $\mathsf{Q}^\top\mathsf{Q} = \mathsf{I}$), and

$$\mathsf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n) := \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} \ .$$

By convention the eigenvalues are numbered ascendingly, that is, $\lambda_1 \leq \ldots \leq \lambda_n$.

Not all applications require computing the full eigendecomposition. Sometimes if suffices to compute only the eigenvalues, sometimes only selected eigenpairs are needed (e.g., eigenpairs for all non-negative eigenvalues or for the 100 largest eigenvalues), etc.

In the *symmetric generalized eigenvalue problem* one wants to compute eigenpairs $(\lambda_i, \mathbf{q}_i)$ satisfying

$$\mathsf{A} \cdot \mathbf{q}_i = \mathsf{B} \cdot \mathbf{q}_i \cdot \lambda_i \ ,$$

where B is another symmetric matrix. In the generalized case the eigenvectors need not be mutually orthogonal. In most applications B is also *positive definite* (i.e., $\mathbf{z}^\top\mathsf{B}\mathbf{z} > 0$ for all $\mathbf{z} \neq \mathbf{0}$ or, equivalently, all eigenvalues of B are positive). In our treatment of the generalized problem we will focus on this case.

1.3 Direct and Iterative Eigensolvers

For the solution of linear systems we can choose between direct methods like Gaussian elimination, which give the solution after a fixed number of operations, and iterative solvers like the *conjugate gradients* method, which produce a sequence of increasingly accurate approximations to the solution. Iterative solvers terminate as soon as the required precision is attained and can thus lead to considerable savings in operations (and in memory as well).

By contrast, a result from Galois theory implies that *there can be no algorithm* that computes the eigenvalues of every matrix in a finite number of operations (additions, subtractions, multiplications, divisions, roots of any order). Thus, every eigensolver must have an iterative component.

Nevertheless some of the methods are called *direct solvers*. As with linear systems, direct eigensolvers *transform* the matrix to obtain the eigensystem, whereas (purely) *iterative solvers* work with the original matrix and try to extract selected eigenvalues and eigenvectors from appropriate low-dimensional subspaces of $\mathbb{R}^n$. Direct solvers are the methods of choice to compute a significant portion of the eigendecomposition for small to medium-sized matrices ($n \lesssim 5000$, say), while iterative solvers are used when only a few (up to 100, say) eigenpairs of very large — and typically sparse — matrices are sought. In this article we discuss only the direct solvers; iterative methods are treated in another contribution.

2 Eigenvalue Computations: How Good, How Fast, and How?

At the beginning of this section we introduce some more notations and definitions. Then we explain why eigenvalue solvers cannot compute the exact eigenvalues and eigenvectors of a matrix, and what kind of accuracy a "good" algorithm can achieve. Issues related to minimizing the computing time on today's high-performance computers are discussed subsequently. Finally, two basic tools for eigenvalue computations are introduced, namely similarity transformations that make the matrix in some way easier to handle, and deflation, which helps to break the problem into smaller ones whose solution is much cheaper.

2.1 More Notation

For contiguous portions of vectors and matrices we use the Matlab-style[30] colon notation: $\mathbf{x}(i_1 : i_2)$ denotes the length-$(i_2 - i_1 + 1)$ vector consisting of $\mathbf{x}$'s entries i_1 through i_2, and $\mathsf{A}(i_1 : i_2, j_1 : j_2)$ is the $(i_2 - i_1 + 1) \times (j_2 - j_1 + 1)$ matrix containing rows $i_1, \ldots, i_2$ of A's columns $j_1, \ldots, j_2$. An isolated colon stands for the whole index range in the respective direction. Thus, $\mathsf{A}(5, :)$ denotes the fifth row of A, whereas $\mathsf{A}(:, 3 : 7)$ contains columns 3 through 7 of the matrix.

The "size" of vectors and matrices is measured with *norms*. For our purposes, the *Euclidean norm*

$$\|\mathbf{x}\|_2 := \sqrt{\mathbf{x}^\top \mathbf{x}} = \sqrt{\sum_{i=1}^{n} \mathbf{x}(i)^2}$$

plays the dominant rôle. Its associated matrix norm is the *spectral norm*

$$\|\mathsf{A}\|_2 := \max\left\{ \frac{\|\mathsf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \ : \ \mathbf{x} \neq \mathbf{0} \right\} \qquad (= \max\{|\lambda_1|, |\lambda_n|\} \text{ if } \mathsf{A} \text{ is symmetric}) \ .$$

Since computing the spectral norm of a matrix is very expensive, often the *Frobenius norm*

$$\|\mathsf{A}\|_F := \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} \mathsf{A}(i,j)^2}$$

is used instead. Another important norm is the *maximum norm*

$$\|\mathbf{x}\|_\infty := \max\{|\, \mathbf{x}(i)| \ : \ i = 1, \ldots, n\} \ .$$

For iterative processes the *order of convergence* indicates how fast the desired values are approached. Let $\|\cdot\|$ denote a vector norm. A sequence of vectors $(\mathbf{x}_k)_{k\geq 0}$ converges *linearly* to a limit $\mathbf{x}^*$ if there is some constant $0 < c < 1$ such that $\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq c \cdot \|\mathbf{x}_k - \mathbf{x}^*\|$ for all k. This means that a constant number of steps is required to obtain one additional correct digit in the approximation $\mathbf{x}_k$. The sequence converges *quadratically* (or *cubically*) if $\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq C \cdot \|\mathbf{x}_k - \mathbf{x}^*\|^2$ (or $\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq C \cdot \|\mathbf{x}_k - \mathbf{x}^*\|^3$) for some $C > 0$. With quadratic and cubic convergence, the number of correct digits is *doubled* or *tripled* in each step. Thus $\mathbf{x}^*$ is approached very fast, once we have come somewhere close to it.

2.2 Accuracy Issues

Two obstacles prevent us from getting the *exact* eigenvalues and eigenvectors of a matrix on a computer.

First, in general the computation of the eigensystem involves an iterative process, which must be interrupted at some point before the correct values are reached, resulting in a so-called *truncation error*.

But even if this were not the case, the eigenvalues typically cannot be stored exactly in the finite number of memory cells that are allocated for each "real" number. The same holds for the results of most intermediate results. E.g., the IEEE standard 754[2] represents double-precision numbers with 64 bits, twelve of them encoding the sign and magnitude, and the remaining 52 bits holding the most significant binary digits of the number — corresponding to roughly 16 decimal places accuracy.

The best we may reasonably expect is that the computed result $\mathsf{fl}(\alpha \circ \beta)$ of each operation, where $\circ$ stands for addition, subtraction, multiplication, or division, is the floating-point number that is closest to the exact result $\alpha \circ \beta$, that is,

$$\mathsf{fl}(\alpha \circ \beta) = (\alpha \circ \beta) \cdot (1 + \epsilon) \ , \tag{3}$$

where ϵ is some tiny quantity depending on the operation $\circ$ and on the operands α and β. If the arithmetic of the computer conforms to the IEEE standard (this is true for all recent workstations and personal computers) then the quantities ϵ in Eq. (3) are guaranteed to be below some constant bound ε, the so-called *machine epsilon*. In double-precision, $\varepsilon \approx 2.22 \cdot 10^{-16}$.

But even if each single operation produces only a tiny relative error, a badly chosen (though mathematically correct) sequence of operations may yield an utterly wrong final result. In addition it is often very difficult to derive good estimates for the error of the computed quantity, the so-called *forward error*.

A much simpler technique for analyzing the behaviour of the algorithms was invented by Wilkinson[39]. Roughly speaking, his *backward error* analysis tries to put the blame for the rounding errors on the initial data. In the context of eigenvalue computations this means proving that the computed eigenvalues $\mathsf{fl}(\lambda_i)$ are the *exact* eigenvalues not of the original matrix A, but of a (slightly) perturbed matrix $\mathsf{A}+\Delta\mathsf{A}$. An eigenvalue algorithm is called *(backward) stable* if the perturbation $\Delta\mathsf{A}$ is very small compared to A:

$$\|\Delta\mathsf{A}\|_2 = \mathcal{O}(\varepsilon) \cdot \|\mathsf{A}\|_2 \ , \tag{4}$$

where $\mathcal{O}(\varepsilon)$ stands for a "small multiple" of the machine epsilon that may grow with a low-degree polynomial in the matrix dimension, like $4n\varepsilon$ or $2n^2\varepsilon$.

In order to derive bounds for the errors $|\mathsf{fl}(\lambda_i) - \lambda_i|$ (λ_i being the exact eigenvalues of A), the backward error analysis must be complemented by *perturbation analysis*, which investigates how much the eigenvalues can change if the matrix is perturbed. For the symmetric case, the following simple bound holds.

Theorem[18]. *Let* A *and* $\tilde{\mathsf{A}}$ *be symmetric matrices with eigenvalues* $\lambda_1 \le \ldots \le \lambda_n$ *and* $\tilde{\lambda}_1 \le \ldots \le \tilde{\lambda}_n$*, respectively. Then*

$$|\tilde{\lambda}_i - \lambda_i| \le \|\tilde{\mathsf{A}} - \mathsf{A}\|_2 \ . \tag{5}$$

The accumulated changes of all eigenvalues can be estimated with the Frobenius norm.

Theorem (Wielandt-Hoffman)[18]. *Let* A *and* $\tilde{\mathsf{A}}$ *be symmetric matrices with eigenvalues* $\lambda_1 \le \ldots \le \lambda_n$ *and* $\tilde{\lambda}_1 \le \ldots \le \tilde{\lambda}_n$*, respectively. Then*

$$\sqrt{\sum_{i=1}^{n} (\tilde{\lambda}_i - \lambda_i)^2} \le \|\tilde{\mathsf{A}} - \mathsf{A}\|_F \ .$$

The changes of eigenvectors corresponding to *simple* eigenvalues can be bounded as follows.

Theorem[11]. *Let* A *and* $\tilde{\mathsf{A}}$ *be symmetric matrices with eigenvalues* $\lambda_1 \le \ldots \le \lambda_n$ *and* $\tilde{\lambda}_1 \le \ldots \le \tilde{\lambda}_n$ *and associated orthonormal eigenvectors* $\mathbf{q}_1, \ldots, \mathbf{q}_n$ *and* $\tilde{\mathbf{q}}_1, \ldots, \tilde{\mathbf{q}}_n$*, respectively. Let* θ_i *denote the (acute) angle between* $\tilde{\mathbf{q}}_i$ *and* $\mathbf{q}_i$*. Then*

$$\frac{1}{2}\sin 2\theta_i \le \frac{\|\tilde{\mathsf{A}} - \mathsf{A}\|_2}{\min\{|\lambda_j - \lambda_i| \ : \ j \ne i\}} \qquad \textit{if} \quad \lambda_j \ne \lambda_i \ \textit{for all} \ j \ne i \ . \tag{6}$$

Therefore the direction of an eigenvector $\mathbf{q}_i$ changes only little, provided that the matrix itself changes very little and that the associated eigenvalue λ_i is simple and well separated from the remaining eigenvalues. Note that when θ_i is small then $\frac{1}{2}\sin 2\theta_i \approx \sin\theta_i \approx \theta_i$.

Eqs. (4), (5), and (6) imply that stable algorithms can compute the eigenvalues of a symmetric matrix A with an error $\mathcal{O}(\varepsilon)\cdot\|\mathsf{A}\|_2$ and the eigenvectors corresponding to well-separated eigenvalues with an error $\mathcal{O}(\varepsilon)$. For the large eigenvalues

($|\lambda_i| \approx \|\mathsf{A}\|_2$) this means that the computed values will be correct except for a few digits at the end. For very small eigenvalues ($|\lambda_i| \approx \mathcal{O}(\varepsilon)\|\mathsf{A}\|_2$), however, very few correct figures can be guaranteed. Indeed, even the magnitude and the sign of the computed eigenvalue may be wrong.

In general, the bounds given in Eqs. (5) and (6) cannot be improved much. For particular classes of matrices, however, the dependence of the eigenvalues on the matrix elements is much stronger: Then *relative* perturbations of the matrix (i.e., each matrix entry is perturbed proportionally to its magnitude) lead to relative eigenvalue perturbations. In these situations appropriate algorithms can compute almost all digits even of extremely tiny eigenvalues[4].

2.3 Performance Issues

Among the stable methods we would like to select one that takes the least time to determine the desired eigenvalues and eigenvectors. The computing time depends on several factors, most notably on the number of operations, on the rate at which these operations can be performed, and on the potential for exploiting parallelism.

In the context of direct eigensolvers, the overall work is adequately captured by counting only the floating-point operations (flop: additions, subtractions, multiplications, divisions, and square roots) and ignoring everything else, like index manipulations etc.

The *performance* (or *execution rate*) of an algorithm is measured in Mflop/s (millions of flop per second). On today's machines with processors running at hundreds of MHz and main memories being almost an order of magnitude slower, the performance is mainly determined by the "data re-use factor" r, which is the number of operations performed, divided by the amount of data that are moved between the main memory and the processor (more precisely, between main memory and the caches, which are small, but fast memory buffers running almost at full processor speed). This issue is best explained with the different levels of the *BLAS* (*Basic Linear Algebra Subprograms*)[13,14,28].

The BLAS define a set of subroutines for performing some simple recurring tasks in linear algebra computations. To consider just three of these routines, the (double-precision) function `DDOT` returns the scalar product $\mathbf{x}^\top\mathbf{y}$ of two vectors. This routine does $2n-1$ flop and reads $2n$ elements from memory (the entries of the two vectors), yielding $r \approx 1$. Another routine, `DGEMV`, computes a matrix–vector product with roughly $2n^2$ flop and n^2 accesses to memory, thus $r \approx 2$. And finally, the routine for computing matrix–matrix products, `DGEMM`, requires about $4n^2$ memory accesses to do $2n^3$ flop, resulting in a much higher ratio $r \approx n/2$. (The "level 1 BLAS" do order-of-n flop on order-of-n data, the level 2 routines do order-of-n^2 flop on order-of-n^2 data, and the level 3 routines do order-of-n^3 flop on order-of-n^2 data.) As can be seen from Table 1, a higher re-use factor can significantly improve the performance. (Note that only the level 3 routine comes anywhere close to the processor's peak performance of 266Mflop/s.)

Therefore the algorithms should be (re)structured in such a way that a major part of their operations can be done with level 3 routines, even if the overall number of operations is slightly increased. Algorithms with this property are often called

Table 1. Performance of selected operations on a 266MHz PentiumII. All vectors had length $n = 1000$, all matrices were n-by-n.

Operation	BLAS Routine	re-use factor r	Mflop/s
$\alpha := \mathbf{x}^\top \mathbf{y}$	`DDOT` (BLAS 1)	1	36.1
$\mathbf{y} := \alpha \mathsf{A}\mathbf{x} + \beta\mathbf{y}$	`DGEMV` (BLAS 2)	2	61.1
$\mathsf{C} := \alpha\mathsf{A}\mathsf{B} + \beta\mathsf{C}$	`DGEMM` (BLAS 3)	$n/2$	195.2

blocked algorithms.

Modern high-performance computers typically feature multiple processors. To fully utilize the potential power of these machines, *parallelism* must be exploited, i.e., the work (and in most cases the data, too) must be distributed among the processors. In particular, the algorithm must contain enough *independent* operations to keep all the processors busy. In addition, a processor cannot work completely on its on, but from time to time it must *synchronize* with other processors in order to exchange information. As synchronization causes significant administrative overhead, one should strive for so-called *coarse-grained parallelism*, where synchronizations occur rarely, thus allowing each processor to do a lot of "useful" operations between them — as opposed to *fine-grained parallelism* with frequent synchronizations and just a few flop in between.

Algorithms that are based mainly on level 3 operations lend themselves in a natural way to coarse-grained parallelism. This is another argument in favor of blocking, besides the fact that these algorithms also achieve high per-node computing performance.

2.4 Similarity Transformations

Almost every direct method for computing eigensystems makes use of the fact that eigenvalues are invariant under *similarity transformations* $\mathsf{A} \mapsto \mathsf{X}^{-1}\mathsf{A}\mathsf{X} =: \tilde{\mathsf{A}}$, where X denotes an arbitrary non-singular matrix. More precisely, if $(\lambda, \mathbf{x})$ is an eigenpair of A then $\mathsf{A}\mathbf{x} = \mathbf{x}\lambda$, and hence

$$(\mathsf{X}^{-1}\mathsf{A}\mathsf{X}) \cdot (\mathsf{X}^{-1}\mathbf{x}) = (\mathsf{X}^{-1}\mathbf{x}) \cdot \lambda \ .$$

Thus $(\lambda, \mathsf{X}^{-1}\mathbf{x})$ is an eigenpair of the transformed matrix $\tilde{\mathsf{A}}$.

Most eigensolvers exploit this property in the following way: First, a suitable similarity transformation $\mathsf{A} \mapsto \mathsf{X}^{-1}\mathsf{A}\mathsf{X} =: \tilde{\mathsf{A}}$ reduces A to a matrix $\tilde{\mathsf{A}}$ that is in some way more easily handled (see Sections 4 and 5), then the desired eigenvalues $\tilde{\lambda}_i$ and associated eigenvectors $\tilde{\mathbf{x}}_i$ of $\tilde{\mathsf{A}}$ are computed, and finally these are back-transformed into the eigenvalues $\lambda_i = \tilde{\lambda}_i$ and eigenvectors $\mathbf{x}_i = \mathsf{X} \cdot \tilde{\mathbf{x}}_i$ of the original matrix A.

Whenever possible, *orthogonal* similarity transformations $\mathsf{A} \mapsto \mathsf{Q}^\top\mathsf{A}\mathsf{Q}$ (Q being an orthogonal matrix) should be used, for three reasons. First, the inverse $\mathsf{Q}^{-1} = \mathsf{Q}^\top$ is readily available. Second, these transformations preserve symmetry: If A is symmetric then $\mathsf{Q}^\top\mathsf{A}\mathsf{Q}$ is symmetric, too. And third, orthogonal transformations are very *stable* in the sense that they induce a small backward error. It is

a nice feature of the symmetric (standard) eigenvalue problem that in principle all computations can be done with orthogonal transformations.

2.5 Deflation

Suppose that we have found an orthogonal matrix Q such that

$$\mathsf{Q}^\top \mathsf{A}\mathsf{Q} = \begin{pmatrix} \mathsf{A}_1 & 0 \\ 0 & \mathsf{A}_2 \end{pmatrix} \tag{7}$$

with $\mathsf{A}_1 \in \mathbb{R}^{k\times k}$ and $\mathsf{A}_2 \in \mathbb{R}^{(n-k)\times(n-k)}$. Then we have $\mathrm{spec}(\mathsf{A}) = \mathrm{spec}(\mathsf{A}_1) \cup \mathrm{spec}(\mathsf{A}_2)$, and therefore the eigenvalues of A can be obtained by computing those of the smaller matrices A_i. This reduction of the size of the matrices that must be transformed further — the so-called *deflation* — leads to considerable flop savings.

The eigenvectors of A may also be obtained from those of the smaller matrices A_i. To this end we partition Q accordingly: $\mathsf{Q} = (\,\mathsf{Q}_1 \,|\, \mathsf{Q}_2\,)$, where Q_1 consists of the first k columns in Q. Then Eq. (7) is equivalent to $\mathsf{A}\mathsf{Q}_i = \mathsf{Q}_i\mathsf{A}_i$, $i = 1, 2$. Given an eigenpair $(\tilde{\lambda}, \tilde{\mathbf{x}})$ of one of the smaller matrices A_i, we thus obtain

$$\mathsf{A}\cdot(\mathsf{Q}_i\tilde{\mathbf{x}}) = \mathsf{Q}_i\cdot(\mathsf{A}_i\tilde{\mathbf{x}}) = (\mathsf{Q}_i\tilde{\mathbf{x}})\cdot\lambda\ ,$$

which means that $(\lambda, \mathsf{Q}_i\tilde{\mathbf{x}})$ is an eigenpair of A.

3 Tools of the Trade: Basic Orthogonal Transformations

Orthogonal matrices play an important rôle in the computation of eigenvalues and eigenvectors (e.g., the eigenvector matrix Q in the eigendecomposition (2) is orthogonal). These orthogonal matrices are built up from two basic types, rotations and Householder transformations, which are introduced in this section. We also discuss techniques for improving the performance by applying these basic transformations in a blocked fashion.

3.1 Rotations

An (i, j) *rotation* in $\mathbb{R}^n$ is given by the matrix

$$\mathsf{R} = \mathsf{R}(i,j,\theta) = \begin{pmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & c & & & s & & \leftarrow i \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & -s & & & c & & \leftarrow j \\ & & & & & & & 1 & \\ & & & & & & & & \ddots \\ & & & & & & & & & 1 \end{pmatrix} \in \mathbb{R}^{n\times n}, \tag{8}$$

where $c = \cos\theta$, $s = \sin\theta$, and all the remaining entries of R are zero. It is easy to verify that R is orthogonal.

Applying R to some column vector $\mathbf{x} \in \mathbb{R}^n$ via $\mathbf{x} \mapsto \mathsf{R}^\top \cdot \mathbf{x}$ corresponds to a counter-clockwise rotation by the angle θ in the (i, j) coordinates plane. The same is achieved for row vectors $\mathbf{x}^\top$ via $\mathbf{x}^\top \mapsto \mathbf{x}^\top \cdot \mathsf{R}$. In both cases only the ith and jth entries of $\mathbf{x}$ change.

By letting

$$c = \frac{\mathbf{x}(i)}{\sqrt{\mathbf{x}(i)^2 + \mathbf{x}(j)^2}} \qquad \text{and} \qquad s = \frac{-\mathbf{x}(j)}{\sqrt{\mathbf{x}(i)^2 + \mathbf{x}(j)^2}} \tag{9}$$

in the above transformations we can zero the jth entry of $\mathbf{x}$. (Note that only the cosine and the sine of the rotation angle θ are present in the formulas (8) and (9); there is no need to compute the angle itself.)

Applying the rotation to a matrix $\mathsf{A} \in \mathbb{R}^{n \times m}$ from the left, $\mathsf{A} \mapsto \mathsf{R}^\top \cdot \mathsf{A}$, affects only the ith and jth row of A,

$$\begin{aligned} \mathsf{tmp} &:= c \cdot \mathsf{A}(i,:) + s \cdot \mathsf{A}(j,:) \\ \mathsf{A}(j,:) &:= -s \cdot \mathsf{A}(i,:) + c \cdot \mathsf{A}(j,:) \\ \mathsf{A}(i,:) &:= \mathsf{tmp} \end{aligned}$$

whereas $\mathbb{R}^{m \times n} \ni \mathsf{A} \mapsto \mathsf{A} \cdot \mathsf{R}$ does the same with columns i and j. In either case, the transformation takes $4m$ multiplications and $2m$ additions. If a symmetric matrix A is transformed from both sides, $\mathsf{A} \mapsto \mathsf{R}^\top \cdot \mathsf{A} \cdot \mathsf{R}$, then the resulting matrix is again symmetric. Therefore the transformations must be applied only to the lower (or upper) triangle of A, thereby approximately halving the overall cost to $6n$ flop.

Rotations are very stable. It can be shown[40] that the backward error corresponding to a transformation $\mathsf{A} \mapsto \mathsf{R}^\top \cdot \mathsf{A}$ is bounded by $6\varepsilon\|\mathsf{A}\|_F$.

There are variants of rotations — called *fast Givens rotations*[35] — that require only half as many multiplications as the "ordinary" rotations described above, resulting in $4m$ flop for the transformation $\mathsf{A} \mapsto \mathsf{R}^\top\mathsf{A}$. These savings are achieved by an appropriate (implicit) scaling of the matrix R.

In rotation–based algorithms, parallelism can be exploited in two ways. Either the work of each rotation is split among the processors (yielding a rather fine-grained parallelism), or in some situations several rotations in disjoint planes can be applied simultaneously[24].

3.2 Householder Transformations

While rotations may be used to introduce single zeros in a vector or a matrix, Householder transformations can zero out more than one entry.

A length-n *Householder transformation* is described by the matrix

$$\mathsf{H} = \mathsf{H}(\mathbf{y}) = \mathsf{I} - \mathbf{y}\tau\mathbf{y}^\top \in \mathbb{R}^{n \times n}, \tag{10}$$

where $\mathbf{y} \in \mathbb{R}^n$ is an arbitrary vector and

$$\tau = \begin{cases} 0, & \text{if } \mathbf{y} = 0 \\ 2/\|\mathbf{y}\|_2^2, & \text{otherwise} \end{cases} \in \mathbb{R}.$$

A short computation reveals that H is orthogonal and that the transformations $\mathbf{x} \mapsto \mathsf{H}^\top \cdot \mathbf{x}$ and $\mathbf{x}^\top \mapsto \mathbf{x}^\top \cdot \mathsf{H}$ correspond to a reflection of $\mathbf{x}$ at the (hyper)plane perpendicular to $\mathbf{y}$, thus inverting $\mathbf{x}$'s component in direction $\mathbf{y}$. (Note that H is symmetric; thus we might write H instead of $\mathsf{H}^\top$.)

To zero out the entries $i+1, \ldots, j$ of a given vector $\mathbf{x} \in \mathbb{R}^n$, we choose

$$\mathbf{y} = (0, \ldots, 0, \mathbf{x}(i) \pm \|\mathbf{x}(i+1:j)\|_2, \mathbf{x}(i+1), \ldots, \mathbf{x}(j), 0, \ldots, 0)^\top , \tag{11}$$

where "$\pm$" is chosen equal to the sign of $\mathbf{x}(i)$ in order to avoid loss of precision through cancellation. This choice gives

$$\mathsf{H}^\top \cdot \mathbf{x} = (\mathbf{x}(1), \ldots, \mathbf{x}(i-1), \mp\|\mathbf{x}(i:j)\|_2, 0, \ldots, 0, \mathbf{x}(j+1), \ldots, \mathbf{x}(n))^\top .$$

In particular, only the entries $i, \ldots, j$ of $\mathbf{x}$ are affected by the transformation.

Analogously the transformations $\mathbb{R}^{n \times m} \ni \mathsf{A} \mapsto \mathsf{H}^\top \cdot \mathsf{A}$ and $\mathbb{R}^{m \times n} \ni \mathsf{A} \mapsto \mathsf{A} \cdot \mathsf{H}$ affect only rows (columns, resp.) $i, \ldots, j$ of A: Making use of Eqs. (10) and (11), in the first case one computes

$$\mathbf{z}^\top := \tau \cdot \mathbf{y}(i:j)^\top \cdot \mathsf{A}(i:j,:) \in \mathbb{R}^{1 \times m} \tag{12}$$

$$\mathsf{A}(i:j,:) := \mathsf{A}(i:j,:) - \mathbf{y}(i:j) \cdot \mathbf{z}^\top . \tag{13}$$

The matrix–vector product (12) and the rank-1-update (13) each require approximately $2\ell m$ operations ($\ell = j - i + 1$ is the "active length" of the transformation), for a total of $4\ell m$ flop. (Note that one *never* computes $\mathsf{H}^\top \cdot \mathsf{A}$ as a matrix–matrix product, which would require approximately $2\ell^2 m$ flop.)

As with rotations, symmetry can be exploited to save operations. The two-sided transformation $\mathbb{R}^{n \times n} \ni \mathsf{A} \mapsto \mathsf{H}^\top \cdot \mathsf{A} \cdot \mathsf{H}$ of a symmetric matrix A is performed as follows:

$$\mathbf{z} := \mathsf{A} \cdot \mathbf{y} \cdot \tau \tag{14}$$

$$\mathbf{v} := \mathbf{z} - \mathbf{y} \cdot \frac{\tau(\mathbf{y}^\top \mathbf{z})}{2}$$

$$\mathbf{A} := \mathbf{A} - \mathbf{y} \cdot \mathbf{v}^\top - \mathbf{v} \cdot \mathbf{y}^\top . \tag{15}$$

Due to the symmetry, only one triangle of A must be updated in the symmetric rank-2-update (15), provided that only one triangle is used in the symmetric matrix–vector product (14). Again, the two-sided transformation of A can be effected at the same cost as a one-sided transformation of a non-symmetric matrix, that is, $4n^2$ flop.

Householder transformations are very stable, too[20]. The backward error corresponding to a transformation $\mathsf{A} \mapsto \mathsf{H}^\top \cdot \mathsf{A}$ is bounded by $cn\varepsilon\|\mathsf{A}\|_F$, where c is some small constant.

With Householder transformations, parallelism is exploited by working on a distributed matrix A and doing each of the steps (14) and (15) in parallel.

3.3 Blocked Householder Transformations

Applying single rotations or Householder transformations does not allow using any level 3 BLAS. The situation changes if a *sequence* of transformations must be applied to the same matrix.

Bischof and Van Loan[6] discovered that the product $\mathsf{Q} = \mathsf{H}_1 \cdot \ldots \cdot \mathsf{H}_k$ of k length-n Householder transformations $\mathsf{H}_j = \mathsf{I} - \mathbf{y}_j \tau_j \mathbf{y}_j^\top$ can be written in the form

$$\mathsf{Q} = \mathsf{I} - \mathsf{W}\mathsf{Y}^\top \qquad (\textit{WY representation}), \tag{16}$$

where W and Y are suitable n-by-k matrices.

For $k = 1$ the representation (16) is obviously valid with $\mathsf{W} = \mathbf{y}_1 \tau_1$ and $\mathsf{Y} = \mathbf{y}_1$. To proceed from k to $k+1$, we assume that the matrices W and Y in (16) are known and that $\tilde{\mathsf{Q}} = \mathsf{H}_1 \cdot \ldots \cdot \mathsf{H}_{k+1} = \mathsf{Q} \cdot \mathsf{H}_{k+1}$. Then a short calculation reveals that $\tilde{\mathsf{Q}} = \mathsf{I} - \tilde{\mathsf{W}}\tilde{\mathsf{Y}}^\top$ with the n-by-$(k+1)$ matrices

$$\tilde{\mathsf{W}} = (\,\mathsf{W} \,|\, \mathsf{Q}\mathbf{y}_{k+1}\tau_{k+1}\,) \qquad \text{and} \qquad \tilde{\mathsf{Y}} = (\,\mathsf{Y} \,|\, \mathbf{y}_{k+1}\,)\ .$$

Thus each additional transformation H_j requires only appending one new column to W and Y. Note that in particular the columns of Y are just the vectors $\mathbf{y}_j$ defining the transformations H_j.

Given the representation (16), applying the k transformations to a matrix A amounts to two matrix–matrix products,

$$\mathsf{H}_k^\top \cdot \ldots \cdot \mathsf{H}_1^\top \cdot \mathsf{A} = \mathsf{Q}^\top \cdot \mathsf{A} = \mathsf{A} - \mathsf{Y} \cdot (\mathsf{W}^\top \cdot \mathsf{A})\ ,$$

which performs significantly better than applying the single Householder transformations and requires only marginally more operations.

Later, Schreiber and Van Loan[36] refined the WY representation to

$$\mathsf{Q} = \mathsf{I} - \mathsf{Y}\mathsf{T}\mathsf{Y}^\top \qquad (\textit{compact WY representation}), \tag{17}$$

where Y is again n-by-k and T is an upper triangular k-by-k matrix. Here, the case $k = 1$ is covered by setting $\mathsf{Y} = \mathbf{y}_1$ and $\mathsf{T} = \tau_1$, and the step from k to $k+1$ is done by letting

$$\tilde{\mathsf{Y}} = (\,\mathsf{Y} \,|\, \mathbf{y}_{k+1}\,) \qquad \text{and} \qquad \tilde{\mathsf{T}} = \left(\begin{array}{c|c} \mathsf{T} & -\tau_{k+1}\mathsf{T}\mathbf{y}_{k+1} \\ \hline \mathbf{0}^\top & \tau_{k+1} \end{array}\right)\ .$$

The compact WY representation costs significantly less additional storage than the original WY representation does (T instead of W), but applying the transformations in the compact representation requires a third matrix–matrix product.

Under favorable conditions, applying a sequence of rotations to a matrix can also be organized in such a way that most of the work is done in matrix–matrix products[27].

4 Phase I: Reduction to Tridiagonal Form

Most eigenvalue solvers first reduce the symmetric matrix A to a symmetric tridiagonal matrix T (i.e., $\mathsf{T}(i,j) = 0$ whenever $|i-j| > 1$) in order to make the ensuing iterative process simpler and cheaper. This section first presents the standard algorithm for tridiagonalizing a full matrix, as well as a blocked variant of the algorithm. Then the back-transformation of the eigenvectors is discussed, which is a final step after computing the eigenvalues and eigenvectors of the tridiagonal matrix T (cf. Section 5). Specialized reduction methods for banded matrices and generalized eigenvalue problems are described at the end of the section.

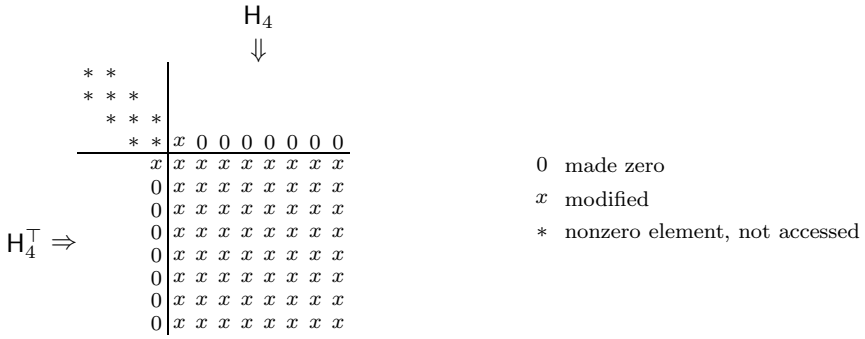


Figure 1. The fourth step in the reduction of a 12-by-12 matrix to tridiagonal form.

4.1 Householder Tridiagonalization

The standard algorithm for reducing an n-by-n symmetric full matrix to tridiagonal form, $\mathsf{A} \mapsto \mathsf{Q}_1^\top \mathsf{A}\mathsf{Q}_1 = \mathsf{T}$, proceeds in $n-2$ steps:

$$\begin{aligned} \mathsf{A} =: \mathsf{A}_0 \;\mapsto\; \mathsf{H}_1^\top \mathsf{A}_0 \mathsf{H}_1 =: \mathsf{A}_1 \;\mapsto\; \mathsf{H}_2^\top \mathsf{A}_1 \mathsf{H}_2 =: \mathsf{A}_2 \;\mapsto \\ \mapsto\; \ldots \;\mapsto\; \mathsf{H}_{n-2}^\top \mathsf{A}_{n-3} \mathsf{H}_{n-2} =: \mathsf{A}_{n-2} = \mathsf{T}\ , \end{aligned} \tag{18}$$

the kth step transforming column and row k to the desired shape with a suitable Householder transformation H_k, cf. Figure 1 and Algorithm 1.

Algorithm 1 Householder reduction to tridiagonal form.

for $k = 1$ **to** $n-2$
  determine a Householder transformation $\mathsf{H}_k = \mathsf{I} - \mathbf{y}_k \tau_k \mathbf{y}_k^\top$ that reduces $\mathsf{A}(k+1:n,k)$ to the form $(x, 0, \ldots, 0)^\top$
  $\mathbf{z}_k := \mathsf{A}\mathbf{y}_k \tau_k$
  $\mathbf{v}_k := \mathbf{z}_k - \mathbf{y}_k \cdot (\tau_k(\mathbf{y}_k^\top \mathbf{z}_k)/2)$
  $\mathsf{A} := \mathsf{A} - \mathbf{y}_k \cdot \mathbf{v}_k^\top - \mathbf{v}_k \cdot \mathbf{y}_k^\top$

Note that only the submatrix $\mathsf{A}(k+1:n, k+1:n)$ (and, due to symmetry, only one triangle of this matrix) is modified in the last line of the algorithm. Thus the reduction requires approximately $\frac{4}{3}n^3$ flop, which are done mainly within the level 2 BLAS.

Before Householder developed the method just described, the tridiagonalization had been done with a rotation–based algorithm due to Givens[40]. Householder's approach is superior for full matrices because it requires fewer operations, whereas Givens' algorithm can make better use of sparsity.

4.2 Blocked Householder Tridiagonalization

It is not necessary to build all the intermediate matrices A_k from Eq. (18) explicitly[15]. Instead, they can be represented in the so-called *factored form*

$$\mathsf{A}_s = \mathsf{A}_0 - \mathbf{y}_1\mathbf{v}_1^\top - \mathbf{v}_1\mathbf{y}_1^\top - \ldots - \mathbf{y}_s\mathbf{v}_s^\top - \mathbf{v}_s\mathbf{y}_s^\top = \mathsf{A}_0 - \mathsf{Y}_s\mathsf{V}_s^\top - \mathsf{V}_s\mathsf{Y}_s^\top \ , \qquad (19)$$

where the vectors $\mathbf{y}_k$ and $\mathbf{v}_k$ are defined as in Algorithm 1, and $\mathsf{Y}_s = (\,\mathbf{y}_1 \,|\, \ldots \,|\, \mathbf{y}_s\,)$ and $\mathsf{V}_s = (\,\mathbf{v}_1 \,|\, \ldots \,|\, \mathbf{v}_s\,)$ are n-by-s matrices. If the matrices A_k are built only every n_bth step then we arrive at Algorithm 2 (lines 4 through 6 and line 8 of the algorithm reflect the fact that $\mathbf{v}_k$ and $\mathbf{z}_k$ are computed from A_{k-1} and not from A_{s-1}, which is currently held in A).

Algorithm 2 Blocked Householder tridiagonalization.

for $s = 1$ **to** $n-2$ **step** n_b
  $\mathsf{Y} := ()$, $\mathsf{V} := ()$ { matrices with 0 columns }
  for $k = s$ **to** $\min\{s+n_b-1, n-2\}$
    if $k > s$
      compute the kth column of A_{k-1} according to Eq. (19):
        $\mathsf{A}(:,k) := \mathsf{A}(:,k) - \mathsf{Y}\mathsf{V}^\top(:,k) - \mathsf{V}\mathsf{Y}^\top(:,k)$
    determine the Householder transformation H_k as in Algorithm 1
    $\mathbf{z}_k := \mathsf{A}\mathbf{y}_k\tau_k - \mathsf{Y}(\mathsf{V}^\top\mathbf{y}_k\tau_k) - \mathsf{V}(\mathsf{Y}^\top\mathbf{y}_k\tau_k)$
    compute $\mathbf{v}_k$ as in Algorithm 1
    $\mathsf{Y} := (\,\mathsf{Y}\,|\,\mathbf{y}_k\,)$, $\mathsf{V} := (\,\mathsf{V}\,|\,\mathbf{v}_k\,)$
  $\mathsf{A} := \mathsf{A} - \mathsf{Y}\mathsf{V}^\top - \mathsf{V}\mathsf{Y}^\top$ { after n_b steps rebuild A according to Eq. (19) }

Note that the rank-2-updates (last line in Algorithm 1) have been replaced with matrix–matrix products, whereas the symmetric matrix–vector products $\mathsf{A}\cdot\mathbf{y}$ in the computation of $\mathbf{z}_k$ persist. Therefore, the blocked algorithm does roughly one half of its $\frac{4}{3}n^3$ operations with level-3 BLAS while the remaining operations are still confined to the level-2 BLAS.

The portion of matrix–matrix operations can be further increased if the reduction to tridiagonal form is done in two phases[5]. First the matrix is reduced to banded form (almost completely with level 3 BLAS), and then the banded matrix is tridiagonalized (no level 3 BLAS, but significantly lower flop count than for the first phase). This approach typically outperforms the direct tridiagonalization if *no eigenvectors* are required.

4.3 Back-Transformation of the Eigenvectors

After the reduction of A to tridiagonal form, $\mathsf{A} \mapsto \mathsf{Q}_1^\top\mathsf{A}\mathsf{Q}_1 = \mathsf{T}$, all or selected eigenpairs $(\lambda_i, \mathbf{v}_i)$ of the tridiagonal matrix T are computed. While the eigenvalues λ_i of T are also eigenvalues of A, the associated eigenvectors must be back-transformed in order to obtain A's eigenvectors via $\mathbf{v}_i \mapsto \mathsf{Q}_1\cdot\mathbf{v}_i = \mathbf{q}_i$, cf. Section 2.4.

Let $\mathsf{H}_k = \mathsf{I} - \mathbf{y}_k\tau_k\mathbf{y}_k^\top$, $k = 1, \ldots, n-2$, be the Householder transformations that were used in the reduction $\mathsf{A} \mapsto \mathsf{T}$, i.e., $\mathsf{Q}_1 = \mathsf{H}_1 \cdot \ldots \cdot \mathsf{H}_{n-2}$, and let $\mathsf{V} =$

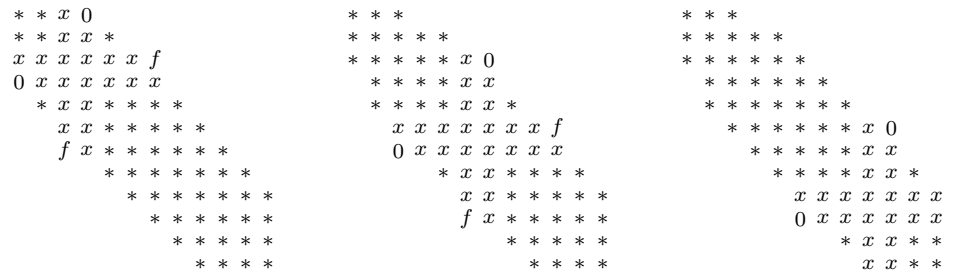


Figure 2. Rotations for zeroing the $(4,1)$ entry and for chasing the intermediate fill-in elements (denoted by f) in Schwarz' algorithm.

$(\,\mathbf{v}_1 \mid \ldots \mid \mathbf{v}_m\,)$ denote those eigenvectors of T that were computed and must be back-transformed. Then the corresponding eigenvectors of A are obtained as

$$(\,\mathbf{q}_1 \mid \ldots \mid \mathbf{q}_m\,) =: \mathsf{Q} = \mathsf{Q}_1 \cdot \mathsf{V} = \mathsf{H}_1 \cdot \ldots \cdot \mathsf{H}_{n-2} \cdot \mathsf{V}\ .$$

In contrast to the tridiagonal reduction, almost all of the roughly $2n^2m$ operations can be done with matrix–matrix products if we resort to the (compact or original) WY representation for applying $n_b > 1$ transformations at a time, cf. Algorithm 3.

Algorithm 3 Blocked back-transformation of selected eigenvectors.

$\mathsf{Q} := \mathsf{V}$
for $s = n-2$ **to** 1 **step** $-n_b$
  determine the compact WY representation for the next $n_b' = \min\{n_b, s\}$ transformations: $\mathsf{H}_{s-n_b'+1} \cdot \ldots \cdot \mathsf{H}_s = \mathsf{I} - \mathsf{YTY}^\top$
  apply these transformations via $\mathsf{Q} := \mathsf{Q} - \mathsf{YTY}^\top\mathsf{Q}$

Like any algorithm involving mainly products of large matrices, the back-transformation can be easily and efficiently parallelized.

4.4 Reduction of Banded Matrices

A symmetric matrix A is *banded* with *semibandwidth* b if $\mathsf{A}(i,j) = 0$ whenever $|i-j| > b$. For narrow-banded matrices ($b \ll n$), Algorithms 1 and 2 are not optimal because they completely destroy the sparsity.

Such matrices are reduced with a rotation–based algorithm by Schwarz[37]. To understand how this algorithm works we consider a 12-by-12 matrix with semibandwidth $b = 3$, see Figure 2.

First, the outmost entry $(4,1)$ in the first column is made zero with a rotation $\mathsf{A} \mapsto \mathsf{R}^\top\mathsf{AR}$ in the $(3,4)$ plane (left picture in Figure 2). This creates a new *fill-in* entry f at position $(7,3)$, just outside the band. Then a second rotation in the $(6,7)$ plane is used to remove the fill-in entry (center picture), only to have another fill element appear at position $(10,6)$, which in turn is removed by the next rotation,

and so on. Each rotation "chases" the fill element b positions down the band until the end of the matrix is reached and no further fill-in is created (right picture).

Then we can zero the next entry $(3,1)$ in the first column with a $(2,3)$ rotation, again followed by a sequence of rotations for chasing the fill-in. When the first column of A is reduced to tridiagonal form we repeat the procedure for the second column, and so on. The whole method is summarized in Algorithm 4.

Algorithm 4 Schwarz' algorithm for tridiagonalizing banded matrices.

for $j = 1$ **to** $n-2$ { proceed by columns }
  for $d = \min\{b, n-j\}$ **to** 2 **step** -1 { zero the entry in the dth subdiagonal }
  make $\mathsf{A}(j+d, j)$ zero with a suitable rotation in the $(j+d-1, j+d)$ plane
  while the most recent rotation created a fill-in entry at some position (k, ℓ)
    make this entry zero by a suitable rotation in the $(k-1, k)$ plane

If A's eigenvectors are needed, too, then in the banded case the orthogonal matrix Q_1 (cf. Section 4.3), which is the product of all rotations that are used for the reduction, is built explicitly *during* the reduction. This is achieved by applying each plane-$(k-1,k)$ rotation R not only to the banded matrix, $\mathsf{A} \mapsto \mathsf{R}^\top \mathsf{A} \mathsf{R}$, but also to the columns $k-1$ and k of an n-by-n matrix Q_1 via $\mathsf{Q}_1 \mapsto \mathsf{Q}_1 \mathsf{R}$, where Q_1 has been initialized as the identity matrix.

The reduction of A requires $6bn^2$ flop. If the eigenvectors of A must be computed then the costs for accumulating Q_1, $3n^3$ flop, by far dominate the reduction costs. In this method some parallelism can be exploited by applying several rotations simultaneously[24].

A more recent reduction algorithm is based on Householder transformations[32], each transformation affecting just b rows and columns of A. This method allows coarser-grained parallelism than Schwarz' algorithm[26], and the accumulation of Q_1 can be done in a blocked fashion[5].

4.5 Reduction of the Generalized Eigenvalue Problem

The symmetric generalized eigenvalue problem $\mathsf{A}\mathbf{q}_i = \mathsf{B}\mathbf{q}_i\lambda_i$ with a symmetric *positive definite* matrix B can be transformed into a symmetric standard eigenvalue problem as follows. Let

$$\mathsf{B} = \mathsf{L}\mathsf{L}^\top \ , \tag{20}$$

where L is a lower triangular matrix with positive diagonal entries, be the *Cholesky decomposition* of B. (Algorithms for computing the *Cholesky factor* L are given below.) Then the condition $\mathsf{A}\mathbf{q}_i = \mathsf{B}\mathbf{q}_i\lambda_i$ is equivalent to

$$(\mathsf{L}^{-1}\mathsf{A}\mathsf{L}^{-\top}) \cdot (\mathsf{L}^\top \mathbf{q}_i) = (\mathsf{L}^{-1}\mathsf{B}\mathsf{L}^{-\top}) \cdot (\mathsf{L}^\top \mathbf{q}_i) \cdot \lambda_i = (\mathsf{L}^\top \mathbf{q}_i) \cdot \lambda_i \ ,$$

where $\mathsf{L}^{-\top}$ is a shorthand for $(\mathsf{L}^\top)^{-1}$ $(= (\mathsf{L}^{-1})^\top)$. Therefore, an eigenpair $(\lambda_i, \mathbf{q}_i)$ of the generalized eigenvalue problem corresponds to the eigenpair $(\lambda_i, \mathsf{L}^\top \mathbf{q}_i)$ of the symmetric standard eigenvalue problem for the matrix $\mathsf{M} = \mathsf{L}^{-1}\mathsf{A}\mathsf{L}^{-\top}$. Thus we arrive at Algorithm 5, which takes approximately $12n^3$ flop for solving the

generalized eigenvalue problem. (Note that the explicit calculation of L^{-1} and $\mathsf{L}^{-\top}$ is avoided by solving triangular systems with multiple right-hand sides, which is implemented in the level 3 BLAS routine `DTRSM`.)

Algorithm 5 Generalized eigenvalue problem via Cholesky decomposition of B.

compute the Cholesky decomposition $\mathsf{B} = \mathsf{L}\mathsf{L}^\top$ { Algorithm 7 }
compute $\mathsf{M} = \mathsf{L}^{-1}\mathsf{A}\mathsf{L}^{-\top}$ by solving two triangular systems with multiple right-hand sides, $\mathsf{X}\mathsf{L}^\top = \mathsf{A}$ (for X), and $\mathsf{L}\mathsf{M} = \mathsf{X}$ (for M)
use blocked Householder tridiagonalization (Algorithm 2) and an algorithm from Section 5 to compute the eigendecomposition $\mathsf{M} = \tilde{\mathsf{Q}}\mathsf{\Lambda}\tilde{\mathsf{Q}}^\top$
compute the eigenvectors $\mathbf{q}_i = \mathsf{L}^{-\top}\tilde{\mathbf{q}}_i$ of the generalized problem by solving the triangular system $\mathsf{L}^\top\mathsf{Q} = \tilde{\mathsf{Q}}$ for Q

In general, the matrix M will be full even if A and B (and therefore B's Cholesky factor L, too) are *banded*. An algorithm by Crawford[9] avoids building M explicitly by interleaving its computation with the ensuing reduction to tridiagonal form. This leads to considerable flop and memory savings.

If B is ill-conditioned (i.e., its eigenvalues vary over many orders of magnitude) then severe loss of accuracy may happen because the backward error of Algorithm 5 grows with $\|\mathsf{B}^{-1}\|_2$. In this case the Cholesky decomposition should be replaced with the *eigendecomposition* $\mathsf{B} = \mathsf{S}\mathsf{\Delta}^2\mathsf{S}^\top = (\mathsf{S}\mathsf{\Delta})(\mathsf{S}\mathsf{\Delta})^\top$, where S is orthogonal and $\mathsf{\Delta}$ contains the roots of B's (positive!) eigenvalues. This leads to $\mathsf{M} = \mathsf{\Delta}^{-1}\mathsf{S}^\top\mathsf{A}\mathsf{S}\mathsf{\Delta}^{-1}$. In practice, the use of orthogonal matrices gives better results, albeit at higher cost.

The generalized eigenvalue problem with a matrix B that is not positive definite requires completely different techniques[18,34].

Let us finally give two algorithms for computing the Cholesky decomposition. Eq. (20) is equivalent to

$$\begin{aligned} \mathsf{B}(1,1) &= \mathsf{L}(1,1)\cdot\mathsf{L}(1,1)\ , \\ \mathsf{B}(2:n,1) &= \mathsf{L}(2:n,1)\cdot\mathsf{L}(1,1)\ , \text{ and} \\ \mathsf{B}(2:n,2:n) &= \mathsf{L}(2:n,2:n) + \mathsf{L}(2:n,1)\cdot\mathsf{L}(2:n,1)^\top\ , \end{aligned}$$

i.e., $\mathsf{L}(1,1) = \sqrt{\mathsf{B}(1,1)}$, $\mathsf{L}(2:n,1) = \mathsf{B}(2:n,1)/\mathsf{L}(1,1)$, and $\mathsf{L}(2:n,2:n)$ is the Cholesky factor of $\mathsf{B}(2:n,2:n) - \mathsf{L}(2:n,1)\cdot\mathsf{L}(2:n,1)^\top$. Resolving this recursion into a loop leads to Algorithm 6, in which the lower triangle of the matrix B is overwritten with the Cholesky factor L.

By replacing matrix entries $\mathsf{B}(i,j)$ with n_b-by-n_b blocks $\mathsf{B}[i,j] := \mathsf{B}((i-1)n_b+1 : in_b, (j-1)n_b+1 : jn_b)$ we arrive at Algorithm 7 ($N = \lceil n/n_b \rceil$ is the number of blocks).

Both algorithms require roughly $\frac{1}{3}n^3$ flop. The operations of Algorithm 7 are done almost exclusively with level 3 BLAS.

Algorithm 6 Cholesky decomposition.

for $k = 1$ **to** n
$\quad \mathsf{B}(k,k) := \sqrt{\mathsf{B}(k,k)}$
$\quad \mathsf{B}(k+1:n,k) := \mathsf{B}(k+1:n,k) \cdot \dfrac{1}{\mathsf{B}(k,k)}$
$\quad \mathsf{B}(k+1:n,k+1:n) := \mathsf{B}(k+1:n,k+1:n)$
$\qquad\qquad\qquad\qquad - \mathsf{B}(k+1:n,k) \cdot \mathsf{B}(k+1:n,k)^\top$

Algorithm 7 Blocked Cholesky decomposition.

for $k = 1$ **to** N
$\quad \mathsf{L}[k,k] :=$ Cholesky factor of $\mathsf{B}[k,k]$ { Algorithm 6, overwriting $\mathsf{B}[k,k]$ }
$\quad \mathsf{B}[k+1:N,k] := \mathsf{B}[k+1:N,k] \cdot \mathsf{L}[k,k]^{-\top}$
$\quad \mathsf{B}[k+1:N,k+1:N] := \mathsf{B}[k+1:N,k+1:N]$
$\qquad\qquad\qquad\qquad - \mathsf{B}[k+1:N,k] \cdot \mathsf{B}[k+1:N,k]^\top$

5 Phase II: Methods for Tridiagonal Matrices

Once the symmetric matrix A is reduced to tridiagonal form, the eigenvalues λ_i and the eigenvectors $\mathbf{v}_i$ of the tridiagonal matrix T must be found. There is a large variety of algorithms for solving this problem.

Since its invention in the 1960's, the QR iteration has been the the method of choice for computing *all* eigenvalues (and, optionally, all eigenvectors). If only *selected eigenvalues* are required then bisection is the adequate method, otherwise QR-style algorithms tend to be faster. Inverse iteration may be used to compute *selected eigenvectors*, the subset being chosen after having determined and inspected the eigenvalues. In the 1990's the new divide-and-conquer algorithm, which has been developed with the aim of exposing parallelism, has proved by far superior to the QR iteration even on serial computers. It requires, however, significantly more memory than QR. Finally, there are other techniques — like homotopy algorithms — that are still in an experimental state.

Thoughout this section, we will use the shorthands $\mathsf{T}(i,i) =: \alpha_i$ and $\mathsf{T}(i,i-1) =: \beta_i$ for the diagonal and subdiagonal entries, respectively, of the symmetric tridiagonal matrix T, so

$$\mathsf{T} = \begin{pmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_n \\ & & & \beta_n & \alpha_n \end{pmatrix} .$$

We assume that all β_i are nonzero because otherwise the problem splits into two smaller subproblems that can be handled independently, cf. Section 2.5.

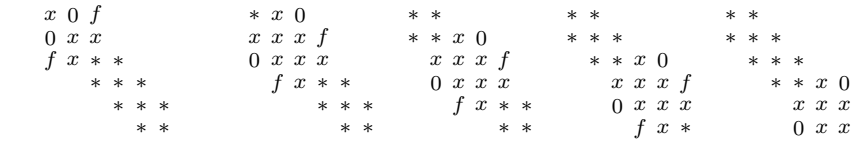


Figure 3. Rotations in one sweep of the QR iteration ($n = 6$).

5.1 QR Iteration

The QR iteration[17] for symmetric tridiagonal matrices is summarized in Algorithm 8. One pass through the **repeat** loop is called a *sweep* of the iteration.

Algorithm 8 QR iteration for tridiagonal matrices.

---

repeat
  determine a suitable shift $\sigma \in \mathbb{R}$
  apply a rotation in the $(1, 2)$ plane that zeroes the second entry of the vector $(\alpha_1 - \sigma, \beta_2, 0, \ldots, 0)^\top$. This creates a fill-in element at position $(3, 1)$.
  for $i = 2$ **to** $n - 1$
    apply a rotation in the $(i, i + 1)$ plane that zeroes the fill-in element at position $(i + 1, i - 1)$ and creates a new fill-in element at $(i + 2, i)$
until some subdiagonal entry β_i becomes negligible
set $\beta_i := 0$ and apply the whole algorithm to the submatrices $\mathsf{T}(1 : i, 1 : i)$ and $\mathsf{T}(i + 1 : n, i + 1 : n)$ { deflation }

Each sweep is initiated with a rotation $\mathsf{T} \mapsto \mathsf{R}^\top \mathsf{T}\mathsf{R}$, where the rotation angle depends on a certain parameter σ (the *shift*). This rotation produces a fill-in element below the subdiagonal, and a whole sequence of additional rotations is used to chase the fill element down the band until the tridiagonal structure is restored, cf. Figure 3. (See also Section 4.4 for a similar chasing strategy.)

If the shifts are chosen appropriately then the subdiagonal entries β_i tend to zero. Typically, the last subdiagonal entry β_n is the first to become negligible so that the last line in the algorithm reduces to continuing with the matrix $\mathsf{T}(1 : n - 1, 1 : n - 1)$, and that one eigenvalue may be read off from the diagonal entry α_n. But it is also possible that the matrix splits somewhere in the middle. The QR iteration tends to compute the eigenvalues by increasing absolute value, but this order may be broken.

The speed of convergence is determined by the shifts. For the most popular choice, *Wilkinson's shifts*, one can prove that the convergence is *global* (i.e., the subdiagonal elements are guaranteed to tend to zero) and — with rare exceptions — ultimately *cubic*. (Using Wilkinson's shifts means setting σ to one of the two eigenvalues of the current matrix T's trailing 2-by-2 subblock $\begin{pmatrix} \alpha_{n-1} & \beta_n \\ \beta_n & \alpha_n \end{pmatrix}$, namely

the eigenvalue closer to α_n.) The high order of convergence explains why — after some initial "warming up" — only two sweeps (with order-of-n flop per sweep) are needed on average in order to split off another eigenvalue. Therefore only order-of-n^2 flop are required to compute all eigenvalues of T. If the eigenvectors of T are needed, too, then each rotation must also be applied to an n-by-n matrix V, $\mathsf{V} \mapsto \mathsf{VR}$, where V has been initialized as the orthogonal matrix Q_1 that reduced the full or banded matrix A to tridiagonal form. (In the full case Algorithm 3, initialized with $\mathsf{Q} := \mathsf{I}$, can be used to build the matrix Q_1 with $\frac{4}{3}n^3$ flop.) Accumulating V requires a total of roughly $6n^3$ flop. As the QR iteration relies completely on rotations it is backward stable.

The name of the method stems from the fact that one sweep of the iteration corresponds to first computing a *QR decomposition* $\mathsf{T} - \sigma\mathsf{I} =: \tilde{\mathsf{Q}}\mathsf{R}$ into an orthogonal matrix $\tilde{\mathsf{Q}}$ and an upper triangular matrix R, and then replacing T with $\mathsf{R}\tilde{\mathsf{Q}} + \sigma\mathsf{I}$.

This idea can also be applied to matrices with semibandwidth $b > 1$ (indeed, even to full matrices), but for complexity reasons (one sweep then takes order-of-b^2n flop without the work on the eigenvectors) such matrices are typically reduced to tridiagonal form before the iteration is started. If many eigenvalues but only a few eigenvectors of the banded matrix A are required then the following hybrid technique is may pay: First reduce A to tridiagonal form *without* accumulating the transformations, then compute the eigenvalues of T, and finally use the computed eigenvalues as "perfect" shifts in the QR iteration on the original A (with accumulation of the transformations in a matrix V, initialized as $\mathsf{V} = \mathsf{I}$) in order to obtain the required eigenvectors.

There is a variety of mostly newer (and often more efficient) methods that are in some way similar to the tridiagonal QR iteration, in particular the LR iteration, which relies on LR (i.e., LU) decompositions of the matrix, the QL iteration, which works bottom-up instead of top-down, the Pal-Walker-Kahan QR variant for computing eigenvalues only, which requires no squares roots, and the new qd algorithms[33], which essentially perform LR iteration on a factored representation of the tridiagonal matrix.

All these variants bear very limited potential for parallelism in the work on T. By contrast, the accumulation of the eigenvector matrix V is easily parallelized.

5.2 Bisection

Bisection is a versatile method for computing all or selected eigenvalues of T. It is based on the fact that for any $\mu \in \mathbb{R}$, the number $\nu(\mu)$ of negative elements in the sequence

$$\begin{aligned} \delta_1 &:= \alpha_1 - \mu \ , \\ \delta_i &:= (\alpha_i - \mu) - \frac{\beta_i^2}{\delta_{i-1}} \ , \quad i = 2, \ldots, n \ , \end{aligned} \tag{21}$$

is equal to the number of eigenvalues of T that are smaller than μ. (This is just Sylvester's law of inertia[34], applied to the decomposition $\mathsf{T} - \mu\mathsf{I} = \mathsf{L}\mathsf{D}\mathsf{L}^\top$ with a lower triangular matrix L having all ones on the diagonal, and a diagonal matrix D.)

Therefore, given an interval $[a, b)$ that is known to include the k-smallest eigenvalue λ_k of T, we may locate this eigenvalue to very high precision by repeatedly cutting the interval into two halves and testing which of the halves contains λ_k, see Algorithm 9. Note that a suitable initial "search interval" is provided by Gershgorin's theorem, which states that *all* eigenvalues of T are contained in the interval

$$I = [\, \min\{\alpha_i - \rho_i \;:\; i = 1, \ldots, n\} \;,\;\; \max\{\alpha_i + \rho_i \;:\; i = 1, \ldots, n\} \,] \;,$$

where $\rho_i = |\beta_i| + |\beta_{i+1}|$, and $\beta_1 := 0$ and $\beta_{n+1} := 0$ for convenience.

Algorithm 9 Bisection for locating eigenvalue λ_k in an interval $[a, b)$.

---

compute $\nu(a)$ and $\nu(b)$ by building the sequences (21) for $\mu = a$ and $\mu = b$
if $\nu(a) < k \leq \nu(b)$ { otherwise λ_k is not contained in $[a, b]$ }
  while the interval width $b - a$ is too large
    let $c = (a + b)/2$ and compute $\nu(c)$ by evaluating (21) for $\mu = c$
    if $\nu(c) \geq k$ { λ_k lies in the left half of $[a, b]$ }
      $b := c$
    else { λ_k lies in the right half of $[a, b]$ }
      $a := c$
  return $\lambda_k \approx (a + b)/2$

Algorithm 9 is easily modified to compute a sequence $\lambda_j, \ldots, \lambda_k$ of consecutive eigenvalues, or all eigenvalues in a given interval $[a, b)$. Although bisection does not rely on orthogonal transformations (indeed, it is related to the unstable Gaussian elimination *without pivoting*), it is a perfectly stable algorithm featuring a very low backward error. In addition, bisection is efficiently parallelizable by having different processors compute disjoint subsets of the desired eigenvalues.

The major drawback of the method is its slow (namely, linear) convergence. This problem can be alleviated to some extent by using bisection only for *isolating* the eigenvalues, i.e., for narrowing down the initial interval until each subinterval contains exactly one eigenvalue. Then we switch to a superlinearly convergent root-finder, like Newton's method[25] or `zeroin`[8], for obtaining these eigenvalues to higher accuracy. Nevertheless, QR-type methods are superior on serial machines if more than one third, say, of the spectrum is required.

5.3 Inverse Iteration

Inverse iteration complements bisection in that it allows determining eigenvectors to selected previously computed eigenvalues. Inverse iteration is based on the *power iteration*, which is shown in Algorithm 10. The normalization (last line of the algorithm) is necessary to avoid overflow.

If there is a simple *dominant eigenvalue* (i.e., $|\lambda_{\max}| > |\lambda_i|$ for the $n - 1$ other eigenvalues) then the vectors $\mathrm{sign}(\lambda_{\max}) \cdot \mathbf{v}$ tend to an eigenvector corresponding to $\lambda_{\max}$ (the *dominant eigenvector*), and $\|\tilde{\mathbf{v}}\|$ approaches $|\lambda_{\max}|$. The convergence is linear with factor $c = |\lambda_{\max}/\lambda_{\max_2}|$, where $\lambda_{\max_2}$ is the eigenvalue with second largest modulus. Therefore, the more dominant $\lambda_{\max}$ is, the faster the convergence.

Now, if $(\lambda_i, \mathbf{v}_i)$ is an eigenpair of T, and $\mu \in \mathbb{R}$, then $(\lambda_i - \mu, \mathbf{v}_i)$ is an eigenpair of $\mathsf{T} - \mu\mathsf{I}$, and $(1/(\lambda_i - \mu), \mathbf{v}_i)$ is an eigenpair of $(\mathsf{T} - \mu\mathsf{I})^{-1}$. Therefore, if μ is a very

Algorithm 10 Power iteration for approximating the dominant eigenvector.

select a suitable starting vector $\mathbf{v}$
repeat
    $\tilde{\mathbf{v}} := \mathsf{T} \cdot \mathbf{v}$
    $\mathbf{v} := \tilde{\mathbf{v}}/\|\tilde{\mathbf{v}}\|$
until $\mathbf{v}$ is "good enough"

good approximation to λ_i (i.e., $|\lambda_i - \mu| \ll |\lambda_j - \mu|$ for all $j \neq i$) then $1/(\lambda_i - \mu)$ is a strongly dominant eigenvalue of $(\mathsf{T} - \mu\mathsf{I})^{-1}$. This is the basis of inverse iteration, which is just the power iteration applied to $(\mathsf{T} - \mu\mathsf{I})^{-1}$ and therefore can be obtained by substituting "solve the system $(\mathsf{T} - \mu\mathsf{I}) \cdot \tilde{\mathbf{v}} = \mathbf{v}$" for the third line in Algorithm 10.

Under favorable circumstances inverse iteration is a very flexible and fast method, able to compute just the needed eigenvectors and requiring only order-of-n flop per eigenvector, one order of magnitude less than QR iteration! This is the case if the eigenvalues of T are well separated and if good approximations to them have been computed (e.g., with bisection or QR iteration). Then experience shows that for each eigenvalue *only one or two* iteration steps are needed to obtain an excellent approximation to the corresponding eigenvector. In addition, eigenvectors to different eigenvalues can be computed in parallel.

Problems occur with *clustered* eigenvalues, i.e., $\lambda_i \approx \lambda_{i+1} \approx \ldots \approx \lambda_j$. Then the orthogonality of the corresponding computed eigenvectors $\mathsf{fl}(\mathbf{v}_i), \ldots, \mathsf{fl}(\mathbf{v}_j)$ gets impaired because the clustered eigenvalues lead to almost identical linear systems $(\mathsf{T} - \mu\mathsf{I})\tilde{\mathbf{v}} = \mathbf{v}$, which in turn lead to similar solutions $\tilde{\mathbf{v}}$. This is particularly true if the starting vector $\mathbf{v}$ always remains the same. Therefore, one partial remedy is to use a new random starting vector for each eigenvalue. In addition, the vectors are explicitly *orthogonalized* against each other. Suppose that we already have computed mutually orthogonal, normalized eigenvectors for the eigenvalues $\lambda_i, \ldots, \lambda_\ell$ of the cluster. Then, in each iteration step for eigenvector $\mathbf{v}_{\ell+1}$, we apply the *modified Gram-Schmidt process* to the intermediate vector $\tilde{\mathbf{v}}$ before it is normalized. That is, for $r = i, \ldots, \ell$ the component of $\tilde{\mathbf{v}}$ in direction $\mathbf{v}_r$ is eliminated by replacing $\tilde{\mathbf{v}}$ with $\tilde{\mathbf{v}} - (\mathbf{v}_r^\top \tilde{\mathbf{v}}) \cdot \mathbf{v}_r$. Note that the explicit orthogonalization may sum up to order-of-n^3 flop if large clusters of eigenvalues are present. Both cures cannot preclude another type of failure, which amounts to matching the eigenvectors with wrong eigenvalues[23].

Currently a new variant of inverse iteration is being developed that always gives orthonormal eigenvectors *without* explicit orthogonalization[12]. If these efforts are successful then the combination of bisection (or QR iteration) with the new method will make all other tridiagonal eigensolvers obsolete.

5.4 Divide-and-Conquer

The divide-and-conquer algorithm[10,16] was developed with the aim of exploiting parallelism. As it turned out, this method typically beats the QR iteration and bisection/inverse iteration even on serial machines.

The first step in the divide-and-conquer algorithm consists of "tearing" the tridiagonal matrix into two halves via a suitable rank-1-modification. For a 6-by-6 matrix, this tearing might look like

$$\mathsf{T} = \begin{pmatrix} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_3 & & & \\ & \beta_3 & \alpha_3 & \beta_4 & & \\ & & \beta_4 & \alpha_4 & \beta_5 & \\ & & & \beta_5 & \alpha_5 & \beta_6 \\ & & & & \beta_6 & \alpha_6 \end{pmatrix} = \left(\begin{array}{ccc|ccc} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_3 & & & \\ & \beta_3 & \alpha_3 - \beta_4 & & & \\ \hline & & & \alpha_4 - \beta_4 & \beta_5 & \\ & & & \beta_5 & \alpha_5 & \beta_6 \\ & & & & \beta_6 & \alpha_6 \end{array}\right) + \begin{pmatrix} & & & & & \\ & & & & & \\ & & & \beta_4 & \beta_4 & \\ & & & \beta_4 & \beta_4 & \\ & & & & & \\ & & & & & \end{pmatrix}$$

$$=: \left(\begin{array}{c|c} \mathsf{T}_1 & \\ \hline & \mathsf{T}_2 \end{array}\right) + \rho \mathbf{w}\mathbf{w}^\top$$

with $\rho = \beta_4$ and $\mathbf{w} = (0,0,1,1,0,0)^\top$. In general, T_1 and T_2 are m-by-m and $(n-m)$-by-$(n-m)$, resp., where $m \approx n/2$, $\rho = \beta_{m+1}$, and $\mathbf{w} = (0,\ldots,0,1,1,0,\ldots,0)^\top$.

Then the eigendecompositions $\mathsf{T}_1 = \mathsf{X}_1\Delta_1\mathsf{X}_1^\top$ and $\mathsf{T}_2 = \mathsf{X}_2\Delta_2\mathsf{X}_2^\top$ are computed by applying the divide-and-conquer algorithm recursively to the smaller matrices T_i (or, if these are small enough, QR iteration is used instead). This yields

$$\begin{aligned} \mathsf{T} &= \begin{pmatrix} \mathsf{X}_1\Delta_1\mathsf{X}_1^\top & \\ & \mathsf{X}_2\Delta_2\mathsf{X}_2^\top \end{pmatrix} + \rho\mathbf{w}\mathbf{w}^\top \\ &= \begin{pmatrix} \mathsf{X}_1 & \\ & \mathsf{X}_2 \end{pmatrix} \cdot \left(\begin{pmatrix} \Delta_1 & \\ & \Delta_2 \end{pmatrix} + \rho\mathbf{z}\mathbf{z}^\top \right) \cdot \begin{pmatrix} \mathsf{X}_1^\top & \\ & \mathsf{X}_2^\top \end{pmatrix} \\ &=: \mathsf{X} \cdot (\Delta + \rho\mathbf{z}\mathbf{z}^\top) \cdot \mathsf{X}^\top \ , \end{aligned}$$

where

$$\mathbf{z} = \mathsf{X}^\top \cdot \mathbf{w} = \begin{pmatrix} \text{last column of } \mathsf{X}_1^\top \\ \text{first column of } \mathsf{X}_2^\top \end{pmatrix} \ .$$

Then the eigendecomposition

$$\Delta + \rho\mathbf{z}\mathbf{z}^\top = \mathsf{Y}\Lambda\mathsf{Y}^\top \tag{22}$$

of a rank-1-perturbed diagonal matrix must be computed (see below), and finally the eigendecomposition of T can be recovered via $\mathsf{T} = (\mathsf{X}\mathsf{Y}) \cdot \Lambda \cdot (\mathsf{X}\mathsf{Y})^\top$.

The crucial step in the overall algorithm is the efficient and stable computation of the eigendecomposition (22), the remaining operations being expensive but trivial.

Suppose for a moment that all the entries of $\mathbf{z} = (\zeta_1,\ldots,\zeta_n)^\top$ are nonzero and that the eigenvalues (i.e., the diagonal entries) of Δ are distinct: $\delta_1 < \delta_2 < \ldots < \delta_n$. Then it is easy to show that the eigenvalues λ_i of $\Delta + \rho\mathbf{z}\mathbf{z}^\top$ are the roots of the so-called *secular equation*

$$f(\lambda) := 1 + \rho \sum_{i=1}^{n} \frac{\zeta_i^2}{\delta_i - \lambda} = 0 \ , \tag{23}$$

and that for any eigenvalue λ_i of $\Delta + \rho\mathbf{z}\mathbf{z}^\top$,

$$\mathbf{y}_i := (\Delta - \lambda_i \mathsf{I})^{-1} \cdot \mathbf{z} \tag{24}$$

is a corresponding eigenvector. (Note that computing $\mathbf{y}_i$ involves just an appropriate entry-wise scaling of $\mathbf{z}$.)

Eq. (23) implies that the δ_i "interlace" the sought eigenvalues, i.e., for $\rho > 0$ we have

$$\delta_1 < \lambda_1 < \delta_2 < \lambda_2 < \delta_3 < \ldots < \delta_n < \lambda_n \ .$$

This property, together with the fact that on each interval (δ_i, δ_{i+1}) the function f can be approximated by simple rational expressions, leads to a globally and quadratically convergent root finder that can compute all the λ_i in order-of-n^2 time.

Unfortunately, Eq. (24) is not an adequate means to compute the eigenvectors because orthogonality is severely impaired in the presence of close δ_i. It took more than ten years from the invention of the divide-and-conquer method until a technique was discovered[19] that did not need resorting to extended precision in the eigenvector computations. Roughly speaking, we can compute orthogonal eigenvectors for $\Delta + \rho\mathbf{z}\mathbf{z}^\top$ by applying Formula (24) to a slightly modified vector $\mathbf{z}$.

In practice our assumption that all the ζ_i are nonzero and all the δ_i are distinct is seldom fulfilled. The superiority of the divide-and-conquer method comes from the fact that it even can take advantage from a violation of this assumption. In fact, if some ζ_i is zero then δ_i and the ith column of X already are an eigenpair of the tridiagonal matrix T. Similarly, if some of the δ_i are (almost) identical then all but one of them are also very good approximations to eigenvalues λ_i of T, and the corresponding eigenvalues can be computed cheaply. Thus, the eigenvalue problem for $\Delta + \rho\mathbf{z}\mathbf{z}^\top$ and the ensuing multiplication $\mathsf{X} \cdot \mathsf{Y}$ are effectively reduced in size. Fortunately this type of *deflation* is quite frequent.

Parallelism can be exploited in two ways. First, all the solutions of the secular equation and the corresponding eigenvectors of $\Delta + \rho\mathbf{z}\mathbf{z}^\top$ may be computed independently from each other, and second, the matrix–matrix product $\mathsf{X} \cdot \mathsf{Y}$ lends itself naturally to a coarse-grained parallelization.

There are also attempts to apply the divide-and-conquer technique to banded matrices[3], but the resulting algorithms are still highly experimental.

5.5 Homotopy Methods

Homotopy methods[29] try to follow the paths of the eigenvalues and eigenvectors through a whole sequence of matrices

$$\mathsf{T}_0 \ \mapsto \ \mathsf{T}_1 = \mathsf{T}_0 + \theta_1(\mathsf{T} - \mathsf{T}_0) \ \mapsto \ \ldots \ \mapsto \ \mathsf{T}_{k-1} = \mathsf{T}_0 + \theta_{k-1}(\mathsf{T} - \mathsf{T}_0) \ \mapsto \ \mathsf{T}_k = \mathsf{T} \ ,$$

where T_0 is some initial matrix whose eigensystem is readily computed (e.g., a diagonal matrix), T is the matrix whose eigensystem is sought, and $0 = \theta_0 < \theta_1 < \ldots < \theta_{k-1} < \theta_k = 1$. If the step-sizes $\theta_i - \theta_{i-1}$ are small enough then the eigenvalues of T_{i-1} are good starting values for computing the eigenvalues of T_i, and inverse iteration for T_i's eigenvectors can be started with the eigenvectors of T_{i-1}.

The same idea may also be applied to other (e.g., banded) matrices and to the generalized eigenvalue problem. Note that the homotopy methods are still in an experimental state.

6 Methods Without Initial Tridiagonalization

In contrast to the methods described above, the algorithms discussed in this section do not rely on an initial tridiagonalization of the matrix, but apply the iterative process to the matrix A itself.

6.1 Jacobi's Method

Jacobi's method is based on the idea of reducing A's "off-diagonal norm",

$$\mathrm{off}(\mathsf{A}) := \sqrt{\sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \mathsf{A}(i,j)^2} \ ,$$

until it is negligibly small. Then we can read off A's eigenvalue from the diagonal entries and the eigenvectors from the columns of the orthogonal transformation matrix that was used to attain the almost-diagonal form.

This form is achieved by repeatedly zeroing selected entries $\mathsf{A}(i,j)$ with suitable rotations. If we choose the rotation angle θ such that

$$\tan\theta = \frac{\mathrm{sign}(\tau)}{|\tau| + \sqrt{1+\tau^2}} \ , \qquad \text{where} \quad \tau = \frac{\mathsf{A}(i,i) - \mathsf{A}(j,j)}{2\mathsf{A}(i,j)} \ ,$$

then a short computations shows that the two-sided rotation $\mathsf{A} \mapsto \mathsf{R}^\top \mathsf{A} \mathsf{R}$ with $\mathsf{R} = \mathsf{R}(i,j,\theta)$ indeed zeroes A's (i,j) and (j,i) entries and that $\mathrm{off}(\mathsf{A})$ drops by $2 \cdot \mathsf{A}(i,j)^2$. Note that again the rotation angle is not needed explicitly because the parameters c and s may be obtained via

$$c = \frac{1}{\sqrt{1+t^2}} \ , \quad s = t \cdot c \ .$$

Unfortunately, zeros introduced this way do not persist but are made nonzero again in later rotations. Thus most entries of A have to be made zero several times during the whole process. There are many different strategies for selecting the order of the entries (i,j) to be zeroed.

Obviously, zeroing the off-diagonal entry $\mathsf{A}(i,j)$ with the largest absolute value will lead to the largest reduction of $\mathrm{off}(\mathsf{A})$. This is the *classical Jacobi method*[22], which is slowed down by organizational overhead since the roughly $n^2/2$ comparisons for determining the maximum entry cost by far more time than the $6n$ ensuing arithmetic operations.

Therefore most often cheaper schemes are used, in particular the *row cyclic* and *column cyclic* elimination orders. In the former, the entries of the strictly lower triangle of A are made zero row-by-row, that is, in the order $\mathsf{A}(2,1)$, $\mathsf{A}(3,1)$, $\mathsf{A}(3,2)$, $\mathsf{A}(4,1)$, $\mathsf{A}(4,2)$, $\mathsf{A}(4,3)$, $\ldots$, $\mathsf{A}(n,1)$, $\mathsf{A}(n,2)$, $\ldots$, $\mathsf{A}(n,n-1)$. When all these entries have been made zero once (this is called a *sweep* of the method), then the process is started anew.

It is easy to see that each rotation in the classical method must reduce $\mathrm{off}(\mathsf{A})$ by a factor $\leq 1 - \frac{2}{n(n-1)} < 1$, thus implying at least linear convergence. A more involved analysis reveals that the convergence is indeed much faster, namely

quadratic[34]. This is also true for the cyclic schemes, provided that some precautions concerning the rotation angles are taken.

Experimental evidence suggests that in practice roughly $\log n$ sweeps are necessary to achieve adequate accuracy. As the methods discussed in Sections 4 and 5 require much less work (corresponding to just two Jacobi sweeps), Jacobi's method is usually not competitive. If, however, the matrix A is already strongly *diagonally dominant* (i.e., its diagonal entries are much larger than the off-diagonals) then Jacobi's method needs only a few sweeps to converge and may even beat the reduction-based methods.

Two other facts have revived the interest in Jacobi's method. First, it was observed that in some cases this algorithm — carefully implemented — can deliver much more accurate eigensystems than the other techniques. And second, appropriate cyclic elimination schemes allow exploiting parallelism by applying several rotations simultaneously[31].

There are also variants of Jacobi's method for the generalized eigenvalue problem, but their convergence properties are not sufficiently known.

6.2 Invariant Subspace Decomposition

The *invariant subspace decomposition algorithm*[21] (ISDA) is in some sense dual to the bisection/inverse iteration approach since it extracts information about the eigenvectors *before* the eigenvalues. This algorithm relies on the fact that eigenvectors are invariant under polynomial transformations of the matrix: If $(\lambda, \mathbf{q})$ is an eigenpair of A then for any polynomial p, $(p(\lambda), \mathbf{q})$ is an eigenpair of $p(\mathsf{A})$.

First, lower and upper bounds for $\mathrm{spec}(\mathsf{A})$ are determined (e.g., with Gershgorin's theorem for full matrices), and then a linear transformation $\mathsf{A} \mapsto \alpha\mathsf{A} + \beta\mathsf{I} =: \mathsf{A}_1$ is applied that maps the lower half of the eigenvalues, $\lambda_1, \ldots, \lambda_{n/2}$, into the interval $[0, \frac{1}{2}]$ and the upper half of the eigenvalues into $[\frac{1}{2}, 1]$.

Then further polynomial transformations $\mathsf{A}_k \mapsto p_k(\mathsf{A}_k) =: \mathsf{A}_{k+1}$ are applied, where the polynomial p_k is designed such that $p_k(x) \approx 0$ for $x \in [0, \frac{1}{2})$ and $p_k(x) \approx 1$ for $x \in (\frac{1}{2}, 1]$, i.e., p_k pushes the lower half of the eigenvalues toward 0 and the upper half toward 1. This process is repeated until $\mathsf{A}_k \approx \mathsf{A}_{k-1}$, implying that all eigenvalues of the final A_k are approximately 0 or 1.

Next, a so-called *rank-revealing QR decomposition* $\mathsf{A}_k = \mathsf{QR}$ of this matrix into an orthogonal matrix Q and an upper triangular matrix R is determined. Then the first (last) $n/2$ columns of Q are orthonormal eigenvectors to the eigenvalue 1 (0, respectively) of A_k. This implies

$$\mathsf{Q}^\top \mathsf{A} \mathsf{Q} = \begin{pmatrix} \mathsf{A}' & \\ & \mathsf{A}'' \end{pmatrix} ,$$

where A' and A'' are symmetric $(n/2)$-by-$(n/2)$ matrices. According to Section 2.5, the eigenvalues and eigenvectors of A may now be obtained from the eigendecompositions of A' and A'', which in turn are computed by recursively applying the ISDA to these two matrices, or with QR iteration if A' and A'' are small enough.

This approach does most of its computations with matrix–matrix products (e.g., in the evaluation of $p_k(\mathsf{A}_k)$) and therefore achieves high $\mathsf{Mflop/s}$ rates. In addition,

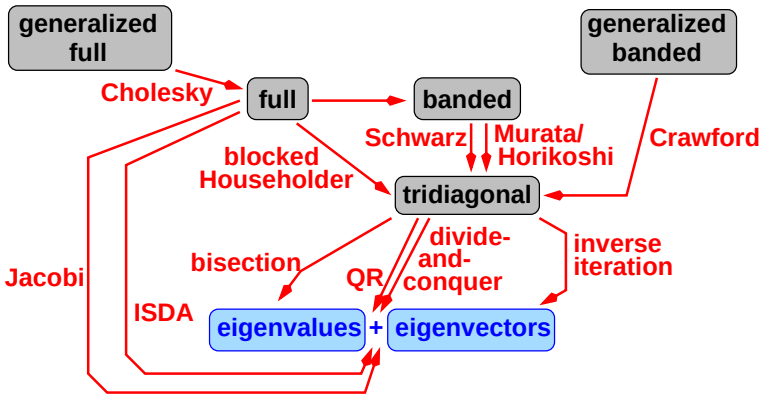


Figure 4. Computational paths for the symmetric eigenvalue problem.

efficient parallelization is possible. On the other hand, the ISDA requires several times more flop than the reduction–based techniques and is therefore not competitive on serial machines.

7 Synopsis

Figure 4 summarizes the algorithms used in direct symmetric eigensolvers. Depending on the initial problem (standard or generalized, full or banded), on the required information (eigenvalues only or eigenvectors, too), and on a priori knowledge (are the eigenvalues clustered or not?), different computational paths through the diagram are taken.

8 Available Software

Whenever possible, eigenvalue solvers from established libraries should be used because much effort went into optimizing their performance and making them robust (e.g., appropriate scalings for "balancing" the eigenvalues). There are many pitfalls awaiting the ambitious but unexperienced programmer.

8.1 Serial and Shared-Memory Machines

As modern software for dense or banded matrices relies heavily on the BLAS, obtaining an optimized implementation of the BLAS is extremely important. For most high-performance machines, optimized BLAS are provided by the manufacturer. If this is not the case, the public-domain BLAS from the ATLAS project (`http://www.netlib.org/atlas`) are a viable alternative. They often perform almost as well as (and sometimes even better than) proprietary implementations.

The public-domain *LAPACK* library[1] (`http://www.netlib.org/lapack`) contains optimized implementations for almost all non-experimental algorithms described in this article. (LAPACK also solves linear systems and least squares and

related problems, and most routines are available for real and complex matrices in either single-precision or double-precision arithmetic.) Preferably the libary's comfortable *driver routines* should be used. E.g., `DSYEV` computes the eigendecomposition of a full matrix, whereas `DSBEV` handles the banded case. Similar drivers exist for the generalized (full or banded) eigenvalue problem, as well as so-called "expert" drivers with additional options (e.g., computing only parts of the eigendecomposition). In addition, the computational routines may be called directly. Thus, `DSYTRD` (`DSBTRD`) reduces a full (banded) matrix to tridiagonal form with Algorithm 2 (with a modified version of Algorithm 4).

At `http://www-unix.mcs.anl.gov/prism`, additional software for the ISDA (Section 6.2) and for the two-phase reduction (mentioned at the end of Section 4.2) is available.

Because of the high quality of the LAPACK library, many vendor-supplied numerical packages and commercial libraries (like NAG) are based on these routines. This is also the case for *multithreaded* parallel libraries that come with shared-memory parallel machines. Here the parallelism is often confined within the BLAS.

8.2 Distributed-Memory Systems

For distributed-memory machines with the message-passing programming model, the situation is more complicated. The direct analogue to LAPACK is the *ScaLAPACK* library[7] (`http://www.netlib.org/scalapack`), which contains the functionality of many LAPACK routines. ScaLAPACK is based on the *PBLAS*, a parallelized implementation of the BLAS. ScaLAPACK contains solvers for the symmetric standard and generalized eigenvalue problem with full matrices, but *no banded solvers*. Another problem with this library is the fact that inverse iteration performs explicit orthogonalization only against eigenvectors within the same processor. Thus excessive memory may be required on a single processor, or orthogonality may be lost if explicit orthogonalization is turned off. (ScaLAPACK also includes QR iteration, which does not suffer from this problem.)

PeIGS (`http://www.emsl.pnl.gov:2080/docs/global/peigs.html`) is another library offering eigensystem functionality comparable to that of ScaLAPACK; in particular, banded problems are not addressed. While ScaLAPACK relies on a two-dimensional data layout (i.e., the matrices are split along rows *and* columns), PeIGS works with matrices that are distributed by whole columns or whole rows. Such distributions are typically inferior. On the other hand, inverse iteration in PeIGS orthogonalizes against all eigenvectors of a cluster — also on different processors — and therefore gives better results.

Finally, there are packages that do not contain complete eigensolvers but rather provide infrastructure for easily putting such methods together, namely *PLAPACK*[38] (`http://www.cs.utexas.edu/users/plapack`) and *Global Arrays* `http://www.emsl.pnl.gov:2080/docs/global/ga.html`). Both packages facilitate the manipulation of distributed matrices.

References

1. E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney and D. Sorensen, *LAPACK Users' Guide*, 3rd ed., SIAM, Philadelphia, PA (1999).
2. ANSI/IEEE, IEEE Standard for Binary Floating Point Arithmetic, Std. 754, New York (1985).
3. P. Arbenz, Divide and conquer algorithms for the bandsymmetric eigenvalue problem, *Parallel Comput.* **18**, 1105–1128 (1992).
4. J. Barlow and J. Demmel, Computing accurate eigensystems of scaled diagonally dominant matrices, *SIAM J. Numer. Anal.* **27(3)**, 762–791 (1990).
5. C. Bischof, B. Lang, and X. Sun, Parallel tridiagonalization through two-step band reduction, in *Proceedings of the Scalable High-Performance Computing Conference, Knoxville, Tennessee, May 23–25, 1994*, IEEE Computer Society, Los Alamitos, CA, 23–27 (1994).
6. C.H. Bischof and C.F. Van Loan, The WY representation for products of Householder matrices, *SIAM J. Sci. Stat. Comput.* **8 (1)**, s2–s13 (1987).
7. L.S. Blackford, J. Choi, A. Cleary, E. d'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker and R.R. Whaley, *ScaLAPACK Users' Guide*, SIAM, Philadelphia, PA (1997).
8. R.P. Brent, *Algorithms for Minimization Without Derivatives*, Prentice-Hall, Englewood Cliffs, NJ (1973).
9. C.R. Crawford, Reduction of a band symmetric generalized eigenvalue problem, *Comm. Assoc. Comp. Mach.* **16**, 41–44 (1973).
10. J.J.M. Cuppen, A divide and conquer method for the symmetric tridiagonal eigenproblem, *Numer. Math.* **36**, 177–195 (1981).
11. J.W. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, PA (1997).
12. I.S. Dhillon, *A New $O(n^2)$ Algorithm for the Symmetric Tridiagonal Eigenvalue/Eigenvector Problem*, Ph.D. thesis, Computer Science Division, University of California at Berkeley (1997).
13. J.J. Dongarra, J. Du Croz, S. Hammarling, and I. Duff, A set of level 3 basic linear algebra subprograms, *ACM Trans. Math. Softw.* **16 (1)**, 1–17 (1990).
14. J.J. Dongarra, J. Du Croz, S. Hammarling, and R.J. Hanson, An extended set of FORTRAN basic linear algebra subprograms, *ACM Trans. Math. Softw.* **14 (1)**, 1–17 (1988).
15. J.J. Dongarra, S.J. Hammarling, and D.C. Sorensen, Block reduction to condensed forms for eigenvalue computations, *J. Comput. Appl. Math.* **27**, 215–227 (1989).
16. J.J. Dongarra and D.C. Sorensen, A fully parallel algorithm for the symmetric eigenvalue problem, *SIAM J. Sci. Stat. Comput.* **8 (2)**, s139–s154 (1987).
17. J.G.F. Francis, The QR transformation: A unitary analogue to the LR transformation, part I and II, *Computer J.* **4**, 265–272 and 332–345 (1961/62).
18. G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD (1996).
19. M. Gu and S.C. Eisenstat, A stable and efficient algorithm for the rank-1

modification of the symmetric eigenvalue problem, *SIAM J. Matrix Anal. Appl.* **15**, 1266–1276 (1994).
20. N.J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, PA (1996).
21. S. Huss-Lederman, A. Tsao and G. Zhang, A parallel implementation of the invariant subspace decomposition algorithm for dense symmetric matrices, in *Proc. Sixth SIAM Conf. on Parallel Processing for Scientific Computing in Norfork, VA*, SIAM, Philadelphi, PA (1993).
22. C.G.J. Jacobi, Über ein leichtes Verfahren, die in der Theorie der Säculärstörungen vorkommenden Gleichungen zu lösen, *Crelle's J.* **30**, 51–94 (1846).
23. E.R. Jessup and I.C.F. Ipsen, Improving the accuracy of inverse iteration, *SIAM J. Sci. Stat. Comput.* **13**, 550–572 (1992).
24. L. Kaufman, Banded eigenvalue solvers on vector machines, *ACM Trans. Math. Softw.*, **10 (1)**, 73–86 (1984).
25. D. Kincaid and W. Cheney, *Numerical Analysis*, 2nd ed., Brooks/Cole Publishing Company, Pacific Grove, CA (1996).
26. B. Lang, A parallel algorithm for reducing symmetric banded matrices to tridiagonal form, *SIAM J. Sci. Comput.* **14 (6)**, 1320–1338 (1993).
27. B. Lang, Using level 3 BLAS in rotation-based algorithms, *SIAM J. Sci. Comput.* **19 (2)**, 626–634 (1998).
28. C.L. Lawson, R.J. Hanson, D.R. Kincaid, and F.T. Krogh, Basic linear algebra subprograms for FORTRAN usage, *ACM Trans. Math. Softw.* **5 (3)**, 308–323 (1979).
29. T.Y. Li and N. Rhee, Homotopy algorithm for symmetric eigenvalue problems, *Numer. Math.* **55**, 265–280 (1989).
30. The MathWorks, Inc., *MATLAB Reference Guide*, Natick, MA (1999).
31. J.J. Modi, *Parallel Algorithms and Matrix Computation*, Clarendon Press, Oxford (1988).
32. K. Murata and K. Horikoshi, A new method for the tridiagonalization of the symmetric band matrix, *Information Processing in Japan* **15**, 108–112 (1975).
33. B.N. Parlett, The new qd algorithms, *Acta Numerica*, 459–491 (1995).
34. B.N. Parlett, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, PA (1998). Updated reprint of the 1980 Prentice-Hall edition.
35. W. Rath, Fast Givens rotations for orthogonal similarity transformations, *Numer. Math.* **40**, 47–56 (1982).
36. R. Schreiber and C. Van Loan, A storage efficient WY representation for products of Householder transformations, *SIAM J. Sci. Stat. Comput.* **10 (1)**, 53–57 (1989).
37. H.R. Schwarz, Tridiagonalization of a symmetric band matrix, *Numer. Math.* **12**, 231–241 (1968).
38. R.A. van de Geijn, *Using PLAPACK*, MIT Press, Cambridge (1997).
39. J. Wilkinson, *Rounding Errors in Algebraic Processes*, Prentice-Hall (1964).
40. J.H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford (1965).

SEMIEMPIRICAL METHODS

WALTER THIEL
Max-Planck-Institut für Kohlenforschung
Kaiser-Wilhelm-Platz 1
45470 Mülheim
Germany
E-mail: thiel@mpi-muelheim.mpg.de

Recent methodological advances in semiempirical quantum chemistry and associated algorithmic developments are reviewed. After a brief overview over the current status of established semiempirical methods, the following topics are covered: new general-purpose methods with orthogonalization corrections, MNDO–NMR chemical shifts, analytic derivatives, parallelization of semiempirical codes, and linear scaling vs combined QM/MM approaches.

1 Introduction

This article outlines a lecture held at the winterschool on "Modern Methods and Algorithms of Quantum Chemistry" at Jülich (February 2000). It summarizes some recent developments in the field of semiempirical quantum chemistry, with emphasis on the work from our group. Theoretical derivations and detailed numerical results are generally not presented since they can be found in the original publications.

Over the past decades the semiempirical molecular orbital (MO) methods have been used widely in computational studies. There are several books [1,2,3,4,5,6] and reviews [7,8,9,10,11,12,13,14] which describe the underlying theory, the different variants of semiempirical methods, and the numerical results. Semiempirical approaches are normally formulated within the same conceptual framework as ab initio methods, but they neglect many smaller integrals to speed up the calculations. In order to compensate for the errors caused by these approximations, empirical parameters are introduced into the remaining integrals and calibrated against reliable experimental or theoretical reference data. This strategy can only be successful if the semiempirical model retains the essential physics to describe the properties of interest. Provided that this is the case, the parameterization can account for all other effects in an average sense, and it is then a matter of validation to establish the numerical accuracy of a given approach.

In current practice, semiempirical methods serve as efficient computational tools which can yield fast quantitative estimates for a number of properties. This may be particularly useful for correlating large sets of experimental and theoretical data, for establishing trends in classes of related molecules, and for scanning a computational problem before proceeding with higher-level treatments. Compared with ab initio or density functional methods, semiempirical calculations are much faster, typically by several orders of magnitude[14], but they are also less accurate, with errors that are less systematic and thus harder to correct. Hence, there remains the need to improve semiempirical methods with regard to their accuracy and range of applicability, without compromising their computational efficiency. In addition, there is the need to develop new algorithms in order to exploit modern computer architectures and

to extend semiempirical calculations to ever larger molecules.

2 Established methods

Quantum-chemical semiempirical treatments are defined by the following specifications:

(a) *The underlying theoretical approach:* Most current general-purpose semiempirical methods are based on MO theory and employ a minimal basis set for the valence electrons. Electron correlation is treated explicitly only if this is necessary for an appropriate zero-order description.

(b) *The integral approximation and the types of interactions included:* Traditionally there are three levels of integral approximation [2,15] – CNDO (complete neglect of differential overlap), INDO (intermediate neglect of differential overlap), and NDDO (neglect of diatomic differential overlap). NDDO is the best of these approximations since it retains the higher multipoles of charge distributions in the two-center interactions (unlike CNDO and INDO which truncate after the monopole).

(c) *The integral evaluation:* At a given level of integral approximation, the integrals are either determined directly from experimental data or calculated from the corresponding analytical formulas or computed from suitable parametric expressions. The first option is generally only feasible for the one-center integrals which may be derived from atomic spectroscopic data. The choice between the second and third option is influenced by the ease of implementation of the analytical formulas, but mainly depends on an assessment of how to model the essential interactions.

(d) *The parameterization:* Semiempirical MO methods are parameterized to reproduce experimental reference data (or, possibly, accurate high-level theoretical predictions as substitutes for experimental data). The reference properties are best selected such that they are representative for the intended applications. The quality of semiempirical results is strongly influenced by the effort put into the parameterization.

In our terminology, the specifications (a)–(b) define a semiempirical model, (a)–(c) an implementation of a given model, and (a)–(d) a particular method.

Over the years, a large number of methods with different choices for (a)–(d) and different acronyms have been published, including CNDO/2 [16], CNDO/S [17], INDO [18], MINDO/3 [19], INDO/S [20,21], SINDO1 [22,23], MSINDO [24,25], MNDO [26,27], MNDOC [28], AM1 [29], PM3 [30], SAM1 [31,32], MNDO/d [33,34], PM3/tm [35], and NDDO–G [36]. The most popular semiempirical methods for studying ground-state potential surfaces are based on the MNDO model [26]. As a point of reference for the further discussion, we therefore first outline the MNDO formalism for closed-shell molecules.

MNDO is a valence-electron self-consistent-field (SCF) MO treatment which employs a minimal basis of atomic orbitals (AOs, ϕ_μ) and the NDDO integral

approximation. The molecular orbitals ψ_i and the corresponding orbital energies ε_i are obtained from the solution of the secular equations ($S_{\mu\nu} = \delta_{\mu\nu}$ for NDDO):

$$\psi_i = \sum_\mu c_{\mu i}\phi_\mu \quad , \tag{1}$$

$$0 = \sum_\nu \left(F_{\mu\nu} - \delta_{\mu\nu}\varepsilon_i\right) c_{\nu i} \quad . \tag{2}$$

Using superscripts to assign an AO (with index μ, ν, λ, σ) to an atom A or B, the NDDO Fock matrix elements $F_{\mu\nu}$ are given as

$$F_{\mu^A\nu^A} = H_{\mu^A\nu^A} + \sum_{\lambda^A}\sum_{\sigma^A} P_{\lambda^A\sigma^A}\left[\left(\mu^A\nu^A,\lambda^A\sigma^A\right) - \frac{1}{2}\left(\mu^A\lambda^A,\nu^A\sigma^A\right)\right] + \sum_B\sum_{\lambda^B}\sum_{\sigma^B} P_{\lambda^B\sigma^B}\left(\mu^A\nu^A,\lambda^B\sigma^B\right) \quad , \tag{3}$$

$$F_{\mu^A\nu^B} = H_{\mu^A\nu^B} - \frac{1}{2}\sum_{\lambda^A}\sum_{\sigma^B} P_{\lambda^A\sigma^B}\left(\mu^A\lambda^A,\nu^B\sigma^B\right) \quad , \tag{4}$$

where $H_{\mu\nu}$ and $P_{\lambda\sigma}$ are elements of the one-electron core Hamiltonian and the density matrix, respectively, and $(\mu\nu,\lambda\sigma)$ denotes a two-electron integral. The total energy E_{tot} of a molecule is the sum of its electronic energy E_{el} and the repulsions E_{AB}^{core} between the cores of all atoms A and B.

$$E_{el} = \frac{1}{2}\sum_\mu\sum_\nu P_{\mu\nu}\left(H_{\mu\nu} + F_{\mu\nu}\right) \quad , \tag{5}$$

$$E_{tot} = E_{el} + \sum_{A<}\sum_B E_{AB}^{core} \quad . \tag{6}$$

It is obvious from Eqs. (1)–(6) that the MNDO model includes only one-center and two-center terms which accounts for much of its computational efficiency. Current implementations of the MNDO model (e.g., in the MNDO, AM1, and PM3 methods) are quite similar: Conceptually the one-center terms are taken from atomic spectroscopic data, with the refinement that slight adjustments are allowed in the optimization to account for possible differences between free atoms and atoms in a molecule. The one-center two-electron integrals derived from atomic spectroscopic data are considerably smaller than their analytically calculated values which is (at least partly) attributed to an average incorporation of electron correlation effects. For reasons of internal consistency, these integrals provide the one-center limit ($R_{AB} = 0$) of the two-center two-electron integrals $\left(\mu^A\nu^A,\lambda^B\sigma^B\right)$, whereas the asymptotic limit of $\left(\mu^A\nu^A,\lambda^B\sigma^B\right)$ for $R_{AB} \to \infty$ is determined by classical electrostatics. The semiempirical calculation of $\left(\mu^A\nu^A,\lambda^B\sigma^B\right)$ conforms to these limits and evaluates these integrals from semiempirical multipole-multipole interactions [33,37]: The relevant multipoles are represented by suitable point-charge configurations whose interaction is damped according to the Klopman-Ohno formula. Therefore, at intermediate distances, the semiempirical two-electron integrals are smaller than their analytical counterparts which again reflects some inclusion of

Table 1. Mean Absolute Errors for Organic Molecules (C, H, N, O).

Property[a]	N[b]	MNDO	AM1	PM3
ΔH_f (kcal/mol)	133	6.3	5.5	4.2
R(Å)	228	0.015	0.017	0.011
θ (deg)	92	2.69	2.01	2.22
IP (eV)	51	0.47	0.36	0.43
μ (D)	57	0.32	0.25	0.27

a Heats of formation ΔH_f, bond lengths R, bond angles θ, ionization potentials IP (Koopmans' theorem), dipole moments μ.
b Number of comparisons.

electron correlation effects. Aiming for a reasonable balance between electrostatic attractions and repulsions within a molecule, the core-electron attractions and the core-core repulsions are treated in terms of the corresponding two-electron integrals, neglecting, for example, penetration effects. The additional effective atom-pair potential that is included in the core-core repulsions (with an essentially exponential repulsion in MNDO and a more flexible parametric function in AM1 and PM3) attempts to compensate for errors introduced by the above assumptions, but mainly represents the Pauli exchange repulsions. Covalent bonding arises from the two-center one-electron integrals $H_{\mu^A \nu^B}$ (resonance integrals) which are often taken to be proportional to the corresponding overlap integrals.

The MNDO model in its current standard implementations (MNDO, AM1, PM3) has been parameterized primarily with respect to ground-state properties, with particular emphasis on the energies and geometries of organic molecules. It has become common practice to judge the accuracy of computational methods from statistical evaluations for standard validation sets. Such results are shown in Table 1 for organic compounds using the original MNDO set [27], in Table 2 for second-row and heavier molecules using the MNDO/d set [34], and in Table 3 for first-row and second-row compounds using the recent G2 neutral test set that is mainly employed in ab initio and density functional (DFT) work [38]. The data in Tables 1–3 have been assembled for previous reviews [14,39].

The results in Table 1 indicate that AM1 and PM3 offer some improvement in accuracy over the original MNDO method, but the mean absolute errors remain of the same order of magnitude. Hence, AM1 and PM3 may be viewed as variants of MNDO that explore the limits of the underlying theoretical model through careful parameterization.

In general, the errors in semiempirical calculations for organic compounds (Table 1) are smaller than those for molecules containing second-row and heavier elements (Table 2). As expected, the methods with an sp basis (MNDO, AM1, PM3) fail for hypervalent compounds, which are described more accurately upon extension of the MNDO model to d orbitals (MNDO/d, see Table 2).

Concerning the G2 validation set, it is obvious from Table 3 that the G2 approach is the most accurate one among those studied (as anticipated), followed by G2(MP2) and B3LYP. The semiempirical methods (especially PM3 and MNDO/d)

Table 2. Mean Absolute Errors for Molecules Containing Second-Row and Heavier Elements[a]

Property[b]	N[c]	MNDO	AM1	PM3	MNDO/d
ΔH_f (kcal/mol)	575	29.2	15.3	10.9	5.4
R(Å)	441	0.072	0.063	0.065	0.056
θ (deg)	243	3.7	3.4	7.4	2.5
IP (eV)	200	0.89	0.55	0.64	0.45
μ (D)	133	0.55	0.50	0.60	0.35
Hypervalent compounds only					
ΔH_f (kcal/mol)	67	143.2	61.3	19.9	5.4

a Elements: Na, Mg, Al, Si, P, S, Cl, Br, I, Zn, Cd, and Hg are included. Original data see ref. [34]

b Heats of formation ΔH_f, bond lengths R, bond angles θ, ionization potentials IP (Koopmans' theorem), dipole moments μ.

c Number of comparisons for MNDO/d (slightly lower for the other methods due to missing parameters, see ref. [34] for details).

Table 3. Mean Absolute Errors Δ_{abs}, Largest Positive Errors Δ_{pos}, and Largest Negative Errors Δ_{neg} of Heats of Formation (kcal/mol) for the Molecules from the G2 Neutral Test Set[a].

Method	Δ_{abs}	Δ_{pos}	Δ_{neg}	Ref.
G2	1.58	8.2	-7.1	[38]
G2(MP2)	2.04	10.1	-5.3	[38]
LDA(SVWN)	91.16	228.7	none	[38]
BLYP	7.09	28.4	-24.8	[38]
BP86	20.19	49.7	-6.3	[38]
B3LYP	3.11	8.2	-20.1	[38]
MNDO	9.32	27.6	-116.7	[39]
AM1	7.81	42.5	-58.2	[39]
PM3	7.01	23.1	-32.2	[39]
MNDO/d	7.26	27.6	-33.9	[39]

a The G2 neutral test set contains 148 molecules [38]. Due to the lack of parameters for certain elements, the data for MNDO, AM1, and PM3 refer to 146, 142, and 144 molecules, respectively.

show similar errors as BLYP, whereas BP86 and particularly LDA(SVWN) overbind strongly. To put these results into perspective, it should be noted that the complete geometry optimization of all 148 test molecules took altogether less than 20 seconds on an SGI R10000 workstation, for any of the four semiempirical methods. Since this computational effort is several orders of magnitude lower than that at the ab initio or DFT levels, the overall performance of the semiempirical methods appears acceptable (Table 3).

The statistical evaluations in Tables 1–3 (and many others that are available in the literature [11,12,13,30,34]) indicate that the established semiempirical methods can often be applied with useful accuracy and at very low computational costs. The

following limitations should be kept in mind, however:

(a) In general, errors tend to be more systematic at a given ab initio or DFT level and may therefore often be taken into account by suitable corrections. Errors in semiempirical calculations are normally less uniform and thus harder to correct.

(b) The accuracy of the semiempirical results may be different for different classes of compounds, and there are elements that are more "difficult" than others. Such variations in the accuracy are again less pronounced in high-level ab initio and DFT calculations.

(c) Semiempirical methods can only be applied to molecules containing elements that have been parameterized, while ab initio and DFT methods are generally applicable (apart from technical considerations such as basis set availability).

(d) Semiempirical parameterizations require reliable experimental or theoretical reference data and are impeded by the lack of such data. Such problems do not occur in ab initio or DFT approaches.

(e) Different parameterizations of a given semiempirical model may be required for different properties to obtain useful accuracy, and there is no systematic procedure for improving the results, unlike ab initio methods which provide a convergent path to the exact solution of the nonrelativistic Schrödinger equation.

To overcome some of these limitations, it seems desirable to develop better semiempirical methods. In the next sections, two such attempts are presented: First, a general-purpose parameterization of an approach that goes beyond the MNDO model by explicitly including orthogonalization effects, and secondly, a special parameterization of the MNDO model for the calculation of NMR chemical shifts.

3 Beyond the MNDO model: Orthogonalization corrections

As discussed above, the established MNDO-type methods do not treat the Pauli exchange repulsions explicitly, but attempt to incorporate them through an effective atom-pair potential that is added to the core-core repulsion. When trying to improve the MNDO model, it would seem logical to include the Pauli exchange repulsions explicitly in the electronic calculation and to remove the effective atom-pair potential from the core-core repulsion. For the sake of consistency, other one-electron terms of similar magnitude should then also be treated explicitly, i.e. penetration integrals and core-valence interactions (effective core potentials).

The zero-differential-overlap (ZDO) approximation causes the formal neglect of the Pauli exchange repulsions in semiempirical methods. Its consequences have been analyzed in many studies (see e.g. refs. [40,41,42,43,44,45,46,47]). ZDO-related deficiencies include:

(a) The gaps between bonding and antibonding molecular orbitals, as well as the corresponding excitation energies, are significantly underestimated.

(b) The correct pairing properties for the electronic states of of conjugated hydrocarbons are not reproduced [45].

(c) Barriers to internal rotation as in ethane are underestimated, with implications for other conformational properties (for a detailed discussion see ref. [46]).

(d) The closed-shell repulsions due to four-electron two-orbital interactions are not recovered properly, which causes problems, for example, with antiaromatic systems, intermolecular interaction potentials, and certain transition structures [41,42,43,44,45,46,47].

To account for the Pauli exchange repulsions more explicitly at the semiempirical level, orthogonalization corrections need to be addressed. The chosen strategy is based on the following general considerations [47].

Ab initio SCF methods solve the Roothaan-Hall pseudo-eigenvalue problem

$$\mathbf{F}\ \mathbf{C} = \mathbf{S}\ \mathbf{C}\ \mathbf{E} \quad , \tag{7}$$

where **F**, **C**, and **S** denote the Fock, eigenvector and overlap matrix, respectively, and **E** is the diagonal matrix of orbital energies. Orthogonalization of the basis leads to a standard eigenvalue problem

$$^{\lambda}\mathbf{F}\,^{\lambda}\mathbf{C} = {}^{\lambda}\mathbf{C}\mathbf{E} \quad , \tag{8}$$

where the superscript λ denotes a quantity expressed in an orthogonalized basis. The corresponding transformation can be achieved through a symmetric Löwdin orthogonalization. By contrast, semiempirical methods solve a secular equation,

$$^{NDDO}\mathbf{F}\,^{NDDO}\mathbf{C} = {}^{NDDO}\mathbf{C}\ \mathbf{E} \quad , \tag{9}$$

where the transformation **F** to $^{\lambda}\mathbf{F}$ is not explicitly performed. This suggests that the semiempirical Fock matrix implicitly refers to an orthogonal basis:

$$^{NDDO}\mathbf{F} \approx {}^{\lambda}\mathbf{F} \tag{10}$$

The neglect of all three-center and four-center two-electron integrals in NDDO approximation [2,15] is consistent with this interpretation because these integrals are vanishingly small only in an orthogonalized basis. The Fock matrix contains both one-electron (**H**) and two-electron (**G**) terms, which can be handled separately during Löwdin orthogonalization:

$$^{\lambda}\mathbf{F} = {}^{\lambda}\mathbf{H} + {}^{\lambda}\mathbf{G} \tag{11}$$

$$^{\lambda}\mathbf{H} = \mathbf{S}^{-1/2}\mathbf{H}\mathbf{S}^{-1/2} \tag{12}$$

$$^{\lambda}\mathbf{G} = \mathbf{S}^{-1/2}\mathbf{G}\mathbf{S}^{-1/2}. \tag{13}$$

To account for the orthogonalization effects arising from these transformations, different strategies may be followed in a semiempirical context. The direct use of Eq. (13) is not feasible since it requires the prior calculation of all two-electron integrals and would thus sacrifice the computational efficiency of the NDDO approach. On the other hand, the exact orthogonalization of the one-electron part according to Eq. (12) would be feasible computationally. However, using Eq. (12) without Eq. (13) introduces an imbalance between the one-electron and two-electron parts of the

Fock matrix and is therefore problematic. In the literature, several variants of this approach have been studied, where different parts of the Fock matrix are subjected to the exact Löwdin transformation while the ZDO approximation is applied to the remainder (see e.g. refs. [45,48,49]). To our knowledge, none of these attempts has been successfully incorporated into a general-purpose semiempirical method. Therefore, we have adopted the alternative strategy of representing the dominant one-electron orthogonalization corrections by suitable parametric functions. These corrections can then be adjusted during the parameterization process, as in previous approaches at the CNDO and INDO levels [22,23,24,25,50].

These basic ideas have been implemented in two steps. First, the Pauli exchange repulsions have been introduced as valence-shell orthogonalization corrections only in the one-center part of the core Hamiltonian [51,52]. In the second step, they have also been incorporated in the two-center part of the core Hamiltonian [46,47], i.e. in the resonance integrals. Both developments have been guided by analytic ab initio formulas and numerical ab initio SCF results. Their implementations are actually quite similar: In both cases, a Gaussian minimal basis set is used for technical reasons, and most two-center interactions are evaluated analytically followed by an appropriate Klopman-Ohno scaling. The valence-shell orthogonalization corrections are represented in terms of the resonance integrals through a truncated and parameterized series expansion. The resonance integrals contain a parameterized radial part while the angular part is the same as in the corresponding overlap integral.

The first approach [51,52] which we now call OM1 (orthogonalization method 1) contains only one-center and two-center terms, since the dominant orthogonalization corrections to the one-center part of the core Hamiltonian involve only a second atom. By contrast, the second approach [46,47] labelled OM2 (orthogonalization method 2) includes three-center contributions, since the corrections to the resonance integrals $\beta_{\mu\lambda}$ involve a third atom. The relevant corrections in OM2 are given by [46,47]

$$^{\lambda}H_{\mu\lambda} = \beta_{\mu\lambda} - \frac{1}{2}\sum_{\rho}^{C}(S_{\mu\rho}\beta_{\rho\lambda} + \beta_{\mu\rho}S_{\rho\lambda}) + \frac{1}{8}\sum_{\rho}^{C} S_{\mu\rho}S_{\rho\lambda}(H_{\mu\mu} + H_{\lambda\lambda} - 2H_{\rho\rho}) \ . \quad (14)$$

These three-center contributions reflect the stereochemical environment of each electron pair bond and should thus be important for modeling conformational properties. Technically, their inclusion increases the computational effort, but does not affect its scaling behavior since the three-center terms fall off like the square of an overlap integral and can therefore safely be neglected beyond certain cutoffs.

OM1 has been parameterized for the elements H, C, N, O, and F [51,52]. Significant qualitative improvements over the established MNDO-type methods are found in several areas, particularly for excited states. The mean absolute error in vertical excitation energies is 0.28 eV, much lower than in AM1 (1.20 eV) or PM3 (1.18 eV). This is readily rationalized: The destabilization of antibonding molecular orbitals is greater than the stabilization of bonding molecular orbitals at the ab initio level. This effect is not taken into account in the established MNDO-type methods whereas it is incorporated in OM1 through the orthogonalization correc-

Table 4. Mean Absolute Errors of Heats of Formation (kcal/mol) for First-Row Molecules from the G2 Neutral Test Set[a].

Method	First-row	CH	CHN	CHNO	CHNOF
G2	1.53(93)	1.31(30)	1.04(17)	1.37(33)	3.29(12)
BLYP	7.38	7.16	5.41	8.34	8.70
B3LYP	2.42	2.66	2.23	2.18	2.86
MNDO	7.71	7.42	5.93	9.13	7.62
AM1	7.44	6.57	5.05	9.35	7.89
PM3	6.86	6.43	6.75	6.88	7.46
OM1	4.64	3.20	3.67	5.89	6.37
OM2	3.36(81)	2.37	3.25	4.30	[b]

a The number of molecules in a subgroup is given in parentheses. Results for G2, BLYP, and B3LYP have been derived from the published data [38].
b OM2 parameters for F still missing.

tions [14]. Hence, the excitation energies are raised in a natural manner by correcting for deficiencies inherent to the ZDO approximation.

OM2 has presently been parameterized for H, C, N, and O [46]. As expected theoretically (see above), the numerical results show qualitative improvements for conformational properties, including rotational barriers, relative energies of isomers, and ring conformations. Hydrogen bonds are generally described much better than previously. The barriers for typical pericyclic reactions are realistic: for example, in the Diels-Alder reaction between butadiene and ethylene, the barrier is lower for the concerted pathway than for the biradicaloid one, and it decreases with increasing cyano substitution (in agreement with experiment and ab initio results).

While these qualitative advances are gratifying, it is also important to demonstrate that OM1 and OM2 perform well for the usual ground-state properties (see above). This is indeed the case [39,46,52]. OM1 and OM2 offer consistent small improvements over the established MNDO-type models, as can be seen, for example, from Table 4 which presents a statistical evaluation of thermochemical results for the first-row compounds and suitable subgroups from the G2 neutral test set [38,39].

Judging from the presently available OM1 and OM2 results [38,46,52,53], the explicit inclusion of Pauli exchange repulsions has led to qualitative and quantitative improvements in several important areas, which can partly be traced back to improvements in the underlying theoretical model. More work is needed to explore the limitations of these new approaches and to extend them to heavier elements.

4 NMR chemical shifts

NMR chemical shifts are an important source of information on molecular structure and reactivity in many fields of chemistry. They can be measured with high accuracy and sensitivity for increasingly complex systems, including large biomolecules whose solution structures can be determined from NMR chemical shifts and coupling constants. The accurate theoretical calculation of absolute NMR shieldings

is a challenging task which is best addressed by high-level theoretical methods. A large number of ab initio and density functional approaches to the evaluation of the NMR chemical shifts have indeed become available in recent years. Most of these approaches are based on the GIAO (Gauge-Including Atomic Orbitals) [54] and IGLO (Individual Gauge for Localized Orbitals) [55,56] concepts. Many such ab initio and DFT calculations have been published, and several reviews of such work have appeared [56,57,58].

Despite these tremendous advances at the ab initio and DFT level, a reliable and less costly semiempirical method would clearly be useful for the treatment of larger systems and the rapid estimation of chemical shifts in cases where the full versatility of ab initio and DFT methods is not needed. Even though there has been some previous semiempirical work in this area (for a review of the earlier literature, see refs. [59,60]), a reliable semiempirical method for computing NMR chemical shifts appears to be lacking. We have therefore decided to implement the evaluation of the NMR chemical shift tensor at the MNDO level for an spd basis set using gauge-including atomic orbitals and analytic derivative theory, followed by a specific parameterization of the GIAO–MNDO model with respect to experimental reference data [59,60]. In the following, we outline this development and summarize the results that have been obtained so far (for details see ref. [60]).

The NMR shielding tensor components σ_{ab} are second-order properties given by the mixed derivative of the energy with respect to the components of the nuclear magnetic moment $\vec{\mu}$ and the magnetic field strength $\vec{B}$:

$$\sigma_{ab} = \frac{\partial^2 E}{\partial B_a \partial \mu_b}\Big|_{\vec{\mu},\vec{B}=0} \tag{15}$$

For the evaluation of the shielding tensor it is sufficient to include only the terms linear and bi-linear in $\vec{\mu}$ and $\vec{B}$ in the one-electron Hamiltonian, which is then given by (atomic units) [54,61]:

$$\hat{h} = \hat{h}^{00} + i\sum_a \hat{h}^{a0} B_a + i\sum_b \hat{h}^{0b}\mu_b + \sum_{a,b} \hat{h}^{ab} B_a \mu_b \tag{16}$$

In the GIAO approach, the origin independence of the chemical shift is ensured by introducing a field-dependent factor into the basis functions [62]:

$$\varphi\left(\vec{r}-\vec{R}\right) = \chi\left(\vec{r}-\vec{R}\right)\exp\left(-\frac{i}{2c}\left(\vec{B}\times\vec{R}\right)\cdot\vec{r}\right) \tag{17}$$

where $\chi\left(\vec{r}-\vec{R}\right)$ is a field-independent basis function centered at $\vec{R}$.

GIAO basis functions do not depend on the nuclear magnetic moment, so that differentiation with respect to $\vec{\mu}$ may be performed using the Hellmann-Feynman theorem. Subsequent differentiation with respect to $\vec{B}$ introduces terms due to the derivatives of the Hamiltonian, the GIAOs, and the wave function. In the density matrix formulation, the resulting expression is given by [61]:

$$\sigma_{ab} = \sum_{\mu\nu} P_{\mu\nu} H^{ab}_{\mu\nu} - \sum_{\mu\nu} P^a_{\mu\nu} H^{0b}_{\mu\nu} \tag{18}$$

The purely real matrix elements $H^{ab}_{\mu\nu}$ and $H^{0b}_{\mu\nu}$ are:

$$H^{ab}_{\mu\nu} = \frac{\partial}{\partial B_a} \left\langle \varphi_\mu \middle| \frac{\partial \hat{h}}{\partial \mu_b} \middle| \varphi_\nu \right\rangle \Bigg|_{\vec{\mu},\vec{B}=0} \tag{19}$$

$$H^{0b}_{\mu\nu} = \left\langle \chi_\mu \middle| \frac{\partial \hat{h}}{\partial i\mu_b} \middle| \chi_\nu \right\rangle \Bigg|_{\vec{\mu},\vec{B}=0} \tag{20}$$

The imaginary part of the first-order density matrix $\mathbf{P}^a$ is formally defined as derivative of the zero-order density matrix $\mathbf{P}$:

$$P^a_{\mu\nu} = \frac{\partial P_{\mu\nu}}{\partial i B_a} \tag{21}$$

The perturbed density matrix is obtained by solving the CPHF (coupled perturbed Hartree-Fock) equations, which take a particularly simple form in the MNDO approximation [59,61]:

$$\mathbf{F}^a\mathbf{P} + \mathbf{F}\mathbf{P}^a = \mathbf{P}^a\mathbf{F} + \mathbf{P}\mathbf{F}^a \tag{22}$$

$$\mathbf{P}^a\mathbf{P} + \mathbf{P}\mathbf{P}^a = 2\mathbf{P}^a \tag{23}$$

$$\mathbf{F}^a = \mathbf{H}^a + \mathbf{G}\left(\mathbf{P}^a\right) \tag{24}$$

$\mathbf{H}^a$ is the imaginary part of the static derivative of the one-electron Hamiltonian with respect to the magnetic field:

$$\begin{aligned} H^a_{\mu\nu} &= \frac{\partial}{\partial i B_a} \left\langle \varphi_\mu \middle| \hat{h} \middle| \varphi_\nu \right\rangle \Big|_{\vec{\mu},\vec{B}=0} \\ &= \left\langle \chi_\mu \middle| \hat{h}^{a0} - \frac{1}{2c}\left(\vec{r}\times\vec{R_\mu}\right)_a \hat{h}^{00} + \frac{1}{2c}\hat{h}^{00}\left(\vec{r}\times\vec{R_\nu}\right)_a \middle| \chi_\nu \right\rangle \end{aligned} \tag{25}$$

$\mathbf{G}\left(\mathbf{P}^a\right)$ is the analogue of the two-electron part of the Fock matrix built using the first-order density matrix in place of the unperturbed density. Equation (24) is considerably simpler than its ab initio analogue [61] due to the cancellation of the field-dependent factors in all two-electron integrals that are retained in MNDO approximation. The CPHF equations (22)–(24) can be solved directly in the AO (atomic orbital) basis or transformed into the explicitly linear non-redundant form [59]. In either case, the equations are solved by a rapidly converging iterative procedure.

In ab initio approaches, the matrix elements $H^a_{\mu\nu}$ in Eq. (25) are evaluated analytically. MNDO-type methods assume the two-center matrix elements $H_{\mu\nu}$ of the operator $\hat{h}^{00}$ to be proportional to the corresponding overlap integral $S_{\mu\nu}$:

$$H_{\mu\nu} = \left\langle \chi_\mu | \hat{h}^{00} | \chi_\nu \right\rangle \approx b_{\mu\nu} S_{\mu\nu} = H^{\mathrm{MNDO}}_{\mu\nu} \tag{26}$$

Given this choice, a consistent approximation is required in Eq. (25) in order to preserve the origin independence of the results. For reasons discussed in more detail

elsewhere [59,60], we have adopted the following semiempirical expression for $H^a_{\mu\nu}$:

$$H^a_{\mu\nu}(\mathrm{MNDO}) = \frac{1}{2c}\Bigg\{ \left(\vec{R}_\mu \times \vec{R}_\nu\right) H^{\mathrm{MNDO}}_{\mu\nu} + \left(\vec{R}_\mu - \vec{R}_\nu\right) \times \left(b_{\mu\nu} \left\langle \chi_\mu \left| \vec{r} - \vec{R}_\nu \right| \chi_\nu \right\rangle\right) - \frac{1}{2}\left\langle \chi_\mu \left| \hat{L}^{R_\nu} \right| \chi_\nu \right\rangle + \frac{1}{2}\left\langle \chi_\nu \left| \hat{L}^{R_\mu} \right| \chi_\mu \right\rangle \Bigg\}_a \tag{27}$$

This completes the outline of the GIAO–MNDO approach. The NMR chemical shift tensor is evaluated from Eqs. (18)–(21), with the first-order density matrix being determined from the CPHF equations (22)–(24) using the approximation of Eq. (27) for $H^a_{\mu\nu}$. It should be stressed that semiempirical approximations only enter the right-hand side of the CPHF equations and thus only influence the paramagnetic terms via the first-order density matrix. The present model does not explicitly account for the contributions of the core electrons to the chemical shift (due to the neglect of core electrons in MNDO-type methods), so that absolute NMR shieldings will not be reproduced. However, since core contributions to the absolute shieldings are constant to within a few ppm [56], and largely cancel when computing shifts, this shortcoming is unlikely to impede a parameterization for NMR chemical shifts, at least for first-row elements (with a 1s core).

In the implementation of the GIAO–MNDO approach, the perturbed density matrix $P^a_{\mu\nu}$, Eq. (21), is obtained by solving the CPHF equations. Unlike in the ab initio case, its determination requires only a minor fraction of the overall computational effort: relatively few one-electron integrals are needed that can be expressed in terms of overlap integrals, and the two-electron integrals are field-independent due to the MNDO approximation (see above). The derivatives $H^{ab}_{\mu\nu}$, Eq. (19), and $H^{0b}_{\mu\nu}$, Eq. (20), both contain two-center and three-center contributions. The former can be determined easily, since there are relatively few two-center terms which can be computed efficiently over Slater orbitals by any of several available techniques [59,60]. The latter represent the computational bottleneck: there are many three-center terms which are hard to compute over Slater orbitals, accounting for far more than 90% of the overall computational effort. In a semiempirical context, it is tempting to neglect these three-center contributions, but this is not possible in general, because they turn out to be important for hydrogen chemical shifts and long-range current effects. We have therefore resorted to the use of STO–4G expansions for the Slater basis functions in MNDO during the evaluation of these three-center integrals (for details see ref. [60]). This speeds up the calculation considerably, even though most of the computational effort is still spent on the three-center terms.

Calculations using standard MNDO parameters overestimate the variation of the paramagnetic contribution to the NMR chemical shifts [59] which is due to the systematic underestimation of excitation energies in MNDO (see above). This failure cannot be rectified without modifying the MNDO parameters [59]. We have therefore carried out two parameterizations for H, C, N, and O where the quantities $H^{\mathrm{MNDO}}_{\mu\nu}$ in Eq. (27) were adjusted to increase the gap between occupied and unoccupied molecular orbitals which decreases the paramagnetic contribution to the NMR chemical shifts. The first parameterization involved a total of 9 orbital

Table 5. Mean Absolute Errors of the Computed NMR Chemical Shifts (ppm)[a].

Element[b]	MA2	MA3	MB2	MB3	Range
Small set					
H (102)	0.57	0.50	0.66	0.46	31
C (93)	11.78	12.12	9.79	9.65	346
N (37)	51.13	50.51	33.53	33.70	933
O (51)	64.30	63.70	59.90	59.75	1650
Large set					
H (345)	0.95	0.66	1.60	0.73	19
C (848)	8.48	8.32	8.31	7.95	359
N (239)	41.90	41.42	28.70	28.74	977
O (168)	47.92	47.79	43.64	43.32	1698

a See text.
b Number of comparisons per element given in parentheses.

exponents and resonance parameters for H, C, N, and O which were optimized to reproduce liquid-phase chemical shift reference data from 299 small organic and inorganic molecules [59] (employing MNDO geometries and neglecting three-center terms during the parameterization for the sake of efficiency). The resulting "Method A" (MA) parameters show a significant improvement over the original MNDO parameters in chemical shift computations, a particularly good agreement with experiment being observed for ^{13}C shifts in hydrocarbons [59]. The second parameterization added the one-center one-electron energies to the previous list of parameters, for a total of 16 adjustable parameters, to allow for the tuning of both diamagnetic and paramagnetic contributions. The parameters were calibrated against experimental chemical shifts for 97 small ions and molecules (mostly gas-phase data) using B3LYP geometries and including three-center terms during the parameterization. The final "Method B" (MB) parameters turn out to be slightly superior to the MA parameters and are the recommended choice, but the quality of the computed shifts is not too different.

Table 5 lists the mean absolute errors of the calculated shifts (relative to experiment) for the small gas-phase set (97 molecules, B3LYP geometries) and the largest liquid-phase set (384 molecules, MNDO geometries). Data are given for the MA and MB parameters, both without (MA2, MB2) and with (MA3, MB3) three-center terms (for details see [60]). For both validation sets, the mean absolute errors as well as the rms errors usually remain below 5% of the total chemical shift range for each element. Closer inspection shows that a significant fraction of the error in the small set is due to small molecules with unusual bonding. Since "normal" organic molecules are represented more strongly in the larger set, the errors tend to be somewhat smaller in the latter, at least for C, N, and O. In the case of H, the experimental shifts in the large set (liquid-phase data) are partly influenced by solvation effects which cannot be completely absorbed by the parameterization; this may be the reason why the errors for H are smaller in the smaller set (gas-phase data).

Three-center terms typically contribute a few ppm to the total chemical shift for all four elements studied, but they are essential for a qualitatively correct description of hydrogen chemical shifts (and also of nucleus-independent chemical shifts, see below). These terms are less important for C, N, and O, where they can normally be omitted without significantly degrading the results. Hence, MB2 is the recommended approach for C, N, and O. MB2 calculations are fairly efficient: for example, the ^{13}C shifts of taxol (113 atoms) are computed in 117 seconds on an SGI R10000 workstation.

Nucleus-independent chemical shifts (NICS) have been introduced as an additional magnetic criterion for aromaticity [63]. They are defined as the negative magnetic shielding at some selected point in space, e.g., at a ring center. Given the large number of successful NICS studies at ab initio and DFT levels, we have applied the MNDO–MB3 approach to compute 116 NICS values for a wide range of organic molecules, including [n]annulenes, polycyclic hydrocarbons, heterocycles, cage molecules, fullerenes, and pericyclic transition states [64]. Generally we find reasonable agreement with the ab initio and DFT reference data. The semiempirical NICS values tend to be smaller in absolute value than their ab initio counterparts, but they often show similar trends. The aromatic or antiaromatic character of a given system can normally be assigned correctly on the basis of the MNDO–MB3 NICS values [64].

For a more detailed assessment of the GIAO–MNDO method, the available statistical evaluations [60,64] need to be supplemented by case studies. For example, we have confirmed that GIAO–MNDO satisfactorily reproduces the differences between the nonclassical and classical 2-norbornyl cations [56], the shifts in benzenonium and related carbocations [65], and the individual shielding tensor components in carbonyl compounds [66]. Further such validation work is in progress.

5 Analytic derivatives

Efficient explorations of potential surfaces require the derivatives of the energy with respect to the nuclear coordinates. The first derivatives (gradient) are essential for geometry optimization, while the second derivatives (harmonic force field) characterize stationary points. In principle, these derivatives can be computed either analytically or numerically. The analytic approach is generally more precise, but requires a significant coding effort. By contrast, the numerical approach is easily implemented and also trivially amenable to a coarse-grained parallelization (see below).

For variational semiempirical SCF methods (e.g., closed-shell restricted Hartree-Fock, RHF, or unrestricted Hartree-Fock, UHF) the gradient can be computed at a fraction of the cost for an SCF calculation. Analytic gradients have long been available for this case [67], but a simple finite-difference procedure with a constant density matrix and recalculated two-center integrals is also efficient. For certain nonvariational semiempirical wavefunctions (e.g., open-shell half-electron restricted Hartree-Fock, HE–RHF, or configuration interaction, CI), analytic gradients have also been introduced [68] employing a procedure for solving the CPHF equations that scales as N^4 (N basis functions). Harmonic force fields have traditionally

been determined in semiempirical programs from numerical finite differences of the corresponding gradients.

We have recently implemented analytic first and second derivatives for MNDO-type methods [59,69,70,71]. Compared with the existing codes, there is little to be gained for the gradient in variational SCF methods (see above) whereas large improvements can be achieved for the gradient in nonvariational approaches (HE–SCF, HE–CI, and small CI expansions in general). In this case, the time-determining step is the solution of the CPHF equations which can be reformulated to scale as N^3 when making use of the **Z**-vector method [72]. The chosen representation of the CPHF equations exploits the simplifications arising from the MNDO integral approximations and covers the case of fractional occupation numbers [69]. The implementation provides 18 predefined combinations of options for the solution of the CPHF equations (e.g., direct vs iterative solver, MO vs AO basis, in-core vs out-of-core treatment, alternative preconditioners) and automatically selects the best computational path for given system size and hardware configuration. As a result of these developments, dramatic speedups are observed between the previous N^4 and the new N^3 algorithms: for example, the cpu time for the gradient evaluation on an SGI Indigo R4000 workstation drops

(a) from 24049 s to 62 s in $C_{66}H_{20}$, N=284, SCF–CI time 166 s [70],

(b) from 100139 s to 222 s in $C_{96}H_{24}$, N=408, HE–SCF time 605 s [69].

It is obvious that these algorithmic improvements will greatly facilitate semiempirical studies of open-shell molecules and electronically excited states: since the evaluation of the analytic gradient is now significantly faster than the underlying SCF and CI calculations, routine full geometry optimizations become feasible for fairly large systems.

Analytic second derivatives always contain contributions from integral derivatives (direct terms) and density matrix derivatives (CPHF terms), even at the SCF level. In the ab initio case, the computational cost is often dominated by the direct terms (in typical applications to medium-size molecules) so that the analytic evaluation of the second derivatives will normally be more efficient than a numerical evaluation. By contrast, in the semiempirical case, the computational cost is always dominated by the CPHF terms since there are relatively few integral derivatives due to the MNDO approximation. While it is not trivial to derive and to code these derivatives (especially for the two-electron terms in an spd basis) [59,71], their evaluation involves little computational effort. On the other hand, the traditional finite-difference evaluation of the second derivatives in MNDO-type methods also benefits from the MNDO integral approximation and scales as MN^3 for a molecule with M atoms and N basis functions. The same scaling can be achieved in the analytic approach if the CPHF problem is formulated in the AO basis and solved iteratively [71]. It is thus the prefactors in these scaling laws (and the convergence behavior of the underlying SCF and CPHF solutions) that determine the relative speed of the two approaches. Our implementation [71] of the analytic second derivatives for MNDO-type methods turns out to be faster by factors of 4-8 compared with analogous numerical computations, and it exhibits a reliable convergence over a wider range of molecules. The asymptotic memory and disk storage requirements

can be chosen to scale as low as N^2 without significant degradation of performance. These advances clearly facilitate force constant calculations for larger molecules at the semiempirical SCF level.

6 Parallelization

Both parallel vector processors (PVP) with shared memory and massively parallel (MP) systems with distributed memory are currently used for high-performance computing (HPC). It is evident that production codes should exploit the possibilities offered by these architectures. Like other quantum-chemical software, our present semiempirical program [73] is quite large (more than 170000 lines of source code) and provides the usual core functionality as well as diverse additional options [74]. When adapting such a program to novel architectures, it is common to concentrate on the time-determining steps of standard applications first and to consider other tasks later.

Focusing on the core functionality of semiempirical codes, the most time-consuming step in an MNDO SCF–MO calculation is the solution of the secular equations, see Eq. (2). In matrix notation:

$$\mathbf{FC} = \mathbf{CE} \tag{28}$$

The solution of this eigenvalue problem is an $O(N^3)$ process. The only other $O(N^3)$ step is the calculation of the density matrix

$$\mathbf{P} = 2\ \mathbf{C}_{occ}\mathbf{C}_{occ}^T \quad , \tag{29}$$

where $\mathbf{C}_{occ}$ is an $(N \times N_{occ})$ matrix containing the N_{occ} doubly occupied closed-shell eigenvectors, and $\mathbf{C}_{occ}^T$ its transpose. Both the evaluation of the two-center integrals and the construction of the Fock matrix (see Eqs. (3)–(4)) require a computational effort that scales as $O(N^2)$. These tasks may represent a large part of the computation for smaller molecules, but the $O(N^3)$ steps are bound to dominate for large molecules and thus deserve special attention.

Traditionally the eigenvalue problem, Eq. (28), is solved by diagonalizing the Fock matrix. In the case of semiempirical SCF methods, however, this diagonalization is replaced by a pseudodiagonalization scheme whenever possible [75]. In essence, the Fock matrix is transformed from the AO basis to the MO basis (from $\mathbf{F}_{AO} = \mathbf{F}$ to $\mathbf{F}_{MO}$) using the currently available trial eigenvectors (occupied and virtual vectors being collected in $\mathbf{C}_{occ}$ and $\mathbf{C}_{virt}$, respectively):

$$\mathbf{F}_{MO} = \mathbf{C}_{occ}^T\mathbf{F}_{AO}\mathbf{C}_{virt} \quad . \tag{30}$$

The elements in the resulting $(N_{occ} \times N_{virt})$ matrix $\mathbf{F}_{MO}$ that exceed a predefined threshold are approximately annihilated by noniterative 2×2 Jacobi-type rotations (with corresponding modification of the trial eigenvectors). This pseudodiagonalization scheme is much faster than the full diagonalization and usually does not slow down SCF convergence [75]. Full diagonalizations are still required at the beginning and the end of the computation.

Measurements for medium-size molecules show that in typical applications (e.g., closed-shell RHF geometry optimizations), more than 50% of the total cpu time is

spent for matrix multiplications (see Eqs. (29)–(30)), while non-negligible portions of around 10% may be required for integral evaluation, Fock matrix construction, full diagonalization, and gradient evaluation. These are very approximate order-of-magnitude figures, of course, and actual timings will show considerable scatter depending on several factors that need not be discussed here. In any event, the share of the total cpu time spent on matrix multiplications will increase with increasing molecular size.

This is an ideal situation for PVP architectures since large matrix multiplications (DGEMM) vectorize very well and can efficiently be parallelized on shared-memory machines, running close to peak performance. Likewise, efficient library routines are available on PVP machines for full diagonalizations (EISPACK, LAPACK). The remaining minor tasks can be vectorized (Fock matrix construction) and parallelized (Fock matrix construction, integral and gradient evaluation). Hence, high performance is reached in such jobs: for example [76], a full geometry optimization of the fullerene C_{960} at the MNDO SCF level runs at a speed of 23.9 GFLOPS/s on the NEC SX-4/16 at CSCS (75% of the hardware limit of 32 GLOPS/s for 16 cpus, speedup factor of 13.8 relative to 1 cpu, wallclock time 2319 s).

The situation is less advantageous for distributed-memory MP systems. A coarse-grained parallelization is possible for certain applications where essentially independent jobs can be distributed over different nodes. Examples of this kind include the finite-difference evaluation of gradients and force constants as well as reaction path calculations. These applications show almost linear speedup with the number of processors [74], but they are not prevalent in practice, and the size of molecules that can be handled in this manner is limited by the memory available at each single node [74].

It is thus necessary to attempt a fine-grained parallelization of the basic MNDO SCF procedure on MP systems such that the relevant data are distributed over all p available processors. We start by considering the square symmetric Fock matrix $\mathbf{F}$ which is distributed by rows over the processors. The division of $\mathbf{F}$ is determined once and for all at the beginning of the computation such that the boundaries are always between basis functions of different atoms. Hence, each processor conceptually owns a partial square Fock matrix $\mathbf{F}_p$ of dimension $(N_p \times N)$ with $N_p \approx N/p$. This partial square matrix is associated with a subset $\{A_p\}$ of atom pairs, i.e. those involving atoms whose basis functions correspond to the row indices of $\mathbf{F}_p$. Inspection of Eqs. (3)–(4) shows that the construction of $\mathbf{F}_p$ requires only the one-electron integrals from the corresponding partial square matrix $\mathbf{H}_p$ and the two-electron integrals $\mathbf{G}_p$ from the subset $\{A_p\}$ of atom pairs. Moreover, the calculation of $\mathbf{F}_p$ from Eqs. (3)–(4) employs only two-center density matrix elements that belong to the corresponding partial square matrix $\mathbf{P}_p$ whereas the required one-center density matrix elements $\mathbf{P}^{AA}$ refer to all atoms in the molecule.

Based on these observations, the partial Fock matrix $\mathbf{F}_p$ can be constructed locally at the corresponding processor using one-electron and two-electron integrals which can also be evaluated locally in an integral-direct manner. The one-center density matrix elements $\mathbf{P}^{AA}$ must be communicated globally, whereas the partial density matrix $\mathbf{P}_p$ needs to be communicated only to the associated processor. This

scheme [74] of constructing the Fock matrix in parallel exploits the simplifications inherent to the MNDO approximation.

The remaining and most time-consuming steps of the MNDO SCF procedure, Eqs. (28)–(30), are handled by calls to standard parallel library routines for matrix multiplication (PBLAS) and full diagonalization (ScaLAPACK), with a corresponding adaptation of the data structures [77]. Benchmark calculations with the resulting parallel code have been carried out some time ago on a Cray T3D system using up to 64 nodes [77]. These calculations showed that large molecules can be treated on this modest machine: Using 64 nodes, the fullerene C_{960} required less than 44 MB per node, with wallclock times of 2275 s and 13 s for the MNDO SCF energy and gradient evaluation, respectively. The MNDO geometry optimization of the fullerene C_{540} showed a speedup of 1.59 when going from 16 to 32 nodes, and another speedup of 1.35 upon moving to 64 nodes; these factors are mostly determined by the performance of the library routine (PSGEMM and PSSYEVX) where more than 75% of the cpu time are spent in this case. Apparently, the fullerene examples chosen are two small (matrix dimensions of 2160 for C_{540} and 3840 for C_{960}) to obtain a convincing scaling behavior from the linear algebra routines.

In summary, our semiempirical code has been ported both to shared-memory PVP and distributed-memory MP platforms. Both versions have been tested for molecules containing up to about 1000 non-hydrogen atoms. The performance is excellent on PVP machines up to 16 nodes, and satisfactory on MP systems up to 64 nodes. In the latter case, the code would seem to require further optimization, and larger molecules would need to be tested with more than 64 nodes. On the other hand, it is probably not worthwhile to do conventional semiempirical SCF calculations (with diagonalization) on molecules with more than 1000 atoms, even on MP systems, because alternative theoretical and algorithmic approaches are available for such large systems. This will be discussed in the next section.

7 Linear scaling and combined QM/MM approaches

In recent years, it has been a common goal of many groups to perform theoretical calculations on ever larger molecules. In the field of quantum chemistry, one common theme in this endeavour has been to exploit the locality of interactions whenever possible and to introduce numerically well-controlled simplifications for long-range interactions. This leads to computational procedures where the scaling of the computational effort with system size is improved over the formal scaling that characterizes a given method. The ultimate goal of such research are methods that exhibit linear scaling with system size.

It is clearly beyond the scope of this article to survey the widespread activities on linear scaling approaches. Some of the ab initio and DFT work has been reviewed recently [78]. In semiempirical quantum chemistry, the primary objective of the linear scaling algorithms is to avoid the diagonalization bottleneck, i.e. to avoid the steps that scale as N^3 (see above). Three different approaches have been proposed which accomplish this goal:

(a) Localized molecular orbital (LMO) techniques [79],

(b) Divide-and-conquer methods [80,81,82],

(c) Conjugate gradient density matrix search (CG–DMS) [83,84,85].

In the LMO approach, 2×2 rotations are applied to annihilate the interactions between occupied and virtual LMOs that are located within a certain cutoff radius (typically 8–10 Å), whereas all other interactions are considered to be negligible and therefore not treated. Small numerical errors result from these approximations, but they can be controlled by a renormalization of the LMOs and a suitable choice of the cutoff radius. The SCF procedure for LMOs is found to converge even faster than the conventional one. Using this algorithm, it is possible to carry out semiempirical SCF–MO calculations on standard workstations for proteins with several thousand atoms. The largest system treated in the original paper is rhizomucor miehei lipase with 4037 atoms [79].

The divide-and-conquer methods are based on a partitioning of the density matrix. The overall electronic structure calculation is decomposed into a series of relatively inexpensive calculations for a set of smaller, overlapping subsystems, each of which involves the diagonalization of a comparatively small Fock matrix. A global description of the full system is then obtained by combining the information from all subsystem density matrices. The accuracy of this approach can be controlled by the manner in which the system is partitioned. Proteins that have been treated by the divide-and-conquer method include HIV protease 6-mer with 9378 atoms [81].

The CG–DMS method avoids diagonalizations by using a direct conjugate gradient search for the density matrix. An acceptable density matrix must be normalized and idempotent, and it must commute with the Fock matrix after SCF convergence. In the CG–DMS treatment, a suitable functional of the density matrix is minimized with respect to the density matrix such that the resulting density matrix satisfies the above criteria; idempotency is enforced through McWeeny purification transformations. By neglecting density and Fock matrix elements close to zero, sparse matrices are obtained, and linear scaling can be approached by applying sparse matrix techniques. The accuracy of this algorithm depends on the chosen cutoffs. Initial benchmarks with this method include nucleic acids up to 6304 atoms and polyglycine chains up to 19995 atoms [85].

The methods outlined above share several common features. First, all of them introduce some approximations so that the results from such treatments will show some deviations from the conventional results obtained by a full diagonalization. These deviations will increase when cutoffs are made less stringent to speed up the calculations, and it is obviously necessary to ensure that they remain tolerable. Secondly, all three methods require some overhead so that the conventional calculations with full diagonalization remain faster for small molecules. The crossover point beyond which the described algorithms become faster depends on a number of factors (e.g., the chosen cutoffs and the molecular shapes), but 200–300 atoms seems to be a typical range. In view of this situation, it makes little sense to invest in the optimization of conventional codes to treat systems with more than thousand atoms more efficiently (see above). Finally, it should be kept in mind that none of these algorithms can exhibit true linear scaling without addressing other parts of semiempirical computations that scale formally as N^2, such as integral evaluation.

However, in practice this is only a minor point since the N^2 steps are generally very fast so that the removal of the diagonalization bottleneck is indeed of decisive importance.

The strength of the semiempirical "linear scaling" methods is most obvious for large systems with long-range charge transfer or long-range charge fluctuations, which may, for example, occur in proteins. Such effects can be captured by these approaches. On the other hand, there are many processes where the electronically active part of a large system is more localized, as in many chemical reactions or in localized electronic excitations. For such systems, an alternative theoretical treatment may be appropriate where the active center is described by quantum mechanics (QM) and the environment by molecular mechanics (MM). Such combined QM/MM methods are not the subject of this article, but they are briefly mentioned here to put the linear scaling developments into perspective.

Combined QM/MM methods (for reviews see refs. [86,87,88,89]) are computationally much less demanding than pure QM methods, even when compared with semiempirical linear scaling approaches. In addition, they offer the possibility for a system-specific modeling of large systems, by selecting suitable QM and MM components for the chemical problem being studied. On the other hand, QM/MM methods require a careful definition of the QM/MM boundary (especially when covalent bonds need to be cut), a physically sensible QM/MM coupling scheme, and possibly a calibration of QM/MM interaction terms [86,87,88,89,90,91]. Moreover, the standard QM/MM approaches do not allow for charge transfer between the QM and MM regions.

At present, it seems that the linear scaling QM methods and the combined QM/MM methods are complementary since they may be used to address different questions. Therefore, our group is active in both these areas which are expected to coexist and supplement each other for some time to come.

Due to the advances in linear scaling and QM/MM methods, the semiempirical methods of quantum chemistry have come to the point where they can be applied to complex systems with thousands of atoms such as enzymes. A relatively large number of enzymatic reactions has already been studied at the semiempirical QM/MM level (for a survey see ref. [88]), and analogous investigations with semiempirical linear scaling approaches are anticipated.

8 Conclusions

The methodological and algorithmic developments outlined in this article promise to open new areas of application for semiempirical methods which will therefore continue to be valuable tools for studying electronic effects in large molecules. Whenever technically feasible, such investigations should be supplemented with appropriate higher-level calculations because the synergetic use of several computational tools is often expected to provide the best computational solution for a given chemical problem.

Acknowledgements

The author wishes to thank his colleagues and coworkers whose names are listed in the references for their essential contributions to the work described. Special thanks are due to F. Terstegen for his help with the manuscript.

References

1. M.J.S. Dewar, *The Molecular Orbital Theory of Organic Chemistry*, McGraw-Hill, New York, (1969).
2. J.A. Pople and D.L. Beveridge,*Approximate Molecular Orbital Theory*, Academic Press, New York, (1970).
3. J.N. Murrell and A.J. Harget, *Semiempirical Self-Consistent-Field Molecular Orbital Theory of Molecules*, Wiley, New York, (1972).
4. G.A. Segal, Ed., *Modern Theoretical Chemistry*, Plenum, New York, (1977), Vols. 7-8.
5. M. Scholz and H.J. Köhler,*Quantenchemie*, Hüthig, Heidelberg, (1981), Vol. 3.
6. T. Clark,*A Handbook of Computational Chemistry*, Wiley, New York, (1985).
7. M.J.S. Dewar, Science **187**, 1037-1044 (1975).
8. K. Jug, Theor. Chim. Acta **54**, 263-300 (1980).
9. M.J.S. Dewar, J. Phys. Chem. **89**, 2145-2150 (1985).
10. W. Thiel, Tetrahedron **44**, 7393-7408 (1988).
11. J.J.P. Stewart, J. Comp.-Aided Mol. Design **4**, 1-103 (1990).
12. J.J.P. Stewart, *in Reviews in Computational Chemistry* (Eds. K.B. Lipkowitz and D.B. Boyd), Vol. **1**, VCH Publishers, New York, (1990), pp. 45-81.
13. M.C. Zerner, *in Reviews in Computational Chemistry* (Eds. K.B. Lipkowitz and D.B. Boyd), Vol. **2**, VCH Publishers, New York, (1991), pp. 313-365.
14. W. Thiel, Adv. Chem. Phys. **93**, 703-757 (1996).
15. J.A. Pople, D.P. Santry, and G.A. Segal, J. Chem. Phys. **43**, S129-S135 (1965).
16. J.A. Pople and G.A. Segal, J. Chem. Phys. **44**, 3289-3296 (1966).
17. J. Del Bene and H.H. Jaffé, J. Chem. Phys. **48**, 1807-1813 (1968).
18. J.A. Pople, D.L. Beveridge, and P.A. Dobosh, J. Chem. Phys. **47**, 2026-2033 (1967).
19. R.C. Bingham, M.J.S. Dewar, and D.H. Lo, J. Am. Chem. Soc. **97**, 1285-1293 (1975).
20. M.C. Zerner and J. Ridley, Theor. Chim. Acta **32**, 111-134 (1973).
21. A.D. Bacon and M.C. Zerner, Theor. Chim. Acta **53**, 21-54 (1979).
22. D.N. Nanda and K. Jug, Theor. Chim. Acta **57**, 95-106 (1980).
23. K. Jug, R. Iffert, and J. Schulz, Int. J. Quantum Chem. **32**, 265-277 (1987).
24. B. Ahlswede and K. Jug, J. Comput. Chem. **20**, 563-571 (1999).
25. B. Ahlswede and K. Jug, J. Comput. Chem. **20**, 572-578 (1999).
26. M.J.S. Dewar and W. Thiel, J. Am. Chem. Soc. **99**, 4899-4907 (1977).
27. M.J.S. Dewar and W. Thiel, J. Am. Chem. Soc. **99**, 4907-4917 (1977).
28. W. Thiel, J. Am. Chem. Soc. **103**, 1413-1420 (1981).
29. M.J.S. Dewar, E. Zoebisch, E.F. Healy, and J.J.P. Stewart, J. Am. Chem. Soc.

107, 3902-3909 (1985).
30. J.J.P. Stewart, J. Comput. Chem. **10**, 209-220 (1989); J. Comput. Chem. **10**, 221-264 (1989).
31. M.J.S. Dewar, C. Jie, and J. Yu, Tetrahedron **49**, 5003-5038 (1993).
32. A.J. Holder, R.D. Dennington, and C. Jie, Tetrahedron **50**, 627-638 (1994).
33. W. Thiel and A.A. Voityuk, Theor. Chim. Acta **81**, 391-404 (1992); Theor. Chim. Acta **93**, 315 (1996).
34. W. Thiel and A.A. Voityuk, J. Phys. Chem. **100**, 616-626 (1996).
35. SPARTAN 4.0 (1995), Wavefunction Inc., 18401 Von Karman Avenue, Irvine, CA 92715.
36. A.A. Voityuk, M.C. Zerner, and N. Rösch, J. Phys. Chem. **A 103**, 4553-4559 (1999).
37. M.J.S. Dewar and W. Thiel, Theor. Chim. Acta **46**, 89-104 (1977).
38. L.A. Curtiss, K. Raghavachari, P.C. Redfern, and J.A. Pople, J. Chem. Phys. **106**, 1063-1079 (1997).
39. W. Thiel, *in "Computational Thermochemistry"* (Eds. K.K. Irikura and D.J. Frurip), ACS Symposium Series 667, American Chemical Society, Washington, DC, (1998), pp.142-161.
40. C. Kollmar and M.C. Böhm, Theor. Chim. Acta **92**, 13-47 (1995).
41. S. de Bruijn, Stud. Phys. Theor. Chem. **21**, 251-272 (1982).
42. A.R. Gregory and M.N. Paddon-Row, J. Am. Chem. Soc. **98**, 7521-7523 (1976).
43. M.V. Basilevsky, A.G. Shamov, and V.A. Tikhomirov, J. Am. Chem. Soc. **99**, 1369-1372 (1977).
44. P. Caramella, K.N. Houk, and L.N. Domelsmith, J. Am. Chem. Soc. **99**, 4511-4514 (1977).
45. J. Spanget-Larsen, Theor. Chim. Acta **98**, 137-153 (1997).
46. W. Weber, Ph.D. Thesis, Universität Zürich, (1996).
47. W. Weber and W. Thiel, Theor. Chem. Acc. , in press (2000).
48. J. Spanget-Larsen, Theor. Chim. Acta **55**, 165-172 (1980).
49. C. Kollmar, Chem. Phys. Lett. **269**, 215-221 (1997).
50. M.J. Filatov, O.V. Gritsenko, and G.M. Zhidomirov, Theor. Chim. Acta **72**, 211-222 (1987).
51. M. Kolb, Ph.D. Thesis, Universität Wuppertal, (1991).
52. M. Kolb and W. Thiel, J. Comput. Chem. **14**, 775-789 (1993).
53. M. Beck, K. Möhle, M. Paech, R. Steiger, W. Thiel, E.U. Wallenborn, and W. Weber, to be published.
54. R. Ditchfield, Mol. Phys. **27**, 789-807 (1974).
55. W. Kutzelnigg, Isr. J. Chem. **19**, 193-200 (1980).
56. W. Kutzelnigg, U. Fleischer, and M. Schindler, NMR Basic Principles Progress **23**, 165-262 (1990).
57. J. Gauss, Ber. Bunsenges. Phys. Chem. **99**, 1001-1008 (1995).
58. G. Schreckenbach and T. Ziegler, Theor. Chem. Acc. **99**, 71-82 (1998).
59. S. Patchkovskii, Ph.D. Thesis, Universität Zürich, (1997).
60. S. Patchkovskii and W. Thiel, J. Comput. Chem. **20**, 1220-1245 (1999).
61. K. Wolinski, J.F. Hinton, and P. Pulay, J. Am. Chem. Soc. **112**, 8251-8260

(1990).
62. F. London, J. Phys. Radium **8**, 397 (1937).
63. P.v.R. Schleyer, C. Maerker, A. Dransfeld, H. Jiao, and N.J.R. van Eikema Hommes, J. Am. Chem. Soc. **118**, 6317-6318 (1996).
64. S. Patchovskii and W. Thiel, J. Mol. Model. , in press (2000).
65. M. Kollwitz and J. Gauss, Chem. Phys. Lett. **260**, 639-646 (1996).
66. N. Gonzales and J. Simons, Int. J. Quantum Chem. **63**, 875-894 (1997).
67. M.J.S. Dewar and Y. Yamaguchi, Comput. Chem. **2**, 25-29 (1978).
68. M.J.S. Dewar and D.A. Liotard, J. Mol. Struct. (Theochem) **206**, 123-133 (1990).
69. S. Patchkovskii and W. Thiel, Theoret. Chim. Acta **93**, 87-99 (1996).
70. S. Patchkovskii and W. Thiel, Theor. Chem. Acc. **98**, 1-4 (1997).
71. S. Patchkovskii and W. Thiel, J. Comput. Chem. **17**, 1318-1327 (1996).
72. N.C. Handy and H.F. Schaefer, J. Chem. Phys. **81**, 5031-5033 (1984).
73. W. Thiel, Program MNDO99, Mülheim, (1999).
74. W. Thiel and D.G. Green, *in: Methods and Techniques in Computational Chemistry, METECC-95* (Eds. E. Clementi and G. Corongiu), STEF, Cagliari, (1995), pp. 141-168.
75. J.J.P. Stewart, P. Csaszar, and P. Pulay, J. Comput. Chem. **3**, 227-228 (1982).
76. C. Henriet, NEC Corporation, private communication (September 1998).
77. R. Graham, Cray Research, private communication (November 1995).
78. G.E. Scuseria, J. Phys. Chem. **A 103**, 4782-4790 (1999).
79. J.J.P. Stewart, Int. J. Quantum Chem. **58**, 133-146 (1996).
80. W. Yang and T.-S. Lee, J. Chem. Phys. **103**, 5674-5678 (1995).
81. T.S. Lee, D.M. York, and W. Yang, J. Chem. Phys. **105**, 2744-2750 (1996).
82. S.L. Dixon and K.M. Merz, J. Chem. Phys. **104**, 6643-6649 (1996).
83. X.-P. Li, R.W. Nunes, and D. Vanderbilt, Phys. Rev. **B 47**, 10891-10894 (1993).
84. J.M. Millam and G.E. Scuseria, J. Chem. Phys. **106**, 5569-5577 (1997).
85. A.D. Daniels, J.M. Millam and G.E. Scuseria, J. Chem. Phys. **107**, 425-431 (1997).
86. J. Aqvist and A. Warshel, Chem. Rev. **93**, 2523-2544 (1993).
87. J. Gao, *in Reviews in Computational Chemistry* (Eds. K.B. Lipkowitz and D.B. Boyd), Vol. 7, VCH Publishers, New York, (1995), pp. 119-185.
88. T.Z. Mordasini and W. Thiel, Chimia **52**, 288-291 (1998).
89. G. Monard and K.M. Merz, Acc. Chem. Res. **32**, 904-911 (1999).
90. D. Bakowies and W. Thiel, J. Phys. Chem. **100**, 10580-10594 (1996).
91. I. Antes and W. Thiel, J. Phys. Chem. **A 103**, 9290-9295 (1999).

HYBRID QUANTUM MECHANICS/MOLECULAR MECHANICS APPROACHES

PAUL SHERWOOD
CLRC Daresbury Laboratory
Warrington, WA4 4AD
United Kingdom
E-mail: p.sherwood@dl.ac.uk

An overview is provided of the range of approaches to hybrid QM/MM (quantum mechanics/molecular mechanics) calculations. The factors considered include the choice of QM and MM methods, the construction of the total QM/MM energy expression, the nature of the QM/MM coupling, and the treatment of bonds between QM and MM regions. The practical issues associated with handling the increased conformational complexity of macromolecular systems, and the construction of QM/MM codes are discussed.

1 Introduction

The modelling of complex chemical systems is still a daunting challenge. We have at our disposal sophisticated first-principles methods for simulating reactions and electronic processes to high accuracy but these are limited by their computational cost to small molecules. The systems of chemical interest in computational biology and catalysis are often condensed phase systems with many thousands of participating atoms. While significant progress is being made in the development of quantum chemical approaches applicable to large systems[1], it is clear that to treat complex biological and catalytic systems we still need to be able to integrate a range of computational chemistry methodologies with differing accuracies and cost. By embedding a quantum mechanics calculation in a classical molecular mechanics model of the environment, the hybrid QM/MM schemes attempt to incorporate environmental effects at an atomistic level, including such influences as mechanical constraints, electrostatic perturbations and dielectric screening. Since the first published example from the field of computational enzymology[2] many QM/MM schemes have been implemented and applied in a wide variety of chemical applications.

The subject has steadily developed, and the last couple of years in particular have seen rapid increase in the rate of publication of QM/MM applications. The availability of implementations within commerical packages (e.g. CHARMM and Gaussian98), and improvements in the available computational resources will doubtless contribute to a continued increase in popularity. Recent reviews of the subject include those by Gao[3] and Mordasini and Thiel[4].

2 Terminology

The first, trivial, step is to divide the entire system (E), into inner (I) and outer (O) regions (Figure 1a). The objective in all cases is to use the QM calculation to model processes in the inner region and to use MM to model the outer region. As always

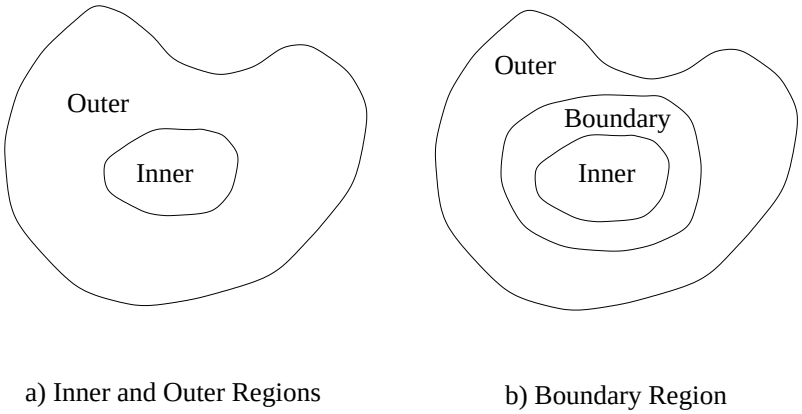


Figure 1. Terminology for regions used in the QM/MM scheme

the devil is in the detail, and it is rarely possible to simply write down the total energy in terms of two, non-overlapping subsystems, Very often the interactions between the systems are sufficiently strong (the obvious example being the presence of a chemical bond) to ensure that a QM calculation on the inner region alone is not sensible. Some form of termination, or treatment of the boundary is required. For the purpose of this article, we will classify the approaches to this QM termination into two groups;

1. those based on **link atoms**[5,6], additional centres added to the QM calculation but not present in the entire system(E). The position of centres in the region (L) are either viewed as independent variables or as a function of the positions of atoms in both (I) and (O) regions (*vide infra*). Link atoms are generally invisible to the MM calculation.

2. those having a **boundary region**, a subset of the centres of the systems which will feature in both QM and MM calculations (Figure 1b). In the QM calculation these centres can have a range of roles, ranging from a re-parameterised semi-empirical Hamiltonian to an *ab-initio* pseudopotential or a frozen hybrid orbital.

3 Overview of QM/MM Schemes

In this section we discuss some of the choices that distinguish the range of possible QM/MM schemes.

3.1 The QM/MM Energy Expression

Given the definitions given in Section 2 we can group QM/MM schemes into two broad classes:

3.1.1 Additive Schemes

This general classification is applied to schemes for which the QM and MM energies are considered complementary, the total energy of the system is obtained by adding them together and applying any coupling terms or corrections.

For link atom based schemes the total energy expression takes the form of Equation 1 (E = entire, I = inner, O = outer and L = link);

$$E^{E}_{QM/MM} = E^{O}_{MM} + E^{I,L}_{QM} + E^{I,O}_{QM-MM} - E^{I,L}_{Corr} \tag{1}$$

Here the term $E^{I,O}_{QM-MM}$ is QM-MM coupling term includes all terms that couple the two regions, for example MM-style bonding and van der Waals interactions and modifications to the QM Hamiltonian to reflect the influence of some or all of the atoms in the outer region.

The correction term $E^{I,L}_{Corr}$ represents terms designed to reduce the dependency of the total energy on the centres in the link atom region[7]. How this term is handled depends on the the choice of link atom coordinates, as discussed below in Section 3.5.2. However, this term is often neglected.

Additive schemes are probably the most widely adopted approaches to QM/MM calculations, particularly in the biomolecular area, with the AMBER[5,8,9] and CHARMM [6,10,11,12] based implementations being important examples.

The main problems with the scheme arise because it is difficult to compute the coupling term $E^{I,O}_{QM-MM}$ accurately in the presence of the link atoms[13], particularly if electrostatic perturbations to the QM Hamiltonian are included.

For boundary-region based methods, which are usually additive in type, the energy total may be written

$$E^{E}_{QM/MM} = E^{O,B}_{MM} + E^{I,B}_{QM} + E^{I,O,B}_{QM-MM} \tag{2}$$

There is no need for any link atom corrections, but since the boundary atoms are treated by both QM and MM methods it is important that the classical energy expression be modified to avoid multiple counting of interactions.

This class of scheme has been adopted most widely for studies involving strongly ionic materials where the boundary region is treated by model- or pseudo-potentials. However, as discussed below in Section 3.5.1, a number of treatments designed for more covalent systems also eliminate link atoms, placing a re-parameterised atomic description or a frozen orbital at the site of the first MM atom.

3.1.2 Subtractive Schemes

Here the entire system is treated by MM, and a third calculation on the inner region at the MM level is performed to eliminate multiple counting[7]. The approach is generally applied to link-atom based schemes, in which case the total energy may be written:

$$E^{E}_{QM/MM} = E^{E}_{MM} + E^{I,L}_{QM} - E^{I,L}_{MM} \tag{3}$$

The coupling term $E^{I,O,L}_{QM-MM}$ is no longer required as all interactions between inner and outer regions are handled at the MM level of theory, in the E^{E}_{MM} term. The handling of link atom corrections here occurs implicitly as a result of the subtraction. It is necessary that the forces in the link region arising from the difference between the QM and MM representations ($E^{I,L}_{QM} - E^{I,L}_{MM}$) remain small for all reasonable positions of the link atoms. It is therefore particularly valuable in this case to use a forcefield designed to reproduce forces at the particular QM approximation used for the inner region.

If the process under investigation involves changes in chemical bonding it will become more difficult to provide a suitable forcefield. However, if the inner region is large enough, it is possible to ensure that the contribution to the total energy from atoms in the interior of the inner region completely cancel when the subtraction $E^{E}_{MM} - E^{I,L}_{MM}$ is performed, and there is therefore no requirement for the forcefield to model the energetics of the reaction. Nevertheless, the forcefield must be able to compute the interaction between the reacting centre and the outer region at the MM level of theory, which requires, for example, partial charges for the former. Since the charge density of the inner region may change during the course of the reaction this can be a demanding requirement. Subtractive schemes are clearly not suitable for cases in which the electronic structure of the QM region is expected to be significantly perturbed by interaction with the environment. However, in most application areas explored so far this approximation has not proved problematic[14]. Where good quality forcefields are available the approach can be very accurate since there are no problems with interactions between the link atom region and the classical environment.

To date the main applications have been in the areas of organometallic[15] and zeolite chemistry[16,14,17] This scheme is actually quite general and can be used to coupling different levels of QM theory, as exemplified by the IMOMO [18,19] scheme of Morokuma.

The subtractive and additive models as defined here are closely related, as discussed by Bakowies and Thiel[7], where the subtractive model is used as a starting point for the derivation of link-atom corrected additive model.

3.2 The Choice of QM model

The choice of QM method will not be dealt with in detail here as it does not a fundamentally affect the design of a QM/MM scheme and will largely be governed by the same critera that apply to pure QM calculations.

Since the first Warshel and Levitt[2], study schemes based on semi-empirical methods have dominated the field for biological applications, and for reasons of computational cost such schemes are likely to remain important for applications incorporating molecular dynamics.

Approaches incorporating parameterised Hamiltonians include the MOLARIS implementation[20] (using an empirical valence bond (EVB) scheme) and the MM-VB scheme of Bernardi, Robb and co-workers[21,22]

A large number of *ab-initio* schemes based on Hartree-Fock[5,10,23,24] and density functional[9,25,26,11,16] approaches have been implemented. Recently a number of

approaches based on Car-Parrinello DFT codes have been reported[27,28,30].

3.3 The Choice of MM model

The significance of the choice of MM scheme depends on whether the additive or subtractive schemes are chosen, since within the subtractive scheme any forcefield can be used (see the discussion in Section 3.1.2).

Within additive schemes, type of MM model can have significant influence on the treatment of the boundary, since different classical approaches differ markedly with respect to the handling of both bonded and non-bonded interactions. The most important distinction is that between:

- valence force fields, exemplified by the biomolecular force fields (CHARMM[31], AMBER[32]) and a number of more general purpose forcefields including MM3[33] and the consistent force field (CFF[34]) constructed from energy terms such as bond stretches, angle bends *etc.*

- ionic forcefields in which the principal terms are the electrostatic and short-range (van der Waals) forces, exemplified by force fields based on the shell model[35,36,14].

The choice affects the construction of the MM model in two major respects:

1. The choice of forcefield influences the atomic partial charges thus affecting the long-range QM/MM interactions, as for most current implementations the same charges are used for MM...MM and MM...QM interactions (in principle it would be possible to construct schemes based on two sets of MM charges but this approach has not been widely adopted). For a given material, (for example the zeolites[37,38]) the shell model forcefields tend to be based on larger charges than those generated by fitting to the electrostatic potential. In many case the ionic forcefields employ formal ionic charges.

2. Handling of bonding and close interionic contacts between QM and MM regions will generally follow the same approach as treatment of similar interactions within the MM region. In the valence forcefield case it is easy to identify the terms involved, typically bond-stretch, angle-bend and torsion terms that are needed, and it is simple to delete those that correspond to terms handled by the QM interaction. For ionic force-fields the short-range QM..MM attractive terms will come from the presence of the MM charges in the QM Hamiltonian, and cannot readily be separated from the long-range interactions.

For these reasons, additive schemes based on link atoms are easier to construct with valence forcefields. The ionic class of forcefields can be used in boundary-region additive schemes, but only if the forcefield charges can generate the correct electrostatic potential in the QM region and thus the correct interionic forces. This is more likely to be true for highly ionic materials, unless significant parameterisation of the boundary region is carried out.

3.4 Handling of the QM/MM non-bonded coupling terms

Within most classical modelling schemes, the non-bonded interactions comprise electrostatic and short-range (or van der Waals) forces. In the context of the additive QM/MM schemes, the same decomposition is applied to the non-bonded interaction between QM and MM centres.

3.4.1 Short-range or Lennard-Jones terms

The treatment of the short-range QM/MM interaction generally follows the model used in the MM calculation. Re-fitting of the non-bonded parameters is often carried out, particularly in the case of solvation studies[39,40,41] or where the details of non-bonded contacts are particulary important[42].

3.4.2 Electrostatic terms

Bakowies and Thiel[7] defined three ways of treating QM/MM electrostatic interaction, labelled A-C, as follows.

A mechanical embedding, in which the QM calculation is essentially performed in the gas phase, without electronic coupling to the environment. The electrostatic interaction between QM and MM regions is either omitted or performed by the MM code, using a classical point charge model for the QM charge distribution (*e.g.* a potential derived charge model).

B electrostatic embedding, in which the classical partition appears as an external charge distribution (*e.g.* a set of point charges) in the QM Hamiltonian. The polarisation of the QM region by the MM charge distribution thus occurs as part of the QM electronic structure calculation. The partial charges used to describe the MM distribution are frequently taken to be those used in the forcefield [5,6,43], relying on the use of electrostatic properties in the forcefield charge derivation. When using an aluminosilicate CFF forcefield[37] for electrostatic QM/MM modelling of zeolites[44] it was found necessary to replace the original MM charges with those derived by fitting to electrostatic potentials. Charge equilibration schemes, which determine the MM charges as a function of geometry have also been employed[45,46].

   In *ab-initio* schemes it is clear that the electrostatic embedding scheme should be implemented, at least at long range, by adding the contribution of the MM point charges to the 1-electron Hamiltonian. However, within the semi-empirical formalism the definition of the electrostatic potential is more ambiguous as a result of the overlap approximations used, and alternative formulations for the 1-electron integral terms have been suggested[6,47,45].

C polarised embedding, in which the polarisation of the MM region in response to the the QM charge distribution is also included. Intuitively this makes most sense when the forcefield incorporates polarisation as unpolarised forcefields implicitly incorporate MM polarisation in their parameterisation, and care must be taken to ensure such implicit contributions to not occur in

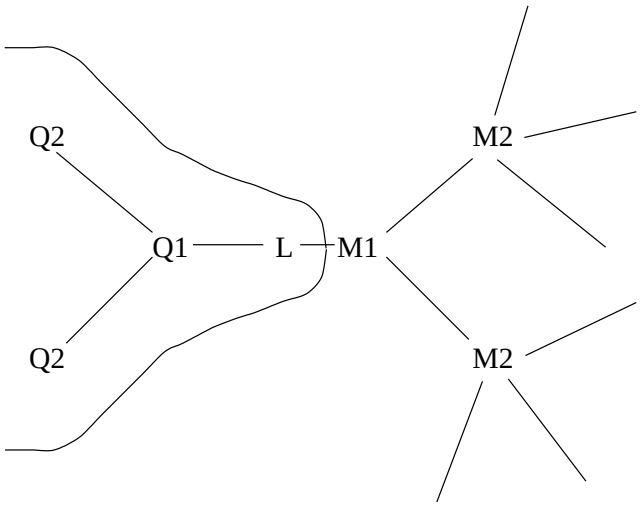


Figure 2. Labelling of atoms in the link atom region

the QM/MM potential. A variety of models for the classical polarisation are available, including the shell model[35], and coupled distributed atomic polarizabilities[48]. Polarisation of MM atoms close to the QM region (*e.g.* those connected by link-atom terminated bonds) were found to be unphysically large, leading to the suggestion that these atoms be treated as unpolarizable[7].

In model C the result of the change in MM charge distribution from the classical polarisation is not propagated to the QM calculation resulting in a non variational total energy.

D While not part of the original definition, model D[49] is defined as an extension to model C where QM and MM polarisations are made self-consistent, either by iterative solution of the SCF and polarizability problems[50], or by matrix inversion techniques, as exemplified by the the Direct Reaction Field (DRF) model[51,52,53,54,55].

3.5 Termination of the QM region

In those cases where there are bonds or strong ionic interactions between the QM and MM regions it is necessary to introduce some termination of the QM calculation, either through the link atom or boundary region approaches. Figure 2 indicates the labelling adopted in the discussion below for those models incorporating link atoms.

3.5.1 Chemical Nature of the Termination

For termination of sites where a covalent bond has been broken addition of a link atom is the most popular approach. An extra nuclear centre is introduced, together

with basis functions and electrons required to form a covalent bond to the QM region that will mimic the bond to the MM region. The simplest and still most popular choice is the use of a hydrogen atom[5].

There are clearly chemical differences between hydrogen and the chemical group it replaces. Within empirical and semi-empirical schemes, an obvious enhancement is to adjust the link atom parameterisation to mimic more closely the modelled group. Recently, Antes and Thiel[57] have described the semi-empirical Adjusted Connection Atom (ACA) scheme, in which the link atom is replaced with a boundary atom with parameters chosen to model a methyl group. A related approach within *ab-initio* based QM/MM methodologies is to place a pseudopotential at the MM site to mimic the electronic properties of the replaced bond[58].

For embedding treatments of highly ionic materials the main requirement of the termination is to stop the unphysical polarisation of the charge density of the QM cluster by the adjacent cations. "Leakage" of charge from the cluster will clearly occur to an increasing extent as the QM basis set is extended. Most treatments therefore include a pseudo- or model- potential at the cationic sites, without any valence electrons or basis functions[59,60]. Modelling the anions by model potentials[24,61] is a further refinement, which may be more important for materials with a more significant covalent character. Here the use of bare anionic charges might be expected to lead to poor structural predictions which can be ameliorated by adjustments to the potential.

All the approaches listed above assume that once the termination potential has been set up a full SCF calculation is performed on the resulting cluster. An alternative approach is to constrain the SCF solution to reflect the influence of the bonds that have been omitted. The local self-consistent field (LSCF) scheme[62,63] of Rivail and coworkers involves the preliminary calculation of the localised bond orbital, which is then frozen during the calculation. The generalised hybrid orbital (GHO) scheme of Gao *et al.*[64] constrains the hybridisation of the terminating centre but allows adjustment of the QM-MM bond itself, allowing changes in local geometry to be handled.

Similar approaches based on *ab-initio* wavefunctions have been less widely used to date, but include adaption of the LSCF scheme [65] and a recent implementation by Philipp and Friesner[43] which incorporates geometry changes at the boundary.

While the frozen hybrid orbital approaches promise to mimic the electronic properties of the extended system, problems with the calculation of the $E^{I,O}_{QM-MM}$ term in Equation 1 still remain. As described by Philipp and Friesner[43] it is probably necessary to combine the definition of the terminating orbitals with reparameterisation of the link atom region to obtain accurate conformational energies.

3.5.2 Coordinates and forces for the termination sites

In contrast to the boundary atom schemes, the appearance of additional QM centres in the link atom approaches leads to additional variability in the definition of the coordinates and forces. Inital coordinates can be chosen by placing the link atom on the bond that is being terminated, but once a geometry optimisation or molecular dynamics run has been started there are a number of different ways of updating

the coordinates and handling the forces.

1. Optimised Link Atoms

   In these approaches, the link atom coordinates are added to the atom list used in the geometry optimisation or molecular dynamics scheme, and the coordinates are free to vary[6].

   Sometimes some additional forcefield terms may be added to favour the positioning along the M1-Q1 bond[66] This is particularly important when link atom corrections or (*vide supra*) lead to small and unphysical forces on the link atoms.

2. Constrained Link Atom Procedures

   If the link atom coordinates can be written as a function of the real atom coordinates it is possible to eliminate them from the set of coordinates used in the optimisation or dynamics. Such an elimination is particularly desirable for molecular dynamics and the evaluation of vibrational frequencies, which would be modified by coupling to any independent link atom motions.

   Since the link atoms have non-zero forces it is necessary add a term to the real atom forces to account for the changes in the link atom position resulting from movement of the real atoms. In the IMOMM scheme of Maseras and Morokuma[68] this was implicitly performed by using working in internal coordinates such that the same internal coordinates were used to position the link atoms and to define the corresponding real atom (M1) coordinates. The internal coordinate force can then be obtained by adding the QM and MM contributions.

   When working in pure cartesian coordinates the same effect can be realised by using the chain rule to establish the contribution to the forces on the real atoms X_i.

$$\frac{\partial E}{\partial X_i} = \frac{\partial E}{\partial X_i} + \frac{\partial E}{\partial X_l} . \frac{\partial X_l}{\partial X_i} \tag{4}$$

   where the derivative $\frac{\partial X_l}{\partial X_i}$ is a 3x3 matrix describing the coupling of the link atom and real atom motions as a function of the constraint term. There is a term for each atom i which appears in the definition of the position of the link atom l, For the case of a link atom placed at a fixed position along the QM–MM bond there will be a correction term for the atom at each end of the bond in question (M1 and Q1 in Figure 2). This approach is used by the ChemShell implementation[69] and by the QM-Pot scheme of Sauer *et al.*[16]. A similar adaptation to the IMOMM scheme has been published by Woo *et al.*[70].

   When using constrained link atoms in additive QM schemes some adjustment to the forcefield is required. The MM force constant for the angle bend Q2-Q1-M1 is in effect supplemented by the bending potential for the group Q2-Q1-H,

since the restoring forces acting on H are transferred to atom M1 as described in equation 4. The approach adopted in the ChemShell scheme is to delete the MM term on the basis that the link atom bending potential will replace it. Greater accuracy could be achieved by fitting a modified MM parameter to reflect the presence of the link atom. Similar considerations apply to the torsion angles of the form Q3-Q2-Q1-M1. As a general rule, the Q1-H distance is kept fixed. It has been suggested that variation of the Q1-H distance provides a simple way to tune the electronic characteristics of the termination[69].

Similar considerations apply to the boundary atom schemes. It is generally assumed, or considered as a requirement for the boundary atom parameterisation, that the QM calculation incorporating a model potential will generate a suitable geometry for the short QM...MM contacts.

3.6 Modifications to the Classical charge distribution at the boundary

Within all of the polarised QM/MM schemes (models B-D in Section 3.4.2) problems can be expected when point charges modelling the MM region closely approach the QM region. In the absence of link atoms, close approach is usually prevented by the non-bonded interaction potential which is repulsive at short range. However in the region of bonds across the QM-MM boundary some adjustment to the classical charge distribution is essential, as the nearest point charges to the QM region will be at most a single bond distance away. In the case of terminating link atom (e.g. hydrogen) the link atom will be almost superimposed on the first classical atom (M1 in Figure 2).

Clearly this problem will be more severe when large basis sets are used, and in fact it is possible to disregard it in the semi-emprical case[6]. Antes and Thiel[71,49] have discussed a variety of approaches to the problem and suggested the L1-L3 classification included below.

3.6.1 Selective deletion of one-electron integral terms

For QM calculations with small basis sets the leading spurious interaction is that of the basis functions on the link atom with the classical MM charges. Since the link atom is an artefact of the QM/MM scheme it has been suggested that the model can be improved simply by deleting the 1-electron Hamiltonian contributions involving link atom basis functions and the full set of MM charges (scheme L1[49]).

Antes and Thiel also defined a further scheme, denoted L3, in which all 1-electron terms are included with the exception of those of the that involve basis functions on the link atom *and* the charge on centre M1.

3.6.2 Deletion of selected atomic charges

Perhaps the simplest way of dealing with the charge on the nearby classical centres is to delete them from the Hamiltonian. The QUASI Gaussian/AMBER scheme[5] omitted any MM charges less than 4 bonds removed from any QM atom. Waszkowycz *et al.*[8] modified this approach so that only a single MM charge at the M1 site was excluded.

The danger of these schemes is that simply deleting charges according to the connectivity will often result in the remaining MM atoms, as experienced by the QM region, appearing to have a net charge. Such an artefact will have particularly serious effect on computed energies for processes in which the total charge of the QM region is modified, such as protonation reactions.

Many biochemical forcefields have the feature that sets of neighbouring atomic charges can be grouped together such that the total group charge is an integer (usually zero)[72]. This is a convenient feature for a forcefield as it enables a molecule built by combining these charge groups to have an integral charge without any adjustment of the atomic charges being necessary. In the L2 scheme[49] the charges on the charge group containing atom M1 (in Figure 2) are neglected when building the QM Hamiltonian. While this will clearly remove some significant physical interactions, the fact that the charges removed will sum to zero will ensure that the total MM charge experienced by the QM calculation is correct. The leading term that is missing will be the dipole moment of the first charge group.

A series of tests on protonation reactions were used to evaluate the schemes L1 to L3[49], using semi-empirical, DFT and MP2 wavefunctions. The differences between the schemes were observed to be more pronounced for the *ab-initio* wavefunction, as expected from an analysis of the influence of the integral approximations[49]. L1 was recommended only at the semi-empirical level, for *ab-initio* wavefunctions the selective integral deletion was considered to lead to problems of imbalance in the electrostatics for the link atom region. The L2 scheme was found to be robust for all types of wavefunctions, and for *ab-initio* studies using forcefields based on neutral groups it appears the obvious choice. The results for L3 were less consistent and this scheme was not recommeded.

3.6.3 Charge Shifting Schemes

Within our calculations on zeolites[44,73,69] a different scheme has been adopted. Since the aluminosilicate forcefield used[37] does not consist of charge groups, there is no simple subset of MM atoms that can be deleted without associating an artificial total charge with the local MM environment.

The first stage of the approach is to adjust the charge on centre M1 to account for the deletion of the M1-Q1 bond. For an aluminosilicate forcefield with silicon charges of $+2x$ and oxygen charges of $-x$, each Si-O bond can be considered to contribute a charge of magnitude $0.5x$ to the atom at each end, since there are 2 bonds to each O and 4 to each Si. The idea of deriving electrostatic models from a sum of dipolar contributions from the bonds to each atom has been used for more general chemical sytems, for example the MM3 forcefield[33] uses an electrostatic model including bond dipole terms. For each QM-MM bond each atom M1 is involved in, the charge on M1 is reduced by $0.5x$, thus ensuring that the defect created in the MM lattice is electrically neutral.

Since the M1 centres still have finite charge, further adjustments are necessary. The approach adopted sets M1 to zero, but for each of the connected atoms M2 a charge adjustment is made to conserve the total charge of the MM system. As M1 and M2 will usually have charges of opposite sign the M2 atoms will have reduced

charges. To compensate for the dipole that has been created by the charge shift a pair of equal and opposite point charges are added close to each M2 centre along the M1-M2 bond direction. The resulting MM charge distribution therefore has the same charge and dipole moment to that of the MM defect created by the first stage, but the charge distribution close to the link atoms has been eliminated.

3.6.4 Gaussian Blur

Brooks has suggested[66] that the problems of close approach of the MM charges to the QM region are largely the result of the representation of this charge distribution by a point charge model. The "Gaussian blur" approach replaces the point charges for selected MM centres with a Gaussian charge distribution. An implementation of this approach[67] is included in the coupling of GAMESS-UK[84] and CHARMM[31].

4 The Issue of Conformational Complexity

One reason for the slow uptake of the *ab-initio* based QM/MM methods has been the computational costs. Although the cost of each energy and gradient evaluation is similar to that for a calculation on a small-molecule cluster model, the number of energy and gradient evaluations needed is likely to be much greater as a result of the larger number of degrees of freedom of the system.

In favourable cases it is possible to concentrate on a particular conformation for the environment, perhaps by simulated annealing or a related conformational search technique, using a cheaper energy function. The QM/MM study can then proceed without any attempt to repeat the conformational search, and relaxation of the MM coordinates can be repeated for each geometry change of the QM core. An example is the IMOMM scheme[68] which restricts the main geometry optimisation loop to those coordinates describing the QM geometry. At each step of this optimisation the remaining (MM) coordinates were optimised. Since IMOMM is a mechanical embedding scheme this relaxation does not change the QM energy and the QM calculation can therefore be skipped. By ensuring that the derivative of the energy with respect to the MM coordinates was zero at all times, standard optimisation algorithms, including transition state (TS) searching could be used. In favorable cases the number of QM calculations required will be similar to a pure QM calculation.

A related approach, but based on cartesian coordinates and polarised embedding was adopted by Turner *et al.*[12] for geometry optimisation and transition state determination for reactions in enzymes and solution. The MM relaxation was performed in the field of a classical model for the QM region, obtained by fitting to the QM contribution to the forces on the MM atoms at each QM geometry. As well as the savings in computational cost that arise from the reduced number of QM evaluations, the division of the coordinate set also proves useful for TS optimisations. If the TS search can be restricted to a reduced number of coordinates the stability of algorithms such as P-RFO[74] can be improved. This is especially true when the MM coordinate set is characterised by many soft eigenmodes and incipient transion states, as the case for solvated systems.

Recently Billeter *et al.*[75] have developed a geometry optimisation scheme based on delocalised internal co-ordinates[76]. The decomposition of the system into two or three sets of coordinates[12] is extended to an arbitrary number of residues, each subject to optimisation in turn. This "divide and conquer" approach enables linear scaling properties with system size to be obtained.

In cases where a number of QM/MM energies and forces are required for geometries with similar QM structures it is possible to reduce the number of QM energy evaluations required, interpolation approaches have been suggested for this purpose[77].

5 Software Implementation

In many respects the issues governing implementation of QM/MM computer codes are similar to those associated with the individual QM and MM methods. Most of the coupling terms are readily computed using the machinery present in either the QM or MM packages. However, it is worth giving brief consideration to a couple of implementational issues.

5.1 Program architecture

Given that the starting point is working QM and MM codes, QM/MM implementations can be considered to fall into three groups:

(i) those based on classical modelling packages, with a QM code integrated as a force-field extension

(ii) those based on a QM packages, incorporating the MM environment as a perturbation

(iii) modular schemes in which a central control program is provided and a choice of both QM and MM methods is left open.

Probably the most popular approach to date has been (i), for the sound reason that the modelling tools present in a typical MM/MD package are well suited for manipulation of large, complex chemical systems. A good example of this approach is the series of QM implementations within the CHARMM package [6,10,11,67]. However, one area in which the functionality of a traditional macromolecular program may need enhancing is the search for transition states. Turner *et al.*[12] extended the QM/MM capabilities of CHARMM by adding a driver package (GRACE) with capabilities including eigenvector-following search methods, and on-the-fly charge model fitting.

Option (ii) is particulary well suited when the tools required are those associated with small-molecule quantum chemistry, for example internal (Z-matrix coordinates) and Hessian-based transition-state searching using active coordinates which involve only QM atoms, By keeping the environment stationary, or fully relaxed at each step, the conformational complexity is hidden and the problem resembles a QM optimisation. Morokuma's IMOMM scheme[68] has the appearance of being

designed in this way. A coupling of GAMESS-UK and AMBER[8] followed a similar philosophy. Breuer *et al.*[78] used the framework of the MSI Cerius-2 package to introduce a mechanically-coupled MM model for the environment in an implementation exploiting the RI-DFT and redundant internal coordinate optimisation capabilities of TURBOMOLE[79].

We will discuss option (iii) in a little more detail, not because it is intrinsically superior, but because the QM/MM development work at Daresbury has been concentrated on this approach, and we therefore have more experience with the benefits and pitfalls of the approach.

The benefits can be summarised as follows

(a) A modular construction offers the greatest flexibility, and is particularly valuable if the same program system is to be used for investigation of an number of different types of chemical systems, each requiring different classes of forcefields. A range of QM schemes can readily be supported within a common environment, and commercial packages can be exploited in some of the simpler models.

(b) If care is taken to ensure that the component packages are modified to a minimal extent this approach also promises to provide the best hope for substituting up-to-date versions of the QM and MM packages when they become available.

The drawbacks are also significant, including

(a) additional programming complexity is introduced by the need to try and generalise the interfaces

(b) the need to provide, as part of the system a set of tools (*e.g.* structure manipulation, geometry optimisation, MD) which can be used with any choice of QM and MM methods.

(c) there are efficiency implications of attempting to keep the QM and MM calculations independent, particulary with respect to data transfer between the programs.

Examples of systems of modular design include the coupling of AMBER[32] to CADPAC[81] and MNDO[82] by Antes and Thiel[49] and the ChemShell package from our Laboratory[83]. The ChemShell package currently has a range of interfaces, including GAMESS-UK[84], DL_POLY[86], MNDO94[82], TURBOMOLE[79], GULP[85], CHARMM[31], GROMOS[87] and MOPAC[80] Interfaces with a variety of codes including Gaussian94[88], CADPAC[81] and AMBER[32] are in various stages of development. Program control is provided by a scripting language (Tcl[89]), which can be used to construct dynamics simulation protocols and geometry optimisation algorthms (using a suite of standard 2nd-order algorithms). Geometry optimisation using the HDLCOpt scheme[75] is also under development within the QUASI[90] project.

5.2 HPC and parallel implementation

The approach The efficient exploitation of HPC resources provides a challenge to the construction of QM/MM schemes, particularly the those of modular design.

The significant overheads associated with loading programs onto a large parallel computer force the requirement that all of the composite programs be linked into a single executable image, an awkward process since many of the components are themselves very large and complex packages, with different parallelisation models. HPC developments of the ChemShell package are taking place as part of the QUASI project[90], and include integrating the parallel version of a number of the component codes.

6 Summary and Outlook

This brief review of the QM/MM approach has emphasised the variety of ways that QM and MM calculations can be combined. It is by no means exhaustive, and the subject is now expanding rapidly.

The subtractive schemes have the advantage of relative simplicity of implementation, and the fact that there is no need to validate the QM/MM interaction. The increased reliance on the forcefield is a potential problem but this is increasingly being addressed by forcefields derived from *ab-initio* data.

Additive approaches based on link atoms, whether simple implementations with rather *ad hoc* treatments of boundaries, or more complex schemes incorporating extensive QM/MM parameterisation, are likely to suffer for some time from a lack of user confidence. As may be clear from the number of variations that are possible it will probably be difficult to get exactly the same answer with two separarate implementations, and, like the forcefields themselves, the methodology will gradually gain acceptance on the basis of experience. Cross-checks between methods and validation studies (*e.g.* comparison with pure QM results) will continue to be important for some time to come.

One of the chief challenges will be exploration of conformational space, particularly in those applications areas for which semi-empirical methods are not applicable. Large scale parallel computing can be expected to play an increasingly important role here.

References

1. See for example the article of Head-Gordon in this volume.
2. A. Warshel and M. Levitt, Theoretical Studies of Enzymic Reactions: Dielectric, Electrostatic and Steric stabilisation of the carbonium ion in the reaction of Lysozyme. J. Mol. Biol. **103**, 227 (1976).
3. J. Gao, Methods and applications of combined quantum mechanical and molecular mechanical potentials, p.119-185 in *Reviews in Computational Chemistry*, VCH Publishers, Inc., New York (1996).
4. T.Z. Mordasini and W. Thiel, Combined quantum mechanical and molecular mechanical approaches, Chimia **52(6)**, 288-291 (1998).
5. U.C. Singh and P.A. Kollman, A combined *ab initio* quantum mechanical and molecular mechanical method for carrying out simulations on complex molecular systems: Applications to the CH_3Cl+Cl^- exchange reaction and gas-phase protonation of polyethers, J. Comp. Chem. **7(6)**, 718-730 (1986).

6. M.J. Field, P.A. Bash, and M. Karplus, A Combined Quantum Mechanical and molecular mechanical potential for molecular dynamics simulation, J. Comp. Chem. **11(6)**, 700-733 (1990).
7. D. Bakowies and W. Thiel, Hybrid models for combined quantum mechanical and molecular mechanical approaches., J. Phys. Chem. **100(25)**, 10580-10594 (1996).
8. B. Waszkowycz, I.H. Hillier, N. Gensmantel, and D.W. Payling, Combined quantum mechanical-molecular mechanical study of catalysis by the enzyme phospholipase-A2 – an investigation of the potential-energy surface for amide hydrolysis, J. Chem. Soc., Perkin Trans. 2 , 2025-2032 (1991).
9. R.V. Stanton, D.S. Hartsough, and K.M. Merz, Calculation of solvation free energies using a density functional/molecular dynamics coupled potential, J. Phys. Chem. **97**, 11868-11870 (1993).
10. Y.S. Lee, M. Hodoscek, B.R. Brooks, P.F. Kador, Catalytic mechanism of aldose reductase studied by the combined potentials of quantum mechanics and molecular mechanics, Biophys. Chem. **70(3)**, 203-216 (1998).
11. P.D. Lyne, M. Hodoscek and M. Karplus, A hybrid QM-MM potential employing Hartree-Fock or density functional methods in the quantum region, J. Phys. Chem. A **103(18)**, 3462-3471 (1999).
12. A.J. Turner, V. Moliner, and I.H. Williams, Transition-state structural refinement with GRACE and CHARMM: Flexible QM/MM modelling for lactate dehydrogenase, Phys. Chem. Chem. Phys. **1(6)**, 1323-1331 (1999).
13. K.P. Eurenius, D.C. Chatfield, B.R. Brooks, and M. Hodoscek, Enzyme mechanisms with hybrid quantum and molecular mechanical potentials .1. Theoretical considerations, Int. J. Quant. Chem. **60(6)**, 1189-1200 (1996).
14. M. Sierka and J. Sauer, Structure and reactivity of silica and zeolite catalysts by a combined quantum mechanics-shell-model potential approach based on DFT, Faraday. Discuss. **106**, 41-62 (1997).
15. M. Svensson, S. Humbel, R.D.J. Froese, T. Matsubara, S. Sieber, and K. Morokuma, ONIOM: A Multilayered Integrated MO+MM Method for geometry optimizations and single point energy predictions. A test for Diels-Alder reactios and $Pt(P(t\text{-}Bu)_3)_2 + H_2$ oxidative addition, J. Phys. Chem. **100**, 19357-19363 (1996).
16. U. Eichler, C.M. Kölmel, and J. Sauer, Combining *ab initio* techniques with analytical potential functions for structure predictions of large systems: Method and application to crystalline silica polymorphs, J. Comp. Chem. **18(4)**, 463-477 (1997).
17. D. Nachtigallova, P. Nachtigall, M. Sierka, and J. Sauer, Coordination and siting of Cu^+ ions in ZSM-5: A combined quantum mechanics interatomic potential function study, Phys. Chem. Chem. Phys **1(8)**, 2019-2026 (1999).
18. S. Humbel, S. Seiber, and K. Morokuma, The IMOMO method: Integration of different levels of molecular orbital approximations for geometry optimisation of large systems: Test for *n*-butane conformation and S'_n2 reaction: $RCl+Cl^-$, J. Chem. Phys. **105(5)**, 1959-1967 (1996).
19. R.D.J. Froese, S. Humbel, M. Svensson, and K. Morokuma, IMOMO(G2MS): A new high-level G2-like method for large molecules and its application to

Diels-Alder Reactions, J. Phys. Chem. A **101**, 227-233 (1997).
20. A. Warshel and S. Creighton, MOLARIS, in *Computer Simulation of Biomolecular Systems* W.F. van Gunsteren, P.K. Weiner (Eds), ESCOM, Leiden, 1989, vol 2, pp. 120.
21. F. Bernardi, M. Olivucci, and M.A. Robb, Simulation of MC-SCF results on covalent organic multibond reactions - molecular mechanics with valence bond (MM-VB), J. Am. Chem. Soc. **114(5)**, 1606-1616 (1992).
22. M.J. Bearpark, P. Celani, F. Jolibois, M. Olivucci, M.A. Robb, and F. Bernardi, Characterization of the indacene S-0/S-1 conical intersection: an MMVB and CASSCF study, Molec. Phys. **96**, 645-652 (1999).
23. J.H. Harding, A.H. Harker, P.B. Keegstra, R. Pandey, J.M. Vail and C. Woodward, Hartree-Kock cluster computations of defect and perfect ionic-crystal properties, Physica B **131(1-3)**, 151-156 (1985).
24. Z. Barandiaran and L. Seijo, The *ab-initio* model potential representation of the crystalline environment. Theoretical study of the local distortion on $NaCl^{-}Cu^{+}$, J. Chem. Phys. **89(9)**, 5739-5746 (1988).
25. D. Wei and D.R. Salahub, A combined density functional and molecular dynamics simulation of a quantum water molecule in aqueous solution, Chem. Phys. Lett. **224**, 291 (1994).
26. X.P. Long, J.B. Nicholas, M.F. Guest, and R.L. Ornstein, A combined density functional theory molecular mechanics formalism and its application to small water clusters, J. Mol. Struct. **412(1-2)**, 121-133 (1997).
27. T.K. Woo, P.M. Margl, P.E. Blöchl, and T. Zeigler, A Combined Car-Parrinello QM/MM implementation for *ab-initio* molecular dynamics simulations of extended systems: Application to transition metal catalysis, J. Phys. Chem. **101(40)**, 7877-7880 (1997).
28. U. Rothlisberger, Hybrid Car-Parrinello simulations of enzymatic reactions. Presentation[29], for abstract see J. Mol. Graphics. Mod. **16(4-6)**, 275-276 (1998).
29. Molecular Graphics and Modelling Society meeting: *MM/QM methods and applications*, Southampton, 14-16 April, 1999.
30. M. Eichinger, P. Tavan, J. Hutter, and M. Parrinello, A hybrid method for solutes in complex solvents: Density functional theory combined with empirical force fields, J. Chem. Phys **110(21)**, 10452-10467 (1999).
31. B.R. Brooks, R.E. Bruccoleri, B.D. Olafson, D.J. States, S. Swaminathan, and M. Karplus CHARMM - A program for macromolecular energy, minimization, and dynamics calculations, J. Comp. Chem. **4(2)**, 187-217 (1983).
32. W.D. Cornell, P. Ciepak, C.I. Bayly, I.R. Gould, K. Merz, D.M. Ferguson, D.C. Spellmeyer, T. Fox, J.W. Caldwell, and P. Kollman, A 2nd generation force-field for the simulation of proteins, nucleic-acids, and organic-molecules, J. Am. Chem. Soc. **117**, 5179 (1995).
33. The MM3 force-field for amides, polypeptides and proteins, J.H. Lii and N.L. Allinger, J. Comp. Chem. **12(2)**, 186-199 (1991).
34. J.R. Maple, M.-J. Hwang, T.P. Stockfish, U. Dinur, M. Waldman, C.S. Ewig, and A.T. Hagler, Derivation of class-II force-fields .1. Methodology and quantum force-field for the alkyl functional-group and alkane molecules, J. Comp.

Chem. **15(2)**, 162-182 (1994).
35. B.G. Dick and A.W. Overhauser, Chem. Rev. **112**, 90 (1958).
36. R.A. Jackson and C.R.A. Catlow, Molecular Simulation **1**, 207 (1988).
37. J.R. Hill and J. Sauer, Molecular mechanics potential for silica and zeolite catalysts based on *ab initio* calculations 2. Aluminosilicates, J. Phys. Chem. **99**, 9536-9550 (1995).
38. K.P. Schroder and J. Sauer, Potential functions for silica and zeolite catalysts based on ab initio calculations .3. A shell model ion pair potential for silica and aluminosilicates, J. Phys. Chem. **100(26)**, 11043-11049 (1996).
39. J. Gao and X. Xia, A Priori evaluation of aqueous polarization effects through Monte Carlo simulations, Science **258**, 631 (1992).
40. P.L. Cummins and J.E. Gready, Coupled semiempirical molecular orbital and molecular mechanics model (QM/MM) for organic molecules in aqueous solution, J. Comp. Chem. **18(12)**, 1496-1512 (1997).
41. Y.Q. Tu and A. Laaksonen, On the effect of Lennard-Jones parameters on the quantum mechanical and molecular mechanical coupling in a hybrid molecular dynamics simulation of liquid water, J. Chem. Phys. **111(16)**, 7519-7525 (1999).
42. G. Ujaque, F. Maseras, and O. Eisenstein, Different van der Waals radii for organic and inorganic halogen atoms: a significant improvement in IMOMM performance, Theor. Chem. Acc. **96**, 146-150 (1997).
43. D.M. Philipp and R.A. Friesner, Mixed *ab initio* QM/MM modeling using frozen orbitals and tests with alanine dipeptide and tetrapeptide, J. Comp. Chem **20(14)**, 1468-1494 (1999).
44. P. Sherwood, A.H. de Vries, S.J. Collins, S.P. Greatbanks, N.A. Burton, Vincent, and I.H. Hillier, Computer simulation of zeolite structure and reactivity using embedded cluster methods, Faraday. Discuss. **106**, 79-92 (1997).
45. D. Bakowies and W. Thiel, Semiempirical treatment of electrostatic potentials and partial charges in combined quantum mechanical and molecular mechanical approaches, J. Comp. Chem. **17(1)**, 87-108 (1996).
46. R.A. Bryce, M.A. Vincent, and I.H. Hillier, Binding energy of $F(H_2O)(^-)$ and the simulation of fluoride water clusters using a hybrid QM/MM (fluctuating charge) potential, J. Phys. Chem. A **103(20)**, 4094-4100 (1999).
47. P.L. Cummins and J.E. Gready, The electrostatic potential in the semiempirical molecular orbital approximation, Chem. Phys. Lett. **225**, 11-17 (1994).
48. B.T. Thole, Chem. Phys. **59**, 341 (1981).
49. I. Antes and W. Thiel, In *Combined Quantum Mechanical and Molecular Mechanical Methods*; J. Gao and M.A. Thompson (eds), ACS Symp. Ser., vol. **712**, American Chemical Society, Washington, DC 1998, p. 50.
50. M.A. Thompson, QM/MMpol: A consistent model for solute/solvent polarization. Application to the aqueous solvation and spectroscopy of formaldehyde, acetaldehyde, and acetone, J. Phys. Chem. **100(34)**, 14492-14507 (1996).
51. B.T. Thole and P.Th. van Duijnen, On the quantum mechanical treatment of solvent, Theor. Chim. Acta. **55**, 307 (1980).
52. A.H. de Vries, P.Th. van Duijnen, A.H. Juffer, J.A.C. Rullmann, J.P. Dijkman, H. Merenga, and B.T. Thole, Implementation of reaction field methods in

quantum chemistry computer codes, J. Comp. Chem. **16(1)**, 37-55 (1995), correction in J. Comp. Chem. **16(11)**, 1445-1446 (1995).
53. P.Th. van Duijnen and A.H. de Vries, Direct reaction field force field: A consistent way to connect and combine quantum-chemical and classical descriptions of molecules, Int. J. Quant. Chem. **60(6)**, 1111-1132 (1996).
54. A.H. de Vries, P.Th. van Duijnen, Solvatochromism of the $\pi^* \leftarrow n$ transition of acetone by combined quantum mechanical classical mechanical calculations, Int. J. Quant. Chem. **57(6)**, 1067-1076 (1996).
55. P.T. van Duijnen, F. Grozema, and M. Swart, Some applications of the direct reaction field approach, J. Mol. Struct. (THEOCHEM) **464(1-3)**, 191-198 (1999).
56. F.C. Grozema, R.W.J. Zijlstra, and P.T. van Duijnen, Many-body interactions calculated with the direct reaction field model, Chem. Phys. **246(1-3)**, 217-227 (1999).
57. I. Antes and W. Thiel, Adjusted connection atoms for combined quantum mechanical and molecular mechanical methods, J. Phys. Chem. A **103(46)**, 9290-9295 (1999).
58. Y.K. Zhang, T.S. Lee, W.T. Yang, A pseudobond approach to combining quantum mechanical and molecular mechanical methods, J. Chem. Phys. **110(1)**, 46-54 (1999).
59. N.W. Winter and R.M. Pitzer, Configuration-Interaction calculation of the electronic-spectra of MgF_2-V^{+2}, J. Chem. Phys. **89**, 446 (1988).
60. G. Pachioni, A.M. Ferrari, and G. Ierano, Cluster model calculations of oxygen vacancies in SiO_2 and MgO, formation energies, optical transitions and EPR spectra, Faraday. Discuss. **106**, 155-172 (1997).
61. M.A. Nygren, L.G.M. Pettersson, Z. Barandiaran, and L. Seijo, Bonding between CO and the MgO(001) surface - a modified picture, J. Chem. Phys. **100(3)**, 2010-2018 (1994).
62. V. Thery, D. Rinaldi, J.-L. Rivail, B. Maigret, and G.G. Ferenczy, Quantum mechanical computations on very large molecular systems: The local self-consistent field method, J. Comp. Chem. **15(3)**, 269-282 (1994).
63. G. Monard, M. Loos, V. Thery, K. Baka, and J.L. Rivail, Hybrid classical quantum force field for modeling very large molecules, Int. J. Quant. Chem. **58(2)**, 153-159 (1996).
64. J. Gao, P. Amara, C. Alhambra, and M.J. Field, A Generalised Hybrid Orbital (GHO) Method for the treatment of boundary atoms in combined QM/MM calculations, J. Phys. Chem. A **102**, 4714-4721 (1998).
65. X. Assfeld and J.L. Rivail, Quantum chemical computations on parts of large molecules: The *ab initio* local self consistent field method, Chem. Phys. Lett. **263(1-2)**, 100-106 (1996).
66. B.R. Brooks, personal communication.
67. P. Sherwood, E. Billings, and B.R. Brooks, unpublished.
68. F. Maseras and K. Morokuma, IMOMM: A new integrated *ab-initio* + molecular mechanics geometry optimisation scheme of equilibrium structures and transition states, J. Comp. Chem. **16(9)**, 1170-1179 (1995).
69. A.H. de Vries, P. Sherwood, S.J. Collins, A.M. Rigby, M. Rigutto, and

G.J. Kramer, Zeolite structure and reactivity by combined quantum-chemical-classical calculations, J. Phys. Chem. B **103(29)**, 6133-6141 (1999).
70. T.K. Woo, L. Cavallo, T. Ziegler, Implementation of the IMOMM methodology for performing combined QM/MM molecular dynamics simulations and frequency calculations, Theor. Chem. Acc **100(5-6)**, 307-313 (1998).
71. I. Antes, Ph.D Thesis, University of Zurich (1998).
72. P.E. Smith and W.F. van Gunsteren in *Computer Simulation of Biomolecular Systems*, W.F. van Gunsteren, P.K. Weiner, and A.J. Wilkinson (Eds), ESCOM, Leiden, 1993, vol. **2**, pp. 182-212.
73. P.E. Sinclair, A. de Vries, P. Sherwood, C.R.A. Catlow, and R.A. van Santen, Quantum-chemical studies of alkene chemisorption in chabazite: A comparison of cluster and embedded-cluster models, J. Chem. Soc., Faraday Trans **94(22)**, 3401-3408 (1998).
74. J. Baker An Algorithm for the location of transition states, J. Comp. Chem. **7(4)**, 385-395 (1986), and references therein.
75. S.R. Billeter, A.J. Turner and W. Thiel, Linear Scaling Geometry Optimisation and Transition State Search in Hybrid Delocalized Internal Coordinates, *Phys. Chem. Chem. Phys.*, in press.
76. J. Baker, A. Kessi and B. Delley, The generation and use of delocalised internal coordinates in geometry optimisation, J. Chem. Phys **105(1)**, 192-206 (1996).
77. C.D. Berweger, W.F. van Gunsteren, F. Müller-Plathe, Finite element interpolation for combined classical/quantum mechanics dynamics simulations, J. Comp. Chem. **18(12)**, 1484-1495 (1997).
78. K. Breuer, J.H. Teles, D. Demuth, H. Hibst, A. Schafer, S. Brode, and H. Domgorgen, Zinc silicates: Very efficient heterogeneous catalysts for the addition of primary alcohols to alkynes and allenes, Angew. Chem. Int. Ed. Engl. **38(1)**), 1401-1405 (1999).
79. R. Ahlrichs, M. Bär, M. Häser, H. Horn and C. Cölmel, Chem. Phys. Lett **162**, 165 (1989).
80. MOPAC 5, J.J.P. Stewart.
81. CADPAC5.2, The Cambridge Analytic Derivatives Package, issue 5.2, Cambridge, 1992. A suite of quantum chemistry programs developed by R.D. Amos with contributions from I.L. Alberts, S.M. Colwell, N.C. Handy, D. Jayatilaka, P.J. Knowles, R. Kobayashi, N. Koga, K.E. Laidig, P.E. Maslen, C.W. Murray, J.E. Rice, J. Sanz, E.D. Simandiras, A.J. Stone, and M.-D. Su.
82. W. Thiel, Program MNDO94; University of Zürich, 1995.
83. ChemShell Users Manual, P. Sherwood and A.H. de Vries, CLRC Daresbury Laboratory 1997.
84. GAMESS-UK, see `http://www.dl.ac.uk/CFS`
85. J.D. Gale, J. Chem. Soc. Faraday Trans. **93**, 629 (1997), J.D. Gale, *Program GULP (General Utility Lattice Program)* Royal Institution/Imperial College, London, 1992.
86. W. Smith and T. Forester, DL_POLY 2.0: A general purpose parallel molecular dynamics simulation package, J. Mol. Graph. **14**, 136 (1996).
87. W. F. van Gunsteren and S. R. Billeter, A. A. Eising, P. H. Hünenberger, P. Krüger, A. E. Mark, W. R. P. Scott and I. G. Tironi, *Biomolecular Sim-*

ulation: The GROMOS96 Manual and User Guide, Biomos b.v., Zürich and Groningen, VdF Hochschulverlag, ETH Zürich, 1996, and W. R. P. Scott and P. H. Hünenberger, I. G. Tironi, A. E. Mark, S. R. Billeter, J. Fennen, A. E. Torda, T. Huber and P. Krüger and W. F. van Gunsteren, The GROMOS Biomolecular Simulation Program Package, J. Phys. Chem. A **103(19)**, 3596-3607 (1999).
88. Gaussian 94, See `http://www.gaussian.com`
89. J.K. Ousterhout, *Tcl and the Tk toolkit*, Addison Wesley, 1994.
90. QUASI - Quantum Simulation in Industry, EU project 25047. For a summary of the project please see `http://www.cse.clrc.ac.uk/Activity/QUASI`

SUBSPACE METHODS FOR SPARSE EIGENVALUE PROBLEMS

BERNHARD STEFFEN
John von Neumann Institute for Computing
Central Institute for Applied Mathematics
Research Centre Jülich, 52425 Jülich
Germany
E-mail: b.steffen@fz-juelich.de

Subspace methods are the methods of choice for calculating a few eigenvalues and -vectors of a large matrix. They may also be considered for completely diagonalizing a matrix if it is either very sparse or too large to be stored. A subspace method for a $n \times n$ matrix A consists of a scheme to extract approximations to some eigenvalues and -vectors of A from the action of A onto a subspace $\mathbf{V} \subset \mathbb{C}^{\mathbf{n}}$ and a method to update $\mathbf{V}$. We will present extraction schemes for extremal (Ritz projection) as well as inner (residual minimization, harmonic Ritz projection) eigenvalues and discuss advanced update schemes.

1 Introduction

The problem of calculating some or all eigenvalues and -vectors of a large matrix appears in a wide range of applications from biology (meta stable states of ecosystems) to mechanical engineering (oscillations of suspension bridges). Theoretical chemistry methods have an outstanding position here by giving rise to extremely large matrices with very special properties, so that there are methods that are very successful there but rarely in use - and not very effective - with problems from other sources. As eigenvalue problems, and particularly eigenvalue problems from theoretical chemistry, make up a fair share of the supercomputer usage, there is good reason for analyzing and improving the algorithms and implementations as well as for teaching users to make the most of the method chosen.

The classification of eigenvalue problems distinguishes between different types of matrices (general, hermitian, complex symmetric, full, sparse, ...), different requirements (eigenvalues only, eigenvectors too, all eigenvalues, only extremal eigenvalues, certain part of spectrum, eigenvector similar to excitation vector, ...) and numerical properties (normal, diagonally dominant, clustered eigenvalues, ...). While these are extremely important for issues of efficiency and implementation, there are only three basic principles involved in the solution of eigenvalue problems:

Similarity transformations $A = S^{-1}JS$ converting A into some normal form (usually diagonal or Jordan) where eigenvalues can be extracted immediately and eigenvectors are given by columns of S. S is built up iteratively as a product of simple matrices (see[8]).

Nonlinear equation methods treat the problem directly as an $(n+1)$-dimensional nonlinear equation, searching for a solution in the neighborhood of $\bar{\lambda}$. Methods in use are 'shift and invert' and polynomial iteration $x^n = P_n(A)x^{n-1}$, where the polynomials P_n are chosen such that the sequence x^n converges, e.g. to the eigenvector of the largest eigenvalue.

Subspace methods where the action of A on a low (e.g. m) dimensional sub-

space $\mathbf{V}$ is considered. May $\{w_i\}$ be an orthonormal basis of $\mathbf{V}$ and $\{w_i, y_i\}$ of span($\mathbf{V}$, $A\mathbf{V}$).
For each vector $w := a_1 w_1 + \ldots + a_m w_m$ we have
$Aw := b_1 w_1 + \ldots + b_m w_m + c_1 y_1 + \ldots + c_m y_m$. The mapping of $\{a_i\}$ onto $\{b_i\}$ is given by the orthogonal projection of A onto $\mathbf{V}$: $\hat{A} = W^T A W$, where the $\{w_i\}$ form the columns of W. Obviously, any eigenvector of A contained in $\mathbf{V}$ gives an eigenvector of $\hat{A}$ to the same eigenvalue, and conversely, the eigenpairs of $\hat{A}$ give approximations to those of A (see[10]). For hermitian A, these approximations are almost[11] optimal with respect to $\mathbf{V}$ for the extremal eigenvalues of A (Ritz projection); for non hermitian A, they are still about the best thing available. For inner eigenpairs, things are complicated. Reliable approximations λ_i to the inner eigenvalues near a value $\bar{\lambda}$ can be constructed either from the inverse of the orthogonal projection of $(A - \bar{\lambda} I)$ onto $\mathbf{V}$ (harmonic Ritz projection) or by minimizing some function $f((A - \lambda I)v, \lambda - \bar{\lambda})$ over $v \in \mathbf{V}$ and $\lambda \in \mathbb{C}$ (residual minimization). With the information from the $\bar{x_i}$ and $(A - \bar{\lambda_i} I)\bar{x_i}$, a new (possibly larger and hopefully better) subspace $\mathbf{V}'$ can be constructed.
The subspace methods differ primarily in the update procedure. New directions are added to $\mathbf{V}$, old directions may be removed. The new directions may be random (not recommended), $A\mathbf{V}$, (Lanczos, Arnoldi[4,6]), $(A - \bar{\lambda} I)^{-1} \bar{x_i}$, or various approximations to $(A - \bar{\lambda} I)^{-1}((A - \lambda_i I)\bar{x_i})$[5,7,13,3]. The latter are the most interesting methods and – notably for theoretical chemistry problems – give the best performance.

All subspace methods separate the computations into a low dimensional nonlinear problem and a number of linear algebra operations (matrix × vector and scalar products) in n dimensions. The latter make up only a small part of the code, but take most of the time. The implementation has to be carefully tuned to reduce the number of operations in n dimensions, low dimensional computations being of minor importance. The linear algebra operations should be carefully optimized (e.g. using BLAS 3), and they allow a large amount of parallelization, too. Parallelizing the low dimensional part is in the process of development, but not really useful yet[3].

2 Eigenvalue extraction

The goal of the extraction part is to find in a given subspace vectors that are good approximations of eigenvectors of A as well as the corresponding approximations of the eigenvalues. The description is independent of the update method, while the implementation shows some interdependence.

2.1 Ritz projection

The Ritz projection is the most important approach to extract eigenvalue and -vector approximations from a given subspace. The basic idea[11] is: May $\mathbf{V^{(k)}}$ be a subspace of $\mathbb{R}^n$ at iteration step k with an orthonormal basis $\vec{w}_1^{(k)}, \ldots, \vec{w}_m^{(k)}$ and $W^{(k)}$ the matrix with columns $\vec{w}_j^{(k)}$, $S^{(k)} := (W^{(k)})^T A W^{(k)}$, $\bar{\lambda}_j^{(k)}$ the eigenvalues of $S^{(k)}$, and $T^{(k)}$ a matrix with the eigenvectors of $S^{(k)}$ as columns. The columns

$\vec{x}_j^{(k)}$ of $W^{(k)}\,T^{(k)}$ (the Ritz vectors) are approximations to eigenvectors of A with the Ritz values $\bar{\lambda}_j^{(k)} = (\vec{x}_j^{(k)})^T\,A\,\vec{x}_j^{(k)}$ approximating eigenvalues of A. If the subspace allows a good approximation of the extremal eigenvectors of A, the corresponding Ritz vectors will be close to optimal approximations[11].

Example 1: Let A be a diagonal matrix and let $\mathbf{V^{(k)}}$ contain a good approximation to the largest eigenvector of A:

$$A := \begin{bmatrix} -1000 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 6 \end{bmatrix} \qquad \mathbf{V^{(k)}} := \mathit{span}\ \left(\begin{bmatrix} 0.001 & -0.005 \\ 0.1 & -1.0 \\ 0 & 1.0 \\ 1 & 0.001 \end{bmatrix} \right)$$

The approximations to $[0,0,0,1]$ calculated by Ritz projection will be $[.00066, .0320, .0677, .9972]$, near optimal, and the Ritz value 5.9929.

For hermitian A, the Ritz vectors are forced to be orthogonal, while the projections of the eigenvectors of A onto $\mathbf{V^{(k)}}$ will not be orthogonal. Now, the Ritz vector $\vec{x}_1^{(k)}$ to the smallest (largest) Ritz value may be askew to all eigenvectors of A. $\vec{x}_2^{(k)}$ will be orthogonal to $\vec{x}_1^{(k)}$, therefore even if a non extremal eigenvector of A has a good approximation in $\mathbf{V^{(k)}}$, this may not be orthogonal to $\vec{x}_1^{(k)}$ and therefore not be close to a Ritz vector. Therefore, the second eigenvector of A can be expected to be well approximated only if the extremal one has at least a decent approximation, and inner eigenvectors of A may be poorly approximated even if a good approximation is contained in $\mathbf{V^{(k)}}$. This effect is pronounced when there appear numerically multiple Ritz values.

Example 2: Let A be as before and exchange the last two rows in the basis vectors of $\mathbf{V^{(k)}}$, which now contains a good approximation to an inner eigenvector of A. The approximations to $[0,0,1,0]$ calculated by Ritz projection will be $[-.00184, -.4423, .7338, .5156]$, which is almost 43^0 off the desired eigenvector and much inferior to the starting approximation. The Ritz value of 4.92 is almost correct. The situation is not necessarily improved by improving the subspace. Changing the first column of $\mathbf{V^{(k)}}$ to $[0.0001, 0.01, 1, 0]$ does not help.

2.2 Methods for interior eigenvalues

If Ritz projection performs poorly, inner eigenvalues may be approximated from a subspace containing a good approximation of the eigenvector by either of two methods depending on information available. Instead of calculating the projection of A onto $\mathbf{V^{(k)}}$, an inverse projection of $(A - \bar{\lambda}I)^{-1}$ onto $\mathbf{W^{(k)}} := (A - \bar{\lambda}I)\mathbf{V^{(k)}}$ is calculated with only marginally increased effort. Now the formerly interior eigenvalues transform to extremal ones, and if $\bar{\lambda}$ is chosen properly, the corresponding eigenvector approximations (in $\mathbf{W^{(k)}}$) are good. Applying $(A - \bar{\lambda}I)^{-1}$ to these approximations is easy, just a linear combination of the basis vectors in $\mathbf{V^{(k)}}$, and yields good approximations to eigenvectors in the neighborhood of $\bar{\lambda}$ (harmonic Ritz projection, [13]). With $\bar{\lambda} = 4.9$, the previous example returns $[0.0017, 0.2390, .9605, -0.1429]$, a much better but not optimal approxima-

tion. Changing $\bar{\lambda}$ to 4.995 gives $[0.0013, 0.1630, 0.9845, -0.0646]$, quite good. This will work well when eigenvalues in a well-known range are looked for.
There are, however, problems where the eigenvalues are not known with sufficient accuracy, but approximations to the eigenvectors are known, e.g. from low accuracy computations or from observation. In this case, a residual minimization gives better results:
Let $\bar{v} \in \mathbf{V^{(k)}}$ be an approximation to an eigenvector $\bar{x}$ with $||\bar{v}|| = 1$. Choose $x^{(k)}$ as the local minimum of $||(A - x^T Ax\ I)x||$ for all $x \in \mathbf{V^{(k)}}$, $x^T v/(||x||||\bar{v}||) > \alpha$, that is closest to $\bar{v}$.
This means looking for an eigenvector that is closer to $\bar{v}$ than any other eigenvector. Usually, $\alpha > 0.9$ is safe, $\sqrt{0.5}$ being the theoretical lower limit that garantees that an eigenvector found almost accurately is not discarded of again. In the non-hermitian case, the threshold may have to be increased depending on the angle between eigenvectors of A. As this is a (low dimensional) nonlinear problem, some approximation is needed. A simple but almost always sufficient linearization is minimizing $||(A - \bar{v}^T A\bar{v}\ I)(\bar{v} + x)||$ over $x^T\bar{v} = 0$. This yields $[0.00006, -0.0861, .9791, .1840]^T$ for the above example, only slightly less accurate than harmonic Ritz projection with $\bar{\lambda} = 4.995$. Changing the $V[1, 2]$ to 0.0001 changes the picture, the harmonic Ritz projection will perform only slightly better than simple Ritz projection, while the residuum minimization will be near perfect. In general, residuum minimization is more robust than harmonic Ritz projection and therefore may be a good choice for starting steps, but it is in most cases tested inferior in final convergence.

3 Update procedures

The update procedure creates $\mathbf{V^{(k+1)}}$ from $\mathbf{V^{(k)}}$ by adding some directions and possibly reducing the dimensions again. The reduction of dimensions is increasing the number of iteration steps, but as the steps get computationally cheaper, there will usually be some gain in computing time. Even if not, the reduction in memory requirement may be helpful. The reduction – sometimes termed restart – usually retains the approximations to the eigenvectors required plus those to neighboring eigenvectors.

3.1 Krylow space updates of subspace

The simplest sequence of subspaces $\mathbf{V^{(k)}}$ is given by the Krylow construction $\mathbf{V^{(k)}} = span(x_0, Ax_0, \ldots, A^{(k)}x_0)$. This is the basis of the methods of Lanczos and Arnoldi, which are about the best possible for black box solvers for a few extreme eigenvalues. The Krylow space allows a construction of an orthogonal basis via a three term recurrence which is extremely efficient. There are quite a number of computational shortcuts available with this choice of updates, such that the performance is better than an iteration count would suggest. If only eigenvalues are required, they need little storage, while the computation of eigenvectors is either very memory-consuming or needs a second pass. There are stability problems, but those can be dealt with nicely, and efficient implementations are available [4,6,1].
A related choice is $\mathbf{V^{(k+1)}} = span(\ Aw_1{}^{(k)}, \ldots, Aw_m{}^{(k)})$. This has the advantage

of a search subspace with constant dimension which reduces memory requirement and enhances stability, but converges only to the largest eigenvalues[11]. Improvements use Chebychev acceleration $\mathbf{V^{(k+1)}} = span(\, P(A)w_1{}^{(k)}, \ldots, P(A)w_m{}^{(k)})$ [11,14,12].

3.2 Approximate inverse updates of subspace

Approximate inverse updates use special features of the matrix and can therefore be very efficient if properly implemented. The idea is to define a linearized correction equation of the eigenvalue approximation and utilize a computationally cheap approximation of this equation. With $\lambda_1 = x_i^T A x_i$ and e_i the correct eigenvector, this equation reads $(A - \lambda_1 I)(e_i + q_i) \approx (A - \lambda_1 I)x_i$, which with $\lambda_1 \approx \lambda, \quad (A - \lambda I)e_i = 0$ seems to give a reasonable way to construct improved subspace updates. May B_{λ_i} be an (easy to compute) approximation to $(A - \lambda_i I)^{-1}$. Add the approximate inverses applied to the residue to the search space: $\mathbf{V^{(k+1)}} = span(\mathbf{V^{(k)}}, B_{\lambda_1} r_1, \ldots, B_{\lambda_m} r_m)$, where $r_i := (A - x_i^T A x_i \; I)x_i$ with x_i the best eigenvector approximations available. An alternative is using $B_{\lambda_1} x_i$ directly, thus approximating 'shift and invert', but this has obvious stability problems. Until recently, the only method using approximate inverses was Davidson's method which simply uses the diagonal entries of A to compute B, and it was useful only for matrices from theoretical chemistry. While convergence was demonstrated to be rather fast, no analysis was available, and attempts to improve it by using better approximate inverses failed. In hindsight, the reason for this is quite clear, and some idea was there right from the start. If B is exact, then $B_{\lambda_i} r_i = x_i$, obviously not a good choice. So B must not be too good. On the other hand, if B is a poor approximation, this is not much better than taking r_i itself, which is the Krylow subspace calculation without the computational shortcuts. The annoying problem that improving B might reduce convergence was understood and overcome in [13], where it was proved that the exact way to define a correction equation is to project the correction problem into the space orthogonal to e_i, and e_i not being available, the space orthogonal to x_i will do fine, too. This leads to the improved definition of q_i:

$$[(I - x_i\, x_i{}^T)\; (\bar{A} - \bar{\lambda}_j^{(k)} I)\; (I - x_i\, x_i{}^T)]\; q_i = r_i$$

The projection $(I - x_i\, x_i{}^T)$ is not easy to incorporate into the matrix, but there is no need to do so. Because of $r_i \perp -x_i$,

$$q_i = (\epsilon + \bar{\lambda}_i) B^{-1}\, \vec{x}_{\,i} - B^{-1}\, A\, x_i \qquad \text{with} \qquad \epsilon = \frac{x_i{}^T\, B^{-1}\, r_i}{x_i{}^T\, B^{-1}\, x_i}$$

gives the proper solution of the projected equation.

There is no need to use the same type of approximate inverses throughout the computation. In some finite element test cases, the best efficiency has been achieved by starting out with a rather crude and simple choice for B (diagonal only) and getting more accurate as the eigenvector approximations improve [3]. This leaves the field of tuning the algorithm wide open.

Approximate inverses are, strictly speaking, not part of the eigenvalue algorithm but only a plug-in, but of course of highest importance for the efficiency. Therefore a few words on the topic seem appropriate. To build a good method, the literature on approximate inverses should be consulted.
In most cases, an approximate inverse will be constructed by extracting from A a structurally simpler (e.g. narrow banded, very sparse, ...) matrix that contains most of the information of A, and invert this accurately or again approximately (incomplete decomposition). With most matrices from quantum chemistry, taking for $\bar{A}$ a (possibly tapered) band gives a useful approximate inverse:
$\bar{a}_{ij} = 0$ for $|i-j| > k_i, \quad \bar{a}_{ij} = a_{ij}$ for $|i-j| \leq k_i, \quad k_1 \ll n\,,\ k_{i+1} \leq k_i$
The Davidson method is a special example, and often near optimal.
If A is not a stored matrix but only given as a procedure to calculate Ax from x, it rarely pays to extract $\bar{A}$. Here the approximate inverse can be calculated using a conjugate gradient method for solving an equation $(A-\lambda I)q = r$, and stopped at appropriate accuracy. This approach is widely used for eigenvalue problems from PDE's.

4 Problems of implementation and parallelization

The eigenvalue computation can be separated into actions in the n-dimensional space and those in the projected space. The former are generally simple, but time consuming, while the latter may be very complicated indeed, but take up little time. The n-dimensional operations consist of calculation of Ax, the solution of $(A-\lambda I)q = r$, calculation of scalar products and linear combinations of vectors. Except for very peculiar data structure of A, all this is done best by using the existing efficient implementations of linear algebra, BLAS and LAPACK for the sequential and PBLAS and ScaLAPACK for parallel computing. The ARPACK and PARPACK packages [1,9] give careful implementations of the Lanczos and Arnoldi method and may be used either as is or serve as a guideline and provide building blocks for other implementations.
There are some not so obvious details that need special attention. In concept, an orthonormal basis $\vec{w}_1^{(k)}, \ldots, \vec{w}_m^{(k)}$ of the subspace $\mathbf{V^{(k)}}$ is required to build the matrix $\hat{A}^{(k)} := (W^{(k)})^T\, A\, W^{(k)}$,(k) and this is usually done by applying the modified Gram-Schmidt method to a basis of $\vec{v}_1^{(k)}, \ldots, \vec{v}_m^{(k)}$ of $\mathbf{V^{(k)}}$. This is not necessary. May $V := [v_1, \ldots, v_m]$, T the eigenvectors of $\hat{A}$, $F = V^T V$, C the Cholesky decomposition of F and $R = C^{-1}$, then $W = V\,R$, and $S := R^T\,((AV)^T\,V)\,R$. The calculation of the Ritz vectors can be done via $W^{(k)}\,T^{(k)} := V(RT)$, so that there is no need to actually compute the $\vec{w}_i^{(k)}$. All that is needed in the n-dimensional space is $V^T V$ and $V^T A\, V$, all other calculations being only in the low-dimensional subspace.
This saves about half the computations of the Gram-Schmidt method, but there is a problem. The Ritz projection is rather sensitive to errors in orthogonality in W, so the basis $\vec{v}_1^{(k)}, \ldots, \vec{v}_m^{(k)}$ must not be near degenerate. Otherwise, the culprit vector has to be removed or replaced by a new, truly independent direction. This can be organized by using an incremental Cholesky decomposition. The method

is computationally similar to the original Gram-Schmidt method, so the numerical stability is less than that of the modified Gram-Schmidt method. Especially if a restart is to be done, the computation of the vectors forming the basis of the reduced space is a numerically sensitive step. Here, actual orthogonalization may pay.
The only computations in low dimensional space that may contribute to the computational load are the solution of the eigenvalue problem of $\hat{A}^{(k)}$ (e.g. by using some LAPACK procedure) and possibly the process to choose the approximations that will be put to further use.
All the n-dimensional linear algebra calculations can be distributed with benefit over different processors of parallel machines. As they have predictable computational effort, static load balancing will do. There are full codes, building blocks and development tools available for almost any architecture, but writing efficient parallel programs still requires skill and insight. The easy-to-use methods like HPF or virtual shared memory are considerably less efficient than explicit message passing, which is not an easy-to-use method. The speedup available depends on problem size, but there are examples of a speedup of 500 on a 512 processor machine.

5 Conclusions

The calculation of eigenvalues and -vectors of a large matrix is an old topic of mathematics, but there is still progress. While matrix transformation, direct iteration and the use of exact inverses have long been understood and are cast into up-to-date implementations, the proper use of approximate inverses is quite recent, and the implementations are less mature and not generally available. The building blocks for state-of-the-art code are there, but it still has to be put together. Parallel implementations may well exploit parallelism for the n-dimensional operations, while the parallelism for the low dimensional computations is tough to use at all.

References

1. R.B. Lehoucq, D.C. Sorensen, C. Yang, ARPACK Users' Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods. `http://www.netlib.org/scalapack/arpack_ug.ps.gz`
2. A. Basermann and B. Steffen, New Preconditioned Solvers for Large Sparse Eigenvalue Problems on Massively Parallel Computers, *Proceedings of the Eighth SIAM Conference on Parallel Processing for Scientific Computing*, CD-ROM, SIAM, Philadelphia, (1997).
3. A. Basermann and B. Steffen, Preconditioned Solvers for Large Eigenvalue Problems on Massively Parallel Computers and Workstation Clusters, *Parallel Computing: Fundamentals, Applications and New Directions*, eds. E.H. D'Hollander, G.R. Joubert, F.J. Peters, and U. Trottenberg, Elsevier Science B. V., 565 (1998).
4. J.K. Cullum and R.A. Willoughby, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, Volume I: Theory, Birkhäuser, Boston Basel Stuttgart, 1985.

5. E.R. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real symmetric matrices, *J. Comp. Phys.* **17**, 87-94 (1975).
6. R.W. Freund and N.M. Nachtigal, QMR, A quasi minimal residual method for non-hermitian linear systems, Numer. Math. **60**, 315 (1991).
7. N. Kosugi, Modification of the Liu-Davidson method for obtaining one or simultaniously several eigensolutions of a large real symmetric matrix, *Comput. Phys.* **55**, 426-436 (1984).
8. B. Lang, Direct solvers for symmetric eigenvalue problems, *this volume*, pp 203-229.
9. K.J. Maschhoff, D.C. Sorensen, A Portable Implementation of ARPACK for Distributed Memory Parallel Architectures,
`http://www.caam.rice.edu/~kristyn/parpack_home.html`
10. B.N. Parlett, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, PA, 1998 (updated reprint of the 1980 Prentice Hall edition).
11. H. Rutishauser, Computational aspects of F.L. Bauer's simultaneous iteration method, Numer. Math. **13**, 4-13 (1969).
12. B. Steffen, *An Improved Version of the Semi-analytical Eigenvector Processor*, KFA-ZAM-IB-9105, (1991).
13. G.L.G. Sleijpen and H.A. van der Vorst, A Jacobi-Davidson iteration for linear eigenvalue problems, SIAM J. Matrix Anal. Appl. **17**, 401 (1996).
14. J. Tückmantel, *An Improved Version of the Semi-analytical Eigenvector Processor SAP*, CERN/EF/RF 85-4, (1985).

COMPUTING DERIVATIVES OF COMPUTER PROGRAMS

CHRISTIAN H. BISCHOF AND H. MARTIN BÜCKER
Institute for Scientific Computing
Aachen University of Technology
D-52056 Aachen
Germany
E-mail: {bischof,buecker}@sc.rwth-aachen.de

Automatic differentiation is introduced as a powerful technique to compute derivatives of functions given in the form of a computer program in a high-level programming language such as Fortran, C, or C++. In contrast to traditional approaches such as handcoding of analytic expressions, numerical approximation by divided differences, or manipulation of symbolic algebraic expressions by computer algebra systems, automatic differentiation offers the following substantial benefits: it is accurate up to machine precision, efficient in terms of computational cost, applicable to a 1-line formula as well as to a 100,000-line code, and can be produced with minimal human effort.

1 Introduction

Numerical simulations arising in large-scale scientific applications such as quantum chemistry often require the evaluation of derivatives of some objective function. An example is given in this conference proceedings[1] where the need for derivatives in quantum chemical calculations of molecular properties is demonstrated. Derivatives play a crucial role not only in quantum chemistry but in numerical computing in general. Examples include the solution of nonlinear systems of equations, stiff ordinary differential equations, partial differential equations, and differential-algebraic equations. Derivatives are also ubiquitous in the areas of sensitivity analysis of computer models, inverse problems, and (multidisciplinary) design optimization.

Traditionally, such problems with derivatives have been addressed by using techniques of numerical and analytical differentiation as discussed by Gauss[1]. Here, we will discuss another powerful technique called automatic differentiation (AD) for computing derivative information, say, gradients or Hessians. AD has been successfully applied[2,3], it is currently less well known than and sometimes confused with symbolic differentiation. The purpose of this note is to call attention to automatic differentiation, to provide some background information on the technique, and to highlight its advantages over other techniques of differentiation.

To abstract from the particular area of interest, let

$$f : \mathbb{R}^n \to \mathbb{R}^m \qquad \text{with} \qquad \mathbf{x} \mapsto \mathbf{y}$$

denote any vector-valued objective function whose derivatives are sought. We call $\mathbf{x}$ the vector of *independent variables* and $\mathbf{y}$ the vector of *dependent variables*. In large-scale applications, the objective function f is typically not available in analytic form but is given by a computer code written in a high-level programming language such as Fortran, C, or C++. Think of f as a function computed by, say, one of the modules of the TURBOMOLE program system to compute and analyze the electronic structure of molecules[4]. Given such a representation of the objective

function $f(\mathbf{x}) = \big(y_1(\mathbf{x}), y_2(\mathbf{x}), \ldots, y_m(\mathbf{x})\big)^T$, computational methods often demand the evaluation of the Jacobian matrix

$$J(\mathbf{x}) := \begin{pmatrix} \frac{\partial}{\partial x_1} y_1(\mathbf{x}) & \ldots & \frac{\partial}{\partial x_n} y_1(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial x_1} y_m(\mathbf{x}) & \ldots & \frac{\partial}{\partial x_n} y_m(\mathbf{x}) \end{pmatrix} \in \mathbb{R}^{m \times n} \tag{1}$$

at some point of interest $\mathbf{x} \in \mathbb{R}^n$.

A well-known and widely used approach for the approximation of the Jacobian matrix is the use of divided differences (DD). For the sake of simplicity, we only mention first-order forward DD but stress that the following discussion applies to DD as a technique of numerical differentiation in general. Using first-order forward DD, one approximates the ith column of the Jacobian matrix Eq. (1) by

$$\frac{f(\mathbf{x} + h_i \mathbf{e}_i) - f(\mathbf{x})}{h_i}, \tag{2}$$

where h_i is a suitably chosen step size and $\mathbf{e}_i \in \mathbb{R}^n$ is the ith Cartesian unit vector. An advantage of the DD approach is that the function f needs to be evaluated only at some suitably chosen points. Roughly speaking, f is used as a black-box. The main disadvantage of DD is that the accuracy of the approximation depends crucially on a suitable choice of these points, that is, of the step size h_i. However, any strategy to determine a step size faces the dilemma of mutual influence of truncation and cancellation error: The step size should be small to decrease the error of Eq. (2) in approximating Eq. (1) even if infinite-precision arithmetic were used; the step size should be large to avoid cancellation of significant digits when using finite-precision arithmetic in the computation of Eq. (2).

Another traditional approach for computing derivatives is handcoding of analytic expressions. Here, an analytic expression for the Jacobian matrix $J(\mathbf{x})$ is identified first and then implemented by hand using any high-level programming language. If care is taken, handcoding results in highly optimized implementations. However, analytic expressions are not always available. Furthermore, handcoding is smooth only for "simple" objective functions, is substantially error-prone, and requires considerable human effort.

Computer algebra systems such as MACSYMA can, in principle, also be used to find an explicit expression for the Jacobian matrix $J(\mathbf{x})$. A disadvantage of symbolic differentiation is that the length of the representation of the resulting derivative expressions increases rapidly with the number n of independent variables. This property is extremely painful when higher-order derivatives are considered. For instance, the Hessian of an objective function of some complexity in more than three variables can easily result in expressions filling several pages. Moreover, symbolic differentiation is inherently inefficient in terms of computing time, because of the rapid growth of the underlying expressions. The reader is referred to an article by Griewank[5] for a more detailed discussion of computing derivatives symbolically. Another computer algebra system, Maple, is unusual in that it does offer the additional option of automatic differentiation. However, the intention of automatic differentiation of Maple procedures is the development of efficient programs in Maple and other programming languages (Fortran, C). On the other hand,

in this note we consider automatic differentiation for generating derivatives of large production codes written in virtually any high-level programming language.

Automatic differentiation is another option for computing the Jacobian matrix $J(\mathbf{x})$. Virtually any computer program written in a high-level programming language such as Fortran, C, or C++ can be differentiated by this black-box mechanism. Given a program for the evaluation of the objective function f, this technique generates, in a completely automatic fashion, another computer program, called the extended program, that evaluates $f(\mathbf{x})$ and $J(\mathbf{x})$ simultaneously. The key concept behind AD is the fact that every computation, no matter how complicated, is executed on a computer as a (potentially very long) sequence of a limited set of elementary arithmetic operations such as additions, multiplications, and intrinsic functions such as `sin()` and `cos()`. By applying the chain rule over and over again to the composition of these elementary operations, the extended program can be generated accurately evaluating $f(\mathbf{x})$ and $J(\mathbf{x})$ up to machine precision. AD techniques are discussed in a monograph[6] and a forthcoming book[7]. Differentiating a computer program by AD meets all of the following requirements:

Reliability: The computed derivatives should ideally be accurate to machine precision. If the functional relation between $\mathbf{x}$ and $\mathbf{y}$ is not necessarily smooth, the user should get a warning that something might be amiss.

Computational Cost: In many applications, the computation of derivatives is the dominant computational burden. Hence, the amount of memory and runtime required for the derivative code should be minimized as much as possible and in any case be bounded a priori.

Scalability: The approach should give correct results for a 1-line formula as well as a 100,000-line code.

Human Effort: Derivatives are a means to an end. Hence a user should not spend much time in preparing a code for differentiation, in particular in situations in which computer models are bound to change frequently.

Handcoding, divided-difference approximations, and symbolic manipulators fall short with respect to the previously mentioned criteria. The main drawbacks of divided-difference approximations are their numerical unpredictability and their computational cost. In contrast, both the handcoding and symbolic approaches suffer from a lack of scalability and require considerable human effort.

In the next section, we give a brief overview of automatic differentiation. Section 3 discusses issues that arise in the design of software packages implementing the AD technology. In Section 4, we discuss some issues concerning the use of AD tools. In the last section, we summarize AD's advantages and provide pointers to AD tools.

2 Basic Modes of Automatic Differentiation

Traditionally, two basic approaches to automatic differentiation have been employed: the so-called forward mode and reverse mode, which date back to the

early sixties and seventies, respectively. These modes are distinguished by how the chain rule is used to propagate derivatives through the computation. We briefly summarize the main points about these two approaches; a more detailed description can be found in the literature[5,6,8].

The forward mode propagates derivatives of intermediate variables with respect to the independent variables and follows the control flow of the original program. By exploiting the linearity of differentiation, the forward mode allows us to compute arbitrary linear combinations $J\,S$ of columns of the Jacobian matrix J. In matrix-matrix multiplication, the symbol S denotes an arbitrary $n \times p$ matrix. The effort required to compute not only the objective function but also $J\,S$ is roughly p times the runtime and memory of the original program. In particular, when $p = 1$ and thus the matrix S reduces to a vector $\mathbf{s}$, we compute the directional derivative

$$J\,\mathbf{s} = \lim_{h \to 0} \frac{f(\mathbf{x} + h\mathbf{s}) - f(\mathbf{x})}{h},$$

where h is some step size.

In contrast, the reverse mode of automatic differentiation propagates derivatives of the final result with respect to an intermediate quantity, so-called adjoint quantities. To propagate adjoints, one must be able to reverse the flow of the program and must remember or recompute any intermediate value that nonlinearly affects the final result. In particular, one must store the intermediate values that have been involved in nonlinear operations before they are overwritten or go out of scope. Sometimes some of these intermediates can be recomputed during the reverse sweep, but in any case one has to keep a log of the branch directions taken.

For an $m \times q$ matrix W, the reverse mode allows us to compute arbitrary linear combinations $W^T J$ of rows of the Jacobian matrix J with roughly q times as many floating-point operations as required for the evaluation of f. In a straightforward implementation, however, the storage requirements may be proportional to the number of floating-point operations required for the evaluation of f, as a result of the tracing required to make the program "reversible." When $q = 1$ and thus the matrix W^T reduces to a row vector w^T, we compute the derivative $w^T J$. The reverse mode is particularly attractive for the computation of long gradients, as its operations count does not depend on the number n of independent variables.

The forward mode can be naturally extended to second or third (and even higher) derivatives, but the complexity grows like the square or cube p, respectively. Especially for Hessian-vector products, a combined forward and reverse sweep is attractive, since it still has essentially the same complexity as a single evaluation of the underlying scalar function. In any case, automatic differentiation produces code that computes derivatives accurate to machine precision[5]. The techniques of automatic differentiation are directly applicable to computer programs of arbitrary length containing branches, loops, and subroutines.

The weighting and combining of derivatives through the matrices W and S are natural and useful for many applications, especially if sparsity in J can be exploited. Unfortunately, many existing AD tools are (like computer algebra packages) still exclusively oriented toward the evaluation of Cartesian derivatives, that is, the partials of certain dependent variables with respect to certain independent variables.

3 Design of Automatic Differentiation Tools

Automatic differentiation can be viewed as a particular semantic transformation problem: Given a code for computing a function, we would like to generate a code that computes the derivatives of that function. To effect this transformation, two approaches have been employed:

Operator Overloading: Modern computer languages such as C++ or Fortran 90 make it possible to redefine the meaning of elementary operators. We can, for example, define a type for floating-point numbers that have gradient objects associated with them (let's call this new type `adouble`), and for each elementary operation such as a multiplication, we can define the meaning of the operator "$*$" for variables of type `adouble` as follows. An assignment $z = x * y$ not only computes the product of x and y but also updates the associated gradient object in a product rule fashion $\nabla z = x\nabla y + y\nabla x$. So, each occurrence of a multiplication of two `adouble`s in the code will also effect the update of the associated derivatives in a transparent fashion.

Source Transformation: Another way of changing the semantics of the code is to rewrite it explicitly. For example, the assignment $z = x * y$ is rewritten into a piece of code that contains not only the computation of z but also an implementation of the vector linear combination $\nabla z = x\nabla y + y\nabla x$, implemented either as a do-loop or as a subroutine call.

Each of these approaches has its advantages and disadvantages. The advantages of operator overloading are threefold.

Terseness: All that is required for a new data type, such as `adouble`s, is a new class definition. While such a class definition can be substantial, comprising several thousand lines of code, it hides this complexity from the user of an AD tool.

Flexibility: If we want to change an implementation strategy associated with a particular class, the source code remains unaffected. All that changes is the class definition itself. So, for example, whether we compute first- or second-order derivatives is reflected in the class definition but not in the code being differentiated.

Full Access to Runtime Information: The reverse mode of AD requires the ability to reverse the partial flow of program execution. One way to do this is to use operator overloading to generate a tape that logs all the operations actually performed, and use this tape as the input for a derivative interpreter, which then can compute any derivatives desired using either the forward or reverse mode of automatic differentiation. This approach is, for example, chosen in the ADOL-C package[9].

The drawbacks of operator overloading are the following

Lack of Transparency: While it is aesthetically pleasing that the source code does not change, even though its meaning does, it does not aid in debugging,

since one has to deduce the meaning of the operations implied by the source code and the associated class definitions.

Implementation Overhead: The actions associated with a class definition can be viewed as an implied subroutine call, and although much progress has been made recently in the compilation of operator overloading, the runtime overhead of this technique can be substantial depending on the sophistication of the compiler.

Dusty Deck Assimilation: Many existing computer codes are written in languages such as Fortran 77 or ANSI-C that do not support operator overloading. In particular, assimilating large codes into the supposedly backwards-compatible Fortran 90 or C++ languages turns out to be a thorny task.

On the other hand, the advantages of the source transformation approach are as follows.

Simplicity of Generated Code: Since the derivative code is spelled out exactly, usually in the same language as the input code, it is easier to follow the actions of the derivative code as long as the chain rule is applied in a basic local fashion. This simplicity also facilitates compiler optimizations and hence faster execution of the generated code.

Dusty Deck Assimilation: The source transformation approach requires traditional compiler infrastructure such as parsers, generators and manipulators of intermediate languages, and unparsers. These tools are readily available for languages such as Fortran 77 or ANSI-C, at least in the commercial world.

Variable Scope: Operator overloading inherently sees one elementary operation at a time. Source transformation approaches, on the other hand, have access to the context of a particular computation and hence have more flexibility in applying derivative rules. For example, the ADIFOR[10,11] and ADIC[12] tools view a program as a sequence of assignment statements, applying the reverse mode at this level and the forward mode overall.

The disadvantages of the source transformation approach are the following.

Implementation Complexity: Source transformation approaches, at least at the moment, require considerable tool infrastructure, in particular for the processing of language-dependent features. Also, the lack of a standardized language description makes changing the semantics of a particular automatic differentiation tool a potentially rather involved task.

Code Expansion or Subroutine Interface Swell: A "pure" source transformation approach is infeasible when the action associated with a particular statement exceeds a certain level of complexity. In this case, either the length of the generated code grows too large for a compiler to digest, or rather extensive subroutine library interfaces must be maintained to encapsulate the basic computational kernels. The latter approach, in many ways, is similar to operator overloading, albeit considerably less elegant.

Of course, the relevance of these advantages and disadvantages depends to a great extent on the particular application.

Given the mathematical underpinnings of the concept of derivatives, the "black-box" application of an AD tool usually raises several questions that we briefly address here.

Question: How do you know that the code represents a globally differentiable function?

Answer: We don't. AD computes the derivative defined by the sequence of assignment statements executed in the course of a function evaluation. Hence, for a branch (if-statement), which potentially introduces a nondifferentiability, AD will compute a one-sided directional derivative. This problem is further discussed by Fischer[13].

Question: How do you deal with intrinsics?

Answer: Some intrinsics functions, such as `abs()` and `sqrt()`, are not differentiable in all points of their domain. Some tools invoke an extension handler flagging such occurrences; others ignore such occurrences.

Question: What happens when you differentiate through iterative processes?

Answer: It depends. AD generates a new iteration, and it is not clear a priori whether the new iteration will converge and what it will converge to, although empirically AD leads to the desired result. However, derivative convergence may lag, or derivatives may diverge. For some commonly used approaches for solving nonlinear systems of equations, this issue is discussed by Griewank *et al.*[14]. This problem clearly requires more research, but the emergence of robust AD tools has made it possible to tackle this problem for sophisticated numerical methods.

4 Using Automatic Differentiation Tools

Based on our experience with the ADIFOR[10,11] and ADIC[12] tools for automatic differentiation, this section explores some of the subtler issues related to the use of AD and the implications for numerical software design. In particular, we focus on the issues that arise from the fact that AD differentiates a given computer program step by step, in a fashion that is oblivious of the overall semantics of a program. This "myopic" view gives AD tools the power to deal with programs of arbitrary length, but it also implies that users of AD tools may have to communicate some of their knowledge to an AD tool to arrive at a desired solution. Specifically, we illustrate the issues arising in the context of nondifferentiable language intrinsics such as `max()` and numerical integrators.

4.1 Intrinsic Functions

Since the derivative of $\sin(x)$ with respect to x is given by $\cos(x)$, an AD tool might transform the statement

$$\texttt{y} = \texttt{sin(x)}$$

into the derivative statement

$$\nabla\texttt{y = cos(x) * } \nabla\ \texttt{x}.$$

Here, the notation ∇y denotes the derivatives of variable `y` with respect to some chosen set of variables. In this case, there is no difficulty, since `sin()` is everywhere differentiable.

Most computer languages do, however, contain intrinsic functions that are not differentiable in some points in their domain, as for example the Fortran 77 intrinsics `abs(x)` and `sqrt(x)` when the value of the argument is zero. We call such a point an "exceptional point." We cannot simply claim that the function in question is not differentiable, since a computer program executing such instructions may well represent a smooth function, such as $g(x, y) = \sqrt{x^4 + y^4}$. Moreover, intrinsics may be used to guard against unphysical values of simulation parameters. For example, in a weather model one might see code such as

$$\texttt{rain} = \texttt{max}(\texttt{rain}, 0.0).$$

This statement reflects the fact that rainfall cannot be negative and is intended to convert a small negative number, which may have arisen from floating-point roundoff, to the physically sensible number 0 (i.e., no rain).

The function $\max(x, y)$ is not differentiable for $x = y$. However, in the previously described case, it makes sense to define partial derivatives for the exceptional cases as $\frac{\partial \max(x,y)}{\partial x}|_{x=y} := 1.0$ and $\frac{\partial \max(x,y)}{\partial y}|_{x=y} := 0.0$. These definitions do not change ∇`rain` when `rain` is set to zero in the induced AD statement

$$\nabla\texttt{rain} = \frac{\partial \max(x, y)}{\partial x} \nabla\texttt{rain}.$$

However, these definitions would not lead to the desired result if the order of arguments in the `max()` call was reversed, namely,

$$\texttt{rain} = \texttt{max}(0.0, \texttt{rain}).$$

In this case, the derivative of `rain` would be zeroed out when the value of the variable was zero, and it would have been appropriate to exchange the definitions of $\frac{\partial \max}{\partial x}$ and $\frac{\partial \max}{\partial y}$. In other contexts, an argument could also be made for setting $\frac{\partial \max(x,y)}{\partial x}|_{x=y} = 0.5$ and $\frac{\partial \max(x,y)}{\partial y}|_{x=y} = 0.5$, since then automatic differentiation provides a so-called subgradient, which is useful in nonsmooth numerical optimization, as described, for example, by Clark[15].

These examples demonstrate the following points:

i. No choice of derivative values for exceptional points will always be correct.

ii. There is no "automatic" way to decide what sensible choices are.

```
Given: parameter p, current time t, current solution x_c ≈ x(t,p),
       suggested time step Δt.
1) Compute x_1 ≈ x(t + Δt, p) using Method A.
2) Compute x_2 ≈ x(t + Δt, p) using Method B.
3) Compute δ = ‖x_1 − x_2‖ for some norm ‖ · ‖.
4) if ( δ < some given threshold )
      Accept the higher-order of x_1 and x_2
      and update t ← t + Δt.
   else
      Δt = g(Δt, δ);
      goto 1)
   endif
```

Figure 1. Simplified Description of a Numerical Integrator

iii. User insight into the problem is essential.

Thus, potential users of AD tools need to be aware of these facts and provide "hints" for an AD tool in the code to be eventually differentiated. Such hints are particularly important for numerical libraries, since these codes typically embody subtle numerics and will be reused often. To this end, the ADIFOR and ADIC systems employ the completely user-customizable ADIntrinsics system for dealing with Fortran and ANSI-C intrinsics. Surprisingly, in most cases the derivatives turn out to be the ones intended without the need for derivatives intrinsics modifications.

4.2 Numerical Integrators

Another problem arises from the fact that an AD tool, when applied to a code embodying a numerical method, will not only differentiate the solution produced by this method, but also take into account the *way by which one arrived at the solution*. As an illustration, consider a parameter-dependent initial value problem

$$\dot{x}(p) = f(x, p, t) \qquad \text{with} \qquad x(t = 0) = x_0, \tag{3}$$

where p is a parameter. Figure 1 shows a simplified version of the time-stepping loop of a typical explicit numerical integrator with step size control. In this figure, the notation Method A and Method B is used for two integration methods of different order, and g is some function that adjusts the time step Δt. For simplicity, we ignore the fact that the time step will be adjusted upward if there is a good fit.

If, for a given p, we are interested in $\frac{\partial x}{\partial p}\big|_{t=T}$, where T is the final time, we can employ an AD tool to differentiate this code with respect to p. If we differentiate with respect to p and use ∇ to denote $\frac{\mathrm{d}}{\mathrm{d}\,p}$, the chain rule of differential calculus now implies that

$$\nabla(\Delta t) = \frac{\partial\, g}{\partial\,(\Delta t)}\nabla(\Delta t) + \frac{\partial\, g}{\partial\,\delta}\nabla\delta.$$

Clearly, $\nabla\delta \neq 0$ in general, since δ depends on x, which in turn depends on p. Thus we have the interesting situation that $\nabla(\Delta t) \neq 0$ when $\frac{\partial\, g}{\partial\,\delta} \neq 0$; that is, the

computational equivalent of time, Δt, will have a nonzero derivative with respect to the parameter p. Viewed from an analytical perspective, this is nonsense – the values of time and the parameter are not related. From a computational perspective however, it does make sense – depending on the value of the parameter, we may choose a different time discretization. Thus, what we really compute as the final value $x_T(p)$ is

$$x_T(p) = x(t(p), p)|_{t(p)=T}$$

(note the dependence of t on p). Thus, we obtain

$$\nabla x_{t=T} = \left.\frac{\partial x}{\partial t}\right|_{t=T} \cdot \nabla t_{t=T} + \left.\frac{\partial x}{\partial p}\right|_{t=T},$$

and with Eq. (3)

$$\nabla x_{t=T} = f(x_T, p, T) \cdot \nabla t_{t=T} + \left.\frac{\partial x}{\partial p}\right|_{t=T}.$$

Note that ∇x and ∇t will have been computed by the AD-generated derivative code. We observe the following:

i. Depending on how the time discretization was chosen, we will obtain different values for $\nabla t_{t=T}$ and thus for $\nabla x_{t=T}$. Most certainly, we will *not* obtain $\left.\frac{\partial x}{\partial p}\right|_{t=T}$ which is the result desired by most users.

ii. If Δt would had been zero at every step, we would have $\nabla t_{t=T} = 0$ and thus $\nabla x_{t=T} = \left.\frac{\partial x}{\partial p}\right|_{t=T}$, as desired by the user. By default, this happens in methods using a fixed step size. This case is also discussed by Sandu *et al.*[16]

iii. Independent of how the time discretization was chosen, we can recover the desired solution as

$$\left.\frac{\partial x}{\partial p}\right|_{t=T} = \nabla x_{t=T} - f(x_T, p, T) \cdot \nabla t_{t=T}. \tag{4}$$

These issues are discussed in more detail by Eberhard and Bischof[17].

Note that approaches (ii) and (iii) are really geared toward the library developer and the sophisticated AD user, respectively. When an integrator code is written, it is probably feasible to indicate the places where the next time step is assigned and to indicate that an AD tool should treat this statement as constant with respect to differentiation, resulting in the assignment of a zero gradient. Current AD tools do not have such facilities built-in yet, but will so soon. At any rate, unless the developer of the integrator provides this information, the considerable sophistication of these codes makes it difficult for others to extract this information from the code.

While one might take the attitude that this was not an issue, given the "fix" (iii), this is not really the case. Even when $\frac{\partial x}{\partial p}$ is well behaved, ∇t and ∇x can become very large and can overflow. Furthermore, the user of an AD tool may well be unaware of these issues, or may not be able to localize the problem since the integrator may be buried under other layers of software. However, as shown by Eberhard and Bischof[17], if the final time is prescribed, we are likely to obtain

$\nabla t_{t=T} = 0$, and everything works out; we suspect that this situation has happened in quite a few AD applications.

We note that while (ii) and (iii) will result in the right derivatives $\frac{\partial x}{\partial p}$, there is no guarantee that the derivatives will be obtained at the same accuracy as the solution x, since the guard of the if-statement governing acceptance or rejection of a step will *not* be augmented by AD, and thus still will be governed only by the behavior of x. Thus, the derivatives obtained by Eq. (4) will be consistent, but they may not be as accurate as those obtained by solving the sensitivity equation

$$\dot{x}_p = \frac{\partial f}{\partial x} x_p + \frac{\partial f}{\partial p},$$

where $x_p = \frac{\partial x}{\partial p}$, alongside the original differential equation Eq. (3). It is easy to add the norm of $\nabla\delta$ to the guard for step size control, but an AD tool cannot be expected to do so without user guidance. Similar issues also arise in the context of automatic differentiation of iterative solvers for nonlinear equations and are discussed by Griewank *et al.*[14].

5 Concluding Remarks

This note was meant to give a brief introduction to automatic differentiation. We briefly discussed the advantages of this powerful technique in contrast to the better-known approaches of numerical, analytic, and symbolic differentiation. Broadly speaking, automatic differentiation saves work in comparison with handcoding of analytic derivatives and, by computing accurate derivatives, avoids the hassle caused by inaccurate numerical differentiation. We reviewed the forward and reverse modes of automatic differentiation, gave some background on design issues of automatic differentiation tools, and discussed some subtle issues involved in using these tools.

Even though automatic differentiation tools are still in their infancy, under a wide range of circumstances they already can compute derivatives faster than divided difference approximations[11]. Furthermore, there are examples where the availability of fully accurate derivatives was essential for numerical robustness and convergence[18,19,20]. Another advantage of automatic differentiation tools that we did not discuss in this note is their ability to provide, in a fashion that is transparent to the user, information about the zero/nonzero structure of derivative matrices.[21] This information is required to solve linear systems involving the Jacobian, and the automatic detection of the sparsity pattern avoids the error-prone task of having the user specify the sparsity pattern. This feature is provided in ADIFOR and ADIC through the SparsLinC library and is used, for example, in the NEOS (Network-enabled Optimization Server) problem-solving environment[22], which is accessible at `http://www-neos.mcs.anl.gov/`.

The emergence of robust automatic differentiation tools applicable to functions defined by computer programs in general-purpose computer languages such as Fortran 77, Fortran 90, C, and C++ is putting these tools within the reach of many computational practitioners in any field requiring derivatives, including quantum chemistry. The web site at

`http://www.sc.rwth-aachen.de/Research/AD/subject.html` gives a short description of some available automatic differentiation tools and provides pointers for obtaining these tools.

Acknowledgments

This work was completed while the second author was visiting the Mathematics and Computer Science Division, Argonne National Laboratory, supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

References

1. J. Gauss, in *Modern Methods and Algorithms of Quantum Chemistry*, edited by J. Grotendorst, (John von Neumann Institute for Computing, Jülich, Germany, 2000).
2. M. Berz, C. Bischof, G. Corliss, and A. Griewank, *Computational Differentiation: Techniques, Applications, and Tools* (SIAM, Philadelphia, 1996).
3. A. Griewank and G. Corliss, *Automatic Differentiation of Algorithms* (SIAM, Philadelphia, 1991).
4. R. Ahlrichs, M. Bär, M. Häser, H. Horn, and C. Kölmel, Chemical Physics Letters **162**, 165 (1989).
5. A. Griewank, in *Mathematical Programming: Recent Developments and Applications*, pages 83–108 (Amsterdam, 1989, Kluwer Academic Publishers).
6. L. B. Rall, *Automatic Differentiation: Techniques and Applications*, volume **120** of *Lecture Notes in Computer Science* (Springer Verlag, Berlin, 1981).
7. A. Griewank, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation* (SIAM, Philadelphia, to appear).
8. C. Bischof, A. Carle, P. Hovland, P. Khademi, and A. Mauer, ADIFOR 2.0 user's guide (Revision D), Technical Memorandum ANL/MCS-TM-192, Mathematics and Computer Science Division, Argonne National Laboratory, 1998 (also CRPC Technical Report CRPC-95516-S).
9. A. Griewank, D. Juedes, and J. Utke, ACM Transactions on Mathematical Software **22**, 131 (1996).
10. C. Bischof, A. Carle, G. Corliss, A. Griewank, and P. Hovland, Scientific Programming **1**, 11 (1992).
11. C. Bischof, A. Carle, P. Khademi, and A. Mauer, IEEE Computational Science & Engineering **3**, 18 (1996).
12. C. Bischof, L. Roh, and A. Mauer, Software–Practice and Experience **27**, 1427 (1997).
13. H. Fischer, in *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, edited by A. Griewank and G. F. Corliss, pages 43–50 (SIAM, Philadelphia, Penn., 1991).
14. A. Griewank, C. Bischof, G. Corliss, A. Carle, and K. Williamson, Optimization Methods and Software **2**, 321 (1993).

15. F. Clark, *Optimization and Nonsmooth Analysis* (John Wiley and Sons, New York, 1983).
16. A. Sandu, G. R. Carmichael, and F. A. Potra, Atmospheric Environment **31**, 475 (1997).
17. P. Eberhard and C. Bischof, Mathematics of Computation **68**, 717 (1999).
18. P. Hovland, C. Bischof, D. Spiegelman, and M. Casella, SIAM Journal on Scientific Computing **18**, 1056 (1997).
19. P. Eberhard, in *ICIAM/GAMM 95: Issue 3: Applied Stochastics and Optimization*, edited by O. Mahrenholtz, K. Marti, and R. Mennicken, pages 40–43 (1996), Special Issue of Zeitschrift für Angewandte Mathematik und Mechanik (ZAMM).
20. A. Ibsais and V. Ajjarapu, IEEE Transactions on Power Systems **12**, 592 (1997).
21. C. Bischof, P. Khademi, A. Bouaricha, and A. Carle, Optimization Methods and Software **7**, 1 (1996).
22. J. Czyzyk, M. P. Mesnier, and J. J. Moré, IEEE Computational Science and Engineering **5**, 68 (1998).

AB INITIO MOLECULAR DYNAMICS: THEORY AND IMPLEMENTATION

DOMINIK MARX
Lehrstuhl für Theoretische Chemie, Ruhr–Universität Bochum
Universitätsstrasse 150, 44780 Bochum
Germany
E-mail: dominik.marx@theochem.ruhr-uni-bochum.de

JÜRG HUTTER
Organisch–chemisches Institut, Universität Zürich
Winterthurerstrasse 190, 8057 Zürich
Switzerland
E-mail: hutter@oci.unizh.ch

The rapidly growing field of *ab initio* molecular dynamics is reviewed in the spirit of a series of lectures given at the Winterschool 2000 at the *John von Neumann Institute for Computing, Jülich*. Several such molecular dynamics schemes are compared which arise from following various approximations to the fully coupled Schrödinger equation for electrons and nuclei. Special focus is given to the Car–Parrinello method with discussion of both strengths and weaknesses in addition to its range of applicability. To shed light upon why the Car–Parrinello approach works several alternate perspectives of the underlying ideas are presented. The implementation of *ab initio* molecular dynamics within the framework of plane wave–pseudopotential density functional theory is given in detail, including diagonalization and minimization techniques as required for the Born–Oppenheimer variant. Efficient algorithms for the most important computational kernel routines are presented. The adaptation of these routines to distributed memory parallel computers is discussed using the implementation within the computer code `CPMD` as an example. Several advanced techniques from the field of molecular dynamics, (constant temperature dynamics, constant pressure dynamics) and electronic structure theory (free energy functional, excited states) are introduced. The combination of the path integral method with *ab initio* molecular dynamics is presented in detail, showing its limitations and possible extensions. Finally, a wide range of applications from materials science to biochemistry is listed, which shows the enormous potential of *ab initio* molecular dynamics for both explaining and predicting properties of molecules and materials on an atomic scale.

1 Setting the Stage: Why *Ab Initio* Molecular Dynamics ?

Classical molecular dynamics using "predefined potentials", either based on empirical data or on independent electronic structure calculations, is well established as a powerful tool to investigate many–body condensed matter systems. The broadness, diversity, and level of sophistication of this technique is documented in several monographs as well as proceedings of conferences and scientific schools [12,135,270,217,69,59,177]. At the very heart of any molecular dynamics scheme is the question of how to describe – that is in practice how to approximate – the interatomic interactions. The traditional route followed in molecular dynamics is to determine these potentials in advance. Typically, the full interaction is broken up into two–body, three–body and many–body contributions, long–range and short–range terms etc., which have to be represented by suitable functional forms, see

Sect. 2 of Ref. [253] for a detailed account. After decades of intense research, very elaborate interaction models including the non–trivial aspect to represent them analytically were devised [253,539,584].

Despite overwhelming success – which will however not be praised in this review – the need to devise a "fixed model potential" implies serious drawbacks, see the introduction sections of several earlier reviews [513,472] for a more complete digression on these aspects. Among the most delicate ones are systems where (i) many different atom or molecule types give rise to a myriad of different interatomic interactions that have to be parameterized and / or (ii) the electronic structure and thus the bonding pattern changes qualitatively in the course of the simulation. These systems can be called "chemically complex".

The reign of traditional molecular dynamics *and* electronic structure methods was greatly extended by the family of techniques that is called here "*ab initio* molecular dynamics". Other names that are currently in use are for instance Car–Parrinello, Hellmann–Feynman, first principles, quantum chemical, on–the–fly, direct, potential–free, quantum, etc. molecular dynamics. The basic idea underlying every *ab initio* molecular dynamics method is to compute the forces acting on the nuclei from electronic structure calculations that are performed "on–the–fly" as the molecular dynamics trajectory is generated. In this way, the electronic variables are not integrated out beforehand, but are considered as active degrees of freedom. This implies that, given a suitable approximate solution of the many–electron problem, also "chemically complex" systems can be handled by molecular dynamics. But this also implies that the approximation is shifted from the level of selecting the model potential to the level of selecting a particular approximation for solving the Schrödinger equation.

Applications of *ab initio* molecular dynamics are particularly widespread in materials science and chemistry, where the aforementioned difficulties (i) and (ii) are particularly severe. A collection of problems that were already tackled by *ab initio* molecular dynamics including the pertinent references can be found in Sect. 5. The power of this novel technique lead to an explosion of the activity in this field in terms of the number of published papers. The locus can be located in the late–eighties, see the squares in Fig. 1 that can be interpreted as a measure of the activity in the area of *ab initio* molecular dynamics. As a matter of fact the time evolution of the number of citations of a particular paper, the one by Car and Parrinello from 1985 entitled "Unified Approach for Molecular Dynamics and Density–Functional Theory" [108], parallels the trend in the entire field, see the circles in Fig. 1. Thus, the resonance that the Car and Parrinello paper evoked and the popularity of the entire field go hand in hand in the last decade. Incidentally, the 1985 paper by Car and Parrinello is the last one included in the section "Trends and Prospects" in the reprint collection of "key papers" from the field of atomistic computer simulations [135]. That the entire field of *ab initio* molecular dynamics has grown mature is also evidenced by a separate PACS classification number (71.15.Pd "Electronic Structure: Molecular dynamics calculations (Car–Parrinello) and other numerical simulations") that was introduced in 1996 into the *Physics and Astronomy Classification Scheme* [486].

Despite its obvious advantages, it is evident that a price has to be payed for

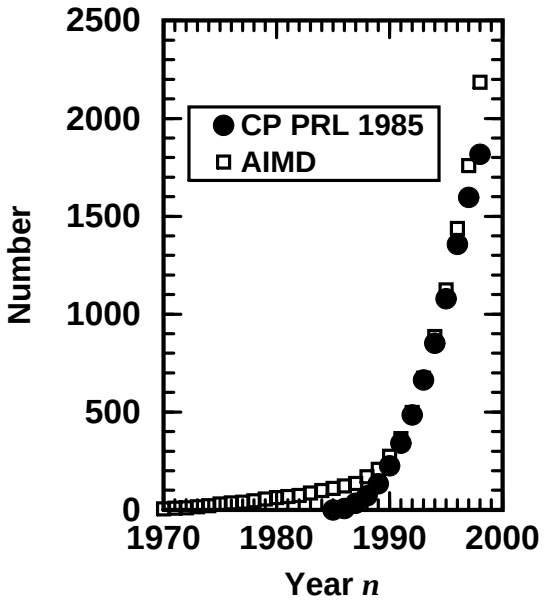


Figure 1. Publication and citation analysis. Squares: number of publications which appeared up to the year n that contain the keyword "*ab initio* molecular dynamics" (or synonyma such as "first principles MD", "Car–Parrinello simulations" etc.) in title, abstract or keyword list. Circles: number of publications which appeared up to the year n that cite the 1985 paper by Car and Parrinello [108] (including misspellings of the bibliographic reference). Self–citations and self–papers are excluded, i.e. citations of Ref. [108] in their own papers and papers coauthored by R. Car and / or M. Parrinello are *not* considered in the respective statistics. The analysis is based on the CAPLUS ("Chemical Abstracts Plus"), INSPEC ("Physics Abstracts"), and SCI ("Science Citation Index") data bases at STN International. Updated statistics from Ref. [405].

putting molecular dynamics on *ab initio* grounds: the correlation lengths and relaxation times that are accessible are much smaller than what is affordable via standard molecular dynamics. Another appealing feature of standard molecular dynamics is less evident, namely the "experimental aspect of playing with the potential". Thus, tracing back the properties of a given system to a simple physical picture or mechanism is much harder in *ab initio* molecular dynamics. The bright side is that new phenomena, which were not forseen before starting the simulation, can simply happen if necessary. This gives *ab initio* molecular dynamics a truly predictive power.

Ab initio molecular dynamics can also be viewed from another corner, namely from the field of classical trajectory calculations [649,541]. In this approach, which has its origin in gas phase molecular dynamics, a *global* potential energy surface is constructed in a first step either empirically or based on electronic structure calculations. In a second step, the dynamical evolution of the nuclei is generated by using classical mechanics, quantum mechanics or semi / quasiclassical approx-

imations of various sorts. In the case of using classical mechanics to describe the dynamics – the focus of the present overview – the limiting step for large systems is the first one, why so? There are $3N-6$ internal degrees of freedom that span the global potential energy surface of an unconstrained N–body system. Using for simplicity 10 discretization points per coordinate implies that of the order of 10^{3N-6} electronic structure calculations are needed in order to map such a global potential energy surface. Thus, the computational workload for the first step grows roughly like $\sim 10^N$ with increasing system size. This is what might be called the "dimensionality bottleneck" of calculations that rely on *global* potential energy surfaces, see for instance the discussion on p. 420 in Ref. [254].

What is needed in *ab initio* molecular dynamics instead? Suppose that a useful trajectory consists of about 10^M molecular dynamics steps, i.e. 10^M electronic structure calculations are needed to generate one trajectory. Furthermore, it is assumed that 10^n independent trajectories are necessary in order to average over different initial conditions so that 10^{M+n} *ab initio* molecular dynamics steps are required in total. Finally, it is assumed that each single–point electronic structure calculation needed to devise the global potential energy surface and one *ab initio* molecular dynamics time step requires roughly the same amount of CPU time. Based on this truly simplistic order of magnitude estimate, the advantage of *ab initio* molecular dynamics vs. calculations relying on the computation of a global potential energy surface amounts to about $10^{3N-6-M-n}$. The crucial point is that for a given statistical accuracy (that is for M and n fixed and independent on N) and for a given electronic structure method, the computational advantage of "on–the–fly" approaches grows like $\sim 10^N$ with system size.

Of course, considerable progress has been achieved in trajectory calculations by carefully selecting the discretization points and reducing their number, choosing sophisticated representations and internal coordinates, exploiting symmetry etc. but basically the scaling $\sim 10^N$ with the number of nuclei remains a problem. Other strategies consist for instance in reducing the number of active degrees of freedom by constraining certain internal coordinates, representing less important ones by a (harmonic) bath or friction, or building up the global potential energy surface in terms of few–body fragments. All these approaches, however, invoke approximations beyond the ones of the electronic structure method itself. Finally, it is evident that the computational advantage of the "on–the–fly" approaches diminish as more and more trajectories are needed for a given (small) system. For instance extensive averaging over many different initial conditions is required in order to calculate quantitatively scattering or reactive cross sections. Summarizing this discussion, it can be concluded that *ab initio* molecular dynamics is the method of choice to investigate large and "chemically complex" systems.

Quite a few review articles dealing with *ab initio* molecular dynamics appeared in the nineties [513,223,472,457,224,158,643,234,463,538,405] and the interested reader is referred to them for various complementary viewpoints. In the present overview article, emphasis is put on both broadness of the approaches and depth of the presentation. Concerning the broadness, the discussion starts from the Schrödinger equation. Classical, Ehrenfest, Born–Oppenheimer, and Car–Parrinello molecular dynamics are "derived" from the time–dependent mean–field approach that is ob-

tained after separating the nuclear and electronic degrees of freedom. The most extensive discussion is related to the features of the basic Car–Parrinello approach but all three *ab initio* approaches to molecular dynamics are contrasted and partly compared. The important issue of how to obtain the correct forces in these schemes is discussed in some depth. The most popular electronic structure theories implemented within *ab initio* molecular dynamics, density functional theory in the first place but also the Hartree–Fock approach, are sketched. Some attention is also given to another important ingredient in *ab initio* molecular dynamics, the choice of the basis set.

Concerning the depth, the focus of the present discussion is clearly the implementation of both the basic Car–Parrinello and Born–Oppenheimer molecular dynamics schemes in the `CPMD` package [142]. The electronic structure approach in `CPMD` is Hohenberg–Kohn–Sham density functional theory within a plane wave / pseudopotential implementation and the Generalized Gradient Approximation. The formulae for energies, forces, stress, pseudopotentials, boundary conditions, optimization procedures, parallelization etc. are given for this particular choice to solve the electronic structure problem. One should, however, keep in mind that a variety of other powerful *ab initio* molecular dynamics codes are available (for instance `CASTEP` [116], `CP-PAW` [143], `fhi98md` [189], `NWChem` [446], `VASP` [663]) which are partly based on very similar techniques. The classic Car–Parrinello approach [108] is then extended to other ensembles than the microcanonical one, other electronic states than the ground state, and to a fully quantum–mechanical representation of the nuclei. Finally, the wealth of problems that can be addressed using *ab initio* molecular dynamics is briefly sketched at the end, which also serves implicitly as the "Summary and Conclusions" section.

2 Basic Techniques: Theory

2.1 Deriving Classical Molecular Dynamics

The starting point of the following discussion is non–relativistic quantum mechanics as formalized via the time–dependent Schrödinger equation

$$i\hbar\frac{\partial}{\partial t}\Phi(\{\mathbf{r}_i\},\{\mathbf{R}_I\};t) = \mathcal{H}\Phi(\{\mathbf{r}_i\},\{\mathbf{R}_I\};t) \tag{1}$$

in its position representation in conjunction with the standard Hamiltonian

$$\begin{aligned}\mathcal{H} &= -\sum_I \frac{\hbar^2}{2M_I}\nabla_I^2 - \sum_i \frac{\hbar^2}{2m_\mathrm{e}}\nabla_i^2 + \sum_{i<j}\frac{\mathrm{e}^2}{|\mathbf{r}_i-\mathbf{r}_j|} - \sum_{I,i}\frac{\mathrm{e}^2 Z_I}{|\mathbf{R}_I-\mathbf{r}_i|} + \sum_{I<J}\frac{\mathrm{e}^2 Z_I Z_J}{|\mathbf{R}_I-\mathbf{R}_J|} \\ &= -\sum_I \frac{\hbar^2}{2M_I}\nabla_I^2 - \sum_i \frac{\hbar^2}{2m_\mathrm{e}}\nabla_i^2 + V_{\mathrm{n-e}}(\{\mathbf{r}_i\},\{\mathbf{R}_I\}) \\ &= -\sum_I \frac{\hbar^2}{2M_I}\nabla_I^2 + \mathcal{H}_\mathrm{e}(\{\mathbf{r}_i\},\{\mathbf{R}_I\})\end{aligned} \tag{2}$$

for the electronic $\{\mathbf{r}_i\}$ and nuclear $\{\mathbf{R}_I\}$ degrees of freedom. The more convenient atomic units (a.u.) will be introduced at a later stage for reasons that will soon

become clear. Thus, only the bare electron–electron, electron–nuclear, and nuclear–nuclear Coulomb interactions are taken into account.

The goal of this section is to derive classical molecular dynamics [12,270,217] starting from Schrödinger's wave equation and following the elegant route of Tully [650,651]. To this end, the nuclear and electronic contributions to the total wavefunction $\Phi(\{\mathbf{r}_i\},\{\mathbf{R}_I\};t)$, which depends on *both* the nuclear and electronic coordinates, *have* to be separated. The simplest possible form is a product ansatz

$$\Phi(\{\mathbf{r}_i\},\{\mathbf{R}_I\};t) \approx \Psi(\{\mathbf{r}_i\};t)\,\chi(\{\mathbf{R}_I\};t)\,\exp\left[\frac{i}{\hbar}\int_{t_0}^{t} dt'\tilde{E}_{\rm e}(t')\right] \quad , \tag{3}$$

where the nuclear and electronic wavefunctions are separately normalized to unity at every instant of time, i.e. $\langle\chi;t|\chi;t\rangle = 1$ and $\langle\Psi;t|\Psi;t\rangle = 1$, respectively. In addition, a convenient phase factor

$$\tilde{E}_{\rm e} = \int d\mathbf{r}d\mathbf{R}\;\; \Psi^{\star}(\{\mathbf{r}_i\};t)\,\chi^{\star}(\{\mathbf{R}_I\};t)\,\mathcal{H}_{\rm e}\,\Psi(\{\mathbf{r}_i\};t)\,\chi(\{\mathbf{R}_I\};t) \tag{4}$$

was introduced at this stage such that the final equations will look nice; $\int d\mathbf{r}d\mathbf{R}$ refers to the integration over all $i = 1,\ldots$ and $I = 1,\ldots$ variables $\{\mathbf{r}_i\}$ and $\{\mathbf{R}_I\}$, respectively. It is mentioned in passing that this approximation is called a one–determinant or single–configuration ansatz for the *total* wavefunction, which at the end must lead to a mean–field description of the coupled dynamics. Note also that this product ansatz (excluding the phase factor) differs from the Born–Oppenheimer ansatz [340,350] for separating the fast and slow variables

$$\Phi_{\rm BO}(\{\mathbf{r}_i\},\{\mathbf{R}_I\};t) = \sum_{k=0}^{\infty}\tilde{\Psi}_k(\{\mathbf{r}_i\},\{\mathbf{R}_I\})\tilde{\chi}_k(\{\mathbf{R}_I\};t) \tag{5}$$

even in its one–determinant limit, where only a single electronic state k (evaluated for the nuclear configuration $\{\mathbf{R}_I\}$) is included in the expansion.

Inserting the separation ansatz Eq. (3) into Eqs. (1)–(2) yields (after multiplying from the left by $\langle\Psi|$ and $\langle\chi|$ and imposing energy conservation $d\langle\mathcal{H}\rangle/dt \equiv 0$) the following relations

$$i\hbar\frac{\partial\Psi}{\partial t} = -\sum_i \frac{\hbar^2}{2m_{\rm e}}\nabla_i^2\Psi + \left\{\int d\mathbf{R}\,\chi^{\star}(\{\mathbf{R}_I\};t)V_{\rm n-e}(\{\mathbf{r}_i\},\{\mathbf{R}_I\})\chi(\{\mathbf{R}_I\};t)\right\}\Psi \tag{6}$$

$$i\hbar\frac{\partial\chi}{\partial t} = -\sum_I \frac{\hbar^2}{2M_I}\nabla_I^2\chi + \left\{\int d\mathbf{r}\;\Psi^{\star}(\{\mathbf{r}_i\};t)\mathcal{H}_{\rm e}(\{\mathbf{r}_i\},\{\mathbf{R}_I\})\Psi(\{\mathbf{r}_i\};t)\right\}\chi \;\; . \tag{7}$$

This set of coupled equations defines the basis of the time–dependent self–consistent field (TDSCF) method introduced as early as 1930 by Dirac [162], see also Ref. [158]. Both electrons and nuclei move quantum–mechanically in time–dependent effective potentials (or self–consistently obtained average fields) obtained from appropriate averages (quantum mechanical expectation values $\langle\ldots\rangle$) over the other class of degrees of freedom (by using the nuclear and electronic wavefunctions, respectively). Thus, the single–determinant ansatz Eq. (3) produces, as already anticipated, a mean–field description of the coupled nuclear–electronic quantum dynamics. This is the price to pay for the simplest possible separation of electronic and nuclear variables.

The next step in the derivation of classical molecular dynamics is the task to approximate the nuclei as classical point particles. How can this be achieved in the framework of the TDSCF approach, given one quantum–mechanical wave equation describing all nuclei? A well–known route to extract classical mechanics from quantum mechanics in general starts with rewriting the corresponding wavefunction

$$\chi(\{\mathbf{R}_I\};t) = A(\{\mathbf{R}_I\};t)\, \exp\left[iS(\{\mathbf{R}_I\};t)/\hbar\right] \tag{8}$$

in terms of an amplitude factor A and a phase S which are both considered to be real and $A > 0$ in this polar representation, see for instance Refs. [163,425,535]. After transforming the nuclear wavefunction in Eq. (7) accordingly and after separating the real and imaginary parts, the TDSCF equation for the nuclei

$$\frac{\partial S}{\partial t} + \sum_I \frac{1}{2M_I} \left(\nabla_I S\right)^2 + \int d\mathbf{r}\; \Psi^\star \mathcal{H}_\mathrm{e} \Psi = \hbar^2 \sum_I \frac{1}{2M_I} \frac{\nabla_I^2 A}{A} \tag{9}$$

$$\frac{\partial A}{\partial t} + \sum_I \frac{1}{M_I} \left(\nabla_I A\right) \left(\nabla_I S\right) + \sum_I \frac{1}{2M_I} A \left(\nabla_I^2 S\right) = 0 \tag{10}$$

is (exactly) re–expressed in terms of the new variables A and S. This so–called "quantum fluid dynamical representation" Eqs. (9)–(10) can actually be used to solve the time–dependent Schrödinger equation [160]. The relation for A, Eq. (10), can be rewritten as a continuity equation [163,425,535] with the help of the identification of the nuclear density $|\chi|^2 \equiv A^2$ as directly obtained from the definition Eq. (8). This continuity equation is independent of $\hbar$ and ensures locally the conservation of the particle probability $|\chi|^2$ associated to the nuclei in the presence of a flux.

More important for the present purpose is a more detailed discussion of the relation for S, Eq. (9). This equation contains one term that depends on $\hbar$, a contribution that vanishes if the classical limit

$$\frac{\partial S}{\partial t} + \sum_I \frac{1}{2M_I} \left(\nabla_I S\right)^2 + \int d\mathbf{r}\; \Psi^\star \mathcal{H}_\mathrm{e} \Psi = 0 \tag{11}$$

is taken as $\hbar \to 0$; an expansion in terms of $\hbar$ would lead to a hierarchy of semi-classical methods [425,259]. The resulting equation is now isomorphic to equations of motion in the Hamilton–Jacobi formulation [244,540]

$$\frac{\partial S}{\partial t} + \mathcal{H}\left(\{\mathbf{R}_I\}, \{\nabla_I S\}\right) = 0 \tag{12}$$

of classical mechanics with the classical Hamilton function

$$\mathcal{H}(\{\mathbf{R}_I\}, \{\mathbf{P}_I\}) = T(\{\mathbf{P}_I\}) + V(\{\mathbf{R}_I\}) \tag{13}$$

defined in terms of (generalized) coordinates $\{\mathbf{R}_I\}$ and their conjugate momenta $\{\mathbf{P}_I\}$. With the help of the connecting transformation

$$\mathbf{P}_I \equiv \nabla_I S \tag{14}$$

the Newtonian equation of motion $\dot{\mathbf{P}}_I = -\nabla_I V(\{\mathbf{R}_I\})$ corresponding to Eq. (11)

$$\frac{d\mathbf{P}_I}{dt} = -\nabla_I \int d\mathbf{r}\ \Psi^\star \mathcal{H}_\mathrm{e} \Psi \qquad \text{or}$$

$$M_I \ddot{\mathbf{R}}_I(t) = -\nabla_I \int d\mathbf{r}\ \Psi^\star \mathcal{H}_\mathrm{e} \Psi \tag{15}$$

$$= -\nabla_I V_\mathrm{e}^\mathrm{E}(\{\mathbf{R}_I(t)\}) \tag{16}$$

can be read off. Thus, the nuclei move according to classical mechanics in an effective potential V_e^E due to the electrons. This potential is a function of only the nuclear positions at time t as a result of averaging $\mathcal{H}_\mathrm{e}$ over the electronic degrees of freedom, i.e. computing its quantum expectation value $\langle\Psi|\mathcal{H}_\mathrm{e}|\Psi\rangle$, while keeping the nuclear positions fixed at their instantaneous values $\{\mathbf{R}_I(t)\}$.

However, the nuclear wavefunction still occurs in the TDSCF equation for the electronic degrees of freedom and has to be replaced by the positions of the nuclei for consistency. In this case the classical reduction can be achieved simply by replacing the nuclear density $|\chi(\{\mathbf{R}_I\};t)|^2$ in Eq. (6) in the limit $\hbar \to 0$ by a product of delta functions $\prod_I \delta(\mathbf{R}_I - \mathbf{R}_I(t))$ centered at the instantaneous positions $\{\mathbf{R}_I(t)\}$ of the classical nuclei as given by Eq. (15). This yields e.g. for the position operator

$$\int d\mathbf{R}\ \chi^\star(\{\mathbf{R}_I\};t)\, \mathbf{R}_I\, \chi(\{\mathbf{R}_I\};t) \ \stackrel{\hbar\to 0}{\longrightarrow}\ \mathbf{R}_I(t) \tag{17}$$

the required expectation value. This classical limit leads to a time–dependent wave equation for the electrons

$$i\hbar\frac{\partial\Psi}{\partial t} = -\sum_i \frac{\hbar^2}{2m_\mathrm{e}}\nabla_i^2\Psi + V_\mathrm{n-e}(\{\mathbf{r}_i\},\{\mathbf{R}_I(t)\})\Psi$$

$$= \mathcal{H}_\mathrm{e}(\{\mathbf{r}_i\},\{\mathbf{R}_I(t)\})\ \ \Psi(\{\mathbf{r}_i\},\{\mathbf{R}_I\};t) \tag{18}$$

which evolve self–consistently as the classical nuclei are propagated via Eq. (15). Note that now $\mathcal{H}_\mathrm{e}$ and thus Ψ depend *parametrically* on the classical nuclear *positions* $\{\mathbf{R}_I(t)\}$ at time t through $V_\mathrm{n-e}(\{\mathbf{r}_i\},\{\mathbf{R}_I(t)\})$. This means that feedback between the classical and quantum degrees of freedom is incorporated in both directions (at variance with the "classical path" or Mott non–SCF approach to dynamics [650,651]).

The approach relying on solving Eq. (15) together with Eq. (18) is sometimes called "Ehrenfest molecular dynamics" in honor of Ehrenfest who was the first to address the question [a] of how Newtonian classical dynamics can be derived from Schrödinger's wave equation [174]. In the present case this leads to a hybrid or mixed approach because only the nuclei are forced to behave like classical particles, whereas the electrons are still treated as quantum objects.

Although the TDSCF approach underlying Ehrenfest molecular dynamics clearly is a mean–field theory, transitions between electronic states are included

[a] The opening statement of Ehrenfest's famous 1927 paper [174] reads:
"Es ist wünschenswert, die folgende Frage möglichst elementar beantworten zu können: *Welcher Rückblick ergibt sich vom Standpunkt der Quantenmechanik auf die Newtonschen Grundgleichungen der klassischen Mechanik?*"

in this scheme. This can be made evident by expanding the *electronic* wavefunction Ψ (as opposed to the *total* wavefunction Φ according to Eq. (5)) in terms of many electronic states or determinants Ψ_k

$$\Psi(\{\mathbf{r}_i\},\{\mathbf{R}_I\};t) = \sum_{k=0}^{\infty} c_k(t)\Psi_k(\{\mathbf{r}_i\};\{\mathbf{R}_I\}) \tag{19}$$

with complex coefficients $\{c_k(t)\}$. In this case, the coefficients $\{|c_k(t)|^2\}$ (with $\sum_k |c_k(t)|^2 \equiv 1$) describe explicitly the time evolution of the populations (occupations) of the different states $\{k\}$ whereas interferences are included via the $\{c_k^\star c_{l\neq k}\}$ contributions. One possible choice for the basis functions $\{\Psi_k\}$ is the adiabatic basis obtained from solving the time–independent electronic Schrödinger equation

$$\mathcal{H}_{\rm e}(\{\mathbf{r}_i\};\{\mathbf{R}_I\})\Psi_k = E_k(\{\mathbf{R}_I\})\Psi_k(\{\mathbf{r}_i\};\{\mathbf{R}_I\}) \ , \tag{20}$$

where $\{\mathbf{R}_I\}$ are the instantaneous nuclear positions at time t according to Eq. (15). The actual equations of motion in terms of the expansion coefficients $\{c_k\}$ are presented in Sect. 2.2.

At this stage a further simplification can be invoked by restricting the total electronic wave function Ψ to be the ground state wave function Ψ_0 of $\mathcal{H}_{\rm e}$ at each instant of time according to Eq. (20) and $|c_0(t)|^2 \equiv 1$ in Eq. (19). This should be a good approximation if the energy difference between Ψ_0 and the first excited state Ψ_1 is everywhere large compared to the thermal energy $k_{\rm B}T$, roughly speaking. In this limit the nuclei move according to Eq. (15) on a single potential energy surface

$$V_{\rm e}^{\rm E} = \int d\mathbf{r}\ \Psi_0^\star \mathcal{H}_{\rm e}\Psi_0 \equiv E_0(\{\mathbf{R}_I\}) \tag{21}$$

that can be computed by solving the *time–independent* electronic Schrödinger equation Eq. (20)

$$\mathcal{H}_{\rm e}\Psi_0 = E_0\Psi_0 \ , \tag{22}$$

for the ground state only. This leads to the identification $V_{\rm e}^{\rm E} \equiv E_0$ via Eq. (21), i.e. in this limit the Ehrenfest potential is identical to the ground–state Born–Oppenheimer potential.

As a consequence of this observation, it is conceivable to decouple the task of generating the nuclear dynamics from the task of computing the potential energy surface. In a first step E_0 is computed for many nuclear configurations by solving Eq. (22). In a second step, these data points are fitted to an analytical functional form to yield a global potential energy surface [539], from which the gradients can be obtained analytically. In a third step, the Newtonian equation of motion Eq. (16) is solved on this surface for many different initial conditions, producing a "swarm" of classical trajectories. This is, in a nutshell, the basis of *classical trajectory calculations* on global potential energy surfaces [649,541].

As already alluded to in the general introduction, such approaches suffer severely from the "dimensionality bottleneck" as the number of active nuclear degrees of freedom increases. One traditional way out of this dilemma is to approximate the

global potential energy surface

$$V_{\rm e}^{\rm E} \approx V_{\rm e}^{\rm approx}(\{\mathbf{R}_I\}) = \sum_{I=1}^{N} v_1(\mathbf{R}_I) + \sum_{I<J}^{N} v_2(\mathbf{R}_I, \mathbf{R}_J) + \sum_{I<J<K}^{N} v_3(\mathbf{R}_I, \mathbf{R}_J, \mathbf{R}_K) + \cdots \quad (23)$$

in terms of a truncated expansion of many–body contributions [253,12,270]. At this stage, the electronic degrees of freedom are replaced by interaction potentials $\{v_n\}$ and are not featured as explicit degrees of freedom in the equations of motion. Thus, the mixed quantum / classical problem is reduced to purely classical mechanics, once the $\{v_n\}$ are determined. *Classical molecular dynamics*

$$M_I \ddot{\mathbf{R}}_I(t) = -\nabla_I V_{\rm e}^{\rm approx}(\{\mathbf{R}_I(t)\}) \quad (24)$$

relies crucially on this idea, where typically only two–body v_2 or three–body v_3 interactions are taken into account [12,270], although more sophisticated models to include non–additive interactions such as polarization exist. This amounts to a dramatic simplification and removes the dimensionality bottleneck as the global potential surface is constructed from a manageable sum of additive few–body contributions — at the price of introducing a drastic approximation and of basically excluding chemical transformations from the realm of simulations.

As a result of this derivation, the essential assumptions underlying classical molecular dynamics become transparent: the electrons follow adiabatically the classical nuclear motion and can be integrated out so that the nuclei evolve on a single Born–Oppenheimer potential energy surface (typically but not necessarily given by the electronic ground state), which is in general approximated in terms of few–body interactions.

Actually, classical molecular dynamics for *many*–body systems is only made possible by somehow decomposing the global potential energy. In order to illustrate this point consider the simulation of $N = 500$ Argon atoms in the liquid phase [175] where the interactions can faithfully be described by additive two–body terms, i.e. $V_{\rm e}^{\rm approx}(\{\mathbf{R}_I\}) \approx \sum_{I<J}^{N} v_2(|\mathbf{R}_I - \mathbf{R}_J|)$. Thus, the determination of the pair potential v_2 from *ab initio* electronic structure calculations amounts to computing and fitting a one–dimensional function. The corresponding task to determine a global potential energy surface amounts to doing that in about 10^{1500} dimensions, which is simply impossible (and on top of that not necessary for Nobel gases!).

2.2 Ehrenfest Molecular Dynamics

A way out of the dimensionality bottleneck other than to approximate the global potential energy surface Eq. (23) or to reduce the number of active degrees of freedom is to take seriously the classical nuclei approximation to the TDSCF equations, Eq. (15) and (18). This amounts to computing the Ehrenfest force by actually solv-

ing numerically

$$
\begin{aligned}
M_I \ddot{\mathbf{R}}_I(t) &= -\nabla_I \int d\mathbf{r}\; \Psi^\star \mathcal{H}_\mathrm{e} \Psi \\
&= -\nabla_I \left\langle \Psi \left| \mathcal{H}_\mathrm{e} \right| \Psi \right\rangle \qquad (25) \\
&= -\nabla_I \left\langle \mathcal{H}_\mathrm{e} \right\rangle \\
&= -\nabla_I V_\mathrm{e}^\mathrm{E} \\
i\hbar \frac{\partial \Psi}{\partial t} &= \left[-\sum_i \frac{\hbar^2}{2m_\mathrm{e}} \nabla_i^2 + V_{\mathrm{n-e}}(\{\mathbf{r}_i\}, \{\mathbf{R}_I(t)\}) \right] \Psi \\
&= \mathcal{H}_\mathrm{e} \Psi \qquad (26)
\end{aligned}
$$

the coupled set of equations simultaneously. Thereby, the *a priori* construction of any type of potential energy surface is avoided from the outset by solving the time–dependent electronic Schrödinger equation "on–the–fly". This allows one to compute the force from $\nabla_I \langle \mathcal{H}_\mathrm{e} \rangle$ for each configuration $\{\mathbf{R}_I(t)\}$ generated by molecular dynamics; see Sect. 2.5 for the issue of using the so–called "Hellmann–Feynman forces" instead.

The corresponding equations of motion in terms of the adiabatic basis Eq. (20) and the time–dependent expansion coefficients Eq. (19) read [650,651]

$$
M_I \ddot{\mathbf{R}}_I(t) = -\sum_k |c_k(t)|^2 \nabla_I E_k - \sum_{k,l} c_k^\star c_l \left(E_k - E_l\right) \mathbf{d}_I^{kl} \qquad (27)
$$

$$
i\hbar \dot{c}_k(t) = c_k(t) E_k - i\hbar \sum_{I,l} c_l(t) \dot{\mathbf{R}}_I \mathbf{d}_I^{kl} \quad , \qquad (28)
$$

where the coupling terms are given by

$$
\mathbf{d}_I^{kl}(\{\mathbf{R}_I(t)\}) = \int d\mathbf{r}\; \Psi_k^\star \nabla_I \Psi_l \qquad (29)
$$

with the property $\mathbf{d}_I^{kk} \equiv \mathbf{0}$. The Ehrenfest approach is thus seen to include rigorously non–adiabatic transitions between different electronic states Ψ_k and Ψ_l within the framework of classical nuclear motion and the *mean–field* (TDSCF) approximation to the electronic structure, see e.g. Refs. [650,651] for reviews and for instance Ref. [532] for an implementation in terms of time–dependent density functional theory.

The restriction to one electronic state in the expansion Eq. (19), which is in most cases the ground state Ψ_0, leads to

$$
M_I \ddot{\mathbf{R}}_I(t) = -\nabla_I \left\langle \Psi_0 \left| \mathcal{H}_\mathrm{e} \right| \Psi_0 \right\rangle \qquad (30)
$$

$$
i\hbar \frac{\partial \Psi_0}{\partial t} = \mathcal{H}_\mathrm{e} \Psi_0 \qquad (31)
$$

as a special case of Eqs. (25)–(26); note that $\mathcal{H}_\mathrm{e}$ is time–dependent via the nuclear coordinates $\{\mathbf{R}_I(t)\}$. A point worth mentioning here is that the propagation of the wavefunction is unitary, i.e. the wavefunction preserves its norm and the set of orbitals used to build up the wavefunction will stay orthonormal, see Sect. 2.6.

Ehrenfest molecular dynamics is certainly the oldest approach to "on–the–fly" molecular dynamics and is typically used for collision– and scattering–type problems [154,649,426,532]. However, it was never in widespread use for systems with many active degrees of freedom typical for condensed matter problems for reasons that will be outlined in Sec. 2.6 (although a few exceptions exist [553,34,203,617] but here the number of explicitly treated electrons is fairly limited with the exception of Ref. [617]).

2.3 Born–Oppenheimer Molecular Dynamics

An alternative approach to include the electronic structure in molecular dynamics simulations consists in straightforwardly solving the *static* electronic structure problem in each molecular dynamics step given the set of *fixed* nuclear positions at that instance of time. Thus, the electronic structure part is reduced to solving a *time–independent* quantum problem, e.g. by solving the time–independent Schrödinger equation, concurrently to propagating the nuclei via classical molecular dynamics. Thus, the time–dependence of the electronic structure is a consequence of nuclear motion, and not intrinsic as in Ehrenfest molecular dynamics. The resulting Born–Oppenheimer molecular dynamics method is defined by

$$M_I \ddot{\mathbf{R}}_I(t) = -\nabla_I \min_{\Psi_0} \{\langle \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle\} \tag{32}$$

$$E_0 \Psi_0 = \mathcal{H}_\mathrm{e} \Psi_0 \tag{33}$$

for the electronic ground state. A deep difference with respect to Ehrenfest dynamics concerning the nuclear equation of motion is that the minimum of $\langle \mathcal{H}_\mathrm{e} \rangle$ has to be reached in each Born–Oppenheimer molecular dynamics step according to Eq. (32). In Ehrenfest dynamics, on the other hand, a wavefunction that minimized $\langle \mathcal{H}_\mathrm{e} \rangle$ initially will also stay in its respective minimum as the nuclei move according to Eq. (30)!

A natural and straightforward extension [281] of ground–state Born–Oppenheimer dynamics is to apply the same scheme to any excited electronic state Ψ_k without considering any interferences. In particular, this means that also the "diagonal correction terms" [340]

$$D_I^{kk}(\{\mathbf{R}_I(t)\}) = -\int d\mathbf{r}\, \Psi_k^\star \nabla_I^2 \Psi_k \tag{34}$$

are *always* neglected; the inclusion of such terms is discussed for instance in Refs. [650,651]. These terms renormalize the Born–Oppenheimer or "clamped nuclei" potential energy surface E_k of a given state Ψ_k (which might also be the ground state Ψ_0) and lead to the so–called "adiabatic potential energy surface" of that state [340]. Whence, Born–Oppenheimer molecular dynamics should not be called "adiabatic molecular dynamics", as is sometime done.

It is useful for the sake of later reference to formulate the Born–Oppenheimer equations of motion for the special case of effective one–particle Hamiltonians. This might be the Hartree–Fock approximation defined to be the variational minimum of the energy expectation value $\langle \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle$ given a single Slater determinant $\Psi_0 = \det\{\psi_i\}$ subject to the constraint that the one–particle orbitals ψ_i are orthonormal

$\langle \psi_i \, | \psi_j \, \rangle = \delta_{ij}$. The corresponding constraint minimization of the total energy with respect to the orbitals

$$\min_{\{\psi_i\}} \{ \langle \Psi_0 \, | \mathcal{H}_\mathrm{e} | \, \Psi_0 \rangle \} \Big|_{\{\langle \psi_i | \psi_j \, \rangle = \delta_{ij}\}} \tag{35}$$

can be cast into Lagrange's formalism

$$\mathcal{L} = - \langle \Psi_0 \, | \mathcal{H}_\mathrm{e} | \, \Psi_0 \rangle + \sum_{i,j} \Lambda_{ij} \left(\langle \psi_i \, | \psi_j \, \rangle - \delta_{ij} \right) \tag{36}$$

where Λ_{ij} are the associated Lagrangian multipliers. Unconstrained variation of this Lagrangian with respect to the orbitals

$$\frac{\delta \mathcal{L}}{\delta \psi_i^\star} \stackrel{!}{=} 0 \tag{37}$$

leads to the well–known Hartree–Fock equations

$$\mathcal{H}_\mathrm{e}^\mathrm{HF} \psi_i = \sum_j \Lambda_{ij} \psi_j \tag{38}$$

as derived in standard text books [604,418]; the diagonal canonical form $\mathcal{H}_\mathrm{e}^\mathrm{HF} \psi_i = \epsilon_i \psi_i$ is obtained after a unitary transformation and $\mathcal{H}_\mathrm{e}^\mathrm{HF}$ denotes the effective one–particle Hamiltonian, see Sect. 2.7 for more details. The equations of motion corresponding to Eqs. (32)–(33) read

$$M_I \ddot{\mathbf{R}}_I(t) = -\nabla_I \min_{\{\psi_i\}} \left\{ \left\langle \Psi_0 \left| \mathcal{H}_\mathrm{e}^\mathrm{HF} \right| \Psi_0 \right\rangle \right\} \tag{39}$$

$$0 = -\mathcal{H}_\mathrm{e}^\mathrm{HF} \psi_i + \sum_j \Lambda_{ij} \psi_j \tag{40}$$

for the Hartree–Fock case. A similar set of equations is obtained if Hohenberg–Kohn–Sham density functional theory [458,168] is used, where $\mathcal{H}_\mathrm{e}^\mathrm{HF}$ has to be replaced by the Kohn–Sham effective one–particle Hamiltonian H_e^KS, see Sect. 2.7 for more details. Instead of diagonalizing the one–particle Hamiltonian an alternative but equivalent approach consists in directly performing the constraint minimization according to Eq. (35) via nonlinear optimization techniques.

Early applications of Born–Oppenheimer molecular dynamics were performed in the framework of a semiempirical approximation to the electronic structure problem [669,671]. But only a few years later an *ab initio* approach was implemented within the Hartree–Fock approximation [365]. Born–Oppenheimer dynamics started to become popular in the early nineties with the availability of more efficient electronic structure codes in conjunction with sufficient computer power to solve "interesting problems", see for instance the compilation of such studies in Table 1 in a recent overview article [82].

Undoubtedly, the breakthrough of Hohenberg–Kohn–Sham density functional theory in the realm of chemistry – which took place around the same time – also helped a lot by greatly improving the "price / performance ratio" of the electronic structure part, see e.g. Refs. [694,590]. A third and possibly the crucial reason that boosted the field of *ab initio* molecular dynamics was the pioneering introduction of

the Car–Parrinello approach [108], see also Fig. 1. This technique opened novel avenues to treat large–scale problems via *ab initio* molecular dynamics and catalyzed the entire field by making "interesting calculations" possible, see also the closing section on applications.

2.4 Car–Parrinello Molecular Dynamics

2.4.1 Motivation

A non–obvious approach to cut down the computational expenses of molecular dynamics which includes the electrons in a single state was proposed by Car and Parrinello in 1985 [108]. In retrospect it can be considered to combine the advantages of both Ehrenfest and Born–Oppenheimer molecular dynamics. In Ehrenfest dynamics the time scale and thus the time step to integrate Eqs. (30) and (31) simultaneously is dictated by the intrinsic dynamics of the electrons. Since electronic motion is much faster than nuclear motion, the largest possible time step is that which allows to integrate the electronic equations of motion. Contrary to that, there is no electron dynamics whatsoever involved in solving the Born–Oppenheimer Eqs. (32)–(33), i.e. they can be integrated on the time scale given by nuclear motion. However, this means that the electronic structure problem has to be solved self–consistently at each molecular dynamics step, whereas this is avoided in Ehrenfest dynamics due to the possibility to propagate the wavefunction by applying the Hamiltonian to an initial wavefunction (obtained e.g. by one self–consistent diagonalization).

From an algorithmic point of view the main task achieved in ground–state Ehrenfest dynamics is simply to keep the wavefunction automatically minimized as the nuclei are propagated. This, however, might be achieved – in principle – by another sort of deterministic dynamics than first–order Schrödinger dynamics. In summary, the "Best of all Worlds Method" should (i) integrate the equations of motion on the (long) time scale set by the nuclear motion but nevertheless (ii) take intrinsically advantage of the smooth time–evolution of the dynamically evolving electronic subsystem as much as possible. The second point allows to circumvent explicit diagonalization or minimization to solve the electronic structure problem for the next molecular dynamics step. Car–Parrinello molecular dynamics is an efficient method to satisfy requirement (ii) in a numerically stable fashion and makes an acceptable compromise concerning the length of the time step (i).

2.4.2 Car–Parrinello Lagrangian and Equations of Motion

The basic idea of the Car–Parrinello approach can be viewed to exploit the quantum–mechanical adiabatic time–scale separation of fast electronic and slow nuclear motion by transforming that into classical–mechanical adiabatic energy–scale separation in the framework of dynamical systems theory. In order to achieve this goal the two–component quantum / classical problem is mapped onto a two–component purely classical problem with two separate energy scales at the expense of loosing the explicit time–dependence of the quantum subsystem dynamics. Furthermore, the central quantity, the energy of the electronic subsystem $\langle\Psi_0|\mathcal{H}_\mathrm{e}|\Psi_0\rangle$

evaluated with some wavefunction Ψ_0, is certainly a function of the nuclear positions $\{\mathbf{R}_I\}$. But at the same time it *can* be considered to be a functional of the wavefunction Ψ_0 and thus of a set of one–particle orbitals $\{\psi_i\}$ (or in general of other functions such as two–particle geminals) used to build up this wavefunction (being for instance a Slater determinant $\Psi_0 = \det\{\psi_i\}$ or a combination thereof). Now, in classical mechanics the force on the nuclei is obtained from the derivative of a Lagrangian with respect to the nuclear positions. This suggests that a functional derivative with respect to the orbitals, which are interpreted as classical fields, might yield the force on the orbitals, given a suitable Lagrangian. In addition, possible constraints within the set of orbitals have to be imposed, such as e.g. orthonormality (or generalized orthonormality conditions that include an overlap matrix).

Car and Parrinello postulated the following class of Lagrangians [108]

$$\mathcal{L}_{\mathrm{CP}} = \underbrace{\sum_I \frac{1}{2} M_I \dot{\mathbf{R}}_I^2 + \sum_i \frac{1}{2} \mu_i \left\langle \dot{\psi}_i \middle| \dot{\psi}_i \right\rangle}_{\text{kinetic energy}} - \underbrace{\langle \Psi_0 | \mathcal{H}_{\mathrm{e}} | \Psi_0 \rangle}_{\text{potential energy}} + \underbrace{\textit{constraints}}_{\text{orthonormality}} \tag{41}$$

to serve this purpose. The corresponding Newtonian equations of motion are obtained from the associated Euler–Lagrange equations

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{R}}_I} = \frac{\partial \mathcal{L}}{\partial \mathbf{R}_I} \tag{42}$$

$$\frac{d}{dt} \frac{\delta \mathcal{L}}{\delta \dot{\psi}_i^\star} = \frac{\delta \mathcal{L}}{\delta \psi_i^\star} \tag{43}$$

like in classical mechanics, but here for both the nuclear positions and the orbitals; note $\psi_i^\star = \langle \psi_i |$ and that the constraints are holonomic [244]. Following this route of ideas, generic Car–Parrinello equations of motion are found to be of the form

$$M_I \ddot{\mathbf{R}}_I(t) = -\frac{\partial}{\partial \mathbf{R}_I} \langle \Psi_0 | \mathcal{H}_{\mathrm{e}} | \Psi_0 \rangle + \frac{\partial}{\partial \mathbf{R}_I} \{\textit{constraints}\} \tag{44}$$

$$\mu_i \ddot{\psi}_i(t) = -\frac{\delta}{\delta \psi_i^\star} \langle \Psi_0 | \mathcal{H}_{\mathrm{e}} | \Psi_0 \rangle + \frac{\delta}{\delta \psi_i^\star} \{\textit{constraints}\} \tag{45}$$

where μ_i $(= \mu)$ are the "fictitious masses" or inertia parameters assigned to the orbital degrees of freedom; the units of the mass parameter μ are energy times a squared time for reasons of dimensionality. Note that the constraints within the total wavefunction lead to "constraint forces" in the equations of motion. Note also that these constraints

$$\textit{constraints} = \textit{constraints}\ (\{\psi_i\}, \{\mathbf{R}_I\}) \tag{46}$$

might be a function of both the set of orbitals $\{\psi_i\}$ and the nuclear positions $\{\mathbf{R}_I\}$. These dependencies have to be taken into account properly in deriving the Car–Parrinello equations following from Eq. (41) using Eqs. (42)–(43), see Sect. 2.5 for a general discussion and see e.g. Ref. [351] for a case with an additional dependence of the wavefunction constraint on nuclear positions.

According to the Car–Parrinello equations of motion, the nuclei evolve in time at a certain (instantaneous) physical temperature $\propto \sum_I M_I \dot{\mathbf{R}}_I^2$, whereas a "fictitious temperature" $\propto \sum_i \mu_i \langle \dot{\psi}_i | \dot{\psi}_i \rangle$ is associated to the electronic degrees of freedom. In this terminology, "low electronic temperature" or "cold electrons" means that the electronic subsystem is close to its instantaneous minimum energy $\min_{\{\psi_i\}} \langle \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle$, i.e. close to the exact Born–Oppenheimer surface. Thus, a ground–state wavefunction optimized for the initial configuration of the nuclei will stay close to its ground state also during time evolution if it is kept at a sufficiently low temperature.

The remaining task is to separate in practice nuclear and electronic motion such that the fast electronic subsystem stays cold also for long times but still follows the slow nuclear motion adiabatically (or instantaneously). Simultaneously, the nuclei are nevertheless kept at a much higher temperature. This can be achieved in nonlinear classical dynamics via decoupling of the two subsystems and (quasi–) adiabatic time evolution. This is possible if the power spectra stemming from both dynamics do not have substantial overlap in the frequency domain so that energy transfer from the "hot nuclei" to the "cold electrons" becomes practically impossible on the relevant time scales. This amounts in other words to imposing and maintaining a metastability condition in a complex dynamical system for sufficiently long times. How and to which extend this is possible in practice was investigated in detail in an important investigation based on well–controlled model systems [467,468] (see also Sects. 3.2 and 3.3 in Ref. [513]), with more mathematical rigor in Ref. [86], and in terms of a generalization to a second level of adiabaticity in Ref. [411].

2.4.3 Why Does the Car–Parrinello Method Work ?

In order to shed light on the title question, the dynamics generated by the Car–Parrinello Lagrangian Eq. (41) is analyzed [467] in more detail invoking a "classical dynamics perspective" of a simple model system (eight silicon atoms forming a periodic diamond lattice, local density approximation to density functional theory, normconserving pseudopotentials for core electrons, plane wave basis for valence orbitals, 0.3 fs time step with $\mu = 300$ a.u., in total 20 000 time steps or 6.3 ps, for full details see Ref. [467]); a concise presentation of similar ideas can be found in Ref. [110]. For this system the vibrational density of states or power spectrum of the electronic degrees of freedom, i.e. the Fourier transform of the statistically averaged velocity autocorrelation function of the classical fields

$$f(\omega) = \int_0^\infty dt \cos(\omega t) \sum_i \left\langle \dot{\psi}_i; t \left| \dot{\psi}_i; 0 \right. \right\rangle \tag{47}$$

is compared to the highest–frequency phonon mode $\omega_\mathrm{n}^\mathrm{max}$ of the nuclear subsystem in Fig. 2. From this figure it is evident that for the chosen parameters the nuclear and electronic subsystems are dynamically separated: their power spectra do not overlap so that energy transfer from the hot to the cold subsystem is expected to be prohibitively slow, see Sect. 3.3 in Ref. [513] for a similar argument.

This is indeed the case as can be verified in Fig. 3 where the conserved energy E_cons, physical total energy E_phys, electronic energy V_e, and fictitious kinetic energy

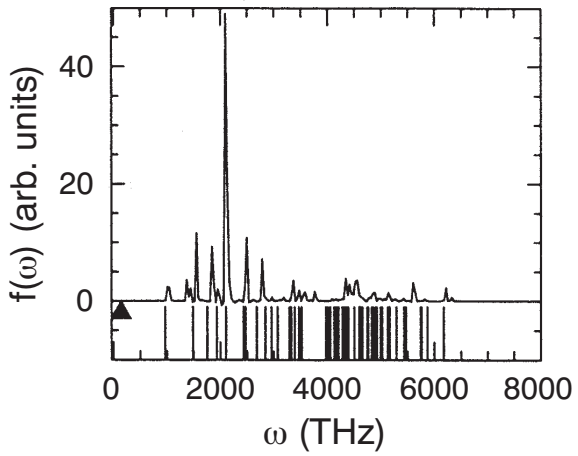


Figure 2. Vibrational density of states Eq. (47) (continuous spectrum in upper part) and harmonic approximation thereof Eq. (52) (stick spectrum in lower part) of the electronic degrees of freedom compared to the highest–frequency phonon mode $\omega_{\rm n}^{\rm max}$ (triangle) for a model system; for further details see text. Adapted from Ref. [467].

of the electrons $T_{\rm e}$

$$E_{\rm cons} = \sum_i \frac{1}{2}\mu_i \left\langle \dot{\psi}_i \middle| \dot{\psi}_i \right\rangle + \sum_I \frac{1}{2} M_I \dot{\mathbf{R}}_I^2 + \langle \Psi_0 | \mathcal{H}_{\rm e} | \Psi_0 \rangle \tag{48}$$

$$E_{\rm phys} = \sum_I \frac{1}{2} M_I \dot{\mathbf{R}}_I^2 + \langle \Psi_0 | \mathcal{H}_{\rm e} | \Psi_0 \rangle = E_{\rm cons} - T_{\rm e} \tag{49}$$

$$V_{\rm e} = \langle \Psi_0 | \mathcal{H}_{\rm e} | \Psi_0 \rangle \tag{50}$$

$$T_{\rm e} = \sum_i \frac{1}{2}\mu_i \left\langle \dot{\psi}_i \middle| \dot{\psi}_i \right\rangle \tag{51}$$

are shown for the same system as a function of time. First of all, there should be a conserved energy quantity according to classical dynamics since the constraints are holonomic [244]. Indeed "the Hamiltonian" or conserved energy $E_{\rm cons}$ is a constant of motion (with relative variations smaller than 10^{-6} and with no drift), which serves as an extremely sensitive check of the molecular dynamics algorithm. Contrary to that the electronic energy $V_{\rm e}$ displays a simple oscillation pattern due to the simplicity of the phonon modes.

Most importantly, the fictitious kinetic energy of the electrons $T_{\rm e}$ is found to perform *bound* oscillations around a *constant*, i.e. the electrons "do not heat up" systematically in the presence of the hot nuclei; note that $T_{\rm e}$ is a measure for deviations from the exact Born–Oppenheimer surface. Closer inspection shows actually two time scales of oscillations: the one visible in Fig. 3 stems from the drag exerted by the moving nuclei on the electrons and is the mirror image of the $V_{\rm e}$ fluctuations. Superimposed on top of that (not shown, but see Fig. 4(b)) are small–amplitude high frequency oscillations intrinsic to the fictitious electron dynamics with a period of only a fraction of the visible mode. These oscillations are actually instrumental

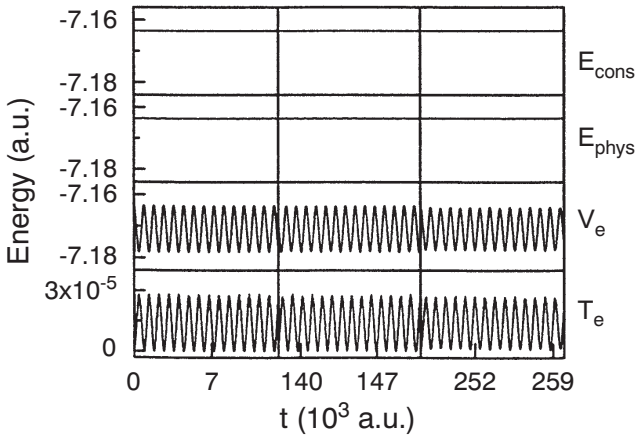


Figure 3. Various energies Eqs. (48)–(51) for a model system propagated via Car–Parrinello molecular dynamics for at short (up to 300 fs), intermediate, and long times (up to 6.3 ps); for further details see text. Adapted from Ref. [467].

for the stability of the Car–Parrinello dynamics, vide infra. But already the visible variations are three orders of magnitude smaller than the physically meaningful oscillations of V_e. As a result, $E_{\rm phys}$ defined as $E_{\rm cons} - T_e$ or equivalently as the sum of the nuclear kinetic energy and the electronic total energy (which serves as the potential energy for the nuclei) is essentially constant on the relevant energy and time scales. Thus, it behaves approximately like the strictly conserved total energy in classical molecular dynamics (with only nuclei as dynamical degrees of freedom) or in Born–Oppenheimer molecular dynamics (with fully optimized electronic degrees of freedom) and is therefore often denoted as the "physical total energy". This implies that the resulting physically significant dynamics of the nuclei yields an excellent approximation to microcanonical dynamics (and assuming ergodicity to the microcanonical ensemble). Note that a different explanation was advocated in Ref. [470] (see also Ref. [472], in particular Sect. VIII.B and C), which was however revised in Ref. [110]. A discussion similar in spirit to the one outlined here [467] is provided in Ref. [513], see in particular Sect. 3.2 and 3.3.

Given the adiabatic separation and the stability of the propagation, the central question remains if the forces acting on the nuclei are actually the "correct" ones in Car–Parrinello molecular dynamics. As a reference serve the forces obtained from full self–consistent minimizations of the electronic energy $\min_{\{\psi_i\}}\langle\Psi_0|\mathcal{H}_e|\Psi_0\rangle$ at each time step, i.e. Born–Oppenheimer molecular dynamics with extremely well converged wavefunctions. This is indeed the case as demonstrated in Fig. 4(a): the physically meaningful dynamics of the x–component of the force acting on one silicon atom in the model system obtained from stable Car–Parrinello fictitious dynamics propagation of the electrons and from iterative minimizations of the electronic energy are extremely close.

Better resolution of one oscillation period in (b) reveals that the gross deviations are also oscillatory but that they are four orders of magnitudes smaller than

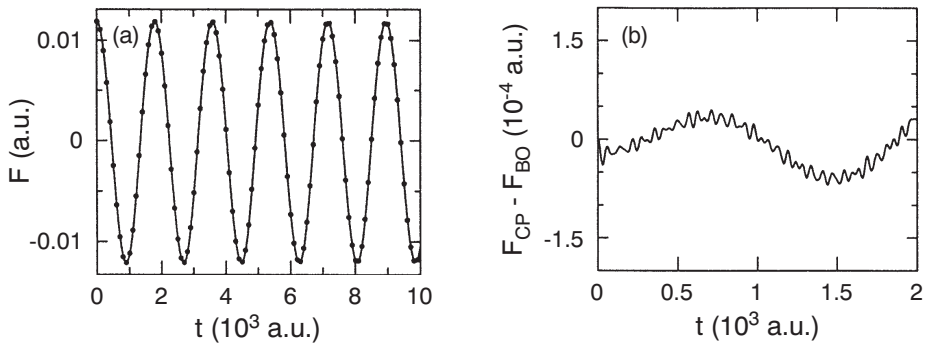


Figure 4. (a) Comparison of the x–component of the force acting on one atom of a model system obtained from Car–Parrinello (solid line) and well–converged Born–Oppenheimer (dots) molecular dynamics. (b) Enlarged view of the difference between Car–Parrinello and Born–Oppenheimer forces; for further details see text. Adapted from Ref. [467].

the physical variations of the force resolved in Fig. 4(a). These correspond to the "large–amplitude" oscillations of $T_{\rm e}$ visible in Fig. 3 due to the drag of the nuclei exerted on the quasi–adiabatically following electrons having a *finite* dynamical mass μ. Note that the inertia of the electrons also dampens artificially the nuclear motion (typically on a few–percent scale, see Sect. V.C.2 in Ref. [75] for an analysis and a renormalization correction of M_I) but decreases as the fictitious mass approaches the adiabatic limit $\mu \to 0$. Superimposed on the gross variation in (b) are again high–frequency *bound oscillatory small–amplitude* fluctuations like for $T_{\rm e}$. They lead *on physically relevant time scales* (i.e. those visible in Fig. 4(a)) to "averaged forces" that are very close to the exact ground–state Born–Oppenheimer forces. This feature is an important ingredient in the derivation of adiabatic dynamics [467,411].

In conclusion, the Car–Parrinello force can be said to deviate at most instants of time from the exact Born–Oppenheimer force. However, this does not disturb the physical time evolution due to (i) the smallness and boundedness of this difference *and* (ii) the intrinsic averaging effect of small–amplitude high–frequency oscillations within a few molecular dynamics time steps, i.e. on the sub–femtosecond time scale which is irrelevant for *nuclear* dynamics.

2.4.4 How to Control Adiabaticity ?

An important question is under which circumstances the adiabatic separation can be achieved, and how it can be controlled. A simple harmonic analysis of the frequency spectrum of the orbital classical fields close to the minimum defining the ground state yields [467]

$$\omega_{ij} = \left(\frac{2(\epsilon_i - \epsilon_j)}{\mu} \right)^{1/2} \quad , \tag{52}$$

where ϵ_j and ϵ_i are the eigenvalues of occupied and unoccupied orbitals, respectively; see Eq. (26) in Ref. [467] for the case where both orbitals are occupied ones. It can be seen from Fig. 2 that the harmonic approximation works faithfully as compared to the exact spectrum; see Ref. [471] and Sect. IV.A in Ref. [472] for a more general analysis of the associated equations of motion. Since this is in particular true for the lowest frequency $\omega_{\mathrm{e}}^{\mathrm{min}}$, the handy analytic estimate for the lowest possible electronic frequency

$$\omega_{\mathrm{e}}^{\mathrm{min}} \propto \left(\frac{E_{\mathrm{gap}}}{\mu} \right)^{1/2} \quad , \tag{53}$$

shows that this frequency increases like the square root of the electronic energy difference E_{gap} between the lowest unoccupied and the highest occupied orbital. On the other hand it increases similarly for a decreasing fictitious mass parameter μ.

In order to guarantee the adiabatic separation, the frequency difference $\omega_{\mathrm{e}}^{\mathrm{min}} - \omega_{\mathrm{n}}^{\mathrm{max}}$ should be large, see Sect. 3.3 in Ref. [513] for a similar argument. But both the highest phonon frequency $\omega_{\mathrm{n}}^{\mathrm{max}}$ and the energy gap E_{gap} are quantities that a dictated by the physics of the system. Whence, the only parameter in our hands to control adiabatic separation is the fictitious mass, which is therefore also called "adiabaticity parameter". However, decreasing μ not only shifts the electronic spectrum upwards on the frequency scale, but also stretches the entire frequency spectrum according to Eq. (52). This leads to an increase of the maximum frequency according to

$$\omega_{\mathrm{e}}^{\mathrm{max}} \propto \left(\frac{E_{\mathrm{cut}}}{\mu} \right)^{1/2} \quad , \tag{54}$$

where E_{cut} is the largest kinetic energy in an expansion of the wavefunction in terms of a plane wave basis set, see Sect. 3.1.3.

At this place a limitation to decrease μ arbitrarily kicks in due to the maximum length of the molecular dynamics time step Δt^{max} that can be used. The time step is inversely proportional to the highest frequency in the system, which is $\omega_{\mathrm{e}}^{\mathrm{max}}$ and thus the relation

$$\Delta t^{\mathrm{max}} \propto \left(\frac{\mu}{E_{\mathrm{cut}}} \right)^{1/2} \tag{55}$$

governs the largest time step that is possible. As a consequence, Car–Parrinello simulators have to find their way between Scylla and Charybdis and have to make a compromise on the control parameter μ; typical values for large–gap systems are μ = 500–1500 a.u. together with a time step of about 5–10 a.u. (0.12–0.24 fs). Recently, an algorithm was devised that optimizes μ during a particular simulation given a fixed accuracy criterion [87]. Note that a poor man's way to keep the time step large and still increase μ in order to satisfy adiabaticity is to choose heavier nuclear masses. That depresses the largest phonon or vibrational frequency $\omega_{\mathrm{n}}^{\mathrm{max}}$ of the nuclei (at the cost of renormalizing all *dynamical* quantities in the sense of classical isotope effects).

Up to this point the entire discussion of the stability and adiabaticity issues was based on model systems, approximate and mostly qualitative in nature. But recently it was actually proven [86] that the deviation or the absolute error Δ_μ of the Car–Parrinello trajectory relative to the trajectory obtained on the exact Born–Oppenheimer potential energy surface is controlled by μ:
Theorem 1 iv.): There are constants $C > 0$ and $\mu^\star > 0$ such that

$$\Delta_\mu = \left|\mathbf{R}^\mu(t) - \mathbf{R}^0(t)\right| + \left|\left|\,\psi^\mu; t\right\rangle - \left|\,\psi^0; t\right\rangle\right| \leq C\mu^{1/2} \quad , \; 0 \leq t \leq T \tag{56}$$

and the fictitious kinetic energy satisfies

$$T_\mathrm{e} = \frac{1}{2}\mu \left\langle \dot{\psi}^\mu; t \left| \dot{\psi}^\mu; t \right.\right\rangle \leq C\mu \quad , \; 0 \leq t \leq T \tag{57}$$

for all values of the parameter μ satisfying $0 < \mu \leq \mu^\star$, where up to time $T > 0$ there exists a unique nuclear trajectory on the exact Born–Oppenheimer surface with $\omega_\mathrm{e}^\mathrm{min} > 0$ for $0 \leq t \leq T$, i.e. there is "always" a finite electronic excitation gap. Here, the superscript μ or 0 indicates that the trajectory was obtained via Car–Parrinello molecular dynamics using a finite mass μ or via dynamics on the exact Born–Oppenheimer surface, respectively. Note that not only the nuclear trajectory is shown to be close to the correct one, but also the wavefunction is proven to stay close to the fully converged one up to time T. Furthermore, it was also investigated what happens if the initial wavefunction at $t = 0$ is not the minimum of the electronic energy $\langle \mathcal{H}_\mathrm{e} \rangle$ but trapped in an excited state. In this case it is found that the propagated wavefunction will keep on oscillating at $t > 0$ also for $\mu \to 0$ and not even time averages converge to any of the eigenstates. Note that this does not preclude Car–Parrinello molecular dynamics in excited states, which is possible given a properly "minimizable" expression for the electronic energy, see e.g. Refs. [281,214]. However, this finding might have crucial implications for electronic level–crossing situations.

What happens if the electronic gap is very small or even vanishes $E_\mathrm{gap} \to 0$ as is the case for metallic systems? In this limit, all the above–given arguments break down due to the occurrence of zero–frequency electronic modes in the power spectrum according to Eq. (53), which necessarily overlap with the phonon spectrum. Following an idea of Sprik [583] applied in a classical context it was shown that the coupling of separate Nosé–Hoover thermostats [12,270,217] to the nuclear and electronic subsystem can maintain adiabaticity by counterbalancing the energy flow from ions to electrons so that the electrons stay "cool" [74]; see Ref. [204] for a similar idea to restore adiabaticity. Although this method is demonstrated to work in practice [464], this *ad hoc* cure is not entirely satisfactory from both a theoretical and practical point of view so that the well–controlled Born–Oppenheimer approach is recommended for strongly metallic systems. An additional advantage for metallic systems is that the latter is also better suited to sample many **k**–points (see Sect. 3.1.3), allows easily for fractional occupation numbers [458,168], and can handle efficiently the so–called charge sloshing problem [472].

2.4.5 The Quantum Chemistry Viewpoint

In order to understand Car–Parrinello molecular dynamics also from the "quantum chemistry perspective", it is useful to formulate it for the special case of the Hartree–Fock approximation using

$$\mathcal{L}_{\mathrm{CP}} = \sum_I \frac{1}{2} M_I \dot{\mathbf{R}}_I^2 + \sum_i \frac{1}{2} \mu_i \left\langle \dot{\psi}_i \middle| \dot{\psi}_i \right\rangle - \left\langle \Psi_0 | \mathcal{H}_{\mathrm{e}}^{\mathrm{HF}} | \Psi_0 \right\rangle + \sum_{i,j} \Lambda_{ij} \left(\langle \psi_i | \psi_j \rangle - \delta_{ij} \right) \quad . \tag{58}$$

The resulting equations of motion

$$M_I \ddot{\mathbf{R}}_I(t) = -\nabla_I \left\langle \Psi_0 \middle| \mathcal{H}_{\mathrm{e}}^{\mathrm{HF}} \middle| \Psi_0 \right\rangle \tag{59}$$

$$\mu_i \ddot{\psi}_i(t) = -\mathcal{H}_{\mathrm{e}}^{\mathrm{HF}} \psi_i + \sum_j \Lambda_{ij} \psi_j \tag{60}$$

are very close to those obtained for Born–Oppenheimer molecular dynamics Eqs. (39)–(40) except for (i) no need to minimize the electronic total energy expression and (ii) featuring the additional fictitious kinetic energy term associated to the orbital degrees of freedom. It is suggestive to argue that both sets of equations become identical if the term $|\mu_i \ddot{\psi}_i(t)|$ is small at any time t compared to the physically relevant forces on the right–hand–side of both Eq. (59) and Eq. (60). This term being zero (or small) means that one is at (or close to) the minimum of the electronic energy $\langle \Psi_0 | \mathcal{H}_{\mathrm{e}}^{\mathrm{HF}} | \Psi_0 \rangle$ since time derivatives of the orbitals $\{\psi_i\}$ can be considered as variations of Ψ_0 and thus of the expectation value $\langle \mathcal{H}_{\mathrm{e}}^{\mathrm{HF}} \rangle$ itself. In other words, no forces act on the wavefunction if $\mu_i \ddot{\psi}_i \equiv 0$. In conclusion, the Car–Parrinello equations are expected to produce the correct dynamics and thus physical trajectories in the microcanonical ensemble in this idealized limit. But if $|\mu_i \ddot{\psi}_i(t)|$ is small for all i, this also implies that the associated kinetic energy $T_{\mathrm{e}} = \sum_i \mu_i \langle \dot{\psi}_i | \dot{\psi}_i \rangle / 2$ is small, which connects these more qualitative arguments with the previous discussion [467].

At this stage, it is also interesting to compare the structure of the Lagrangian Eq. (58) and the Euler–Lagrange equation Eq. (43) for Car–Parrinello dynamics to the analogues equations (36) and (37), respectively, used to derive "Hartree–Fock statics". The former reduce to the latter if the dynamical aspect and the associated time evolution is neglected, that is in the limit that the nuclear and electronic momenta are absent or constant. Thus, the Car–Parrinello ansatz, namely Eq. (41) together with Eqs. (42)–(43), can also be viewed as a prescription to derive a new class of "dynamical *ab initio* methods" in very general terms.

2.4.6 The Simulated Annealing and Optimization Viewpoints

In the discussion given above, Car–Parrinello molecular dynamics was motivated by "combining" the positive features of both Ehrenfest and Born–Oppenheimer molecular dynamics as much as possible. Looked at from another side, the Car–Parrinello method can also be considered as an ingenious way to perform *global* optimizations (minimizations) of nonlinear functions, here $\langle \Psi_0 | \mathcal{H}_{\mathrm{e}} | \Psi_0 \rangle$, in a high–dimensional parameter space including complicated constraints. The optimization

parameters are those used to represent the total wavefunction Ψ_0 in terms of simpler functions, for instance expansion coefficients of the orbitals in terms of Gaussians or plane waves, see e.g. Refs. [583,375,693,608] for applications of the same idea in other fields.

Keeping the nuclei frozen for a moment, one could start this optimization procedure from a "random wavefunction" which certainly does not minimize the electronic energy. Thus, its fictitious kinetic energy is high, the electronic degrees of freedom are "hot". This energy, however, can be extracted from the system by systematically cooling it to lower and lower temperatures. This can be achieved in an elegant way by adding a non–conservative damping term to the electronic Car–Parrinello equation of motion Eq. (45)

$$\mu_i \ddot{\psi}_i(t) = -\frac{\delta}{\delta \psi_i^\star} \langle \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle + \frac{\delta}{\delta \psi_i^\star} \{ constraints \} - \gamma_\mathrm{e} \mu_i \psi_i \ , \tag{61}$$

where $\gamma_\mathrm{e} \geq 0$ is a friction constant that governs the rate of energy dissipation [610]; alternatively, dissipation can be enforced in a discrete fashion by reducing the velocities by multiplying them with a constant factor < 1. Note that this deterministic and dynamical method is very similar in spirit to simulated annealing [332] invented in the framework of the stochastic Monte Carlo approach in the canonical ensemble. If the energy dissipation is done slowly, the wavefunction will find its way down to the minimum of the energy. At the end, an intricate global optimization has been performed!

If the nuclei are allowed to move according to Eq. (44) in the presence of another damping term a combined or simultaneous optimization of both electrons and nuclei can be achieved, which amounts to a "global geometry optimization". This perspective is stressed in more detail in the review Ref. [223] and an implementation of such ideas within the `CADPAC` quantum chemistry code is described in Ref. [692]. This operational mode of Car–Parrinello molecular dynamics is related to other optimization techniques where it is aimed to optimize simultaneously both the structure of the nuclear skeleton and the electronic structure. This is achieved by considering the nuclear coordinates and the expansion coefficients of the orbitals as variation parameters on the same footing [49,290,608]. But Car–Parrinello molecular dynamics is more than that because even if the nuclei continuously move according to Newtonian dynamics at finite temperature an initially optimized wavefunction will stay optimal along the nuclear trajectory.

2.4.7 The Extended Lagrangian Viewpoint

There is still another way to look at the Car–Parrinello method, namely in the light of so–called "extended Lagrangians" or "extended system dynamics" [14], see e.g. Refs. [136,12,270,585,217] for introductions. The basic idea is to couple additional degrees of freedom to the Lagrangian of interest, thereby "extending" it by increasing the dimensionality of phase space. These degrees of freedom are treated like classical particle coordinates, i.e. they are in general characterized by "positions", "momenta", "masses", "interactions" and a "coupling term" to the particle's positions and momenta. In order to distinguish them from the physical degrees of freedom, they are often called "fictitious degrees of freedom".

The corresponding equations of motion follow from the Euler–Lagrange equations and yield a microcanonical ensemble in the extended phase space where the Hamiltonian of the extended system is strictly conserved. In other words, the Hamiltonian of the physical (sub–) system is no more (strictly) conserved, and the produced ensemble is no more the microcanonical one. Any extended system dynamics is constructed such that time–averages taken in that part of phase space that is associated to the physical degrees of freedom (obtained from a partial trace over the fictitious degrees of freedom) are physically meaningful. Of course, dynamics and thermodynamics of the system are affected by adding fictitious degrees of freedom, the classic examples being temperature and pressure control by thermostats and barostats, see Sect. 4.2.

In the case of Car–Parrinello molecular dynamics, the basic Lagrangian for Newtonian dynamics of the nuclei is actually extended by classical *fields* $\{\psi_i(\mathbf{r})\}$, i.e. functions instead of coordinates, which represent the quantum wavefunction. Thus, vector products or absolute values have to be generalized to scalar products and norms of the fields. In addition, the "positions" of these fields $\{\psi_i\}$ actually have a physical meaning, contrary to their momenta $\{\dot{\psi}_i\}$.

2.5 What about Hellmann–Feynman Forces ?

An important ingredient in all dynamics methods is the efficient calculation of the forces acting on the nuclei, see Eqs. (30), (32), and (44). The straightforward numerical evaluation of the derivative

$$\mathbf{F}_I = -\nabla_I \langle \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle \tag{62}$$

in terms of a finite–difference approximation of the total electronic energy is both too costly and too inaccurate for dynamical simulations. What happens if the gradients are evaluated analytically? In addition to the derivative of the Hamiltonian itself

$$\begin{aligned} \nabla_I \langle \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle = & \langle \Psi_0 | \nabla_I \mathcal{H}_\mathrm{e} | \Psi_0 \rangle \\ & + \langle \nabla_I \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle + \langle \Psi_0 | \mathcal{H}_\mathrm{e} | \nabla_I \Psi_0 \rangle \end{aligned} \tag{63}$$

there are in general also contributions from variations of the wavefunction $\sim \nabla_I \Psi_0$. In general means here that these contributions vanish exactly

$$\mathbf{F}_I^\mathrm{HFT} = - \langle \Psi_0 | \nabla_I \mathcal{H}_\mathrm{e} | \Psi_0 \rangle \tag{64}$$

if the wavefunction is an exact eigenfunction (or stationary state wavefunction) of the particular Hamiltonian under consideration. This is the content of the often–cited Hellmann–Feynman Theorem [295,186,368], which is also valid for many variational wavefunctions (e.g. the Hartree–Fock wavefunction) provided that *complete basis sets* are used. If this is not the case, which has to be assumed for numerical calculations, the additional terms have to be evaluated explicitly.

In order to proceed a Slater determinant $\Psi_0 = \det\{\psi_i\}$ of one–particle orbitals ψ_i, which themselves are expanded

$$\psi_i = \sum_\nu c_{i\nu} \, f_\nu(\mathbf{r}; \{\mathbf{R}_I\}) \tag{65}$$

in terms of a linear combination of basis functions $\{f_\nu\}$, is used in conjunction with an effective one–particle Hamiltonian (such as e.g. in Hartree–Fock or Kohn–Sham theories). The basis functions might depend explicitly on the nuclear positions (in the case of basis functions with origin such as atom–centered orbitals), whereas the expansion coefficients always carry an implicit dependence. This means that from the outset two sorts of forces are expected

$$\nabla_I \psi_i = \sum_\nu \left(\nabla_I c_{i\nu}\right)\ f_\nu(\mathbf{r};\{\mathbf{R}_I\}) + \sum_\nu c_{i\nu}\ \left(\nabla_I f_\nu(\mathbf{r};\{\mathbf{R}_I\})\right) \tag{66}$$

in addition to the Hellmann–Feynman force Eq. (64).

Using such a linear expansion Eq. (65), the force contributions stemming from the nuclear gradients of the wavefunction in Eq. (63) can be disentangled into two terms. The first one is called "incomplete–basis–set correction" (IBS) in solid state theory [49,591,180] and corresponds to the "wavefunction force" [494] or "Pulay force" in quantum chemistry [494,496]. It contains the nuclear gradients of the basis functions

$$\mathbf{F}_I^{\mathrm{IBS}} = -\sum_{i\nu\mu} \left(\left\langle \nabla_I f_\nu \left| \mathcal{H}_{\mathrm{e}}^{\mathrm{NSC}} - \epsilon_i \right| f_\mu \right\rangle + \left\langle f_\nu \left| \mathcal{H}_{\mathrm{e}}^{\mathrm{NSC}} - \epsilon_i \right| \nabla f_\mu \right\rangle \right) \tag{67}$$

and the (in practice non–self–consistent) effective one–particle Hamiltonian [49,591]. The second term leads to the so–called "non–self–consistency correction" (NSC) of the force [49,591]

$$\mathbf{F}_I^{\mathrm{NSC}} = -\int d\mathbf{r}\ \left(\nabla_I n\right) \left(V^{\mathrm{SCF}} - V^{\mathrm{NSC}}\right) \tag{68}$$

and is governed by the difference between the self–consistent ("exact") potential or field V^{SCF} and its non–self–consistent (or approximate) counterpart V^{NSC} associated to $\mathcal{H}_{\mathrm{e}}^{\mathrm{NSC}}$; $n(\mathbf{r})$ is the charge density. In summary, the total force needed in *ab initio* molecular dynamics simulations

$$\mathbf{F}_I = \mathbf{F}_I^{\mathrm{HFT}} + \mathbf{F}_I^{\mathrm{IBS}} + \mathbf{F}_I^{\mathrm{NSC}} \tag{69}$$

comprises in general three qualitatively different terms; see the tutorial article Ref. [180] for a further discussion of core vs. valence states and the effect of pseudopotentials. Assuming that self–consistency is exactly satisfied (which is *never* going to be the case in numerical calculations), the force $\mathbf{F}_I^{\mathrm{NSC}}$ vanishes and $\mathcal{H}_{\mathrm{e}}^{\mathrm{SCF}}$ has to be used to evaluate $\mathbf{F}_I^{\mathrm{IBS}}$. The Pulay contribution vanishes in the limit of using a complete basis set (which is also not possible to achieve in actual calculations).

The most obvious simplification arises if the wavefunction is expanded in terms of originless basis functions such as plane waves, see Eq. (100). In this case the Pulay force vanishes exactly, which applies of course to all *ab initio* molecular dynamics schemes (i.e. Ehrenfest, Born–Oppenheimer, and Car–Parrinello) using that particular basis set. This statement is true for calculations where the number of plane waves is fixed. If the number of plane waves changes, such as in (constant pressure) calculations with varying cell volume / shape where the energy cutoff is strictly fixed instead, Pulay stress contributions crop up [219,245,660,211,202], see Sect. 4.2. If basis sets with origin are used instead of plane waves Pulay forces arise always and have to be included explicitely in force calculations, see e.g. Refs. [75,370,371] for such methods. Another interesting simplification of the same origin is noted in passing:

there is no basis set superposition error (BSSE) [88] in plane wave–based electronic structure calculations.

A non–obvious and more delicate term in the context of *ab initio* molecular dynamics is the one stemming from non–self–consistency Eq. (68). This term vanishes only if the wavefunction Ψ_0 is an eigenfunction of the Hamiltonian *within the subspace spanned by the finite basis set used*. This demands less than the Hellmann–Feynman theorem where Ψ_0 has to be an exact eigenfunction of the Hamiltonian and a complete basis set has to be used in turn. In terms of electronic structure calculations complete self–consistency (within a given incomplete basis set) has to be reached in order that $\mathbf{F}_I^{\mathrm{NSC}}$ vanishes. Thus, in numerical calculations the NSC term can be made arbitrarily small by optimizing the effective Hamiltonian and by determining its eigenfunctions to very high accuracy, but it can never be suppressed completely.

The crucial point is, however, that in Car–Parrinello as well as in Ehrenfest molecular dynamics it is not the minimized expectation value of the electronic Hamiltonian, i.e. $\min_{\Psi_0}\{\langle\Psi_0|\mathcal{H}_{\mathrm{e}}|\Psi_0\rangle\}$, that yields the consistent forces. What is merely needed is to evaluate the expression $\langle\Psi_0|\mathcal{H}_{\mathrm{e}}|\Psi_0\rangle$ with the Hamiltonian and the associated wavefunction available at a certain time step, compare Eq. (32) to Eq. (44) or (30). In other words, it is not required (concerning the present discussion of the contributions to the force!) that the expectation value of the electronic Hamiltonian is actually completely minimized for the nuclear configuration at that time step. Whence, full self–consistency is not required for this purpose in the case of Car–Parrinello (and Ehrenfest) molecular dynamics. As a consequence, the non–self–consistency correction to the force $\mathbf{F}_I^{\mathrm{NSC}}$ Eq. (68) is irrelevant in Car–Parrinello (and Ehrenfest) simulations.

In Born–Oppenheimer molecular dynamics, on the other hand, the expectation value of the Hamiltonian has to be minimized for each nuclear configuration before taking the gradient to obtain the consistent force! In this scheme there is (independently from the issue of Pulay forces) *always* the non–vanishing contribution of the non–self–consistency force, which is unknown by its very definition (if it were know, the problem was solved, see Eq. (68)). It is noted in passing that there are estimation schemes available that correct *approximately* for this systematic error in Born–Oppenheimer dynamics and lead to significant time–savings, see e.g. Ref. [344].

Heuristically one could also argue that within Car–Parrinello dynamics the non–vanishing non–self–consistency force is kept under control or counterbalanced by the non–vanishing "mass times acceleration term" $\mu_i\ddot{\psi}_i(t) \approx 0$, which is small but not identical to zero and oscillatory. This is sufficient to keep the propagation stable, whereas $\mu_i\ddot{\psi}_i(t) \equiv 0$, i.e. an extremely tight minimization $\min_{\Psi_0}\{\langle\Psi_0|\mathcal{H}_{\mathrm{e}}|\Psi_0\rangle\}$, is required by its very definition in order to make the Born–Oppenheimer approach stable, compare again Eq. (60) to Eq. (40). Thus, also from this perspective it becomes clear that the fictitious kinetic energy of the electrons and thus their fictitious temperature is a measure for the departure from the exact Born–Oppenheimer surface during Car–Parrinello dynamics.

Finally, the present discussion shows that nowhere in these force derivations was *made use of* the Hellmann–Feynman theorem as is sometimes stated. Actually, it is known for a long time that this theorem is quite useless for numerical electronic

structure calculations, see e.g. Refs. [494,49,496] and references therein. Rather *it turns out* that in the case of Car–Parrinello calculations using a plane wave basis the resulting relation for the force, namely Eq. (64), looks like the one obtained by simply invoking the Hellmann–Feynman theorem at the outset.

It is interesting to recall that the Hellmann–Feynman theorem as applied to a non–eigenfunction of a Hamiltonian yields only a first–order perturbative estimate of the exact force [295,368]. The same argument applies to *ab initio* molecular dynamics calculations where possible force corrections according to Eqs. (67) and (68) are neglected without justification. Furthermore, such simulations can of course not strictly conserve the total Hamiltonian $E_{\rm cons}$ Eq. (48). Finally, it should be stressed that possible contributions to the force in the nuclear equation of motion Eq. (44) due to *position–dependent* wavefunction *constraints* have to be evaluated following the same procedure. This leads to similar "correction terms" to the force, see e.g. Ref. [351] for such a case.

2.6 Which Method to Choose ?

Presumably the most important question for practical applications is which *ab initio* molecular dynamics method is the most efficient in terms of computer time given a specific problem. An *a priori* advantage of both the Ehrenfest and Car–Parrinello schemes over Born–Oppenheimer molecular dynamics is that no diagonalization of the Hamiltonian (or the equivalent minimization of an energy functional) is necessary, except at the very first step in order to obtain the initial wavefunction. The difference is, however, that the Ehrenfest time–evolution according to the time–dependent Schrödinger equation Eq. (26) conforms to a unitary propagation [341,366,342]

$$\Psi(t_0+\Delta t) = \exp\left[-i\mathcal{H}_{\rm e}(t_0)\Delta t/\hbar\right]\Psi(t_0) \tag{70}$$

$$\begin{aligned}\Psi(t_0+m\,\Delta t) &= \exp\left[-i\mathcal{H}_{\rm e}(t_0+(m-1)\Delta t)\,\Delta t/\hbar\right]\\ &\quad\times\cdots\\ &\quad\times\exp\left[-i\mathcal{H}_{\rm e}(t_0+2\Delta t)\,\Delta t/\hbar\right]\\ &\quad\times\exp\left[-i\mathcal{H}_{\rm e}(t_0+\Delta t)\,\Delta t/\hbar\right]\\ &\quad\times\exp\left[-i\mathcal{H}_{\rm e}(t_0)\,\Delta t/\hbar\right]\Psi(t_0)\end{aligned} \tag{71}$$

$$\Psi(t_0+t^{\rm max}) \stackrel{\Delta t\to 0}{=} \mathsf{T}\exp\left[-\frac{i}{\hbar}\int_{t_0}^{t_0+t^{\rm max}} dt\,\mathcal{H}_{\rm e}(t)\right]\Psi(t_0) \tag{72}$$

for infinitesimally short times given by the time step $\Delta t = t^{\rm max}/m$; here T is the time–ordering operator and $\mathcal{H}_{\rm e}(t)$ is the Hamiltonian (which is *implicitly* time–dependent via the positions $\{\mathbf{R}_I(t)\}$) evaluated at time t using e.g. split operator techniques [183]. Thus, the wavefunction Ψ will conserve its norm and in particular orbitals used to expand it will stay orthonormal, see e.g. Ref. [617]. In Car–Parrinello molecular dynamics, on the contrary, the orthonormality has to be imposed brute force by Lagrange multipliers, which amounts to an additional orthogonalization at each molecular dynamics step. If this is not properly done, the orbitals will become non–orthogonal and the wavefunction unnormalized, see e.g. Sect. III.C.1 in Ref. [472].

But this theoretical disadvantage of Car–Parrinello vs. Ehrenfest dynamics is in reality more than compensated by the possibility to use a much larger time step in order to propagate the electronic (and thus nuclear) degrees of freedom in the former scheme. In both approaches, there is the time scale inherent to the nuclear motion $\tau_{\rm n}$ and the one stemming from the electronic dynamics $\tau_{\rm e}$. The first one can be estimated by considering the highest phonon or vibrational frequency and amounts to the order of $\tau_{\rm n} \sim 10^{-14}$ s (or 0.01 ps or 10 fs, assuming a maximum frequency of about 4000 cm^{-1}). This time scale depends only on the physics of the problem under consideration and yields an upper limit for the timestep $\Delta t^{\rm max}$ that can be used in order to integrate the equations of motion, e.g. $\Delta t^{\rm max} \approx \tau_{\rm n}/10$.

The fasted electronic motion in Ehrenfest dynamics can be estimated within a plane wave expansion by $\omega_{\rm e}^{\rm E} \sim E_{\rm cut}$, where $E_{\rm cut}$ is the maximum kinetic energy included in the expansion. A realistic estimate for reasonable basis sets is $\tau_{\rm e}^{\rm E} \sim 10^{-16}$ s, which leads to $\tau_{\rm e}^{\rm E} \approx \tau_{\rm n}/100$. The analogues relation for Car–Parrinello dynamics reads however $\omega_{\rm e}^{\rm CP} \sim (E_{\rm cut}/\mu)^{1/2}$ according to the analysis in Sect. 2.4, see Eq. (54). Thus, in addition to reducing $\omega_{\rm e}^{\rm CP}$ by introducing a finite electron mass μ, the maximum electronic frequency increases much more slowly in Car–Parrinello than in Ehrenfest molecular dynamics with increasing basis set size. An estimate for the same basis set and a typical fictitious mass yields about $\tau_{\rm e}^{\rm CP} \sim 10^{-15}$ s or $\tau_{\rm e}^{\rm CP} \approx \tau_{\rm n}/10$. According to this simple estimate, the time step can be about one order of magnitude larger if Car–Parrinello second–order fictitious–time electron dynamics is used instead of Ehrenfest first–order real–time electron dynamics.

The time scale and thus time step problem inherent to Ehrenfest dynamics prompted some attempts to releave it. In Ref. [203] the equations of motion of electrons and nuclei were integrated using two different time steps, the one of the nuclei being 20–times as large as the electronic one. The powerful technology of multiple–time step integration theory [636,639] could also be applied in order to ameliorate the time scale disparity [585]. A different approach borrowed from plasma simulations consists in decreasing the nuclear masses so that their time evolution is artificially speeded up [617]. As a result, the *nuclear* dynamics is fictitious (in the presence of real–time electron dynamics!) and has to be rescaled to the proper mass ratio after the simulation.

In both Ehrenfest and Car–Parrinello schemes the explicitly treated electron dynamics limits the largest time step that can be used in order to integrate simultaneously the coupled equations of motion for nuclei and electrons. This limitation does of course not exist in Born–Oppenheimer dynamics since there is no explicit electron dynamics so that the maximum time step is simply given by the one intrinsic to nuclear motion, i.e. $\tau_{\rm e}^{\rm BO} \approx \tau_{\rm n}$. This is formally an order of magnitude advantage with respect to Car–Parrinello dynamics.

Do these back–of–the–envelope estimates have anything to do with reality? Fortunately, several state–of–the–art studies are reported in the literature for physically similar systems where all three molecular dynamics schemes have been employed. Ehrenfest simulations [553,203] of a dilute $K_x{\cdot}(KCl)_{1-x}$ melt were performed using a time step of 0.012–0.024 fs. In comparison, a time step as large as 0.4 fs could be used to produce a stable Car–Parrinello simulation of electrons in liquid ammonia [155,156]. Since the physics of these systems has a similar nature —

"unbound electrons" dissolved in liquid condensed matter (localizing as F–centers, polarons, bipolarons, etc.) — the time step difference of about a factor of ten confirms the crude estimate given above. In a Born–Oppenheimer simulation [569] of again $K_x{\cdot}(KCl)_{1-x}$ but up to a higher concentration of unbound electrons the time step used was 0.5 fs.

The time–scale advantage of Born–Oppenheimer vs. Car–Parrinello dynamics becomes more evident if the nuclear dynamics becomes fairly slow, such as in liquid sodium [343] or selenium [331] where a time step of 3 fs was used. This establishes the above–mentioned order of magnitude advantage of Born–Oppenheimer vs. Car–Parrinello dynamics in advantageous cases. However, it has to be taken into account that in simulations [331] with such a large time step dynamical information is limited to about 10 THz, which corresponds to frequencies below roughly 500 cm^{-1}. In order to resolve vibrations in molecular systems with stiff covalent bonds the time step has to be decreased to less than a femtosecond (see the estimate given above) also in Born–Oppenheimer dynamics.

The comparison of the overall performance of Car–Parrinello and Born–Oppenheimer molecular dynamics in terms of computer time is a delicate issue. For instance it depends crucially on the choice made concerning the accuracy of the conservation of the energy $E_{\rm cons}$ as defined in Eq. (48). Thus, this issue is to some extend subject of "personal taste" as to what is considered to be a "sufficiently accurate" energy conservation. In addition, this comparison might to different conclusions as a function of system size. In order to nevertheless shed light on this point, microcanonical simulations of 8 silicon atoms were performed with various parameters using Car–Parrinello and Born–Oppenheimer molecular dynamics as implemented in the `CPMD` package [142]. This large–gap system was initially extremely well equilibrated and the runs were extended to 8 ps (and a few to 12 ps with no noticeable difference) at a temperature of about 360–370 K (with ±80 K root–mean–square fluctuations). The wavefunction was expanded up to $E_{\rm cut}$ = 10 Ry at the Γ–point of a simple cubic supercell and LDA was used to describe the interactions. In both cases the velocity Verlet scheme was used to integrate the equations of motion, see Eqs. (231). It is noted in passing that also the velocity Verlet algorithm [638] allows for stable integration of the equations of motion contrary to the statements in Ref. [513] (see Sect. 3.4 and Figs. 4–5).

In Car–Parrinello molecular dynamics two different time steps were used, 5 a.u. and 10 a.u. (corresponding to about 0.24 fs), in conjunction with a fictitious electron mass of μ = 400 a.u.; this mass parameter is certainly not optimized and thus the time step could be increased furthermore. Also the largest time step lead to perfect adiabaticity (similar to the one documented in Fig. 3), i.e. $E_{\rm phys}$ Eq. (49) and $T_{\rm e}$ Eq. (51) did not show a systematic drift relative to the energy scale set by the variations of $V_{\rm e}$ Eq. (50). Within Born–Oppenheimer molecular dynamics the minimization of the energy functional was done using the highly efficient DIIS (direct inversion in the iterative subspace) scheme using 10 "history vectors", see Sect. 3.6. In this case, the time step was either 10 a.u. or 100 a.u. and three convergence criteria were used; note that the large time step corresponding to 2.4 fs is already at the limit to be used to investigate typical *molecular* systems (with frequencies up to 3–4000 cm^{-1}). The convergence criterion is based on the

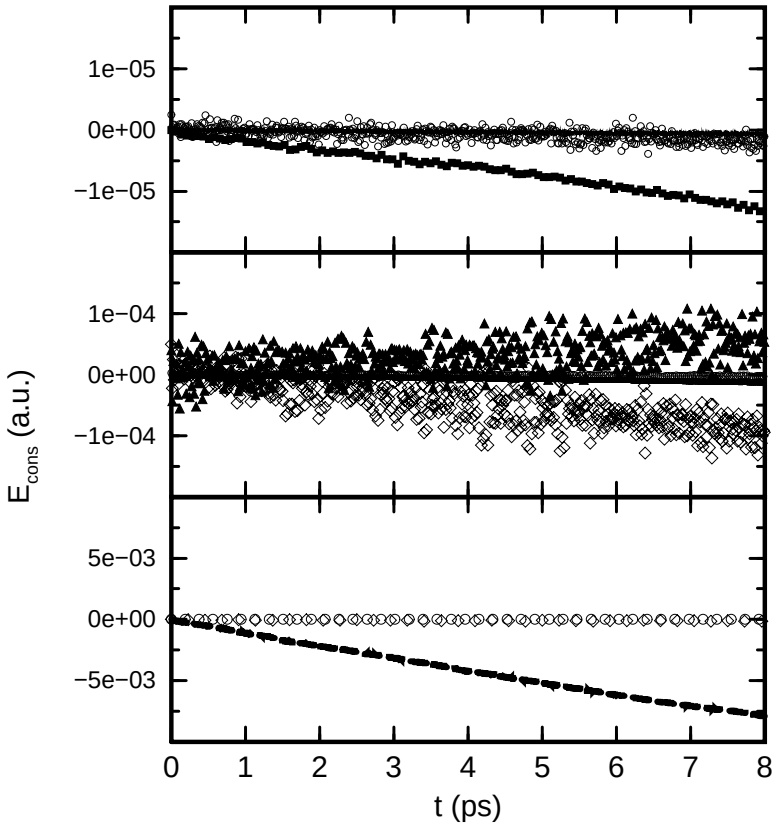


Figure 5. Conserved energy $E_{\rm cons}$ defined in Eq. (48) from Car–Parrinello (CP) and Born–Oppenheimer (BO) molecular dynamics simulations of a model system for various time steps and convergence criteria using the `CPMD` package [142]; see text for further details and Table 1 for the corresponding timings. Top: solid line: CP, 5 a.u.; open circles: CP, 10 a.u.; filled squares: BO, 10 a.u., 10^{-6}. Middle: open circles: CP, 10 a.u.; filled squares: BO, 10 a.u., 10^{-6}; filled triangles: BO, 100 a.u., 10^{-6}; open diamonds: BO, 100 a.u., 10^{-5}. Bottom: open circles: CP, 10 a.u.; open diamonds: BO, 100 a.u., 10^{-5}; dashed line: BO, 100 a.u., 10^{-4}.

largest element of the wavefunction gradient which was required to be smaller than 10^{-6}, 10^{-5} or 10^{-4} a.u.; note that the resulting energy convergence shows roughly a quadratic dependence on this criterion.

The outcome of this comparison is shown in Fig. 5 in terms of the time evolution of the conserved energy $E_{\rm cons}$ Eq. (48) on energy scales that cover more than three orders of magnitude in absolute accuracy. Within the present comparison ultimate energy stability was obtained using Car–Parrinello molecular dynamics with the shortest time step of 5 a.u., which conserves the energy of the total system to about $6{\times}10^{-8}$ a.u. per picosecond, see solid line in Fig. 5(top). Increasing the

Table 1. Timings in CPU seconds and energy conservation in a.u. / ps for Car–Parrinello (CP) and Born–Oppenheimer (BO) molecular dynamics simulations of a model system for 1 ps of trajectory on an IBM RS6000 / model 390 (Power2) workstation using the `CPMD` package [142]; see Fig. 5 for corresponding energy plots.

Method	Time step (a.u.)	Convergence (a.u.)	Conservation (a.u./ps)	Time (s)
CP	5	—	6×10^{-8}	3230
CP	7	—	1×10^{-7}	2310
CP	10	—	3×10^{-7}	1610
BO	10	10^{-6}	1×10^{-6}	16590
BO	50	10^{-6}	1×10^{-6}	4130
BO	100	10^{-6}	6×10^{-6}	2250
BO	100	10^{-5}	1×10^{-5}	1660
BO	100	10^{-4}	1×10^{-3}	1060

time step to 10 a.u. leads to an energy conservation of about 3×10^{-7} a.u./ps and much larger energy fluctuations, see open circles in Fig. 5(top). The computer time needed in order to generate one picosecond of Car–Parrinello trajectory increases – to a good approximation – linearly with the increasing time step, see Table 1. The most stable Born–Oppenheimer run was performed with a time step of 10 a.u. and a convergence of 10^{-6}. This leads to an energy conservation of about 1×10^{-6} a.u./ps, see filled squares in Fig. 5(top).

As the maximum time step in Born–Oppenheimer dynamics is only related to the time scale associated to nuclear motion it could be increased from 10 to 100 a.u. while keeping the convergence at the same tight limit of 10^{-6}. This worsens the energy conservation slightly (to about 6×10^{-6} a.u./ps), whereas the energy *fluctuations* increase dramatically, see filled triangles in Fig. 5(middle) and note the change of scale compared to Fig. 5(top). The overall gain is an acceleration of the Born–Oppenheimer simulation by a factor of about seven to eight, see Table 1. In the Born–Oppenheimer scheme, the computer time needed for a fixed amount of simulated physical time decreases only sublinearly with increasing time step since the initial guess for the iterative minimization degrades in quality as the time step is made larger. Further savings of computer time can be easily achieved by decreasing the quality of the wavefunction convergence from 10^{-6} to 10^{-5} and finally to 10^{-4}, see Table 1. This is unfortunately tied to a significant decrease of the energy conservation from 6×10^{-6} a.u./ps at 10^{-6} (filled triangles) to about 1×10^{-3} a.u./ps at 10^{-4} (dashed line) using the same 100 a.u. time step, see Fig. 5(bottom) but note the change of scale compared to Fig. 5(middle).

In conclusion, Born–Oppenheimer molecular dynamics can be made as fast as (or even faster than) Car–Parrinello molecular dynamics (as measured by the amount of CPU time spent per picosecond) at the expense of sacrificing accuracy in terms of energy conservation. In the "classical molecular dynamics community" there is a general consensus that this conservation law should be taken seriously being a measure of the numerical quality of the simulation. In the "quantum chemistry and total energy communities" this issue is typically of less concern. There, it is rather the quality of the convergence of the wavefunction or energy (as achieved in every individual molecular dynamics step) that is believed to be crucial in order to gauge the quality of a particular simulation.

Finally, it is worth commenting in this particular section on a paper entitled "A comparison of Car–Parrinello and Born–Oppenheimer generalized valence bond molecular dynamics" [229]. In this paper one (computationally expensive) term in the nuclear equations of motion is neglected [648,405]. It is well known that using a basis set with origin, such as Gaussians $f_\nu^{\mathrm{G}}(\mathbf{r};\{\mathbf{R}_I\})$ centered at the nuclei, see Eq. (99), produces various Pulay forces, see Sect. 2.5. In particular a linear expansion Eq. (65) or (97) based on such orbitals introduces a position dependence into the orthogonality constraint

$$\langle\psi_i|\psi_j\rangle = \sum_{\nu\mu} c_{i\nu}^\star c_{j\mu} \underbrace{\left\langle f_\nu^{\mathrm{G}} \middle| f_\mu^{\mathrm{G}} \right\rangle}_{S_{\nu\mu}} = \delta_{ij} \tag{73}$$

that is hidden in the overlap matrix $S_{\nu\mu}(\{\mathbf{R}_I\})$ which involves the basis functions. According to Eq. (44) this term produces a constraint force of the type

$$\sum_{ij} \Lambda_{ij} \sum_{\nu\mu} c_{i\nu}^\star c_{j\mu} \frac{\partial}{\partial \mathbf{R}_I} S_{\nu\mu}(\{\mathbf{R}_I\}) \tag{74}$$

in the correct Car–Parrinello equation of motion *for the nuclei* similar to the one contained in the electronic equation of motion Eq. (45). This term has to be included in order to yield exact Car–Parrinello trajectories and thus energy conservation, see e.g. Eq. (37) in Ref. [351] for a similar situation. In the case of Born–Oppenheimer molecular dynamics, on the contrary, this term is always absent in the nuclear equation of motion, see Eq. (32). Thus, the particular implementation [229] underlying the comparison between Car–Parrinello and Born–Oppenheimer molecular dynamics is an approximate one from the outset concerning the Car–Parrinello part; it can be argued that this was justified in the early papers [281,282] where the basic feasibility of both the Hartree Fock– and generalized valence bond–based Car–Parrinello molecular dynamics techniques was demonstrated [285]. Most importantly, this approximation implies that the energy E_{cons} Eq. (48) *cannot be rigorously conserved* in this particular version of Car–Parrinello molecular dynamics. However, energy conservation of E_{cons} was used in Ref. [229] to compare the efficiency and accuracy of these two approaches to GVB *ab initio* molecular dynamics (using DIIS for the Born–Oppenheimer simulations as done in the above–given comparison). Thus, the final conclusion that for "... approaches that utilize non–space–fixed bases to describe the electronic wave function, Born–Oppenheimer AIMD is the method of choice, both in terms of accuracy and speed" [229] cannot be drawn from this specific comparison for the reasons outlined above (independently of the particular basis set or electronic structure method used).

The toy system investigated here (see Fig. 5 and Table 1), i.e. 8 silicon atoms in a periodic supercell, is for the purpose of comparing different approaches to *ab initio* molecular dynamics quite similar to the system used in Ref. [229], i.e. clusters of 4 or 6 sodium atoms (in addition, qualitatively identical results where reported in Sect. 4 for silicon clusters). Thus, it is admissible to compare the energy conservations reported in Figs. 1 and 2 of Ref. [229] to the ones depicted here in Fig. 5 noting that the longest simulations reported in Ref. [229] reached only 1 ps. It should be

stressed that the energy conservation seen in Fig. 5(top) is routinely achieved in Car–Parrinello molecular dynamics simulations.

2.7 Electronic Structure Methods

2.7.1 Introduction

Up to this point, the electronic structure method to calculate the *ab initio* forces $\nabla_I \langle \Psi | \mathcal{H}_\mathrm{e} | \Psi \rangle$ was not specified in detail. It is immediately clear that *ab initio* molecular dynamics is not tied to any particular approach, although very accurate techniques are of course prohibitively expensive. It is also evident that the strength or weakness of a particular *ab initio* molecular dynamics scheme is intimately connected to the strength or weakness of the chosen electronic structure method. Over the years a variety of different approaches such as density functional [108,679,35,472,343,36], Hartree–Fock [365,254,191,379,281,284,316,293], generalized valence bond (GVB) [282,283,228,229,230], complete active space SCF (CASSCF) [566,567], full configuration interaction (FCI) [372], semiempirical [669,671,91,190,114,666,280] or other approximate [473,454,551,455,170,171,26] methods were combined with molecular dynamics, and this list is certainly incomplete.

The focus of the present review clearly is Car–Parrinello molecular dynamics in conjunction with Hohenberg–Kohn–Sham density functional theory [301,338]. In the following, only those parts of density functional theory are presented that impact directly on *ab initio* molecular dynamics. For a deeper presentation and in particular for a discussion of the assumptions and limitations of this approach (both conceptually and in practice) the reader is referred to the existing excellent literature [591,320,458,168]. For simplicity, the formulae are presented for the spin–unpolarized or restricted special case.

Following the exposition of density functional theory, the fundamentals of Hartree–Fock theory, which is often considered to be the basis of quantum chemistry, are introduced for the same special case. Finally, a glimpse is given at post Hartree–Fock methods. Again, an extensive text–book literature exists for these wavefunction–based approaches to electronic structure calculations [604,418]. The very useful connection between the density–based and wavefunction–based methods goes back to Löwdin's work in the mid fifties and is e.g. worked out in Chapt. 2.5 of Ref. [458], where Hartree–Fock theory is formulated in density–matrix language.

2.7.2 Density Functional Theory

The total ground–state energy of the interacting system of electrons with classical nuclei fixed at positions $\{\mathbf{R}_I\}$ can be obtained

$$\min_{\Psi_0} \{ \langle \Psi_0 | \mathcal{H}_\mathrm{e} | \Psi_0 \rangle \} = \min_{\{\phi_i\}} E^\mathrm{KS}[\{\phi_i\}]$$

as the minimum of the Kohn–Sham energy [301,338]

$$E^\mathrm{KS}[\{\phi_i\}] = T_\mathrm{s}[\{\phi_i\}] + \int d\mathbf{r} \;\; V_\mathrm{ext}(\mathbf{r}) \; n(\mathbf{r}) + \frac{1}{2} \int d\mathbf{r} \;\; V_\mathrm{H}(\mathbf{r}) \; n(\mathbf{r}) + E_\mathrm{xc}[n] \;\; , \quad (75)$$

which is an explicit functional of the set of auxiliary functions $\{\phi_i(\mathbf{r})\}$ that satisfy the orthonormality relation $\langle \phi_i \mid \phi_j \rangle = \delta_{ij}$. This is a dramatic simplification since the minimization with respect to all possible *many–body* wavefunctions $\{\Psi\}$ is replaced by a minimization with respect to a set of orthonormal one–particle functions, the Kohn–Sham orbitals $\{\phi_i\}$. The associated electronic one–body density or charge density

$$n(\mathbf{r}) = \sum_{i}^{\text{occ}} f_i \mid \phi_i(\mathbf{r}) \mid^2 \tag{76}$$

is obtained from a single Slater determinant built from the occupied orbitals, where $\{f_i\}$ are integer occupation numbers.

The first term in the Kohn–Sham functional Eq. (75) is the kinetic energy of a non–interacting reference system

$$T_{\text{s}}[\{\phi_i\}] = \sum_{i}^{\text{occ}} f_i \left\langle \phi_i \left| -\frac{1}{2}\nabla^2 \right| \phi_i \right\rangle \tag{77}$$

consisting of the same number of electrons exposed to the same external potential as in the fully interacting system. The second term comes from the fixed external potential

$$V_{\text{ext}}(\mathbf{r}) = -\sum_{I} \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}|} + \sum_{I<J} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} \tag{78}$$

in which the electrons move, which comprises the Coulomb interactions between electrons and nuclei and in the definition used here also the internuclear Coulomb interactions; this term changes in the first place if core electrons are replaced by pseudopotentials, see Sect. 3.1.5 for further details. The third term is the Hartree energy, i.e. the classical electrostatic energy of two charge clouds which stem from the electronic density and is obtained from the Hartree potential

$$V_{\text{H}}(\mathbf{r}) = \int d\mathbf{r}' \ \frac{n(\mathbf{r}')}{\mid \mathbf{r} - \mathbf{r}' \mid} \ , \tag{79}$$

which in turn is related to the density via

$$\nabla^2 V_{\text{H}}(\mathbf{r}) = -4\pi n(\mathbf{r}) \tag{80}$$

Poisson's equation. The last contribution in the Kohn–Sham functional, the exchange–correlation functional $E_{\text{xc}}[n]$, is the most intricate contribution to the total electronic energy. The electronic exchange and correlation effects are lumped together and basically define this functional as the remainder between the exact energy and its Kohn–Sham decomposition in terms of the three previous contributions.

The minimum of the Kohn–Sham functional is obtained by varying the energy functional Eq. (75) for a fixed number of electrons with respect to the density Eq. (76) or with respect to the orbitals subject to the orthonormality constraint,

see e.g. the discussion following Eq. (35) for a similar variational procedure. This leads to the Kohn–Sham equations

$$\left\{ -\frac{1}{2}\nabla^2 + V_{\rm ext}(\mathbf{r}) + V_{\rm H}(\mathbf{r}) + \frac{\delta E_{\rm xc}[n]}{\delta n(\mathbf{r})} \right\} \phi_i(\mathbf{r}) = \sum_j \Lambda_{ij} \phi_j(\mathbf{r}) \tag{81}$$

$$\left\{ -\frac{1}{2}\nabla^2 + V^{\rm KS}(\mathbf{r}) \right\} \phi_i(\mathbf{r}) = \sum_j \Lambda_{ij} \phi_j(\mathbf{r}) \tag{82}$$

$$H_{\rm e}^{\rm KS} \phi_i(\mathbf{r}) = \sum_j \Lambda_{ij} \phi_j(\mathbf{r}) \ , \tag{83}$$

which are one–electron equations involving an effective *one–particle* Hamiltonian $H_{\rm e}^{\rm KS}$ with the local potential $V^{\rm KS}$. Note that $H_{\rm e}^{\rm KS}$ nevertheless embodies the electronic *many–body* effects by virtue of the exchange–correlation potential

$$\frac{\delta E_{\rm xc}[n]}{\delta n(\mathbf{r})} = V_{\rm xc}(\mathbf{r}) \ . \tag{84}$$

A unitary transformation within the space of the occupied orbitals leads to the canonical form

$$H_{\rm e}^{\rm KS} \phi_i = \epsilon_i \phi_i \tag{85}$$

of the Kohn–Sham equations, where $\{\epsilon_i\}$ are the eigenvalues. In conventional static density functional or "band structure" calculations this set of equations has to be solved self–consistently in order to yield the density, the orbitals and the Kohn–Sham potential for the electronic ground state [487]. The corresponding total energy Eq. (75) can be written as

$$E^{\rm KS} = \sum_i \epsilon_i - \frac{1}{2} \int d\mathbf{r} \ V_{\rm H}(\mathbf{r}) \ n(\mathbf{r}) + E_{\rm xc}[n] - \int d\mathbf{r} \ \frac{\delta E_{\rm xc}[n]}{\delta n(\mathbf{r})} \ n(\mathbf{r}) \ , \tag{86}$$

where the sum over Kohn–Sham eigenvalues is the so–called "band–structure energy".

Thus, Eqs. (81)–(83) together with Eqs. (39)–(40) define Born–Oppenheimer molecular dynamics within Kohn–Sham density functional theory, see e.g. Refs. [232,616,594,35,679,472,36,343,344] for such implementations. The functional derivative of the Kohn–Sham functional with respect to the orbitals, the Kohn–Sham force acting on the orbitals, can be expressed as

$$\frac{\delta E^{\rm KS}}{\delta \phi_i^\star} = f_i H_{\rm e}^{\rm KS} \phi_i \ , \tag{87}$$

which makes clear the connection to Car–Parrinello molecular dynamics, see Eq. (45). Thus, Eqs. (59)–(60) have to be solved with the effective one–particle Hamiltonian in the Kohn–Sham formulation Eqs. (81)–(83). In the case of Ehrenfest dynamics presented in Sect. 2.2, which will not be discussed in further detail at this stage, the Runge–Gross time–dependent generalization of density functional theory [258] has to be invoked instead, see e.g. Refs. [203,617,532].

Crucial to any application of density functional theory is the approximation of the unknown exchange and correlation functional. A discussion focussed on the

utilization of such functionals in the framework of *ab initio* molecular dynamics is for instance given in Ref. [588]. Those exchange–correlation functionals that will be considered in the implementation part, Sect. 3.3, belong to the class of the "Generalized Gradient Approximation"

$$E_{\rm xc}^{\rm GGA}[n] = \int d\mathbf{r}\ \ n(\mathbf{r})\ \ \varepsilon_{\rm xc}^{\rm GGA}(n(\mathbf{r}); \nabla n(\mathbf{r}))\ \ , \tag{88}$$

where the unknown functional is approximated by an integral over a function that depends only on the density and its gradient at a given point in space, see Ref. [477] and references therein. The combined exchange–correlation function is typically split up into two additive terms $\varepsilon_{\rm x}$ and $\varepsilon_{\rm c}$ for exchange and correlation, respectively. In the simplest case it is the exchange and correlation energy density $\varepsilon_{\rm xc}^{\rm LDA}(n)$ of an interacting but homogeneous electron gas at the density given by the "local" density $n(\mathbf{r})$ at space–point $\mathbf{r}$ in the inhomogeneous system. This simple but astonishingly powerful approximation [320] is the famous local density approximation LDA [338] (or local spin density LSD in the spin–polarized case [40]), and a host of different parameterizations exist in the literature [458,168]. The self–interaction correction [475] SIC as applied to LDA was critically assessed for molecules in Ref. [240] with a disappointing outcome.

A significant improvement of the accuracy was achieved by introducing the gradient of the density as indicated in Eq. (88) beyond the well–known straightforward gradient expansions. These so–called GGAs (also denoted as "gradient corrected" or "semilocal" functionals) extended the applicability of density functional calculation to the realm of chemistry, see e.g. Refs. [476,42,362,477,478,479] for a few "popular functionals" and Refs. [318,176,577,322] for extensive tests on molecules, complexes, and solids, respectively.

Another considerable advance was the successful introduction of "hybrid functionals" [43,44] that include to some extent "exact exchange" [249] in addition to a standard GGA. Although such functionals can certainly be implemented within a plane wave approach [262,128], they are prohibitively time–consuming as outlined at the end of Sect. 3.3. A more promising route in this respect are those functionals that include higher–order powers of the gradient (or the local kinetic energy density) in the sense of a generalized gradient expansion beyond the first term. Promising results could be achieved by including Laplacian or local kinetic energy terms [493,192,194,662], but at this stage a sound judgment concerning their "prize / performance ratio" has to await further scrutinizing tests. The "optimized potential method" (OPM) or "optimized effective potentials" (OEP) are another route to include "exact exchange" within density functional theory, see e.g. Sect. 13.6 in Ref. [588] or Ref. [251] for overviews. Here, the exchange–correlation functional $E_{\rm xc}^{\rm OPM} = E_{\rm xc}[\{\phi_i\}]$ depends on the individual orbitals instead of only on the density or its derivatives.

2.7.3 Hartree–Fock Theory

Hartree–Fock theory is derived by invoking the variational principle in a restricted space of wavefunctions. The antisymmetric ground–state electronic wavefunction is approximated by a single Slater determinant $\Psi_0 = \det\{\psi_i\}$ which is constructed

from a set of one–particle spin orbitals $\{\psi_i\}$ required to be mutually orthonormal $\langle \psi_i \, | \psi_j \, \rangle = \delta_{ij}$. The corresponding variational minimum of the total electronic energy $\mathcal{H}_\mathrm{e}$ defined in Eq. (2)

$$\begin{aligned} E^\mathrm{HF}[\{\psi_i\}] = \sum_i \int d\mathbf{r} \; \psi_i^\star(\mathbf{r}) \left[-\frac{1}{2}\nabla^2 + V_\mathrm{ext}(\mathbf{r}) \right] \psi_i(\mathbf{r}) \\ + \frac{1}{2} \sum_{ij} \int \int d\mathbf{r} \; d\mathbf{r}' \; \psi_i^\star(\mathbf{r}) \psi_j^\star(\mathbf{r}') \, \frac{1}{|\mathbf{r} - \mathbf{r}'|} \, \psi_i(\mathbf{r}) \psi_j(\mathbf{r}') \\ + \frac{1}{2} \sum_{ij} \int \int d\mathbf{r} \; d\mathbf{r}' \; \psi_i^\star(\mathbf{r}) \psi_j^\star(\mathbf{r}') \, \frac{1}{|\mathbf{r} - \mathbf{r}'|} \, \psi_j(\mathbf{r}) \psi_i(\mathbf{r}') \end{aligned} \tag{89}$$

yields the lowest energy and the "best" wavefunction within a one–determinant ansatz; the external Coulomb potential V_ext was already defined in Eq. (78). Carrying out the constraint minimization within this ansatz (see Eq. (36) in Sect. 2.3 for a sketch) leads to

$$\left\{ -\frac{1}{2}\nabla^2 + V_\mathrm{ext}(\mathbf{r}) + \sum_j \mathcal{J}_j(\mathbf{r}) - \sum_j \mathcal{K}_j(\mathbf{r}) \right\} \psi_i(\mathbf{r}) = \sum_j \Lambda_{ij} \psi_j(\mathbf{r}) \tag{90}$$

$$\left\{ -\frac{1}{2}\nabla^2 + V^\mathrm{HF}(\mathbf{r}) \right\} \psi_i(\mathbf{r}) = \sum_j \Lambda_{ij} \psi_j(\mathbf{r}) \tag{91}$$

$$H_\mathrm{e}^\mathrm{HF} \psi_i(\mathbf{r}) = \sum_j \Lambda_{ij} \psi_j(\mathbf{r}) \tag{92}$$

the Hartree–Fock integro–differential equations. In analogy to the Kohn–Sham equations Eqs. (81)–(83) these are effective one–particle equations that involve an effective one–particle Hamiltonian H_e^HF, the (Hartree–) Fock operator. The set of canonical orbitals

$$H_\mathrm{e}^\mathrm{HF} \psi_i = \epsilon_i \psi_i \tag{93}$$

is obtained similarly to Eq. (85). The Coulomb operator

$$\mathcal{J}_j(\mathbf{r}) \; \psi_i(\mathbf{r}) = \left[\int d\mathbf{r}' \; \psi_j^\star(\mathbf{r}') \frac{1}{|\mathbf{r} - \mathbf{r}'|} \psi_j(\mathbf{r}') \right] \; \psi_i(\mathbf{r}) \tag{94}$$

and the exchange operator

$$\mathcal{K}_j(\mathbf{r}) \; \psi_i(\mathbf{r}) = \left[\int d\mathbf{r}' \; \psi_j^\star(\mathbf{r}') \frac{1}{|\mathbf{r} - \mathbf{r}'|} \psi_i(\mathbf{r}') \right] \; \psi_j(\mathbf{r}) \tag{95}$$

are most easily defined via their action on a particular orbital ψ_i. It is found that upon acting on orbital $\psi_i(\mathbf{r})$ the exchange operator for the j–th state "exchanges" $\psi_j(\mathbf{r}') \rightarrow \psi_i(\mathbf{r}')$ in the kernel as well as replaces $\psi_i(\mathbf{r}) \rightarrow \psi_j(\mathbf{r})$ in its argument, compare to the Coulomb operator. Thus, $\mathcal{K}$ is a non–local operator as its action on a function ψ_i at point $\mathbf{r}$ in space requires the evaluation and thus the knowledge of that function throughout all space by virtue of $\int d\mathbf{r}' \; \psi_i(\mathbf{r}') \ldots$ the required integration. In this sense the exchange operator does not possess a simple classical interpretation like the Coulomb operator $\mathcal{C}$, which is the counterpart of

the Hartree potential $V_{\rm H}$ in Kohn–Sham theory. The exchange operator vanishes exactly if the antisymmetrization requirement of the wavefunction is relaxed, i.e. only the Coulomb contribution survives if a Hartree product is used to represent the wavefunction.

The force acting on the orbitals is defined

$$\frac{\delta E^{\rm HF}}{\delta \psi_i^\star} = H_{\rm e}^{\rm HF} \psi_i \tag{96}$$

similarly to Eq. (87). At this stage, the various *ab initio* molecular dynamics schemes based on Hartree–Fock theory are defined, see Eqs. (39)–(40) for Born–Oppenheimer molecular dynamics and Eqs. (59)–(60) for Car–Parrinello molecular dynamics. In the case of Ehrenfest molecular dynamics the time–dependent Hartree–Fock formalism [162] has to be invoked instead.

2.7.4 Post Hartree–Fock Theories

Although post Hartree–Fock methods have a very unfavorable scaling of the computational cost as the number of electrons increases, a few case studies were performed with such correlated quantum chemistry techniques. For instance *ab initio* molecular dynamics was combined with GVB [282,283,228,229,230], CASSCF [566,567], as well as FCI [372] approaches, see also references therein. It is noted in passing that Car–Parrinello molecular dynamics can only be implemented straightforwardly if energy and wavefunction are "consistent". This is not the case in perturbation theories such as e.g. the widely used Møller–Plesset approach [292]: within standard MP2 the energy is correct to second order, whereas the wavefunction is the one given by the uncorrelated HF reference. As a result, the derivative of the MP2 energy with respect to the wavefunction Eq. (96) does not yield the correct force on the HF wavefunction in the sense of fictitious dynamics. Such problems are of course absent from the Born–Oppenheimer approach to sample configuration space, see e.g. Ref. [328,317,33] for MP2, density functional, and multireference CI *ab initio* Monte Carlo schemes.

It should be kept in mind that the rapidly growing workload of post HF calculations, although extremely powerful in principle, limits the number of explicitely treated electrons to only a few. The rapid development of correlated electronic structure methods that scale linearly with the number of electrons will certainly broaden the range of applicability of this class of techniques in the near future.

2.8 Basis Sets

2.8.1 Gaussians and Slater Functions

Having selected a specific electronic structure method the next choice is related to which basis set to use in order to represent the orbitals ψ_i in terms of simple analytic functions f_ν with well–known properties. In general a *linear* combination of such basis functions

$$\psi_i(\mathbf{r}) = \sum_\nu c_{i\nu} f_\nu(\mathbf{r}; \{\mathbf{R}_I\}) \tag{97}$$

is used, which represents exactly any reasonable function in the limit of using a complete set of basis functions. In quantum chemistry, Slater–type basis functions (STOs)

$$f^{\mathrm{S}}_{\mathbf{m}}(\mathbf{r}) = N^{\mathrm{S}}_{\mathbf{m}}\, r_x^{m_x} r_y^{m_y} r_z^{m_z}\ \exp\left[-\zeta_{\mathbf{m}}|\mathbf{r}|\right] \tag{98}$$

with an exponentially decaying radial part and Gaussian–type basis functions (GTOs)

$$f^{\mathrm{G}}_{\mathbf{m}}(\mathbf{r}) = N^{\mathrm{G}}_{\mathbf{m}}\, r_x^{m_x} r_y^{m_y} r_z^{m_z}\ \exp\left[-\alpha_{\mathbf{m}} r^2\right] \tag{99}$$

have received widespread use, see e.g. Ref. [292] for a concise overview–type presentation. Here, $N_{\mathbf{m}}$, $\zeta_{\mathbf{m}}$ and $\alpha_{\mathbf{m}}$ are constants that are typically kept fixed during a molecular electronic structure calculation so that only the orbital expansion coefficients $c_{i\nu}$ need to be optimized. In addition, fixed linear combinations of the above–given "primitive" basis functions can be used for a given angular momentum channel $\mathbf{m}$, which defines the "contracted" basis sets.

The Slater or Gaussian basis functions are in general centered at the positions of the nuclei, i.e. $\mathbf{r} \to \mathbf{r} - \mathbf{R}_I$ in Eq. (98)–(99), which leads to the linear combination of *atomic orbitals* (LCAO) ansatz to solve differential equations algebraically. Furthermore, their derivatives as well as the resulting matrix elements are efficiently obtained by differentiation and integration in real–space. However, Pulay forces (see Sect. 2.5) will result for such basis functions that are fixed at atoms (or bonds) if the atoms are allowed to move, either in geometry optimization or molecular dynamics schemes. This disadvantage can be circumvented by using *freely* floating Gaussians that are distributed in space [582], which form an originless basis set since it is localized but not atom–fixed.

2.8.2 Plane Waves

A vastly different approach has its roots in solid–state theory. Here, the ubiquitous periodicity of the underlying lattice produces a periodic potential and thus imposes the same periodicity on the density (implying Bloch's Theorem, Born–von Karman periodic boundary conditions etc., see e.g. Chapt. 8 in Ref. [27]). This heavily suggests to use plane waves as the generic basis set in order to expand the periodic part of the orbitals, see Sect. 3.1.2. Plane waves are defined as

$$f^{\mathrm{PW}}_{\mathbf{G}}(\mathbf{r}) = N \exp\left[i\mathbf{G}\mathbf{r}\right] \quad , \tag{100}$$

where the normalization is simply given by $N = 1/\sqrt{\Omega}$; Ω is the volume of the periodic (super–) cell. Since plane waves form a complete and orthonormal set of functions they can be used to expand orbitals according to Eq. (97), where the labeling ν is simply given by the vector $\mathbf{G}$ in reciprocal space / G–space (including only those $\mathbf{G}$–vectors that satisfy the particular periodic boundary conditions). The total electronic energy is found to have a particularly simple form when expressed in plane waves [312].

It is important to observe that plane waves are originless functions, i.e. they do *not* depend on the positions of the nuclei $\{\mathbf{R}_I\}$. This implies that the Pulay forces Eq. (67) vanish exactly even within a *finite* basis (and using a fixed number of plane waves, see the discussion related to "Pulay stress" in Sect. 2.5), which

tremendously facilitates force calculations. This also implies that plane waves are a very unbiased basis set in that they are "delocalized" in space and do not "favor" certain atoms or regions over others, i.e. they can be considered as an ultimately "balanced basis set" in the language of quantum chemistry. Thus, the only way to improve the quality of the basis is to increase the "energy cutoff" $E_{\rm cut}$, i.e. to increase the largest $|\mathbf{G}|$–vector that is included in the finite expansion Eq. (97). This blind approach is vastly different from the traditional procedures in quantum chemistry that are needed in order to produce reliable basis sets [292]. Another appealing feature is that derivatives in real–space are simply multiplications in G–space, and both spaces can be efficiently connected via Fast Fourier Transforms (FFTs). Thus, one can easily evaluate operators in that space in which they are diagonal, see for instance the flow charts in Fig. 6 or Fig. 7.

According to the well–known "No Free Lunch Theorem" there cannot be only advantages connected to using plane waves. The first point is that the pseudopotential approximation is intimately connected to using plane waves, why so? A plane wave basis is basically a lattice–symmetry–adapted three–dimensional Fourier decomposition of the orbitals. This means that increasingly large Fourier components are needed in order to resolve structures in real space on decreasingly small distance scales. But already orbitals of first row atoms feature quite strong and rapid oscillations close to the nuclei due to the Pauli principle, which enforces a nodal structure onto the wavefunction by imposing orthogonality of the orbitals. However, most of chemistry is ruled by the valence electrons, whereas the core electrons are essentially inert. In practice, this means that the innermost electrons can be taken out of explicit calculations. Instead they are represented by a smooth and nodeless effective potential, the so–called pseudopotential [296,297,484,485,139], see for instance Refs. [487,578,221] for reviews in the context of "solid state theory" and Refs. [145,166] for pseudopotentials as used in "quantum chemistry". The resulting pseudo wavefunction is made as smooth as possible close to the nuclear core region. This also means that properties that depend crucially on the wavefunction close to the core cannot be obtained straightforwardly from such calculations. In the field of plane wave calculations the introduction of "soft" norm–conserving *ab initio* pseudopotentials was a breakthrough both conceptually [274] and in practice [28]. Another important contribution, especially for transition metals, was the introduction of the so–called ultrasoft pseudopotentials by Vanderbilt [661]. This approaches lead to the powerful technique of plane wave–pseudopotential electronic structure calculations in the framework of density functional theory [312,487]. Within this particular framework the issue of pseudopotentials is elaborated in more detail in Sect. 3.1.5.

Another severe shortcoming of plane waves is the backside of the medal of being an unbiased basis set: there is no way to shuffle more basis functions into regions in space where they are more needed than in other regions. This is particularly bad for systems with strong inhomogeneities. Such examples are all–electron calculations or the inclusion of semi–core states, a few heavy atoms in a sea of light atoms, and (semi–) finite systems such as surfaces or molecules with a large vacuum region in order to allow the long–range Coulomb interactions to decay. This is often referred to as the multiple length scale deficiency of plane wave calculations.

2.8.3 Generalized Plane Waves

An extremely appealing and elegant generalization of the plane wave concept [263,264] consists in defining them in curved ξ–space

$$f_{\mathbf{G}}^{\mathrm{GPW}}(\xi) = N \det{}^{1/2} J \exp\left[i\mathbf{G}\,\mathbf{r}(\xi)\right] \tag{101}$$
$$\det J = \left|\frac{\partial r^i}{\partial \xi^j}\right| \ ,$$

where $\det J$ is the Jacobian of the transformation from Cartesian to curvilinear coordinates $\mathbf{r} \rightarrow \xi(\mathbf{r})$ with $\xi = (\xi^1, \xi^2, \xi^3)$ and $N = 1/\sqrt{\Omega}$ as for regular plane waves. These functions are orthonormal, form a complete basis set, can be used for $\mathbf{k}$–point sampling after replacing $\mathbf{G}$ by $\mathbf{G} + \mathbf{k}$ in Eq. (101), are originless (but nevertheless localized) so that Pulay forces are absent, can be manipulated via efficient FFT techniques, and reduce to standard plane waves in the special case of an Euclidean space $\xi(\mathbf{r}) = \mathbf{r}$. Thus, they can be used equally well like plane waves in linear expansions of the sort Eq. (65) underlying most of electronic structure calculations. The Jacobian of the transformation is related to the Riemannian metric tensor

$$g_{ij} = \sum_{k=1}^{3} \frac{\partial \xi^k}{\partial r^i} \frac{\partial \xi^k}{\partial r^j}$$
$$\det J = \det{}^{-1/2} \{g_{ij}\} \tag{102}$$

which defines the metric of the ξ–space. The metric and thus the curvilinear coordinate system itself is considered as a variational parameter in the original fully adaptive–coordinate approach [263,264], see also Refs. [159,275,276,277,278]. Thus, a uniform grid in curved Riemannian space is non–uniform or distorted when viewed in flat Euclidean space (where $g_{ij} = \delta_{ij}$) such that the density of grid points (or the "local" cutoff energy of the expansion in terms of G–vectors) is highest in regions close to the nuclei and lowest in vacuum regions, see Fig. 2 in Ref. [275].

Concerning actual calculations, this means that a lower number of generalized plane waves than standard plane waves are needed in order to achieve a given accuracy [263], see Fig. 1 in Ref. [275]. This allows even for all–electron approaches to electronic structure calculations where plane waves fail [431,497]. More recently, the distortion of the metric was frozen spherically around atoms by introducing deformation functions [265,266], which leads to a concept closely connected to non–uniform atom–centered meshes in real–space methods [431], see below. In such non–fully–adaptive approaches using *predefined* coordinate transformations attention has to be given to Pulay force contributions which have to be evaluated explicitely [265,431].

2.8.4 Wavelets

Similar to using generalized plane waves is the idea to exploit the powerful multiscale–properties of wavelets. Since this approach requires an extensive introductory discussion (see e.g. Ref. [242] for a gentle introduction) and since it seems still quite far from being used in large–scale electronic structure calculations the interested reader is referred to original papers [134,674,699,652,241,25] and the general

wavelet literature cited therein. Wavelet–based methods allow intrinsically to exploit multiple length scales without introducing Pulay forces and can be efficiently handled by fast wavelet transforms. In addition, they are also a powerful route to linear scaling or "order–N" methods [453,243] as first demonstrated in Ref. [241] with the calculation of the Hartree potential for an all–electron uranium dimer.

2.8.5 Mixed and Augmented Basis Sets

Localized Gaussian basis functions on the one hand and plane waves on the other hand are certainly two extreme cases. There has been a tremendous effort to combine such localized and originless basis functions in order to exploit their mutual strengths. This resulted in a rich collection of mixed and augmented basis sets with very specific implementation requirements. This topic will not be covered here and the interested reader is referred to Refs. [75,654,498,370,371] and references given therein for some recent implementations used in conjunction with *ab initio* molecular dynamics.

2.8.6 Wannier Functions

An alternative to the plane wave basis set in the framework of periodic calculations in solid–state theory are Wannier functions, see for instance Sect. 10 in Ref. [27]. These functions are formally obtained from a unitary transformation of the Bloch orbitals Eq. (114) and have the advantage that they can be exponentially localized under certain circumstances. The so–called maximally localized generalized Wannier functions [413] are the periodic analogues of Boys' localized orbitals defined for isolated systems. Recently the usefulness of Wannier functions for numerical purposes was advocated by several groups, see Refs. [339,184,413,10] and references given therein.

2.8.7 Real Space Grids

A quite different approach is to leave conventional basis set approaches altogether and to resort to real–space methods where continuous space is replaced by a discrete space $\mathbf{r} \rightarrow r_{\mathbf{p}}$. This entails that the derivative operator or the entire energy expression has to be discretized in some way. The high–order central–finite difference approach leads to the expression

$$\begin{aligned} -\frac{1}{2}\nabla^2\psi_i(\mathbf{r}) \stackrel{h\rightarrow 0}{=} -\frac{1}{2}\Bigg[& \sum_{n_x=-N}^{N} C_{n_x}\ \psi_i(r_{p_x}+n_x h, r_{p_y}, r_{p_z}) \\ & + \sum_{n_y=-N}^{N} C_{n_y}\ \psi_i(r_{p_x}, r_{p_y}+n_y h, r_{p_z}) \\ & + \sum_{n_z=-N}^{N} C_{n_z}\ \psi_i(r_{p_x}, r_{p_y}, r_{p_z}+n_z h)\Bigg] + \mathcal{O}\left(h^{2N+2}\right) \end{aligned} \quad (103)$$

for the Laplacian which is correct up to the order h^{2N+2}. Here, h is the uniform grid spacing and $\{C_{\mathbf{n}}\}$ are known expansion coefficients that depend on the selected order [130]. Within this scheme, not only the grid spacing h but also the order are

disposable parameters that can be optimized for a particular calculation. Note that the discretization points in continuous space can also be considered to constitute a sort of "finite basis set" – despite different statements in the literature – and that the "infinite basis set limit" is reached as $h \to 0$ for N fixed. A variation on the theme are Mehrstellen schemes where the discretization of the entire differential equation and not only of the derivative operator is optimized [89].

The first real–space approach devised for *ab initio* molecular dynamics was based on the lowest–order finite–difference approximation in conjunction with a equally–spaced cubic mesh in real space [109]. A variety of other implementations of more sophisticated real–space methods followed and include e.g. non–uniform meshes, multigrid acceleration, different discretization techniques, and finite–element methods [686,61,39,130,131,632,633,431,634]. Among the chief advantages of the real–space methods is that linear scaling approaches [453,243] can be implemented in a natural way and that the multiple–length scale problem can be coped with by adapting the grid. However, the extension to such non–uniform meshes induces the (in)famous Pulay forces (see Sect. 2.5) if the mesh moves as the nuclei move.

3 Basic Techniques: Implementation within the CPMD Code

3.1 Introduction and Basic Definitions

This section discusses the implementation of the plane wave–pseudopotential molecular dynamics method within the CPMD computer code [142]. It concentrates on the basics leaving advanced methods to later chapters. In addition all formulas are for the non-spin polarized case. This allows to show the essential features of a plane wave code as well as the reasons for its high performance in detail. The implementation of other versions of the presented algorithms and of the more advanced techniques in Sect. 4 is in most cases very similar.

There are many reviews on the pseudopotential plane wave method alone or in connection with the Car–Parrinello algorithm. Older articles [312,157,487,591] as well as the book by Singh [578] concentrate on the electronic structure part. Other reviews [513,472,223,224] present the plane wave method in connection with the molecular dynamics technique.

3.1.1 Unit Cell and Plane Wave Basis

The unit cell of a periodically repeated system is defined by the Bravais lattice vectors $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$. The Bravais vectors can be combined into a three by three matrix $\mathbf{h} = [\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]$ [459]. The volume Ω of the cell is calculated as the determinant of $\mathbf{h}$

$$\Omega = \det\mathbf{h} \ . \tag{104}$$

Further, scaled coordinates $\mathbf{s}$ are introduced that are related to $\mathbf{r}$ via $\mathbf{h}$

$$\mathbf{r} = \mathbf{h}\mathbf{s} \ . \tag{105}$$

Distances in scaled coordinates are related to distances in real coordinates by the metric tensor $\mathcal{G} = \mathbf{h}^{\mathrm{t}}\mathbf{h}$

$$(\mathbf{r}_i - \mathbf{r}_j)^2 = (\mathbf{s}_i - \mathbf{s}_j)^{\mathrm{t}}\mathcal{G}(\mathbf{s}_i - \mathbf{s}_j) \ . \tag{106}$$

Periodic boundary conditions can be enforced by using

$$\mathbf{r}_{\mathrm{pbc}} = \mathbf{r} - \mathbf{h}\big[\mathbf{h}^{-1}\mathbf{r}\big]_{\mathrm{NINT}} \ , \tag{107}$$

where $[\cdots]_{\mathrm{NINT}}$ denotes the nearest integer value. The coordinates $\mathbf{r}_{\mathrm{pbc}}$ will be always within the box centered around the origin of the coordinate system. Reciprocal lattice vectors $\mathbf{b}_i$ are defined as

$$\mathbf{b}_i \cdot \mathbf{a}_j = 2\pi\,\delta_{ij} \tag{108}$$

and can also be arranged to a three by three matrix

$$[\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3] = 2\pi(\mathbf{h}^{\mathrm{t}})^{-1} \ . \tag{109}$$

Plane waves build a complete and orthonormal basis with the above periodicity (see also the section on plane waves in Sect. 2.8)

$$f_{\mathbf{G}}^{\mathrm{PW}}(\mathbf{r}) = \frac{1}{\sqrt{\Omega}}\exp[i\mathbf{G}\cdot\mathbf{r}] = \frac{1}{\sqrt{\Omega}}\exp[2\pi\,i\,\mathbf{g}\cdot\mathbf{s}] \ , \tag{110}$$

with the reciprocal space vectors

$$\mathbf{G} = 2\pi(\mathbf{h}^{\mathrm{t}})^{-1}\mathbf{g} \ , \tag{111}$$

where $\mathbf{g} = [i, j, k]$ is a triple of integer values. A periodic function can be expanded in this basis

$$\psi(\mathbf{r}) = \psi(\mathbf{r} + \mathbf{L}) = \frac{1}{\sqrt{\Omega}}\sum_{\mathbf{G}}\psi(\mathbf{G})\exp[i\,\mathbf{G}\cdot\mathbf{r}] \ , \tag{112}$$

where $\psi(\mathbf{r})$ and $\psi(\mathbf{G})$ are related by a three-dimensional Fourier transform. The direct lattice vectors $\mathbf{L}$ connect equivalent points in different cells.

3.1.2 Plane Wave Expansions

The Kohn–Sham potential (see Eq. (82)) of a periodic system exhibits the same periodicity as the direct lattice

$$V^{\mathrm{KS}}(\mathbf{r}) = V^{\mathrm{KS}}(\mathbf{r} + \mathbf{L}) \ , \tag{113}$$

and the Kohn–Sham orbitals can be written in Bloch form (see e.g. Ref. [27])

$$\Psi(\mathbf{r}) = \Psi_i(\mathbf{r}, \mathbf{k}) = \exp[i\,\mathbf{k}\cdot\mathbf{r}]\ u_i(\mathbf{r}, \mathbf{k}) \ , \tag{114}$$

where $\mathbf{k}$ is a vector in the first Brillouin zone. The functions $u_i(\mathbf{r}, \mathbf{k})$ have the periodicity of the direct lattice

$$u_i(\mathbf{r}, \mathbf{k}) = u_i(\mathbf{r} + \mathbf{L}, \mathbf{k}) \ . \tag{115}$$

The index i runs over all states and the states have an occupation $f_i(\mathbf{k})$ associated with them. The periodic functions $u_i(\mathbf{r},\mathbf{k})$ are now expanded in the plane wave basis

$$u_i(\mathbf{r},\mathbf{k}) = \frac{1}{\sqrt{\Omega}} \sum_{\mathbf{G}} c_i(\mathbf{G},\mathbf{k}) \exp[i\mathbf{G}\cdot\mathbf{r}] \quad , \tag{116}$$

and the Kohn–Sham orbitals are

$$\phi_i(\mathbf{r},\mathbf{k}) = \frac{1}{\sqrt{\Omega}} \sum_{\mathbf{G}} c_i(\mathbf{G},\mathbf{k}) \exp[i(\mathbf{G}+\mathbf{k})\cdot\mathbf{r}] \quad , \tag{117}$$

where $c_i(\mathbf{G},\mathbf{k})$ are complex numbers. With this expansion the density can also be expanded into a plane wave basis

$$n(\mathbf{r}) = \frac{1}{\Omega} \sum_i \int d\mathbf{k}\; f_i(\mathbf{k}) \sum_{\mathbf{G},\mathbf{G}'} c_i^\star(\mathbf{G}',\mathbf{k}) c_i(\mathbf{G},\mathbf{k}) \exp[i(\mathbf{G}+\mathbf{k})\cdot\mathbf{r}] \tag{118}$$

$$= \sum_{\mathbf{G}} n(\mathbf{G}) \exp[i\,\mathbf{G}\cdot\mathbf{r}] \quad , \tag{119}$$

where the sum over $\mathbf{G}$ vectors in Eq. (119) expands over double the range given by the wavefunction expansion. This is one of the main advantages of the plane wave basis. Whereas for atomic orbital basis sets the number of functions needed to describe the density grows quadratically with the size of the system, there is only a linear dependence for plane waves.

3.1.3 $\mathbf{K}$*–Points and Cutoffs*

In actual calculations the infinite sums over $\mathbf{G}$ vectors and cells has to be truncated. Furthermore, we have to approximate the integral over the Brillouin zone by a finite sum over special $\mathbf{k}$–points

$$\int d\mathbf{k} \rightarrow \sum_{\mathbf{k}} w_{\mathrm{k}} \quad , \tag{120}$$

where w_k are the weights of the integration points. Schemes on how to choose the integration points efficiently are available in the literature [30,123,435] where also an overview [179] on the use of $\mathbf{k}$–points in the calculation of the electronic structure of solids can be found.

The truncation of the plane wave basis rests on the fact that the Kohn–Sham potential $V^{\mathrm{KS}}(\mathbf{G})$ converges rapidly with increasing modulus of $\mathbf{G}$. For this reason, at each $\mathbf{k}$–point, only $\mathbf{G}$ vectors with a kinetic energy lower than a given maximum cutoff

$$\frac{1}{2} |\mathbf{k}+\mathbf{G}|^2 \leq E_{\mathrm{cut}} \tag{121}$$

are included in the basis. With this choice of the basis the precision of the calculation within the approximations of density functional theory is controlled by one parameter E_{cut} only.

The number of plane waves for a given cutoff depends on the unit cell and the **k**–point. An estimate for the size of the basis at the center of the Brillouin zone is

$$N_{\mathrm{PW}} = \frac{1}{2\pi^2}\,\Omega\,E_{\mathrm{cut}}^{3/2}\ , \tag{122}$$

where E_{cut} is in Hartree units. The basis set needed to describe the density calculated from the Kohn-Sham orbitals has a corresponding cutoff that is four times the cutoff of the orbitals. The number of plane waves needed at a given density cutoff is therefore eight times the number of plane waves needed for the orbitals.

It is a common approximation in density functional theory calculations [536,169] to use approximate electronic densities. Instead of using the full description, the density is expanded in an auxiliary basis. An incomplete plane wave basis can be considered as an auxiliary basis with special properties [371]. Because of the filter property of plane waves the new density is an optimal approximation to the true density. No additional difficulties in calculations of the energy or forces appear. The only point to control is, if the accuracy of the calculation is still sufficient.

Finally, sums over all unit cells in real space have to be truncated. The only term in the final energy expression with such a sum is the real space part of the Ewald sum (see Sect. 3.2). This term is not a major contribution to the workload in a density functional calculation, that is the cutoff can be set rather generously.

3.1.4 Real Space Grid

A function given as a finite linear combination of plane waves can also be defined as a set of functional values on a equally spaced grid in real space. The sampling theorem (see e.g. Ref. [492]) gives the maximal grid spacing that still allows to hold the same information as the expansion coefficients of the plane waves. The real space sampling points **R** are defined

$$\mathbf{R} = \mathbf{h}\,\mathbf{N}\mathbf{q}\ , \tag{123}$$

where **N** is a diagonal matrix with the entries $1/N_s$ and **q** is a vector of integers ranging from 0 to $N_s - 1$ (s = x, y, z). To fulfill the sampling theorem N_s has to be bigger than $2\max(\mathbf{g}_s) + 1$. To be able to use fast Fourier techniques, N_s must be decomposable into small prime numbers (typically 2, 3, and 5). In applications the smallest number N_s that fulfills the above requirements is chosen.

A periodic function can be calculated at the real space grid points

$$f(\mathbf{R}) = \sum_{\mathbf{G}} f(\mathbf{G}) \exp[i\,\mathbf{G}\cdot\mathbf{R}] \tag{124}$$

$$= \sum_{\mathbf{g}} f(\mathbf{G}) \exp\left[2\pi\,i\,\left((\mathbf{h}^{\mathrm{t}})^{-1}\mathbf{g}\right)\cdot(\mathbf{h}\mathbf{N}\mathbf{q})\right] \tag{125}$$

$$= \sum_{\mathbf{g}} f(\mathbf{G}) \exp\left[\frac{2\pi}{N_{\mathrm{x}}} i g_{\mathrm{x}} q_{\mathrm{x}}\right] \exp\left[\frac{2\pi}{N_{\mathrm{y}}} i g_{\mathrm{y}} q_{\mathrm{y}}\right] \exp\left[\frac{2\pi}{N_{\mathrm{z}}} i g_{\mathrm{z}} q_{\mathrm{z}}\right]\ . \tag{126}$$

The function $f(\mathbf{G})$ is zero outside the cutoff region and the sum over **g** can be extended over all indices in the cube $-\mathbf{g}_{\mathrm{s}}^{\max}\ldots\mathbf{g}_{\mathrm{s}}^{\max}$. The functions $f(\mathbf{R})$ and

$f(\mathbf{G})$ are related by three–dimensional Fourier transforms

$$f(\mathbf{R}) = \text{inv\_FT}\left[f(\mathbf{G})\right] \tag{127}$$

$$f(\mathbf{G}) = \text{fw\_FT}\left[f(\mathbf{R})\right] \quad . \tag{128}$$

The Fourier transforms are defined by

$$\begin{aligned}\left[\text{inv\_FT}\left[f(\mathbf{G})\right]\right]_{uvw} = & \sum_{j=0}^{N_\text{x}-1}\sum_{k=0}^{N_\text{y}-1}\sum_{l=0}^{N_\text{z}-1} f_{jkl}^{\mathbf{G}} \\ & \exp\left[i\frac{2\pi}{N_\text{x}}\,j\,u\right]\exp\left[i\frac{2\pi}{N_\text{y}}\,k\,v\right]\exp\left[i\frac{2\pi}{N_\text{z}}\,l\,w\right] \end{aligned} \tag{129}$$

$$\begin{aligned}\left[\text{fw\_FT}\left[f(\mathbf{R})\right]\right]_{jkl} = & \sum_{u=0}^{N_\text{x}-1}\sum_{v=0}^{N_\text{y}-1}\sum_{w=0}^{N_\text{z}-1} f_{uvw}^{\mathbf{R}} \\ & \exp\left[-i\frac{2\pi}{N_\text{x}}\,j\,u\right]\exp\left[-i\frac{2\pi}{N_\text{y}}\,k\,v\right]\exp\left[-i\frac{2\pi}{N_\text{z}}\,l\,w\right] \quad , \end{aligned} \tag{130}$$

where the appropriate mappings of $\mathbf{q}$ and $\mathbf{g}$ to the indices

$$[u, v, w] = \mathbf{q} \tag{131}$$

$$\{j, k, l\} = \mathbf{g}_s \quad \text{if} \;\; \mathbf{g}_s \geq 0 \tag{132}$$

$$\{j, k, l\} = N_s + \mathbf{g}_s \quad \text{if} \;\; \mathbf{g}_s < 0 \tag{133}$$

have to be used. From Eqs. (129) and (130) it can be seen, that the calculation of the three–dimensional Fourier transforms can be performed by a series of one dimensional Fourier transforms. The number of transforms in each direction is $N_\text{x}N_\text{y}$, $N_\text{x}N_\text{z}$, and $N_\text{y}N_\text{z}$ respectively. Assuming that the one-dimensional transforms are performed within the fast Fourier transform framework, the number of operations per transform of length n is approximately $5n\log n$. This leads to an estimate for the number of operations for the full three-dimensional transform of $5N\log N$, where $N = N_\text{x}N_\text{y}N_\text{z}$.

3.1.5 Pseudopotentials

In order to minimize the size of the plane wave basis necessary for the calculation, core electrons are replaced by pseudopotentials. The pseudopotential approximation in the realm of solid–state theory goes back to the work on orthogonalized plane waves [298] and core state projector methods [485]. Empirical pseudopotentials were used in plane wave calculations [294,703] but new developments have considerably increased efficiency and reliability of the method. Pseudopotential are required to correctly represent the long range interactions of the core and to produce pseudo–wavefunction solutions that approach the full wavefunction outside a core radius r_c. Inside this radius the pseudopotential and the wavefunction should be as smooth as possible, in order to allow for a small plane wave cutoff. For the pseudo–wavefunction this requires that the nodal structure of the valence wavefunctions is replaced by a smooth function. In addition it is desired that a pseudopotential is transferable [238,197], this means that one and the same pseudopotential can be

used in calculations of different chemical environment resulting in calculations with comparable accuracy.

A first major step to achieve all this conflicting goals was the introduction of "norm–conservation" [622,593]. Norm–conserving pseudopotentials have to be angular momentum dependent. In their most general form they are semi–local

$$V^{\rm PP}(\mathbf{r},\mathbf{r}') = \sum_{lm} Y_{lm}(\mathbf{r}) V_l(\mathbf{r}) \delta_{\mathbf{r},\mathbf{r}'} Y_{lm}(\mathbf{r}') \ , \tag{134}$$

where Y_{lm} are spherical harmonics. A minimal set of requirements and a construction scheme for soft, semi–local pseudopotentials were developed [274,28] . Since then many variations of the original method have been proposed, concentrating either on an improvement in softness or in transferability. Analytic representations of the core part of the potential [326,626,627,509] were used. Extended norm-conservation [564] was introduced to enhance transferability and new concepts to increase the softness were presented [659,509,369]. More information on pseudopotentials and their construction can be found in recent review articles [487,578,221].

Originally generated in a semi-local form, most applications use the fully separable form. Pseudopotentials can be transformed to the separable form using atomic wavefunctions [335,73,659]. Recently [239,288] a new type of separable, norm-conserving pseudopotentials was introduced. Local and non-local parts of these pseudopotentials have a simple analytic form and only a few parameters are needed to characterize the potential. These parameters are globally optimized in order to reproduce many properties of atoms and ensure a good transferability.

A separable non-local pseudopotential can be put into general form (this includes all the above mentioned types)

$$V^{\rm PP}(\mathbf{r},\mathbf{r}') = \left(V_{\rm core}(\mathbf{r}) + \Delta V_{\rm local}(\mathbf{r})\right) \delta_{\mathbf{r},\mathbf{r}'} + \sum_{k,l} P_k^\star(\mathbf{r}) h_{kl} P_l(\mathbf{r}') \ . \tag{135}$$

The local part has been split into a core ($\sim 1/r$ for $r \to \infty$) and a short-ranged local part in order to facilitate the derivation of the final energy formula. The actual form of the core potential will be defined later. The local potential $\Delta V_{\rm local}$ and the projectors P_k are atom-centered functions of the form

$$\varphi(\mathbf{r}) = \varphi(|\mathbf{r} - \mathbf{R}_I|) \ Y_{lm}(\theta,\phi) \ , \tag{136}$$

that can be expanded in plane waves

$$\varphi(\mathbf{r}) = \sum_{\mathbf{G}} \varphi(G) \exp[i\mathbf{G}\cdot\mathbf{r}] \ S_I(\mathbf{G}) \ Y_{lm}(\tilde{\theta},\tilde{\phi}) \ , \tag{137}$$

$\mathbf{R}_I$ denote atomic positions and the so–called structure factors S_I are defined as

$$S_I(\mathbf{G}) = \exp[-i\mathbf{G}\cdot\mathbf{R}_I] \ . \tag{138}$$

The functions $\varphi(G)$ are calculated from $\varphi(r)$ by a Bessel transform

$$\varphi(G) = 4\pi \ (-i)^l \int_0^\infty dr \, r^2 \, \varphi(r) \, j_l(Gr) \ , \tag{139}$$

where j_l are spherical Bessel functions of the first kind. The local pseudopotential and the projectors of the nonlocal part in Fourier space are given by

$$\Delta V_{\rm local}(\mathbf{G}) = \frac{4\pi}{\Omega}\int_0^\infty dr\, r^2\, \Delta V_{\rm local}(r) j_0(Gr) \tag{140}$$

$$P_k(\mathbf{G}) = \frac{4\pi}{\sqrt{\Omega}}\,(-i)^l \int_0^\infty dr\, r^2\, P_k(r)\, j_l(Gr)\; Y_{lm}(\tilde{\theta},\tilde{\phi}) \;\;, \tag{141}$$

where lm are angular momentum quantum numbers associated with projector α.

3.2 Electrostatic Energy

3.2.1 General Concepts

The electrostatic energy of a system of nuclear charges Z_I at positions $\mathbf{R}_I$ and an electronic charge distribution $n(\mathbf{r})$ consists of three parts: the Hartree energy of the electrons, the interaction energy of the electrons with the nuclei and the internuclear interactions

$$\begin{aligned} E_{\rm ES} = & \frac{1}{2}\iint d\mathbf{r}\, d\mathbf{r}'\, \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \\ & + \sum_I \int d\mathbf{r}\, V^I_{\rm core}(\mathbf{r}) n(\mathbf{r}) + \frac{1}{2}\sum_{I\neq J}\frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} \;\;. \end{aligned} \tag{142}$$

The Ewald method (see e.g. Ref. [12]) can be used to avoid singularities in the individual terms when the system size is infinite. In order to achieve this a Gaussian core charge distribution associated with each nuclei is defined

$$n^I_{\rm c}(\mathbf{r}) = -\frac{Z_I}{(\mathrm{R}^{\rm c}_I)^3}\pi^{-3/2}\exp\left[-\left(\frac{\mathbf{r}-\mathbf{R}_I}{\mathrm{R}^{\rm c}_I}\right)^2\right] \;\;. \tag{143}$$

It is convenient at this point to make use of the arbitrariness in the definition of the core potential and define it to be the potential of the Gaussian charge distribution of Eq. (143)

$$V^I_{\rm core}(\mathbf{r}) = \int d\mathbf{r}'\frac{n^I_{\rm c}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} = -\frac{Z_I}{|\mathbf{r}-\mathbf{R}_I|}\mathrm{erf}\left[\frac{|\mathbf{r}-\mathbf{R}_I|}{\mathrm{R}^{\rm c}_I}\right] \;\;, \tag{144}$$

where erf is the error function. The interaction energy of this Gaussian charge

distributions is now added and subtracted from the total electrostatic energy

$$\begin{aligned} E_{\rm ES} = & \frac{1}{2}\int\!\!\int d\mathbf{r}\, d\mathbf{r}'\, \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \\ & + \frac{1}{2}\int\!\!\int d\mathbf{r}\, d\mathbf{r}'\, \frac{n_{\rm c}(\mathbf{r})n_{\rm c}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \\ & + \int\!\!\int d\mathbf{r}\, d\mathbf{r}'\, \frac{n_{\rm c}(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \\ & + \frac{1}{2}\sum_{I\neq J}\frac{Z_I Z_J}{|\mathbf{R}_I-\mathbf{R}_J|} \\ & - \frac{1}{2}\int\!\!\int d\mathbf{r}\, d\mathbf{r}'\, \frac{n_{\rm c}(\mathbf{r})n_{\rm c}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \quad , \end{aligned} \tag{145}$$

where $n_{\rm c}(\mathbf{r}) = \sum_I n_{\rm c}^I(\mathbf{r})$. The first three terms can be combined to the electrostatic energy of a total charge distribution $n_{\rm tot}(\mathbf{r}) = n(\mathbf{r}) + n_{\rm c}(\mathbf{r})$. The remaining terms are rewritten as a double sum over nuclei and a sum over self–energy terms of the Gaussian charge distributions

$$\begin{aligned} E_{\rm ES} = & \frac{1}{2}\int\!\!\int d\mathbf{r}\, d\mathbf{r}'\, \frac{n_{\rm tot}(\mathbf{r})n_{\rm tot}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \\ & + \frac{1}{2}\sum_{I\neq J}\frac{Z_I Z_J}{|\mathbf{R}_I-\mathbf{R}_J|}{\rm erfc}\left[\frac{|\mathbf{R}_I-\mathbf{R}_J|}{\sqrt{{\rm R}_I^{\rm c\,2}+{\rm R}_J^{\rm c\,2}}}\right] - \sum_I \frac{1}{\sqrt{2\pi}}\frac{Z_I^2}{R_I^{\rm c}} \quad , \end{aligned} \tag{146}$$

where erfc denotes the complementary error function.

3.2.2 Periodic Systems

For a periodically repeated system the total energy per unit cell is derived from the above expression by using the solution to Poisson's equation in Fourier space for the first term and make use of the quick convergence of the second term in real space. The total charge is expanded in plane waves with expansion coefficients

$$n_{\rm tot}(\mathbf{G}) = n(\mathbf{G}) + \sum_I n_{\rm c}^I(\mathbf{G}) S_I(\mathbf{G}) \tag{147}$$

$$= n(\mathbf{G}) - \frac{1}{\Omega}\sum_I \frac{Z_I}{\sqrt{4\pi}}\exp\left[-\frac{1}{2}G^2 R_I^{\rm c\,2}\right] S_I(\mathbf{G}) \quad . \tag{148}$$

This leads to the electrostatic energy for a periodic system

$$E_{\rm ES} = 2\pi\,\Omega \sum_{\mathbf{G}\neq 0}\frac{|n_{\rm tot}(\mathbf{G})|^2}{G^2} + E_{\rm ovrl} - E_{\rm self} \quad , \tag{149}$$

where

$$E_{\rm ovrl} = {\sum_{I,J}}' \sum_{\mathbf{L}} \frac{Z_I Z_J}{|\mathbf{R}_I-\mathbf{R}_J-\mathbf{L}|}{\rm erfc}\left[\frac{|\mathbf{R}_I-\mathbf{R}_J-\mathbf{L}|}{\sqrt{{\rm R}_I^{\rm c\,2}+{\rm R}_J^{\rm c\,2}}}\right] \tag{150}$$

and

$$E_{\rm self} = \sum_I \frac{1}{\sqrt{2\pi}} \frac{Z_I^2}{R_I^{\rm c}} \ . \tag{151}$$

Here, the sums expand over all atoms in the simulation cell, all direct lattice vectors $\mathbf{L}$, and the prime in the first sum indicates that $I < J$ is imposed for $\mathbf{L} = \mathbf{0}$.

3.2.3 Cluster Boundary Conditions

The possibility to use fast Fourier transforms to calculate the electrostatic energy is one of the reasons for the high performance of plane wave calculations. However, plane wave based calculations imply periodic boundary conditions. This is appropriate for crystal calculations but very unnatural for molecule or slab calculations. For neutral systems this problem is circumvented by use of the supercell method. Namely, the molecule is periodically repeated but the distance between each molecule and its periodic images is so large that their interaction is negligible. This procedure is somewhat wasteful but can lead to satisfactory results.

Handling charged molecular systems is, however, considerably more difficult, due to the long range Coulomb forces. A charged periodic system has infinite energy and the interaction between images cannot really be completely eliminated. In order to circumvent this problem several solutions have been proposed. The simplest fix-up is to add to the system a neutralizing background charge. This is achieved trivially as the $\mathbf{G} = \mathbf{0}$ term in Eq. (149) is already eliminated. This leads to finite energies but does not eliminate the interaction between the images and makes the calculation of absolute energies difficult. Other solutions involve performing a set of different calculations on the system such that extrapolation to the limit of infinitely separated images is possible. This procedure is lengthy and one cannot use it easily in molecular dynamics applications. It has been shown, that it is possible to estimate the correction to the total energy for the removal of the image charges [378]. Still it seems not easy to incorporate this scheme into the frameworks of molecular dynamics. Another method [60,702,361] works with the separation of the long and short range parts of the Coulomb forces. In this method the low–order multipole moments of the charge distribution are separated out and handled analytically. This method was used in the context of coupling *ab initio* and classical molecular dynamics [76].

The long-range forces in Eq. (146) are contained in the first term. This term can be written

$$\frac{1}{2} \int\int d\mathbf{r}\, d\mathbf{r}' \, \frac{n_{\rm tot}(\mathbf{r}) n_{\rm tot}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} = \frac{1}{2} \int d\mathbf{r}\, V_{\rm H}(\mathbf{r}) n_{\rm tot}(\mathbf{r}) \ , \tag{152}$$

where the electrostatic potential $V_{\rm H}(\mathbf{r})$ is the solution of Poisson's equation (see Eq. (80)). There are two approaches to solve Poisson's equation subject to the boundary conditions $V_{\rm H}(\mathbf{r}) \to 0$ for $\mathbf{r} \to \infty$ implemented in `CPMD`. Both of them rely on fast Fourier transforms, thus keeping the same framework as for the periodic case.

The first method is due to Hockney [300] and was first applied to density functional plane wave calculations in Ref. [36]. In the following outline, for the sake of simplicity,

a one-dimensional case is presented. The charge density is assumed to be non-zero only within an interval L and sampled on N equidistant points. These points are denoted by x_p. The potential can then be written

$$V_{\rm H}(x_p) = \frac{L}{N} \sum_{p'=-\infty}^{\infty} G(x_p - x_{p'}) n(x_{p'}) \tag{153}$$

$$= \frac{L}{N} \sum_{p'=0}^{N} G(x_p - x_{p'}) n(x_{p'}) \tag{154}$$

for $p = 0, 1, 2, \ldots N$, where $G(x_p - x_{p'})$ is the corresponding Green's function. In Hockney's algorithm this equation is replaced by the cyclic convolution

$$\tilde{V}_{\rm H}(x_p) = \frac{L}{N} \sum_{p'=0}^{2N+1} \tilde{G}(x_p - x_{p'}) \tilde{n}(x_{p'}) \tag{155}$$

where $p = 0, 1, 2, \ldots 2N+1$, and

$$\tilde{n}(x_p) = \begin{cases} n(x_p) & 0 \le p \le N \\ 0 & N \le p \le 2N+1 \end{cases} \tag{156}$$

$$\tilde{G}(x_p) = G(x_p) \qquad -(N+1) \le p \le N \tag{157}$$

$$\tilde{n}(x_p) = \tilde{n}(x_p + L) \tag{158}$$

$$\tilde{G}(x_p) = \tilde{G}(x_p + L) \tag{159}$$

The solution $\tilde{V}_{\rm H}(x_p)$ can be obtained by a series of fast Fourier transforms and has the desired property

$$\tilde{V}_{\rm H}(x_p) = V_{\rm H}(x_p) \quad \text{for } 0 \le p \le N \ . \tag{160}$$

To remove the singularity of the Green's function at $x = 0$, $G(x)$ is modified for small x and the error is corrected by using the identity

$$G(x) = \frac{1}{x} {\rm erf}\left[\frac{x}{r_{\rm c}}\right] + \frac{1}{x} {\rm erfc}\left[\frac{x}{r_{\rm c}}\right] \ , \tag{161}$$

where $r_{\rm c}$ is chosen such, that the short-ranged part can be accurately described by a plane wave expansion with the density cutoff. In an optimized implementation Hockney's method requires the double amount of memory and two additional fast Fourier transforms on the box of double size (see Fig. 6 for a flow chart). Hockney's method can be generalized to systems with periodicity in one (wires) and two (slabs) dimensions. It was pointed out [173] that Hockney's method gives the exact solution to Poisson's equation for isolated systems if the boundary condition (zero density at the edges of the box) are fulfilled.

A different, fully reciprocal space based method, that can be seen as an approximation to Hockney's method, was recently proposed [393]. The final expression for the Hartree energy is also based on the splitting of the Green's function in Eq. (161)

$$E_{\rm ES} = 2\pi\,\Omega \sum_{\mathbf{G}} V_{\rm H}^{\rm MT}(\mathbf{G}) n_{\rm tot}^{\star}(\mathbf{G}) + E_{\rm ovrl} - E_{\rm self} \ . \tag{162}$$

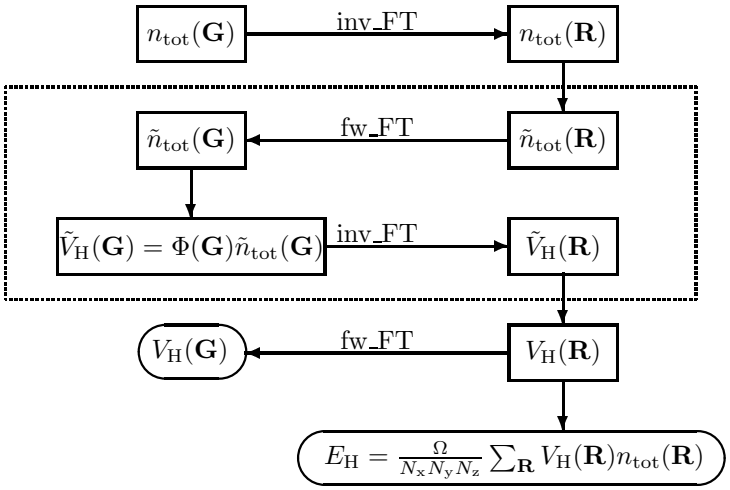


Figure 6. Flow chart for the calculation of long-ranged part of the electrostatic energy using the method by Hockney [300]. The part inside the dashed box is calculated most efficiently with the procedure outlined by Eastwood and Brownrigg [173].

The potential function is calculated from two parts,

$$V_{\rm H}^{\rm MT}(\mathbf{G}) = \bar{V}_{\rm H}(\mathbf{G}) + \tilde{V}_{\rm H}(\mathbf{G}) \ , \tag{163}$$

where $\tilde{V}_{\rm H}(\mathbf{G})$ is the analytic part, calculated from a Fourier transform of erfc

$$\tilde{V}_{\rm H}(\mathbf{G}) = \frac{4\pi}{G^2}\left(1 - \exp\left[-\frac{G^2 r_{\rm c}^2}{4}\right]\right) n(\mathbf{G}) \tag{164}$$

and $\bar{V}_{\rm H}(\mathbf{G})$ is calculated from a discrete Fourier transform of the Green's function on an appropriate grid. The calculation of the Green's function can be done at the beginning of the calculation and has not to be repeated again. It is reported [393] that a cutoff of ten to twenty percent higher than the one employed for the charge density gives converged results. The same technique can also be applied for systems that are periodic in one and two dimensions [394].

If the boundary conditions are appropriately chosen, the discrete Fourier transforms for the calculation of $\bar{V}_{\rm H}(\mathbf{G})$ can be performed analytically [437]. This is possible for the limiting case where $r_{\rm c} = 0$ and the boundary conditions are on a sphere of radius R for the cluster. For a one-dimensional system we choose a torus of radius R and for the two-dimensional system a slab of thickness Z. The electrostatic potential for these systems are listed in Table 2, where $G_{\rm xy} = \left[g_{\rm x}^2 + g_{\rm y}^2\right]^{1/2}$ and J_n and K_n are the Bessel functions of the first and second kind of integer order n.

Hockney's method requires a computational box such that the charge density is negligible at the edges. This is equivalent to the supercell approach [510]. Practical experience tells that a minimum distance of about 3 Å of all atoms to the edges of

Table 2. Fourier space formulas for the Hartree energy, see text for definitions.

Dim.	periodic	$(G^2/4\pi)V_{\rm H}(\mathbf{G})$	$V_{\rm H}(\mathbf{0})$
0	–	$(1-\cos[R\,G])\,n(\mathbf{G})$	$2\pi R^2 n(0)$
1	z	$(1+R\,(G_{\rm xy}J_1(RG_{\rm xy})\,K_0(Rg_{\rm z})$	
		$-g_{\rm z}J_0(RG_{\rm xy})\,K_1(Rg_{\rm z})))\,n(\mathbf{G})$	0
2	x, y	$(1-(-1)^{g_{\rm z}}\exp[-GZ/2])\,n(\mathbf{G})$	0
3	x, y, z	$n(\mathbf{G})$	0

the box is sufficient for most systems. The Green's function is then applied to the charge density in a box double this size. The Green's function has to be calculated only once at the beginning of the calculation. The other methods presented in this chapter require a computational box of double the size of the Hockney method as they are applying the artificially periodic Green's function within the computational box. This can only be equivalent to the exact Hockney method if the box is enlarged to double the size. In plane wave calculations computational costs grow linearly with the volume of the box. Therefore Hockney's method will prevail over the others in accuracy, speed, and memory requirements in the limit of large systems. The direct Fourier space methods have advantages through their easy implementation and for small systems, if not full accuracy is required, i.e. if they are used with smaller computational boxes. In addition, they can be of great use in calculations with classical potentials.

3.3 Exchange and Correlation Energy

Exchange and correlation functionals implemented in the `CPMD` code are of the local type with gradient corrections. These type of functionals can be written as (see also Eqs. (88) and (84))

$$E_{\rm xc} = \int d\mathbf{r}\,\varepsilon_{\rm xc}(n,\nabla n)\,n(\mathbf{r}) = \Omega\sum_{\mathbf{G}}\varepsilon_{\rm xc}(\mathbf{G})n^{\star}(\mathbf{G}) \tag{165}$$

with the corresponding potential

$$V_{\rm xc}(\mathbf{r}) = \frac{\partial F_{\rm xc}}{\partial n} - \sum_s \frac{\partial}{\partial \mathbf{r}_s}\left[\frac{\partial F_{\rm xc}}{\partial(\partial_s n)}\right] \quad , \tag{166}$$

where $F_{\rm xc} = \varepsilon_{\rm xc}(n,\nabla n)n$ and $\partial_s n$ is the s-component of the density gradient.

Exchange and correlation functionals have complicated analytical forms that give rise to high frequency components in $\varepsilon_{\rm xc}(\mathbf{G})$. Although these high frequency components do not enter the sum in Eq. (165) due to the filter effect of the density, they affect the calculation of $\varepsilon_{\rm xc}$. As the functionals are only local in real space, not in Fourier space, they have to be evaluated on a real space grid. The function $\varepsilon_{\rm xc}(\mathbf{G})$ can then be calculated by a Fourier transform. Therefore the exact calculation of $E_{\rm xc}$ would require a grid with infinite resolution. However, the high frequency components are usually very small and even a moderate grid gives accurate results. The use of a finite grid results in an effective redefinition of the exchange and

correlation energy

$$E_{\rm xc} = \frac{\Omega}{N_{\rm x}N_{\rm y}N_{\rm z}} \sum_{\mathbf{R}} \varepsilon_{\rm xc}(n,\nabla n)(\mathbf{R})n(\mathbf{R}) = \Omega \sum_{\mathbf{G}} \tilde{\varepsilon}_{\rm xc}(\mathbf{G})n(\mathbf{G}) \ , \tag{167}$$

where $\tilde{\varepsilon}_{\rm xc}(\mathbf{G})$ is the finite Fourier transform of $\varepsilon_{\rm xc}(\mathbf{R})$. This definition of $E_{\rm xc}$ allows the calculation of all gradients analytically. In most applications the real space grid used in the calculation of the density and the potentials is also used for the exchange and correlation energy. Grids with higher resolution can be used easily. The density is calculated on the new grid by use of Fourier transforms and the resulting potential is transfered back to the original grid. With this procedure the different grids do not have to be commensurate.

The above redefinition has an undesired side effect. The new exchange and correlation energy is no longer translationally invariant. Only translations by a multiple of the grid spacing do not change the total energy. This introduces a small modulation of the energy hyper surface [685], known as "ripples". Highly accurate optimizations of structures and the calculation of harmonic frequencies can be affected by the ripples. Using a denser grid for the calculation of $E_{\rm xc}$ is the only solution to avoid these problems.

The calculation of a gradient corrected functional within the plane wave framework can be conducted using Fourier transforms [685]. The flowchart of the calculation is presented in Fig. 7. With the use of Fourier transforms the calculation of second derivatives of the charge density is avoided, leading to a numerically stable algorithm. To this end the identity

$$\frac{\partial F_{\rm xc}}{\partial(\partial_s n)} = \frac{\partial F_{\rm xc}}{\partial|\nabla n|}\frac{\partial_s n}{|\nabla n|} \tag{168}$$

is used.

This is the place to say some words on functionals that include exact exchange. As mentioned in Sect. 2.7 this type of functional has been very popular recently and improvements of results over GGA–type density functionals for many systems and properties have been reported. However, the calculation of the Hartree–Fock exchange causes a considerable performance problem in plane wave calculations. The Hartree–Fock exchange energy is defined as [604]

$$E_{\rm HFX} = \sum_{ij} \int\int d\mathbf{r}d\mathbf{r}' \frac{\rho_{ij}(\mathbf{r})\rho_{ij}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \ , \tag{169}$$

where

$$\rho_{ij}(\mathbf{r}) = \phi_i(\mathbf{r})\phi_j(\mathbf{r}). \tag{170}$$

From this expression the wavefunction force is easily derived and can be calculated in Fourier space

$$\frac{1}{f_i}\frac{\partial E_{\rm HFX}}{\partial c_i^\star(\mathbf{G})} = \sum_j \sum_{\mathbf{G}'} V_{\rm HFX}^{ij}(\mathbf{G}-\mathbf{G}')c_j(\mathbf{G}') \ . \tag{171}$$

The force calculation is best performed in real space, whereas the potential is calculated in Fourier space. For a system with $N_{\rm b}$ electronic states and N real space

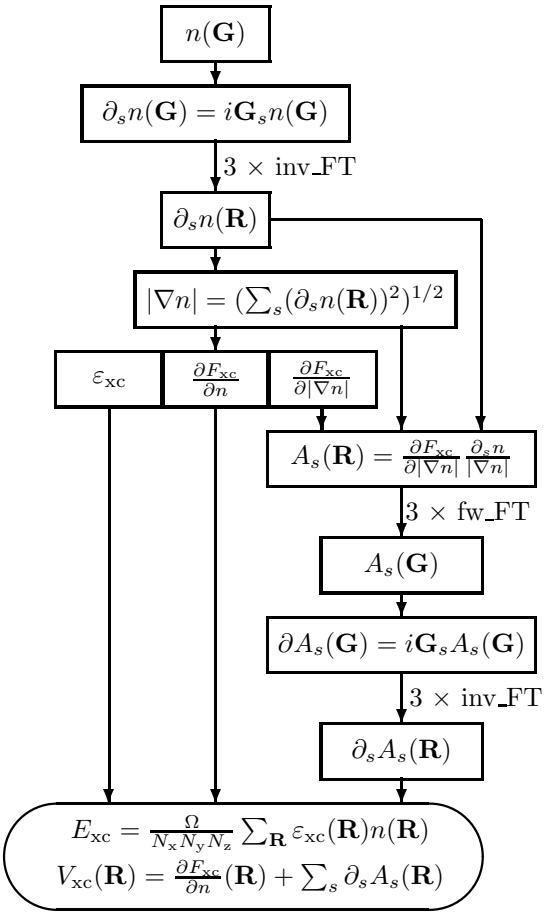


Figure 7. Flow chart for the calculation of the energy and potential of a gradient corrected exchange and correlation functional.

grid points, a total of $5N_{\rm b}^2$ three–dimensional transforms are needed, resulting in approximately $25N_{\rm b}^2 N \log N$ operations needed to perform the calculation. This has to be compared to the $15N_{\rm b} N \log N$ operations needed for the other Fourier transforms of the charge density and the application of the local potential and the $4N_{\rm b}^2 N$ operations for the orthogonalization step. In calculations dominated by the Fourier transforms an additional factor of at least $N_{\rm b}$ is needed. If on the other hand orthogonalization dominates an increase in computer time by a factor of $5 \log N$ is expected. Therefore, at least an order of magnitude more computer time is needed for calculations including exact exchange compared to ordinary density functional calculations. Consequently, hybrid functionals will only be used in exceptional cases together with plane waves [262,128].

3.4 Total Energy, Gradients, and Stress Tensor

3.4.1 Total Energy

Molecular dynamics calculations with interaction potentials derived from density functional theory require the evaluation of the total energy and derivatives with respect to the parameters of the Lagrangian. In this section formulas are given in Fourier space for a periodic system. The total energy can be calculated as a sum of kinetic, external (local and non-local pseudopotential), exchange and correlation, and electrostatic energy (to be compared with Eq. (75))

$$E_{\text{total}} = E_{\text{kin}} + E^{\text{PP}}_{\text{local}} + E^{\text{PP}}_{\text{nonlocal}} + E_{\text{xc}} + E_{\text{ES}} \ . \tag{172}$$

The individual terms are defined by

$$E_{\text{kin}} = \sum_{\mathbf{k}} w_{\mathbf{k}} \sum_{i} \sum_{\mathbf{G}} \frac{1}{2} f_i(\mathbf{k}) \left|\mathbf{G}+\mathbf{k}\right|^2 \left|c_i(\mathbf{G},\mathbf{k})\right|^2 \tag{173}$$

$$E^{\text{PP}}_{\text{local}} = \sum_{I} \sum_{\mathbf{G}} \Delta V^{I}_{\text{local}}(\mathbf{G})\, S_I(\mathbf{G}) n^{\star}(\mathbf{G}) \tag{174}$$

$$E^{\text{PP}}_{\text{nonlocal}} = \sum_{\mathbf{k}} w_{\mathbf{k}} \sum_{i} f_i(\mathbf{k}) \sum_{I} \sum_{\alpha,\beta \in I} \left(F^{\alpha}_{I,i}(\mathbf{k})\right)^{\star} h^{I}_{\alpha\beta} F^{\beta}_{I,i}(\mathbf{k}) \tag{175}$$

$$E_{\text{xc}} = \Omega \sum_{\mathbf{G}} \epsilon_{\text{xc}}(\mathbf{G}) n^{\star}(\mathbf{G}) \tag{176}$$

$$E_{\text{ES}} = 2\pi\,\Omega \sum_{\mathbf{G}\neq 0} \frac{\left|n_{\text{tot}}(\mathbf{G})\right|^2}{G^2} + E_{\text{ovrl}} - E_{\text{self}}. \tag{177}$$

The overlap between the projectors of the non-local pseudopotential and the Kohn–Sham orbitals has been introduced in the equation above

$$F^{\alpha}_{I,i}(\mathbf{k}) = \frac{1}{\sqrt{\Omega}} \sum_{\mathbf{G}} P^{I}_{\alpha}(\mathbf{G})\, S_I(\mathbf{G}+\mathbf{k})\, c^{\star}_i(\mathbf{G},\mathbf{k}) \ . \tag{178}$$

An alternative expression, using the Kohn–Sham eigenvalues $\epsilon_i(\mathbf{k})$ can also be used

$$\begin{aligned} E_{\text{total}} = & \sum_{\text{k}} w_{\mathbf{k}} \sum_{i} f_i(\mathbf{k}) \epsilon_i(\mathbf{k}) \\ & -\Omega \sum_{\mathbf{G}} \left(V_{\text{xc}}(\mathbf{G}) - \varepsilon_{\text{xc}}(\mathbf{G})\right) n^{\star}(\mathbf{G}) \\ & -2\pi\,\Omega \sum_{\mathbf{G}\neq 0} \frac{\left|n(\mathbf{G})\right|^2 - \left|n_{\text{c}}(\mathbf{G})\right|^2}{G^2} + E_{\text{ovrl}} - E_{\text{self}} \\ & +\Delta E_{\text{tot}} \ , \end{aligned} \tag{179}$$

to be compared to Eq. (86). The additional term ΔE_{tot} in Eq. (179) is needed to have an expression for the energy that is quadratic in the variations of the charge density, as it is true for Eq. (172). Without the correction term, which is zero for the exact charge density, small differences between the computed and the exact

density could give rise to large errors in the total energy [129]. The correction energy can be calculated from

$$\Delta E_{\rm tot} = -2\pi\,\Omega \sum_{\mathbf{G}\neq 0} \left(\frac{n^{\rm in}(\mathbf{G})}{G^2} - \frac{n^{\rm out}(\mathbf{G})}{G^2} \right) (n^{\rm out}(\mathbf{G}))^\star \\ -\Omega \sum_{\mathbf{G}} \left(V_{\rm xc}^{\rm in}(\mathbf{G}) - V_{\rm xc}^{\rm out}(\mathbf{G}) \right) (n^{\rm out}(\mathbf{G}))^\star, \tag{180}$$

where $n^{\rm in}$ and $n^{\rm out}$ are the input and output charge densities and $V_{\rm xc}^{\rm in}$ and $V_{\rm xc}^{\rm out}$ the corresponding exchange and correlation potentials. This term leads to the so–called "non–self–consistency correction" of the force, introduced in Eq. (68).

The use of an appropriate **k**–point mesh is the most efficient method to calculate the total energy of a periodic system. Equivalent, although not as efficient, the calculation can be performed using a supercell consisting of replications of the unit cell and a single integration point for the Brillouin zone. In systems where the translational symmetry is broken, e.g. disorder systems, liquids, or thermally excited crystals, periodic boundary conditions can still be used if combined with a supercell approach. Many systems investigated with the here described method fall into these categories, and therfore most calculations using the Car-Parrinello molecular dynamics approach are using supercells and a single **k**–point "integration scheme". The only point calculated is the center of the Brillouin zone (Γ–point; $\mathbf{k} = \mathbf{0}$). For the remainder of this chapter, all formulas are given for the Γ–point approximation.

3.4.2 Wavefunction Gradient

Analytic derivatives of the total energy with respect to the parameters of the calculation are needed for stable molecular dynamics calculations. All derivatives needed are easily accessible in the plane wave pseudopotential approach. In the following Fourier space formulas are presented

$$\frac{1}{f_i}\frac{\partial E_{\rm total}}{\partial c_i^\star(\mathbf{G})} = \frac{1}{2}\,G^2\,c_i(\mathbf{G}) \\ + \sum_{\mathbf{G}'} V_{\rm loc}^\star(\mathbf{G}-\mathbf{G}')c_i(\mathbf{G}') \\ + \sum_I \sum_{\alpha,\beta} \left(F_{I,i}^\alpha\right)^\star h_{\alpha\beta}^I P_\beta^I(\mathbf{G}) S_I(\mathbf{G}) \ , \tag{181}$$

where $V_{\rm loc}$ is the local potential

$$V_{\rm loc}(\mathbf{G}) = \sum_I \Delta V_{\rm local}^I(\mathbf{G}) S_I(\mathbf{G}) + V_{\rm xc}(\mathbf{G}) + 4\pi\,\frac{n_{\rm tot}(\mathbf{G})}{G^2} \ . \tag{182}$$

Wavefunction gradients are needed in optimization calculations and in the Car-Parrinello molecular dynamics approach.

3.4.3 Gradient for Nuclear Positions

The derivative of the total energy with respect to nuclear positions is needed for structure optimization and in molecular dynamics, that is

$$\frac{\partial E_{\text{total}}}{\partial \mathbf{R}_{I,s}} = \frac{\partial E^{\text{PP}}_{\text{local}}}{\partial \mathbf{R}_{I,s}} + \frac{\partial E^{\text{PP}}_{\text{nonlocal}}}{\partial \mathbf{R}_{I,s}} + \frac{\partial E_{\text{ES}}}{\partial \mathbf{R}_{I,s}} \ , \tag{183}$$

as the kinetic energy E_{kin} and the exchange and correlation energy E_{xc} do not depend directly on the atomic positions, the relevant parts are

$$\frac{\partial E^{\text{PP}}_{\text{local}}}{\partial \mathbf{R}_{I,s}} = -\Omega \sum_{\mathbf{G}} i\mathbf{G}_s \, \Delta V^I_{\text{local}}(\mathbf{G}) \, S_I(\mathbf{G}) \, n^\star(\mathbf{G}) \tag{184}$$

$$\frac{\partial E^{\text{PP}}_{\text{nonlocal}}}{\partial \mathbf{R}_{I,s}} = \sum_i f_i \sum_{\alpha,\beta \in I} \left\{ \left(F^\alpha_{I,i}\right)^\star h^I_{\alpha\beta} \left(\frac{\partial F^\beta_{I,i}}{\partial \mathbf{R}_{I,s}} \right) + \left(\frac{\partial F^\alpha_{I,i}}{\partial \mathbf{R}_{I,s}} \right)^\star h^I_{\alpha,\beta} F^\beta_{I,i} \right\} \tag{185}$$

$$\frac{\partial E_{\text{ES}}}{\partial \mathbf{R}_{I,s}} = -\,\Omega \sum_{\mathbf{G}\neq 0} i\mathbf{G}_s \, \frac{n^\star_{\text{tot}}(\mathbf{G})}{G^2} \, n^I_{\text{c}}(\mathbf{G}) \, S_I(\mathbf{G}) + \frac{\partial E_{\text{ovrl}}}{\partial \mathbf{R}_{I,s}} \ . \tag{186}$$

The contribution of the projectors of the non-local pseudopotentials is calculated from

$$\frac{\partial F^\alpha_{I,i}}{\partial \mathbf{R}_{I,s}} = -\frac{1}{\sqrt{\Omega}} \sum_{\mathbf{G}} i\mathbf{G}_s \, P^I_\alpha(\mathbf{G}) \, S_I(\mathbf{G}) \, c^\star_i(\mathbf{G},\mathbf{k}) \ . \tag{187}$$

Finally, the real space part contribution of the Ewald sum is

$$\begin{aligned} \frac{\partial E_{\text{ovrl}}}{\partial \mathbf{R}_{I,s}} = {\sum_J}' \sum_{\mathbf{L}} \Bigg\{ & \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|^3} \text{erfc}\left[\frac{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|}{\sqrt{\text{R}^{\text{c}\,2}_I + \text{R}^{\text{c}\,2}_J}} \right] \\ & + \frac{2}{\sqrt{\pi}} \frac{1}{\sqrt{\text{R}^{\text{c}\,2}_I + \text{R}^{\text{c}\,2}_J}} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|^2} \exp\left[-\frac{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|^2}{\sqrt{\text{R}^{\text{c}\,2}_I + \text{R}^{\text{c}\,2}_J}} \right] \Bigg\} \\ & \times (\mathbf{R}_{I,s} - \mathbf{R}_{J,s} - \mathbf{L}_s) \ . \end{aligned} \tag{188}$$

The self energy E_{self} is independent of the atomic positions and does not contribute to the forces.

3.4.4 Internal Stress Tensor

For calculations where the supercell is changed (e.g. the combination of the Car–Parrinello method with the Parrinello–Rahman approach [201,55]) the electronic internal stress tensor is required. The electronic part of the internal stress tensor is defined as [440,441] (see also Sect. 4.2.3)

$$\Pi_{uv} = -\frac{1}{\Omega} \sum_s \frac{\partial E_{\text{total}}}{\partial \mathbf{h}_{us}} \mathbf{h}^{\text{t}}_{sv} \ . \tag{189}$$

An important identity for the derivation of the stress tensor is

$$\frac{\partial \Omega}{\partial \mathbf{h}_{uv}} = \Omega (\mathbf{h}^{\text{t}})^{-1}_{uv} \ . \tag{190}$$

The derivatives of the total energy with respect to the components of the cell matrix $\mathbf{h}$ can be performed on every part of the total energy individually,

$$\frac{\partial E_{\text{total}}}{\partial \mathbf{h}_{uv}} = \frac{\partial E_{\text{kin}}}{\partial \mathbf{h}_{uv}} + \frac{\partial E^{\text{PP}}_{\text{local}}}{\partial \mathbf{h}_{uv}} + \frac{\partial E^{\text{PP}}_{\text{nonlocal}}}{\partial \mathbf{h}_{uv}} + \frac{\partial E_{\text{xc}}}{\partial \mathbf{h}_{uv}} + \frac{\partial E_{\text{ES}}}{\partial \mathbf{h}_{uv}} \ . \tag{191}$$

Using Eq. (190) extensively, the derivatives can be calculated for the case of a plane wave basis in Fourier space [202],

$$\frac{\partial E_{\text{kin}}}{\partial \mathbf{h}_{uv}} = -\sum_i f_i \sum_{\mathbf{G}} \sum_s \mathbf{G}_u \mathbf{G}_s (\mathbf{h}^{\text{t}})^{-1}_{sv} |c_i(\mathbf{G})|^2 \tag{192}$$

$$\frac{\partial E^{\text{PP}}_{\text{local}}}{\partial \mathbf{h}_{uv}} = \Omega \sum_I \sum_{\mathbf{G}} \left(\frac{\partial \Delta V^I_{\text{local}}(\mathbf{G})}{\partial h_{uv}} \right) S_I(\mathbf{G}) n^\star(\mathbf{G}) \tag{193}$$

$$\frac{\partial E^{\text{PP}}_{\text{nonlocal}}}{\partial \mathbf{h}_{uv}} = \sum_i f_i \sum_I \sum_{\alpha,\beta \in I} \left\{ \left(F^\alpha_{I,i}\right)^\star h^I_{\alpha\beta} \left(\frac{\partial F^\beta_{I,i}}{\partial \mathbf{h}_{uv}} \right) + \left(\frac{\partial F^\alpha_{I,i}}{\partial \mathbf{h}_{uv}} \right)^\star h^I_{\alpha,\beta} F^\beta_{I,i} \right\} \tag{194}$$

$$\begin{aligned} \frac{\partial E_{\text{xc}}}{\partial \mathbf{h}_{uv}} = & -\sum_{\mathbf{G}} n^\star(\mathbf{G}) \left[V_{\text{xc}}(\mathbf{G}) - \varepsilon_{\text{xc}}(\mathbf{G}) \right] (\mathbf{h}^{\text{t}})^{-1}_{uv} \\ & + \sum_s \sum_{\mathbf{G}} i \mathbf{G}_u n^\star(\mathbf{G}) \left(\frac{\partial F_{\text{xc}}(\mathbf{G})}{\partial(\partial_s n)} \right) (\mathbf{h}^{\text{t}})^{-1}_{sv} \end{aligned} \tag{195}$$

$$\begin{aligned} \frac{\partial E_{\text{ES}}}{\partial \mathbf{h}_{uv}} = & \, 2\pi\,\Omega \sum_{\mathbf{G}\neq 0} \sum_s \left\{ -\frac{|n_{\text{tot}}(\mathbf{G})|^2}{G^2} \delta_{us} \right. \\ & + \frac{n^\star_{\text{tot}}(\mathbf{G})}{G^2} \left(\frac{n_{\text{tot}}(\mathbf{G})}{G^2} \right. \\ & \left. \left. + \frac{1}{2} \sum_I n^I_{\text{c}}(\mathbf{G}) (R^I_{\text{c}})^2 \right) \mathbf{G}_u \mathbf{G}_s \right\} \mathbf{G}_u \mathbf{G}_s (\mathbf{h}^{\text{t}})^{-1}_{sv} \\ & + \frac{\partial E_{\text{ovrl}}}{\partial \mathbf{h}_{uv}} \ . \end{aligned} \tag{196}$$

Finally, the derivative of the overlap contribution to the electrostatic energy is

$$\begin{aligned} \frac{\partial E_{\text{ovrl}}}{\partial \mathbf{h}_{uv}} = & -{\sum_{I,J}}' \sum_{\mathbf{L}} \left\{ \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|^3} \text{erfc} \left[\frac{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|}{\sqrt{\text{R}^{\text{c}\,2}_I + \text{R}^{\text{c}\,2}_J}} \right] \right. \\ & \left. + \frac{2}{\sqrt{\pi}\sqrt{\text{R}^{\text{c}\,2}_I + \text{R}^{\text{c}\,2}_J}} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|^2} \exp \left[-\frac{|\mathbf{R}_I - \mathbf{R}_J - \mathbf{L}|^2}{\sqrt{\text{R}^{\text{c}\,2}_I + \text{R}^{\text{c}\,2}_J}} \right] \right\} \\ & \times \sum_s (\mathbf{R}_{I,u} - \mathbf{R}_{J,u} - \mathbf{L}_u)(\mathbf{R}_{I,s} - \mathbf{R}_{J,s} - \mathbf{L}_s)(\mathbf{h}^{\text{t}})^{-1}_{sv} \ . \end{aligned} \tag{197}$$

The local part of the pseudopotential $\Delta V^I_{\text{local}}(\mathbf{G})$ and the nonlocal projector functions depend on the cell matrix $\mathbf{h}$ through the volume, the Bessel transform integral and the spherical harmonics function. Their derivatives are lengthy but easy to cal-

culate from their definitions Eqs. (140) and (141)

$$\frac{\partial \Delta V^I_{\rm local}(\mathbf{G})}{\partial \mathbf{h}_{uv}} = -\Delta V^I_{\rm local}(\mathbf{G})(\mathbf{h}^{\rm t})^{-1}_{uv} + \frac{4\pi}{\Omega}\int_0^\infty dr\, r^2\, \Delta V_{\rm local}(r) \left(\frac{\partial j_0(Gr)}{\partial \mathbf{h}_{uv}}\right) Y_{lm}(\tilde{\theta},\tilde{\phi}) \tag{198}$$

$$\frac{\partial F^\alpha_{I,i}}{\partial \mathbf{h}_{uv}} = \frac{4\pi}{\sqrt{\Omega}}\,(-i)^l \sum_{\mathbf{G}} c^\star_i(\mathbf{G})\, S_I(\mathbf{G}) \left[\left(\frac{\partial Y_{lm}(\tilde{\theta},\tilde{\phi})}{\partial \mathbf{h}_{uv}} - \frac{1}{2} Y_{lm}(\tilde{\theta},\tilde{\phi})(\mathbf{h}^{\rm t})^{-1}_{uv}\right)\int_0^\infty dr\, r^2\, P^I_\alpha(r)\, j_l(Gr) + Y_{lm}(\tilde{\theta},\tilde{\phi})\int_0^\infty dr\, r^2\, P^I_\alpha(r)\, \left(\frac{\partial j_l(Gr)}{\partial \mathbf{h}_{uv}}\right)\right] \ . \tag{199}$$

3.4.5 Non-linear Core Correction

The success of pseudopotentials in density functional calculations relies on two assumptions. The transferability of the core electrons to different environments and the linearization of the exchange and correlation energy. The second assumption is only valid if the frozen core electrons and the valence state do not overlap. However, if there is significant overlap between core and valence densities, the linearization will lead to reduced transferability and systematic errors. The most straightforward remedy is to include "semi–core states" in addition to the valence shell, i.e. one more inner shell (which is from a chemical viewpoint an inert "core level") is treated explicitely. This approach, however, leads to quite hard pseudopotentials which call for large plane wave cutoffs. Alternatively, it was proposed to treat the non–linear parts of the exchange and correlation energy $E_{\rm xc}$ explicitely [374]. This idea does not lead to an increase of the cutoff but ameliorates the above–mentioned problems quite a bit. To achieve this, $E_{\rm xc}$ is calculated not from the valence density $n(\mathbf{R})$ alone, but from a modified density

$$\tilde{n}(\mathbf{R}) = n(\mathbf{R}) + \tilde{n}_{\rm core}(\mathbf{R}) \ , \tag{200}$$

where $\tilde{n}_{core}(\mathbf{R})$ denotes a density that is equal to the core density of the atomic reference state in the region of overlap with the valence density

$$\tilde{n}_{\rm core}(r) = n_{\rm core}(r) \quad \text{if } r > r_0 \ ; \tag{201}$$

with the vanishing valence density inside r_0. Close to the nuclei a model density is chosen in order to reduce the cutoff for the plane wave expansion. Finally, the two densities and their derivatives are matched at r_0. This procedure leads to a modified total energy in Eq. (176), where $E_{\rm xc}$ is replace by

$$E_{\rm xc} = E_{\rm xc}(n + \tilde{n}_{\rm core}) \ , \tag{202}$$

and the corresponding potential is

$$V_{\rm xc} = V_{\rm xc}(n + \tilde{n}_{\rm core}) \ . \tag{203}$$

The sum of all modified core densities

$$\tilde{n}_{core}(\mathbf{G}) = \sum_I \tilde{n}^I_{core}(\mathbf{G}) S_I(\mathbf{G}) \tag{204}$$

depends on the nuclear positions, leading to a new contribution to the forces

$$\frac{\partial E_{\rm xc}}{\partial \mathbf{R}_{I,s}} = -\Omega \sum_{\mathbf{G}} i\mathbf{G}_s V^\star_{\rm xc}(\mathbf{G}) \tilde{n}^I_{\rm core}(\mathbf{G}) S_I(\mathbf{G}) \ , \tag{205}$$

and to the stress tensor

$$\frac{\partial E_{\rm xc}}{\partial \mathbf{h}_{uv}} = \sum_I \sum_{\mathbf{G}} V^\star_{\rm xc}(\mathbf{G}) \frac{\partial \tilde{n}^I_{\rm core}(\mathbf{G})}{\partial \mathbf{h}_{uv}} S_I(\mathbf{G}) \ . \tag{206}$$

The derivative of the core charge with respect to the cell matrix can be performed in analogy to the formula given for the local potential. The method of the non-linear core correction dramatically improves results on systems with alkali and transition metal atoms. For practical applications, one should keep in mind that the non-linear core correction should only be applied together with pseudopotentials that were generated using the same energy expression.

3.5 Energy and Force Calculations in Practice

In Sect. 3.4 formulas for the total energy and forces were given in their Fourier space representation. Many terms are in fact calculated most easily in this form, but some terms would require double sums over plane waves. In particular, the calculation of the charge density and the wavefunction gradient originating from the local potential

$$\sum_{\mathbf{G}'} V^\star_{\rm loc}(\mathbf{G} - \mathbf{G}') c_i(\mathbf{G}') \ . \tag{207}$$

The expression in Eq. (207) is a convolution and can be calculated efficiently by a series of Fourier transforms. The flow charts of this calculations are presented in Fig. 8. Both of these modules contain a Fourier transform of the wavefunctions from $\mathbf{G}$ space to the real space grid. In addition, the calculation of the wavefunction forces requires a back transform of the product of the local potential with the wavefunctions, performed on the real space grid, to Fourier space. This leads to a number of Fourier transforms that is three times the number of states in the system. If enough memory is available on the computer the second transform of the wavefunctions to the grid can be avoided if the wavefunctions are stored in real space during the computation of the density. These modules are further used in the flow chart of the calculation of the local potential in Fig. 9. Additional Fourier transforms are needed in this part of the calculation. However, the number of transforms does not scale with the number of electrons in the system. Additional transforms might be hidden in the module to calculate the exchange and correlation potential (see also Fig. 7) and the Poisson solver in cases when the Hockney method is used (see Fig. 6).

The calculation of the total energy, together with the local potential is shown in Fig. 10. The overlap between the projectors of the nonlocal pseudopotential and

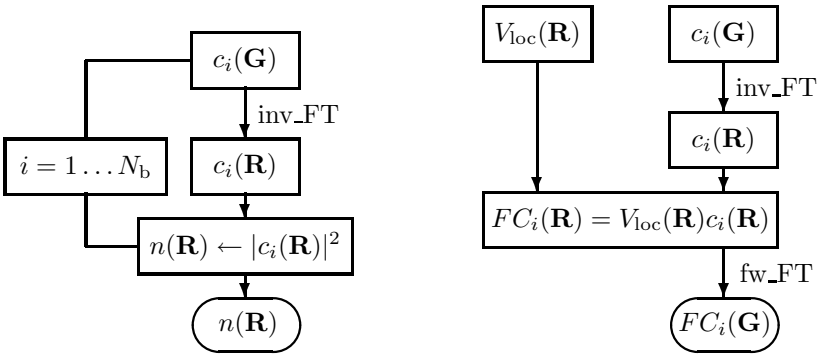


Figure 8. Flow chart for the calculation of the charge density (on the left) and the force on the wavefunction from the local potential (on the right). The charge density calculation requires N_b (number of states) three dimensional Fourier transforms. For the application of the local potential two Fourier transforms per state are needed. If enough memory is available the first transform can be avoided if the wavefunction on the real space grid are stored during the density calculation.

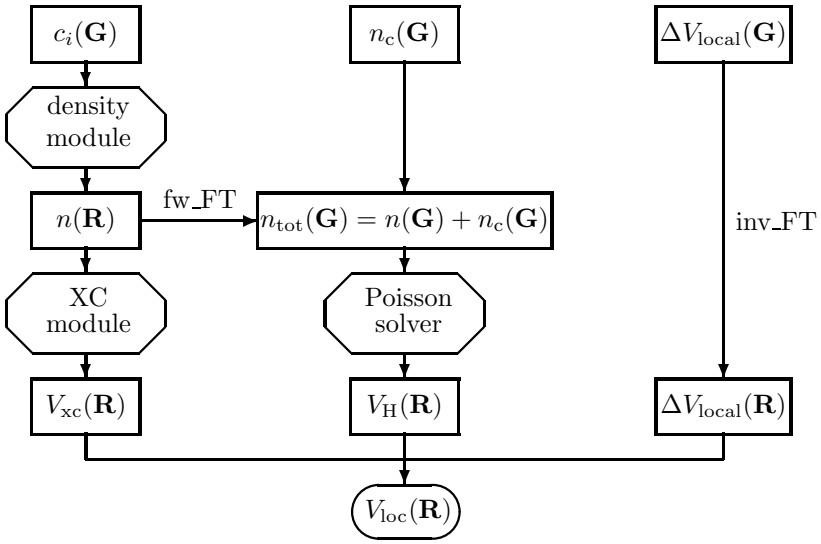


Figure 9. Flow chart for the calculation of the local potential from the Kohn–Sham orbitals. This module calculates also the charge density in real and Fourier space and the exchange and correlation energy, Hartree energy, and local pseudopotential energy.

the wavefunctions calculated in this part will be reused in the calculation of the forces on the wavefunctions. There are three initialization steps marked in Fig. 9. Step one has only to be performed at the beginning of the calculation, as the quantities $\mathbf{g}$ and E_self are constants. The quantities calculated in step two depend on the absolute value of the reciprocal space vectors. They have to be recalculated

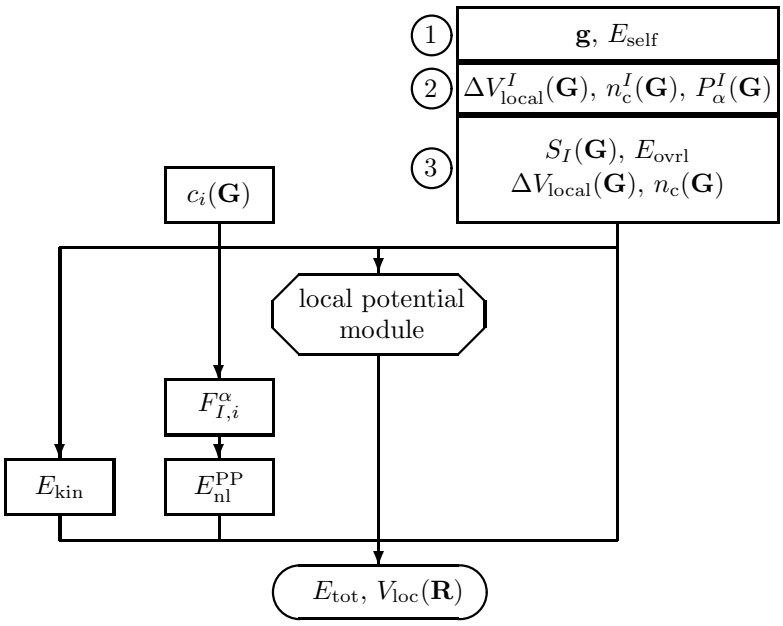


Figure 10. Flow chart for the calculation of the local potential and total energy. Initialization steps are marked with numbers. Step 2 has to be repeated whenever the size of the unit cell changes. Step 3 has to be repeated whenever nuclear positions have changed.

whenever the box matrix $\mathbf{h}$ changes. Finally, the variables in step three depend on the atomic positions and have to be calculated after each change of the nuclear positions. The flow charts of the calculation of the forces for the wavefunctions and the nuclei are shown in Figs. 11 and 12.

3.6 Optimizing the Kohn-Sham Orbitals

Advances in the application of plane wave based electronic structure methods are closely related to improved methods for the solution of the Kohn–Sham equations. There are now two different but equally successful approaches available. Fix–point methods based on the diagonalization of the Kohn–Sham matrix follow the more traditionally ways that go back to the roots of basis set methods in quantum chemistry. Direct nonlinear optimization approaches subject to a constraint were initiated by the success of the Car–Parrinello method. The following sections review some of these methods, focusing on the special problems related to the plane wave basis.

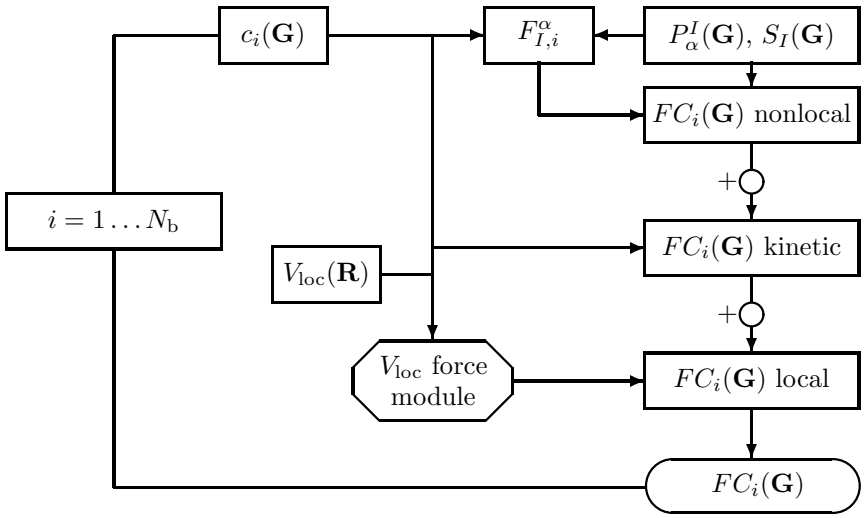


Figure 11. Flow chart for the calculation of the forces on the wavefunctions. Notice that the calculation of the overlap terms $F^{\alpha}_{I,i}$ is done outside the loop over wavefunctions. Besides the wavefunctions and the local potential, the structure factors and the projectors of the nonlocal pseudopotential are input parameters to this module.

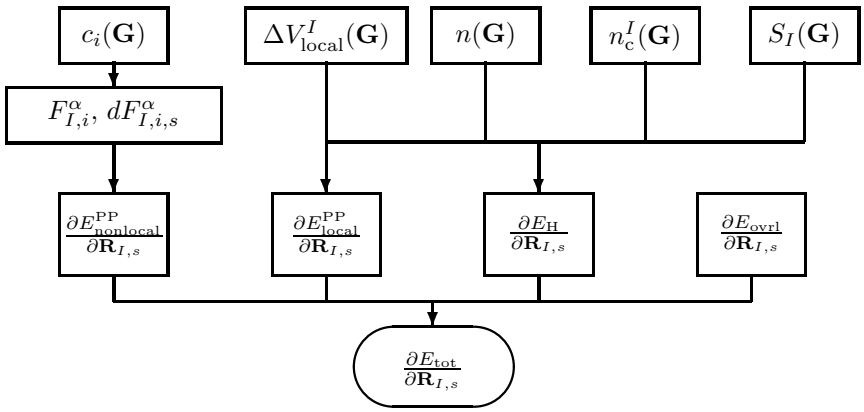


Figure 12. Flow chart for the calculation of the forces on the nuclei.

3.6.1 Initial Guess

The initial guess of the Kohn–Sham orbitals is the first step to a successful calculation. One would like to introduce as much knowledge as possible into the first step of the calculation, but at the same time the procedure should be general and robust. One should also take care not to introduce artifical symmetries that may be preserved during the optimization and lead to false results. The most general

initialization might be, choosing the wavefunction coefficients from a random distribution. It makes sense to weight the random numbers by a function reflecting the relative importance of different basis functions. A good choice is a Gaussian distribution in G^2. This initialization scheme avoids symmetry problems but leads to energies far off the final results and especially highly tuned optimization methods might have problems.

A more educated guess is to use a superposition of atomic densities and then diagonalize the Kohn–Sham matrix in an appropriate basis. This basis can be the full plane wave basis or just a part of it, or any other reasonable choice. The most natural choice of atomic densities and basis sets for a plane wave calculation are the pseudo atomic density and the pseudo atomic wavefunction of the atomic reference state used in the generation of the pseudopotential. In the `CPMD` code this is one possibility, but often the data needed are not available. For this case the default option is to generate a minimal basis out of Slater functions (see Eq. (98) in Sect. 2.8) and combine them with the help of atomic occupation numbers (gathered using the Aufbau principle) to an atomic density. From the superposition of these densities a Kohn–Sham potential is constructed. The Slater orbitals are expanded in plane waves and using the same routines as in the standard code the Kohn–Sham and overlap matrices are calculated in this basis. The general eigenvalue problem is solved and the eigenfunctions can easily be expressed in the plane wave basis that are in turn used as the initial wavefunctions to the optimization routines. Similarly, a given plane wave representation of the total wavefunction can be projected onto an auxiliary set of atom–centered functions. This opens up the possibility to perform population and bond–order analyses (following for instance the schemes of Mulliken or Mayer) in plane wave–pseudopotential calculations [537].

3.6.2 Preconditioning

Optimizations in many dimensions are often hampered by the appearance of different length scales. The introduction of a metric that brings all degrees of freedom onto the same length scale can improve convergence considerably. The application of such a metric is called "preconditioning" and is used in many optimization problems. If the variables in the optimization are decoupled the preconditioning matrix is diagonal and becomes computationally tractable even for very large systems. Fortunately, this is to a large degree the case for a plane wave basis set. For large $\mathbf{G}$ vectors the Kohn–Sham matrix is dominated by the kinetic energy which is diagonal in the plane wave representation. Based on this observation efficient preconditioning schemes have been proposed [616,610,308,344]. The preconditioner implemented in the `CPMD` code is based on the diagonal of the Kohn–Sham matrix $H_{\mathbf{G},\mathbf{G}'}$, which is given by

$$\begin{array}{ll} \mathbf{K}_{\mathbf{G},\mathbf{G}'} = \mathbf{H}_{\mathbf{G},\mathbf{G}}\ \delta_{\mathbf{G},\mathbf{G}'} & \text{if } |\mathbf{G}| \geq G_{\rm c} \\ \mathbf{K}_{\mathbf{G},\mathbf{G}'} = \mathbf{H}_{\mathbf{G}_{\rm c},\mathbf{G}_{\rm c}}\ \delta_{\mathbf{G},\mathbf{G}'} & \text{if } |\mathbf{G}| \leq G_{\rm c} \end{array} \quad , \tag{208}$$

where $G_{\rm c}$ is a free parameter that can be adjusted to accelerate convergence. However, it is found that the actual choice is not very critical and for practical purposes it is convenient not to fix $G_{\rm c}$, but to use an universal constant of 0.5 Hartree for $H_{\mathbf{G}_{\rm c},\mathbf{G}_{\rm c}}$ that in turn defines $G_{\rm c}$ for each system.

3.6.3 Direct Methods

The success of the Car–Parrinello approach started the interest in other methods for a direct minimization of the Kohn–Sham energy functional. These methods optimize the total energy using the gradient derived from the Lagrange function

$$\mathcal{L} = E^{\mathrm{KS}}\left(\{\Phi_i\}\right) - \sum_{ij} \lambda_{ij} \left(\langle \Phi_i | \Phi_j \rangle - \delta_{ij}\right) \tag{209}$$

$$\frac{\partial \mathcal{L}}{\partial \Phi_i} = \mathcal{H}_{\mathrm{e}} \Phi_i - \sum_j \langle \Phi_i | \mathcal{H}_{\mathrm{e}} | \Phi_j \rangle \Phi_j \ . \tag{210}$$

Optimization methods differ in the way orbitals are updated. A steepest descent based scheme

$$c_i(\mathbf{G}) \leftarrow c_i(\mathbf{G}) + \alpha \mathbf{K}^{-1}_{\mathbf{G},\mathbf{G}} \psi_i(\mathbf{G}) \tag{211}$$

can be combined with the preconditioner and a line search option to find the optimal step size α. Nearly optimal α's can be found with an interpolation based on a quadratic polynomial. In Eq. (211) $\psi_i(\mathbf{G})$ denote the Fourier components of the wavefunction gradient.
Improved convergence can be achieved by replacing the steepest descent step with a search direction based on conjugate gradients [594,232,616,23,499]

$$c_i(\mathbf{G}) \leftarrow c_i(\mathbf{G}) + \alpha h_i(\mathbf{G}) \ . \tag{212}$$

The conjugate directions are calculated from

$$h_i^{(n)}(\mathbf{G}) = \begin{cases} g_i^{(n)}(\mathbf{G}) & n = 0 \\ g_i^{(n)}(\mathbf{G}) + \gamma^{(n-1)} h_i^{n-1}(\mathbf{G}) & n = 1, 2, 3, \ldots \end{cases} \tag{213}$$

where

$$g_i^{(n)}(\mathbf{G}) = \mathbf{K}^{-1}_{\mathbf{G},\mathbf{G}} \psi_i^{(n)}(\mathbf{G}) \tag{214}$$

$$\gamma^{(n)} = \frac{\sum_i \langle g_i^{(n+1)}(\mathbf{G}) | g_i^{(n+1)}(\mathbf{G}) \rangle}{\langle g_i^{(n)}(\mathbf{G}) | g_i^{(n)}(\mathbf{G}) \rangle}. \tag{215}$$

A very efficient implementation of this method [616] is based on a band by band optimization. A detailed description of this method can also be found in Ref. [472].

The direct inversion in the iterative subspace (DIIS) method [495,144,308] is a very successful extrapolation method that can be used in any kind of optimization problems. In quantum chemistry the DIIS scheme has been applied to wavefunction optimizations, geometry optimizations and in post–Hartree–Fock applications. DIIS uses the information of n previous steps. Together with the position vectors $c_i^{(k)}(\mathbf{G})$ an estimate of the error vector $e_i^{(k)}(\mathbf{G})$ for each previous step k is stored. The best approximation to the final solution within the subspace spanned by the n stored vectors is obtained in a least square sense by writing

$$c_i^{(n+1)}(\mathbf{G}) = \sum_{k=1}^{n} d_k c_i^{(k)}(\mathbf{G}) \ , \tag{216}$$

where the d_k are subject to the restriction

$$\sum_{k=1}^{n} d_k = 1 \tag{217}$$

and the estimated error becomes

$$e_i^{(n+1)}(\mathbf{G}) = \sum_{k=1}^{n} d_k e_i^{(k)}(\mathbf{G}) \ . \tag{218}$$

The expansion coefficients d_k are calculated from a system of linear equations

$$\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} & 1 \\ b_{21} & b_{22} & \cdots & b_{2n} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} & 1 \\ 1 & 1 & \cdots & 1 & 0 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \\ -\lambda \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{219}$$

where the b_{kl} are given by

$$b_{kl} = \sum_i \langle e_i^k(\mathbf{G}) | e_i^l(\mathbf{G}) \rangle \ . \tag{220}$$

The error vectors are not known, but can be approximated within a quadratic model

$$e_i^{(k)}(\mathbf{G}) = -\mathbf{K}_{\mathbf{G},\mathbf{G}}^{-1} \psi_i^{(k)}(\mathbf{G}) \ . \tag{221}$$

In the same approximation, assuming $\mathbf{K}$ to be a constant, the new trial vectors are estimated to be

$$c_i(\mathbf{G}) = c_i^{(n+1)}(\mathbf{G}) + \mathbf{K}_{\mathbf{G},\mathbf{G}}^{-1} \psi_i^{(n+1)}(\mathbf{G}) \ , \tag{222}$$

where the first derivative of the energy density functional is estimated to be

$$\psi_i^{(n+1)}(\mathbf{G}) = \sum_{k=1}^{n} d_k \psi_i^{(k)}(\mathbf{G}) \ . \tag{223}$$

The methods described in this section produce new trail vectors that are not orthogonal. Therefore an orthogonalization step has to be added before the new charge density is calculated

$$c_i(\mathbf{G}) \leftarrow \sum_k c_j(\mathbf{G}) \mathbf{X}_{ji} \ . \tag{224}$$

There are different choices for the rotation matrix $\mathbf{X}$ that lead to orthogonal orbitals. Two of the computationally convenient choices are the Löwdin orthogonalization

$$\mathbf{X}_{ji} = \mathbf{S}_{ji}^{-1/2} \tag{225}$$

and a matrix form of the Gram–Schmidt procedure

$$\mathbf{X}_{ji} = (\mathbf{G}^T)_{ji}^{-1} \ , \tag{226}$$

where $\mathbf{S}$ is the overlap matrix and $\mathbf{G}$ is its Cholesky decomposition

$$\mathbf{S} = \mathbf{G}\mathbf{G}^T \quad . \tag{227}$$

Recently new methods that avoid the orthogonalization step have been introduced. One of them [483] relies on modified functionals that can be optimized without the orthogonality constraint. These functionals, originally introduced in the context of linear scaling methods [417,452], have the property that their minima coincide with the original Kohn–Sham energy functional. The methods described above can be used to optimize the new functional.

Another approach [309] is to use a variable transformation from the expansion coefficients of the orbitals in plane waves to a set of non–redundant orbital rotation angles. This method was introduced in quantum chemistry [618,149,167] and is used successfully in many optimization problems that involve a set of orthogonal orbitals. A generalization of the orbital rotation scheme allowed the application also for cases where the number of basis functions is orders of magnitudes bigger than the number of occupied orbitals. However, no advantage is gained over the standard methods, as the calculation of the gradient in the transformed variables scales the same as the orthogonalization step. In addition, there is no simple and efficient preconditioner available for the orbital rotation coordinates.

3.6.4 Fix-Point Methods

Originally all methods to find solutions to the Kohn–Sham equations were using matrix diagonalization methods. It became quickly clear that direct schemes can only be used for very small systems. The storage requirements of the Kohn–Sham matrix in the plane wave basis and the scaling proportional to the cube of the basis set size lead to unsurmountable problems. Iterative diagonalization schemes can be adapted to the special needs of a plane wave basis and when combined with a proper preconditioner lead to algorithms that are comparable to the direct methods, both in memory requirements and over all scaling properties. Iterative diagonalization schemes are abundant. Methods based on the Lanczos algorithm [357,151,489] can be used as well as conjugate gradient techniques [616,97]. Very good results have been achieved by the combination of the DIIS method with the minimization of the norm of the residual vector [698,344]. The diagonalization methods have to be combined with an optimization method for the charge density. Methods based on mixing [153,4], quasi-Newton algorithms [92,77,319], and DIIS [495,344,345] are successfully used. Also these methods use a preconditioning scheme. It was shown that the optimal preconditioning for charge density mixing is connected to the charge dielectric matrix [153,4,299,658,48]. For a plane wave basis, the charge dielectric matrix can be approximated by expressions very close to the ones used for the preconditioning in the direct optimization methods.

Fix-point methods have a slightly larger prefactor than most of the direct methods. Their advantage lies in the robustness and capability of treating systems with no or small band gaps.

3.7 Molecular Dynamics

Numerical methods to integrate the equations of motion are an important part of every molecular dynamics program. Therefore an extended literature exists on integration techniques (see Ref. [217] and references in there). All considerations valid for the integration of equations of motion with classical potentials also apply for *ab initio* molecular dynamics if the Born–Oppenheimer dynamics approach is used. These basic techniques will not be discussed here.

A good initial guess for the Kohn–Sham optimization procedure is a crucial ingredient for good performance of the Born–Oppenheimer dynamics approach. An extrapolation scheme was devised [24] that makes use of the optimized wavefunctions from previous time steps. This procedure has a strong connection to the basic idea of the Car–Parrinello method, but is not essential to the method.

The remainder of this section discusses the integration of the Car–Parrinello equations in their simplest form and explains the solution to the constraints equation for general geometric constraints. Finally, a special form of the equations of motion will be used for optimization purposes.

3.7.1 Car–Parrinello Equations

The Car–Parrinello Lagrangian and its derived equations of motions were introduced in Sect. 2.4. Here Eqs. (41), (44), and (45) are specialized to the case of a plane wave basis within Kohn–Sham density functional theory. Specifically the functions ϕ_i are replaced by the expansion coefficients $c_i(\mathbf{G})$ and the orthonormality constraint only depends on the wavefunctions, not the nuclear positions. The equations of motion for the Car–Parrinello method are derived from this specific extended Lagrangian

$$\mathcal{L} = \mu \sum_i \sum_{\mathbf{G}} |\dot{c}_i(\mathbf{G})|^2 + \frac{1}{2} \sum_I M_I \dot{\mathbf{R}}_I^2 - E_{\mathrm{KS}}\left[\{\mathbf{G}\}, \{\mathbf{R_I}\}\right] + \sum_{ij} \Lambda_{ij} \left(\sum_{\mathbf{G}} c_i^\star(\mathbf{G}) c_j(\mathbf{G}) - \delta_{ij} \right) \quad , \tag{228}$$

where μ is the electron mass, and M_I are the masses of the nuclei. Because of the expansion of the Kohn–Sham orbitals in plane waves, the orthonormality constraint does not depend on the nuclear positions. For basis sets that depend on the atomic positions (e.g. atomic orbital basis sets) or methods that introduce an atomic position dependent metric (ultra–soft pseudopotentials [661,351], PAW [143,347], the integration methods have to be adapted (see also Sect. 2.5). Solutions that include these cases can be found in the literature [280,351,143,310]. The Euler–Lagrange equations derived from Eq.(228) are

$$\mu \ddot{c}_i(\mathbf{G}) = -\frac{\partial E}{\partial c_i^\star(\mathbf{G})} + \sum_j \Lambda_{ij} c_j(\mathbf{G}) \tag{229}$$

$$M_I \ddot{\mathbf{R}}_I = -\frac{\partial E}{\partial \mathbf{R}_I}. \tag{230}$$

The two sets of equations are coupled through the Kohn–Sham energy functional and special care has to be taken for the integration because of the orthonormality constraint.

The integrator used in the CPMD code is based on the velocity Verlet / RATTLE algorithm [603,638,15]. The velocity Verlet algorithm requires more operations and more storage than the Verlet algorithm [664]. However, it is much easier to incorporate temperature control via velocity scaling into the velocity Verlet algorithm. In addition, velocity Verlet allows to change the time step trivially and is conceptually easier to handle [638,391]. It is defined by the following equations

$$
\begin{aligned}
\dot{\tilde{R}}_I(t+\delta t) &= \dot{\mathbf{R}}_I(t) + \frac{\delta t}{2M_I}\mathbf{F}_I(t) \qquad (231)\\
\mathbf{R}_I(t+\delta t) &= \mathbf{R}_I(t) + \delta t\, \dot{\tilde{R}}_I(t+\delta t)\\
\dot{\tilde{c}}_I(t+\delta t) &= \dot{\mathbf{c}}_I(t) + \frac{\delta t}{2\mu}\mathbf{f}_i(t)\\
\tilde{\mathbf{c}}_i(t+\delta t) &= \mathbf{c}_i(t) + \delta t\, \dot{\tilde{c}}_i(t+\delta t)\\
\mathbf{c}_i(t+\delta t) &= \tilde{\mathbf{c}}_i(t+\delta t) + \sum_j \mathbf{X}_{ij}\, \mathbf{c}_j(t)\\
\text{calculate} &\quad \mathbf{F}_I(t+\delta t)\\
\text{calculate} &\quad \mathbf{f}_i(t+\delta t)\\
\dot{\mathbf{R}}_I(t+\delta t) &= \dot{\tilde{R}}_I(t+\delta t) + \frac{\delta t}{2M_I}\mathbf{F}_I(t+\delta t)\\
\dot{\mathbf{c}'}_i(t+\delta t) &= \dot{\tilde{c}}_i(t+\delta t) + \frac{\delta t}{2\mu}\mathbf{f}_i(t+\delta t)\\
\dot{\mathbf{c}}_i(t+\delta t) &= \dot{\mathbf{c}'}_i(t+\delta t) + \sum_j \mathbf{Y}_{ij}\, \mathbf{c}_j(t+\delta t)\ ,
\end{aligned}
$$

where $\mathbf{R}_I(t)$ and $\mathbf{c}_i(t)$ are the atomic positions of particle I and the Kohn–Sham orbital i at time t respectively. Here, $\mathbf{F}_I$ are the forces on atom I, and $\mathbf{f}_i$ are the forces on Kohn–Sham orbital i. The matrices $\mathbf{X}$ and $\mathbf{Y}$ are directly related to the Lagrange multipliers by

$$\mathbf{X}_{ij} = \frac{\delta t^2}{2\mu}\Lambda^{\mathrm{p}}_{ij} \qquad (232)$$

$$\mathbf{Y}_{ij} = \frac{\delta t}{2\mu}\Lambda^{\mathrm{v}}_{ij}\ . \qquad (233)$$

Notice that in the RATTLE algorithm the Lagrange multipliers to enforce the orthonormality for the positions Λ^{p} and velocities Λ^{v} are treated as independent variables. Denoting with $\mathbf{C}$ the matrix of wavefunction coefficients $c_i(\mathbf{G})$, the or-

thonormality constraint can be written as

$$\mathbf{C}^\dagger(t+\delta t)\mathbf{C}(t+\delta t) - \mathbf{I} = 0 \tag{234}$$

$$\left[\tilde{\mathbf{C}} + \mathbf{X}\mathbf{C}\right]^\dagger \left[\tilde{\mathbf{C}} + \mathbf{X}\mathbf{C}\right] - \mathbf{I} = 0 \tag{235}$$

$$\tilde{\mathbf{C}}^\dagger\tilde{\mathbf{C}} + \mathbf{X}\tilde{\mathbf{C}}^\dagger\mathbf{C} + \mathbf{C}^\dagger\tilde{\mathbf{C}}\mathbf{X}^\dagger + \mathbf{X}\mathbf{X}^\dagger - \mathbf{I} = 0 \tag{236}$$

$$\mathbf{X}\mathbf{X}^\dagger + \mathbf{X}\mathbf{B} + \mathbf{B}^\dagger\mathbf{X}^\dagger = \mathbf{I} - \mathbf{A} \ , \tag{237}$$

where the new matrices $\mathbf{A}_{ij} = \tilde{\mathbf{c}}_i^\dagger(t+\delta t)\tilde{\mathbf{c}}_j(t+\delta t)$ and $\mathbf{B}_{ij} = \mathbf{c}_i^\dagger(t)\tilde{\mathbf{c}}_j(t+\delta t)$ have been introduced in Eq. (237). The unit matrix is denoted by the symbol $\mathbf{I}$. By noting that $\mathbf{A} = \mathbf{I} + \mathcal{O}(\delta t^2)$ and $\mathbf{B} = \mathbf{I} + \mathcal{O}(\delta t)$, Eq. (237) can be solved iteratively using

$$\mathbf{X}^{(n+1)} = \frac{1}{2}\Big[\mathbf{I} - \mathbf{A} + \mathbf{X}^{(n)}(\mathbf{I} - \mathbf{B}) + (\mathbf{I} - \mathbf{B})\mathbf{X}^{(n)} - \left(\mathbf{X}^{(n)}\right)^2\Big] \tag{238}$$

and starting from the initial guess

$$\mathbf{X}^{(0)} = \frac{1}{2}(\mathbf{I} - \mathbf{A}) \ . \tag{239}$$

In Eq. (238) it has been made use of the fact that the matrices $\mathbf{X}$ and $\mathbf{B}$ are real and symmetric, which follows directly from their definitions. Eq. (238) can usually be iterated to a tolerance of 10^{-6} within a few iterations.

The rotation matrix $\mathbf{Y}$ is calculated from the orthogonality condition on the orbital velocities

$$\dot{\mathbf{c}}_i^\dagger(t+\delta t)\mathbf{c}_j(t+\delta t) + \mathbf{c}_i^\dagger(t+\delta t)\dot{\mathbf{c}}_j(t+\delta t) = 0. \tag{240}$$

Applying Eq. (240) to the trial states $\dot{\mathbf{C}}' + \mathbf{Y}\mathbf{C}$ yields a simple equation for $\mathbf{Y}$

$$\mathbf{Y} = -\frac{1}{2}(\mathbf{Q} + \mathbf{Q}^\dagger), \tag{241}$$

where $\mathbf{Q}_{ij} = \mathbf{c}_i^\dagger(t+\delta t)\dot{\mathbf{c}'}_i^\dagger(t+\delta t)$. The fact that $\mathbf{Y}$ can be obtained without iteration means that the velocity constraint condition Eq. (240) is satisfied exactly at each time step.

3.7.2 Advanced Techniques

One advantage of the velocity Verlet integrator is that it can be easily combined with multiple time scale algorithms [636,639] and still results in reversible dynamics. The most successful implementation of a multiple time scale scheme in connection with the plane wave–pseudopotential method is the harmonic reference system idea [471,639]. The high frequency motion of the plane waves with large kinetic energy is used as a reference system for the integration. The dynamics of this reference system is harmonic and can be integrated analytically. In addition, this can be combined with the basic notion of a preconditioner already introduced in the section on optimizations. The electronic mass used in the Car–Parrinello scheme is a fictitious construct (see Sect. 2.4, Eq. (45)) and it is allowed to generalize the idea by

introducing different masses for different "classical" degrees of freedom [473,610,639]. In agreement with the preconditioner introduced in the optimization section, the new plane wave dependent masses are

$$\mu(\mathbf{G}) = \begin{cases} \mu & \mathbf{H}(\mathbf{G},\mathbf{G}) \leq \alpha \\ (\mu/\alpha)\ (\frac{1}{2}G^2 + \mathbf{V}(\mathbf{G},\mathbf{G})) & \mathbf{H}(\mathbf{G},\mathbf{G}) \geq \alpha \end{cases} , \tag{242}$$

where $\mathbf{H}$ and $\mathbf{V}$ are the matrix elements of the Kohn–Sham matrix and the potential respectively. The reference electron mass is μ and the parameter α has been introduced before in Eq. (208) as $\mathbf{H}_{\mathbf{G_c},\mathbf{G_c}}$. With the preconditioned masses and the harmonic reference system, the equations of motion of the system are

$$\mu(\mathbf{G})\ddot{c}_i(\mathbf{G}) = -\lambda(\mathbf{G})c_i(\mathbf{G}) + \delta\Phi_i(\mathbf{G}) + \sum_j \Lambda_{ij}c_j(\mathbf{G}) \ . \tag{243}$$

where $\delta\Phi_i(\mathbf{G})$ is the force on orbital i minus $-\lambda(\mathbf{G})$. From Eq. (243) it is easy to see that the frequencies $\omega(\mathbf{G}) = \sqrt{\lambda(\mathbf{G})/\mu(\mathbf{G})}$ are independent of $\mathbf{G}$ and that there is only one harmonic frequency equal to $\sqrt{\alpha/\mu}$. The revised formulas for the integration of the equations of motion for the velocity Verlet algorithm can be found in the literature [639].

The implications of the $\mathbf{G}$ vector dependent masses can be seen by revisiting the formulas for the characteristic frequencies of the electronic system Eqs. (52), (53), and (54). The masses μ are chosen such that all frequencies ω_{ij} are approximately the same, thus optimizing both, adiabaticity and maximal time step. The disadvantage of this method is that the average electron mass seen by the nuclei is drastically enhanced, leading to renormalization corrections [75] on the masses M_I that are significantly higher than in the standard approach and not as simple to estimate by an analytical expression.

3.7.3 Geometrical Constraints

Geometrical constraints are used in classical simulations to freeze fast degrees of freedom in order to allow for larger time steps. Mainly distance constraints are used for instance to fix intramolecular covalent bonds. These type of applications of constraints is of lesser importance in *ab initio* molecular dynamics. However, in the simulation of rare events such as many reactions, constraints play an important role together with the method of thermodynamic integration [217]. The "blue–moon" ensemble method [115,589] enables one to compute the potential of mean force. This potential can be obtained directly from the average force of constraint and a geometric correction term during a molecular dynamics simulation as follows:

$$\mathcal{F}(\xi_2) - \mathcal{F}(\xi_1) = \int_{\xi_1}^{\xi_2} d\xi' \left\langle \frac{\partial \mathcal{H}}{\partial \xi} \right\rangle_{\xi'}^{\text{cond.}} , \tag{244}$$

where $\mathcal{F}$ is the free energy and $\xi(\mathbf{r})$ a one–dimensional reaction coordinate, $\mathcal{H}$ the Hamiltonian of the system and $\langle \cdots \rangle_{\xi'}^{\text{cond.}}$ the conditioned average in the constraint ensemble [589]. By way of the blue moon ensemble, the statistical average is replaced by a time average over a constrained trajectory with the reaction coordinate fixed

at special values, $\xi(\mathbf{R}) = \xi'$, and $\dot{\xi}(\mathbf{R}, \dot{\mathbf{R}}) = 0$. The quantity to evaluate is the mean force

$$\frac{d\mathcal{F}}{d\xi'} = \frac{\left\langle \mathrm{Z}^{-1/2} \left[-\lambda + k_{\mathrm{B}} T \mathrm{G} \right] \right\rangle_{\xi'}}{\left\langle \mathrm{Z}^{-1/2} \right\rangle_{\xi'}} \ , \tag{245}$$

where λ is the Lagrange multiplier of the constraint,

$$\mathrm{Z} = \sum_I \frac{1}{M_I} \left(\frac{\partial \xi}{\partial \mathbf{R}_I} \right)^2 \ , \tag{246}$$

and

$$\mathrm{G} = \frac{1}{\mathrm{Z}^2} \sum_{I,J} \frac{1}{M_I M_J} \frac{\partial \xi}{\partial \mathbf{R}_I} \cdot \frac{\partial^2 \xi}{\partial \mathbf{R}_I \partial \mathbf{R}_J} \cdot \frac{\partial \xi}{\partial \mathbf{R}_J} \ , \tag{247}$$

where $\langle \cdots \rangle_{\xi'}$ is the unconditioned average, as directly obtained from a constrained molecular dynamics run with $\xi(\mathbf{R}) = \xi'$ and

$$\mathcal{F}(\xi_2) - \mathcal{F}(\xi_1) = \int_{\xi_1}^{\xi_2} d\xi' \frac{d\mathcal{F}}{d\xi'} \tag{248}$$

finally defines the free energy difference. For the special case of a simple distance constraint $\xi(\mathbf{R}) = |\mathbf{R}_I - \mathbf{R}_J|$ the parameter Z is a constant and $\mathrm{G} = 0$.

The RATTLE algorithm, allows for the calculation of the Lagrange multiplier of arbitrary constraints on geometrical variables within the velocity Verlet integrator. The following algorithm is implemented in the `CPMD` code. The constraints are defined by

$$\sigma^{(i)}(\{\mathbf{R}_I(t)\}) = 0 \ , \tag{249}$$

and the velocity Verlet algorithm can be performed with the following steps.

$$\begin{aligned}
\dot{\tilde{R}}_I &= \dot{\mathbf{R}}_I(t) + \frac{\delta t}{2M_I} \mathbf{F}_I(t) \\
\tilde{\mathbf{R}}_I &= \mathbf{R}_I(t) + \delta t \, \dot{\tilde{R}}_I \\
\mathbf{R}_I(t+\delta t) &= \tilde{\mathbf{R}}_I + \frac{\delta t^2}{2M_I} \mathbf{g}_{\mathrm{p}}(t) \\
\text{calculate} \quad & \mathbf{F}_I(t+\delta t) \\
\dot{\mathbf{R}}'_I &= \dot{\tilde{R}}_I + \frac{\delta t}{2M_I} \mathbf{F}_I(t+\delta t) \\
\dot{\mathbf{R}}_I(t+\delta t) &= \dot{\mathbf{R}}'_I + \frac{\delta t}{2M_I} \mathbf{g}_{\mathrm{v}}(t+\delta t) \ ,
\end{aligned}$$

where the constraint forces are defined by

$$\mathbf{g}_{\mathrm{p}}(t) = -\sum_i \lambda_{\mathrm{p}}^i \frac{\partial \sigma^{(i)}(\{\mathbf{R}_I(t)\})}{\partial \mathbf{R}_I} \tag{250}$$

$$\mathbf{g}_{\mathrm{v}}(t) = -\sum_i \lambda_{\mathrm{v}}^i \frac{\partial \sigma^{(i)}(\{\mathbf{R}_I(t)\})}{\partial \mathbf{R}_I} \ . \tag{251}$$

The Lagrange multiplier have to be determined to ensure that the constraint on the positions and velocities are exactly fulfilled at the end of the time step. For the position, the constraint condition is

$$\sigma^{(i)}(\{\mathbf{R}_I(t+\delta t)\}) = 0 \ . \tag{252}$$

Eq. (252) is in general a system of nonlinear equations in the Lagrange multipliers $\lambda^i_{\rm p}$. These equations can be solved using a generalized Newton algorithm [491] that can be combined with a convergence acceleration scheme based on the direct inversion in the iterative subspace method [495,144]. The error vectors for a given set of Lagrange multipliers λ are calculated from

$$\mathbf{e}_i(\lambda) = -\sum_j \mathbf{J}^{-1}_{ij}(\lambda)\sigma^{(j)}(\lambda) \ . \tag{253}$$

The Jacobian $\mathbf{J}$ is defined by

$$\mathbf{J}_{ij}(\lambda) = \frac{\partial\sigma^{(i)}(\lambda)}{\partial\lambda^j} \tag{254}$$

$$= \sum_I \frac{\partial\sigma^{(i)}(\lambda)}{\partial\mathbf{R}_I(\lambda)}\frac{\partial\mathbf{R}_I(\lambda)}{\partial\lambda^j} \tag{255}$$

$$= -\sum_I \frac{\delta t^2}{2M_I}\mathbf{f}^{\rm c}_I(\lambda)\mathbf{f}^{\rm c}_I(0) \ , \tag{256}$$

where $\mathbf{f}^{\rm c}_I(\lambda) = \sum_i \lambda^i \partial\sigma^{(i)}/\partial\mathbf{R}_I$. Typically only a few iterations are needed to converge the Lagrange multipliers to an accuracy of 1×10^{-8}.

The constraint condition for the velocities can be cast into a system of linear equations. Again, as in the case of the orthonormality constraints in the Car–Parrinello method, the Lagrange multiplier for the velocity update can be calculated exactly without making use of an iterative scheme. Defining the derivative matrix

$$\mathbf{A}_{iI} = \frac{\partial\sigma^{(i)}}{\partial\mathbf{R}_I} \ , \tag{257}$$

the velocity constraints are

$$\dot\sigma^{(i)}(t+\delta t) = 0 \tag{258}$$

$$\sum_I \frac{\partial\sigma^{(i)}}{\partial\mathbf{R}_I}\dot{\mathbf{R}}_I = 0 \tag{259}$$

$$-\sum_j\left(\sum_I \frac{\delta t^2}{2M_I}\mathbf{A}_{iI}\mathbf{A}_{jI}\right)\lambda^{\rm v}_j = \sum_I \mathbf{A}_{iI}\dot{\mathbf{R}}'_I \ . \tag{260}$$

The only information needed to implement a new type of constraint are the formulas for the functional value and its derivative with respect to the nuclear coordinates involved in the constraint.

3.7.4 Using Car-Parrinello Dynamics for Optimizations

By adding a friction term, Car–Parrinello molecular dynamics can be turned into a damped second order dynamics scheme (see also Sect. 2.4.6).

The friction can be applied both to the nuclear degrees of freedom and the electronic coordinates. The resulting dynamics equation are a powerful method to simultaneously optimize the atomic structure and the Kohn–Sham orbitals [472,610]. Harmonic reference system integration and plane wave dependent electron masses, introduced above, are especially helpful in this context, as the derived dynamics does not have a direct physical relevance.

Introducing a friction force proportional to the constants $\gamma_{\rm n}$ and $\gamma_{\rm e}$ the equations of motion can readily be integrated using the velocity Verlet algorithm. The friction terms translate into a simple rescaling of the velocities at the beginning and end of the time step according to

$$\begin{aligned}
\dot{\mathbf{R}}_I(t) &= \gamma_{\rm n}\dot{\mathbf{R}}_I(t) \\
\dot{\mathbf{c}}_i(t) &= \gamma_{\rm e}\dot{\mathbf{c}}_i(t) \\
\texttt{VELOCITY} &\quad \texttt{VERLET UPDATE} \\
\dot{\mathbf{R}}_I(t+\delta t) &= \gamma_{\rm n}\dot{\mathbf{R}}_I(t+\delta t) \\
\dot{\mathbf{c}}_i(t+\delta t) &= \gamma_{\rm e}\dot{\mathbf{c}}_i(t+\delta t) \ .
\end{aligned}$$

It was shown [472,610] that this scheme leads to optimizations that are competitive with other methods described in Sect. 3.6

3.8 Data Structures and Computational Kernels

In the practical implementation of the method, mathematical symbols have to be translated into data structures of the computer language. Then mathematical formulas are set into computer code using the data structures. The layout of the data structures should be such that optimal performance for the algorithms can be achieved. The `CPMD` code is written in FORTRAN77 and in the following sections the most important data structures and computational kernels will be given in pseudo code form. The following variables are used to denote quantities that measure system size.

$N_{\rm at}$	number of atoms
$N_{\rm p}$	number of projectors
$N_{\rm b}$	number of electronic bands or states
$N_{\rm PW}$	number of plane-waves
$N_{\rm D}$	number of plane-waves for densities and potentials
$N_{\rm x}$, $N_{\rm y}$, $N_{\rm z}$	number of grid points in x, y, and z direction
$N = N_{\rm x}N_{\rm y}N_{\rm z}$	total number of grid points

In Table 3 the relative size of this variables are given for two systems. The example for a silicon crystal assumes an energy cutoff of 13 Rydberg and s non-locality for the pseudopotential. In the example of a water system the numbers are given per molecule. The cutoff used was 70 Rydberg and the oxygen pseudopotential has a s nonlocal part, the hydrogen pseudopotential is local.

Table 3. Relative size of characteristic variables in a plane wave calculation. See text for details.

	silicon	water
$N_{\rm at}$	1	3
$N_{\rm p}$	1	1
$N_{\rm b}$	2	4
$N_{\rm PW}$	53	1000
$N_{\rm D}$	429	8000
N	1728	31250

3.8.1 `CPMD` *Program: Data Structures*

Important quantities in the pseudopotential plane–wave method depend either not at all, linearly, or quadratically on the system size. Examples for the first kind of data are the unit cell matrix $\mathbf{h}$ and the cutoff $E_{\rm cut}$. Variables with a size that grows linearly with the system are

`r(3,`N_{at}`)`	nuclear positions
`v(3,`N_{at}`)`	nuclear velocities
`f(3,`N_{at}`)`	nuclear forces
`g(3,`N_{PW}`)`	plane–wave indices
`ipg(3,`N_{PW}`)`	mapping of G–vectors (positive part)
`img(3,`N_{PW}`)`	mapping of G–vectors (negative part)
`rhog(`N_{PW}`)`	densities (n, $n_{\rm c}$, $n_{\rm tot}$) in Fourier–space
`vpot(`N_{PW}`)`	potentials ($V_{\rm loc}$, $V_{\rm xc}$, $V_{\rm H}$) in Fourier–space
`n(`N_x`,`N_y`,`N_z`)`	densities (n, $n_{\rm c}$, $n_{\rm tot}$) in real–space
`v(`N_x`,`N_y`,`N_z`)`	potentials ($V_{\rm loc}$, $V_{\rm xc}$, $V_{\rm H}$) in real–space
`vps(`N_D`)`	local pseudopotential
`rpc(`N_D`)`	core charges
`pro(`N_{PW}`)`	projectors of non-local pseudopotential.

The pseudopotential related quantities `vps`, `rpc`, and `pro` are one–dimensional in system size but also depend on the number of different atomic species. In the following it is assumed that this is one. It is easy to generalize the pseudo codes given to more than one atomic species. For real quantities that depend on G–vectors only half of the values have to be stored. The other half can be recomputed when needed by using the symmetry relation

$$A(\mathbf{G}) = A^{\star}(-\mathbf{G}) \ . \tag{261}$$

This saves a factor of two in memory. In addition $\mathbf{G}$ vectors are stored in a linear array, instead of a three-dimensional structure. This allows to store only non–zero variables. Because there is a spherical cutoff, another reduction of a factor of two is achieved for the memory. For the Fourier transforms the variables have to be prepared in a three-dimensional array. The mapping of the linear array to this structure is provided by the information stored in the arrays `ipg` and `img`.
Most of the memory is needed for the storage of quantities that grow quadratically

with system size.

`eigr`(N_D,N_{at})	structure factors
`fnl`(N_p,N_b)	overlap of projectors and bands
`dfnl`(N_p,N_b,3)	derivative of `fnl`
`smat`(N_b,N_b)	overlap matrices between bands
`cr`(N_{PW},N_b)	bands in Fourier space
`cv`(N_{PW},N_b)	velocity of bands in Fourier space
`cf`(N_{PW},N_b)	forces of bands in Fourier space

In order to save memory it is possible to store the structure factors only for the **G** vectors of the wave function basis or even not to store them at all. However, this requires that the missing structure factors are recomputed whenever needed. The structure factors `eigr` and the wavefunction related quantities `cr, cv, cf` are complex numbers. Other quantities, like the local pseudopotential `vps`, the core charges `rpc`, and the projectors `pro` can be stored as real numbers if the factor $(-i)^l$ is excluded.

3.8.2 `CPMD` *Program: Computational Kernels*

Most of the calculations in a plane wave code are done in only a few kernel routines. These routines are given in this section using a pseudo code language. Where possible an implementation using basic linear algebra (BLAS) routines is given. The first kernel is the calculation of the structure factors. The exponential function of the structure factor separates in three parts along the directions $s_{\rm x}, s_{\rm y}, s_{\rm z}$.

```
MODULE StructureFactor
FOR i=1:N_at
   s(1:3) = 2 * PI * MATMUL[htm1(1:3,1:3),r(1:3,i)]
   dp(1:3) = CMPLX[COS[s(1:3)],SIN[s(1:3)]]
   dm(1:3) = CONJG[dp(1:3)]
   e(0,1:3,i) = 1
   FOR k=1:g_max
      e(k,1:3,i) = e(k-1,1:3,i) * dp
      e(-k,1:3,i) = e(-k+1,1:3,i) * dm
   END
   FOR j=0:N_D
      eigr(j,i) = e(g(1,j),1,i) * e(g(2,j),2,i) * e(g(3,j),3,i)
   END
END
```

In the module above `htm1` is the matrix $(\mathbf{h}^{\rm t})^{-1}$. One of the most important calculation is the inner product of two vectors in Fourier space. This kernel appears for example in the calculation of energies

$$e = \sum_{\mathbf{G}} A^{\star}(\mathbf{G}) B(\mathbf{G}) \ . \tag{262}$$

Making use of the fact that both functions are real the sum can be restricted to half

of the **G** vectors, and only real operations have to be performed. Approximately a factor of three in operations can be save this way. Special care has to be taken for the zero **G** vector. It is assumed that this plane wave component is stored in the first position of the arrays.

```
MODULE DotProduct
e = A(1) * B(1)
FOR i=2:N_D
   ar = REAL(A(i))
   ai = IMAG(A(i))
   br = REAL(B(i))
   bi = IMAG(B(i))
   e = e + 2 * (ar * br + ai * bi)
END
```

This loop structure is available in the BLAS library, optimized on most computer architectures. To use the BLAS routines for real variables, complex numbers have to be stored as two real numbers in contiguous memory locations.

```
e = A(1) * B(1) + 2 * sdot(2 * N_D - 2,A(2),1,B(2),1)
```

The calculation of overlap matrices between sets of vectors in real space is a important task in the orthogonalization step

$$S_{ij} = \sum_{\mathbf{G}} A_i^\star(\mathbf{G}) B_j(\mathbf{G}) \ . \tag{263}$$

It can be executed by using matrix multiply routines from the BLAS library. The special case of the zero **G** vector is handled by a routine that performs a rank 1 update of the final matrix.

```
MODULE Overlap
CALL SGEMM('T','N',N_b,N_b,2*N_PW,2,&
           & ca(1,1),2*N_PW,cb(1,1),2*N_PW,0,smat,N_b)
CALL SDER(N_b,N_b,-1,ca(1,1),2*N_PW,cb(1,1),2*N_PW,smat,N_b)
```

For a symmetric overlap additional time can be saved by using the symmetric matrix multiply routine. The overlap routines scale like $N_\mathrm{b}^2 N_\mathrm{PW}$. It is therefore very important to have an implementation of these parts that performs close to peek performance.

```
MODULE SymmetricOverlap
CALL SSYRK('U','T',N_b,2*N_PW,2,ca(1,1),2*N_PW,0,smat,N_b)
CALL SDER(N_b,N_b,-1,ca(1,1),2*N_PW,cb(1,1),2*N_PW,smat,N_b)
```

Another operation that scales as the overlap matrix calculations is the rotation of

a set of wavefunctions in Fourier space

$$B_i(\mathbf{G}) = \sum_j A_j(\mathbf{G}) S_{ji} \ . \tag{264}$$

Again this kernel can be executed by using the optimized BLAS matrix multiply routines.

```
MODULE Rotation
CALL SGEMM('N','N',2*N_PW,N_b,N_b,1,ca(1,1),2*N_PW,&
           & smat,N_b,0,cb(1,1),2*N_PW)
```

The overlap calculation of the projectors of the nonlinear pseudopotential with the wavefunctions in Fourier space scales as $N_\mathrm{p} N_\mathrm{b} N_\mathrm{PW}$. As the projectors are stored as real quantities, the imaginary prefactor and the structure factor have to be applied before the inner product can be calculated. The following pseudo code calculates M projectors at a time, making use of the special structure of the prefactor. This allows again to do all calculations with real quantities. The code assumes that the total number of projectors is a multiple of M. A generalization of the code to other cases is straightforward. By using batches of projectors the overlap can be calculated using matrix multiplies. The variable `lp(i)` holds the angular momentum of projector i.

```
MODULE FNL
FOR i=1:N_p,M
   IF (MOD(lp(i),2) == 0) THEN
      FOR j=0:M-1
         pf = -1**(lp(i+j)/2)
         FOR k=1:N_PW
            t = pro(k) * pf
            er = REAL[eigr(k,iat(i+j))]
            ei = IMAG[eigr(k,iat(i+j))]
            scr(k,j) = CMPLX[t * er,t * ei]
         END
      END
   ELSE
      FOR j=0:M-1
         pf = -1**(lp(i+j)/2+1)
         FOR k=1:N_PW
            t = pro(k) * pf
            er = REAL[eigr(k,iat(i+j))]
            ei = IMAG[eigr(k,iat(i+j))]
            scr(k,j) = CMPLX[-t * ei,t * er]
         END
      END
   END IF
   scr(1,0:M-1) = scr(1,0:M-1)/2
```

```
   CALL SGEMM('T','N',M,N_b,2*N_PW,2,&
              & scr(1,0),2*N_PW,cr(1,1),2*N_PW,0,fnl(i,1),N_p)
END
```

Fourier transform routines are assumed to work on complex data and return also arrays with complex numbers. The transform of data with the density cutoff is shown in the next two pseudo code sections. It is assumed that a three dimensional fast Fourier transform routine exists. This is in fact the case on most computers where optimized scientific libraries are available. The next two pseudo code segments show the transform of the charge density from Fourier space to real space and back.

```
MODULE INVFFT
scr(1:N_x,1:N_y,1:N_z) = 0
FOR i=1:N_D
   scr(ipg(1,i),ipg(2,i),ipg(3,i)) = rhog(i)
   scr(img(1,i),img(2,i),img(3,i)) = CONJG[rhog(i)]
END
CALL FFT3D("INV",scr)
n(1:N_x,1:N_y,1:N_z) = REAL[scr(1:N_x,1:N_y,1:N_z)]

MODULE FWFFT
scr(1:N_x,1:N_y,1:N_z) = n(1:N_x,1:N_y,1:N_z)
CALL FFT3D("FW",scr)
FOR i=1:N_D
   rhog(i) = scr(ipg(1,i),ipg(2,i),ipg(3,i))
END
```

Special kernels are presented for the calculation of the density and the application of the local potential. These are the implementation of the flow charts shown in Fig. 8. The operation count of these routines is $N_{\rm b}N\log[N]$. In most applications these routines take most of the computer time. Only for the biggest applications possible on todays computers the cubic scaling of the orthogonalization and the nonlocal pseudopotential become dominant. A small prefactor and the optimized implementation of the overlap are the reasons for this.

In the Fourier transforms of the wavefunction two properties are used to speed up the calculation. First, because the wavefunctions are real two transforms can be done at the same time, and second, the smaller cutoff of the wavefunctions can be used to avoid some parts of the transforms. The use of the sparsity in the Fourier transforms is not shown in the following modules. In an actual implementation a mask will be generated and only transforms allowed by this mask will be executed. Under optimal circumstances a gain of almost a factor of two can be achieved.

```
MODULE Density
rho(1:Nx,1:Ny,1:Nz) = 0
FOR i=1:Nb,2
   scr(1:Nx,1:Ny,1:Nz) = 0
   FOR j=1:NPW
      scr(ipg(1,i),ipg(2,i),ipg(3,i)) = c(j,i) + I * c(j,i+1)
      scr(img(1,i),img(2,i),img(3,i)) = CONJG[c(j,i) + I * c(j,i+1)]
   END
   CALL FFT3D("INV",scr)
   rho(1:Nx,1:Ny,1:Nz) = rho(1:Nx,1:Ny,1:Nz) + &
      & REAL[scr(1:Nx,1:Ny,1:Nz)]**2 + IMAG[scr(1:Nx,1:Ny,1:Nz)]**2
END

MODULE VPSI
FOR i=1:Nb,2
   scr(1:Nx,1:Ny,1:Nz) = 0
   FOR j=1:NPW
      scr(ipg(1,i),ipg(2,i),ipg(3,i)) = c(j,i) + I * c(j,i+1)
      scr(img(1,i),img(2,i),img(3,i)) = CONJG[c(j,i) + I * c(j,i+1)]
   END
   CALL FFT3D("INV",scr)
   scr(1:Nx,1:Ny,1:Nz) = scr(1:Nx,1:Ny,1:Nz) * &
      & vpot(1:Nx,1:Ny,1:Nz)
   CALL FFT3D("FW",scr)
   FOR j=1:NPW
      FP = scr(ipg(1,i),ipg(2,i),ipg(3,i)) &
           & + scr(img(1,i),img(2,i),img(3,i))
      FM = scr(ipg(1,i),ipg(2,i),ipg(3,i)) &
           & - scr(img(1,i),img(2,i),img(3,i))
      fc(j,i) = f(i) * CMPLX[REAL[FP],IMAG[FM]]
      fc(j,i+1) = f(i+1) * CMPLX[IMAG[FP],-REAL[FM]]
   END
END
```

3.9 Parallel Computing

3.9.1 Introduction

Ab initio molecular dynamics calculation need large computer resources. Memory and CPU time requirement make it necessary to run projects on the biggest computers available. It is exclusively parallel computers that provide these resources today. There are many different types of parallel computers available. Computers differ in their memory access system and their communication system. Widely different performances are seen for bandwidth and latency. In addition, different programming paradigms are supported. In order to have a portable code that can be used on most of the current computer architectures, CPMD was programmed us-

ing standard communication libraries and making no assumption on the topology of the processor network and memory access system.

Minimizing the communication was the major goal in the implementation of the parallel plane wave code in CPMD. Therefore, the algorithms had to be adapted to the distributed data model chosen. The most important decisions concern the data distribution of the largest arrays in the calculation. These arrays are the ones holding information on the wavefunctions. Three distribution strategies can be envisaged and were used before [90,137,687,688,117].

First, the data are distributed over the bands [687]. Each processor holds all expansion coefficients of an electronic band locally. Several problems arise with this choice. The number of bands is usually of the same magnitude as the number of processors. This leads to a severe load-balancing problem that can only be avoided for certain magic numbers, namely if the number of bands is a multiple of the number of CPU's. Furthermore this approach requires to perform three-dimensional Fourier transforms locally. The memory requirements for the Fourier transform only increase linearly with system size, but their prefactor is very big and a distribution of these arrays is desirable. In addition, all parts of the program that do not contain loops over the number of bands have to be parallelized using another scheme, leading to additional communication and synchronization overhead.

Second, the data is distributed over the Fourier space components and the real space grid is also distributed [90,137,117]. This scheme allows for a straight forward parallelization of all parts of the program that involve loops over the Fourier components or the real space grid. Only a few routines are not covered by this scheme. The disadvantage is that all three-dimensional Fourier transforms require communication.

Third, it is possible to use a combination of the above two schemes [688]. This leads to the most complicated scheme, as only a careful arrangement of algorithms avoids the disadvantages of the other schemes while still keeping their advantages.

Additionally, it is possible to distribute the loop over **k**–points. As most calculation only use a limited number of **k**–points or even only the Γ–point, this method is of limited use. However, combining the distribution of the k-points with one of the other method mentioned above might result in a very efficient approach.

The CPMD program is parallelized using the distribution in Fourier and real space. The data distribution is held fixed during a calculation, i.e. static load balancing is used. In all parts of the program where the distribution of the plane waves does not apply, an additional parallelization over the number of atoms or bands is used. However, the data structures involved are replicated on all processors.

A special situation exists for the case of path integral calculations (see Sect. 4.4), where an inherent parallelization over the Trotter slices is present. The problem is "embarrassingly parallel" in this variable and perfect parallelism can be observed on all types of computers, even on clusters of workstations or supercomputers ("meta–computing"). In practice the parallelization over the Trotter slices will be combined with one of the schemes mentioned above, allowing for good results even on massively parallel machines with several hundred processors.

In addition to the variables used in the serial version, local copies have to be defined. These local variables will be indexed by a superscript indicating the processor number. The total number of processors is P. Each processor has a certain number of plane waves, atoms, electronic bands and real space grid points assigned.

$N_{\rm at}^p$	number of atoms on processor p
$N_{\rm p}^p$	number of projectors on processor p
$N_{\rm b}^p$	number of electronic bands or states on processor p
$N_{\rm PW}^p$	number of plane-waves on processor p
$N_{\rm D}^p$	number of plane-waves for densities and potentials on processor p
$N_{\rm x}^p$, $N_{\rm y}$, $N_{\rm z}$	number of grid points in x, y, and z direction on processor p
N^p=$N_{\rm x}^p N_{\rm y} N_{\rm z}$	total number of grid points on processor p

The real space grid is only distributed over the x coordinates. This decision is related to the performance of the Fourier transform that will be discussed in more detail in the following sections. The distribution algorithm for atoms, projectors and bands just divides the total number of these quantities in equal junks based on their arbitrary numbering. The algorithms that use these parallelization schemes do not play a major role in the overall performance of the program (at least for the systems accessible with the computers available today) and small imperfections in load balancing can be ignored.

Data structures that are replicated on all processors:

`r(3,`N_{at}`)`	nuclear positions
`v(3,`N_{at}`)`	nuclear velocities
`f(3,`N_{at}`)`	nuclear forces
`fnl(`N_p`,`N_b`)`	overlap of projectors and bands
`smat(`N_b`,`N_b`)`	overlap matrices between bands.

Data structures that are distributed over all processors:

`g(3,`N_{PW}^p`)`	plane–wave indices
`ipg(3,`N_{PW}^p`)`	mapping of G–vectors (positive part)
`img(3,`N_{PW}^p`)`	mapping of G–vectors (negative part)
`rhog(`N_{PW}^p`)`	densities (n, $n_{\rm c}$, $n_{\rm tot}$) in Fourier–space
`vpot(`N_{PW}^p`)`	potentials ($V_{\rm loc}$, $V_{\rm xc}$, $V_{\rm H}$) in Fourier–space
`n(`N_x^p`,`N_y`,`N_z`)`	densities (n, $n_{\rm c}$, $n_{\rm tot}$) in real–space
`v(`N_x^p`,`N_y`,`N_z`)`	potentials ($V_{\rm loc}$, $V_{\rm xc}$, $V_{\rm H}$) in real–space
`vps(`N_D^p`)`	local pseudopotential
`rpc(`N_D^p`)`	core charges
`pro(`N_{PW}^p`)`	projectors of non-local pseudopotential
`eigr(`N_D^p`,`N_{at}`)`	structure factors
`dfnl(`N_p`,`N_b^p`,3)`	derivative of `fnl`
`cr(`N_{PW}^p`,`N_b`)`	bands in Fourier space
`cv(`N_{PW}^p`,`N_b`)`	velocity of bands in Fourier space
`cf(`N_{PW}^p`,`N_b`)`	forces of bands in Fourier space.

Several different goals should be achieved in the distribution of the plane waves

over processors. All processors should hold approximately the same number of plane waves. If a plane wave for the wavefunction cutoff is on a certain processor, the same plane wave should be on the same processor for the density cutoff. The distribution of the plane waves should be such that at the beginning or end of a three dimensional Fourier transform no additional communication is needed. To achieve all of these goals the following heuristic algorithm [137] is used. The plane waves are ordered into "pencils". Each pencil holds all plane waves with the same $g_{\rm y}$ and $g_{\rm z}$ components. The pencils are numbered according to the total number of plane waves that are part of it. Pencils are distributed over processors in a "round robin" fashion switching directions after each round. This is first done for the wavefunction cutoff. For the density cutoff the distribution is carried over, and all new pencils are distributed according to the same algorithm. Experience shows that this algorithm leads to good results for the load balancing on both levels, the total number of plane waves and the total number of pencils. The number of pencils on a processor is proportional to the work for the first step in the three-dimensional Fourier transform.
Special care has to be taken for the processor that holds the $\mathbf{G} = \mathbf{0}$ component. This component has to be treated individually in the calculation of the overlaps. The processor that holds this component will be called $p0$.

3.9.3 `CPMD` *Program: Computational Kernels*

There are three communication routines mostly used in the parallelization of the `CPMD` code. All of them are collective communication routines, meaning that all processors are involved. This also implies that synchronization steps are performed during the execution of these routines. Occasionally other communication routines have to be used (e.g. in the output routines for the collection of data) but they do not appear in the basic computational kernels. The three routines are the `Broadcast`, `GlobalSum`, and `MatrixTranspose`. In the `Broadcast` routine data is send from one processor (px) to all other processors

$$x^p \leftarrow x^{\rm px} \ . \tag{265}$$

In the `GlobalSum` routine a data item is replaced on each processor by the sum over this quantity on all processors

$$x^p \leftarrow \sum_p x^p \ . \tag{266}$$

The `MatrixTranspose` changes the distribution pattern of a matrix, e.g. from row distribution to column distribution

$$x(p,:) \leftarrow x(:,p) \ . \tag{267}$$

On a parallel computer with P processors, a typical latency time $t_{\rm L}$ (time for the first data to arrive) and a bandwidth of B, the time spend in the communication routines is

`Broadcast`	$\log_2[P]\,\{t_{\rm L} + N/B\}$
`GlobalSum`	$\log_2[P]\,\{t_{\rm L} + N/B\}$
`MatrixTranspose`	$Pt_{\rm L} + N/(PB)$

Table 4. Distribution of plane waves and "pencils" in parallel runs on different numbers of processors. Example for a cubic box with a volume of 6479.0979 bohr^3 and a 70 Rydberg cutoff for the wavefunctions. This is the simulation box needed for 32 water molecules at normal pressure.

	wavefunction cutoff			
PE	plane waves		pencils	
	max	min	max	min
1	32043	32043	1933	1933
2	16030	16013	967	966
4	8016	8006	484	482
8	4011	4000	242	240
16	2013	1996	122	119
32	1009	994	62	59
64	507	495	32	29
128	256	245	16	14
	density cutoff			
PE	plane waves		pencils	
	max	min	max	min
1	256034	256034	7721	7721
2	128043	127991	3859	3862
4	64022	63972	1932	1929
8	32013	31976	966	964
16	16011	15971	484	482
32	8011	7966	242	240
64	4011	3992	122	119
128	2006	1996	62	59

where it is assumed that the amount of data N is constant. The time needed in `Broadcast` and `GlobalSum` will increase with the logarithm of the number of processors involved. The time for the matrix transposition scales for one part linearly with the number of processors. Once this part is small, then the latency part will be dominant and increase linearly. Besides load balancing problems, the communication routines will limit the maximum speedup that can be achieved on a parallel computer for a given problem size. Examples will be shown in the last part of this section.

With the distribution of the data structures given, the parallelization of the computational kernels is in most cases easy. In the `StructureFactor` and `Rotation` routines the loop over the plane waves N_{D} has to be replaced by N_{D}^p. The routines performing inner products have to be adapted for the $\mathbf{G} = \mathbf{0}$ term and the global summation of the final result.

```
MODULE DotProduct
IF (p == P0) THEN
   ab = A(1) * B(1) + 2 * sdot(2 * (N_D^p − 1),A(2),1,B(2),1)
ELSE
```

```
   ab = 2 * sdot(2 * N_D^p,A(1),1,B(1),1)
END IF
CALL GlobalSum[ab]

MODULE Overlap
CALL SGEMM('T','N',N_b,N_b,2*N_PW^p,2,&
           & ca(1,1),2*N_PW^p,cb(1,1),2*N_PW^p,0,smat,N_b)
IF (p == P0) CALL SDER(N_b,N_b,-1,ca(1,1),2*N_PW^p,&
           & cb(1,1),2*N_PW^p,smat,N_b)
CALL GlobalSum[smat]
```

Similarly, the overlap part of the `FNL` routine has to be changed and the loops restricted to the local number of plane waves.

```
MODULE FNL
FOR i=1:N_p,M
   IF (MOD(lp(i),2) == 0) THEN
      FOR j=0:M-1
         pf = -1**(lp(i+j)/2)
         FOR k=1:N_PW^p
            t = pro(k) * pf
            er = REAL[eigr(k,iat(i+j))]
            ei = IMAG[eigr(k,iat(i+j))]
            scr(k,j) = CMPLX[t * er,t * ei]
         END
      END
   ELSE
      FOR j=0:M-1
         pf = -1**(lp(i+j)/2+1)
         FOR k=1:N_PW^p
            t = pro(k) * pf
            er = REAL[eigr(k,iat(i+j))]
            ei = IMAG[eigr(k,iat(i+j))]
            scr(k,j) = CMPLX[-t * ei,t * er]
         END
      END
   END IF
   IF (p == P0) scr(1,0:M-1) = scr(1,0:M-1)/2
   CALL SGEMM('T','N',M,N_b,2*N_PW^p,2,&
              & scr(1,0),2*N_PW^p,cr(1,1),2*N_PW^p,0,fnl(i,1),N_p)
END
CALL GlobalSum[fnl]
```

The routines that need the most changes are the once that include Fourier transforms. Due to the complicated break up of the plane waves a new mapping has to be introduced. The map `mapxy` ensures that all pencils occupy contiguous memory

locations on each processor.

```
MODULE INVFFT
scr1(1:N_x,1:N_pencil^D) = 0
FOR i=1:N_D^p
   scr1(ipg(1,i),mapxy(ipg(2,i),ipg(3,i))) = rhog(i)
   scr1(img(1,i),mapxy(img(2,i),img(3,i))) = CONJG[rhog(i)]
END
CALL ParallelFFT3D("INV",scr1,scr2)
n(1:N_x^p,1:N_y,1:N_z) = REAL[scr2(1:N_x^p,1:N_y,1:N_z)]

MODULE FWFFT
scr2(1:N_x^p,1:N_y,1:N_z) = n(1:N_x^p,1:N_y,1:N_z)
CALL ParallelFFT3D("FW",scr1,scr2)
FOR i=1:N_D^p
   rhog(i) = scr1(ipg(1,i),mapxy(ipg(2,i),ipg(3,i)))
END
```

Due to the mapping of the y and z direction in Fourier space onto a single dimension, input and output array of the parallel Fourier transform do have different shapes.

```
MODULE Density
rho(1:N_x^p,1:N_y,1:N_z) = 0
FOR i=1:N_b,2
   scr1(1:N_x,1:N_pencil^PW) = 0
   FOR j=1:N_PW^p
      scr1(ipg(1,i),mapxy(ipg(2,i),ipg(3,i))) = &
            & c(j,i) + I * c(j,i+1)
      scr1(img(1,i),mapxy(img(2,i),img(3,i))) = &
            & CONJG[c(j,i) + I * c(j,i+1)]
   END
   CALL ParallelFFT3D("INV",scr1,scr2)
   rho(1:N_x^p,1:N_y,1:N_z) = rho(1:N_x^p,1:N_y,1:N_z) + &
      & REAL[scr2(1:N_x^p,1:N_y,1:N_z)]**2 + &
      & IMAG[scr2(1:N_x^p,1:N_y,1:N_z)]**2
END

MODULE VPSI
FOR i=1:N_b,2
   scr1(1:N_x,1:N_pencil^PW) = 0
   FOR j=1:N_PW^p
      scr1(ipg(1,i),mapxy(ipg(2,i),ipg(3,i))) = &
            & c(j,i) + I * c(j,i+1)
      scr1(img(1,i),mapxy(img(2,i),img(3,i))) = &
            & CONJG[c(j,i) + I * c(j,i+1)]
```

```
   END
   CALL ParallelFFT3D("INV",scr1,scr2)
   scr2(1:N_x^p,1:N_y,1:N_z) = scr2(1:N_x^p,1:N_y,1:N_z) * &
      & vpot(1:N_x^p,1:N_y,1:N_z)
   CALL ParallelFFT3D("FW",scr1,scr2)
   FOR j=1:N_PW^p
      FP = scr1(ipg(1,i),mapxy(ipg(2,i),ipg(3,i))) &
           & + scr1(img(1,i),mapxy(img(2,i),img(3,i)))
      FM = scr1(ipg(1,i),mapxy(ipg(2,i),ipg(3,i))) &
           & - scr1(img(1,i),mapxy(img(2,i),img(3,i)))
      fc(j,i) = f(i) * CMPLX[REAL[FP],IMAG[FM]]
      fc(j,i+1) = f(i+1) * CMPLX[IMAG[FP],-REAL[FM]]
   END
END
```

The parallel Fourier transform routine can be built from a multiple one-dimensional Fourier transform and a parallel matrix transpose. As mentioned above, only one dimension of the real space grid is distributed in the CPMD code. This allows to combine the transforms in y and z direction to a series of two-dimensional transforms. The handling of the plane waves in Fourier space breaks the symmetry and two different transpose routines are needed, depending on the direction. All the communication is done in the routine ParallelTranspose. This routine consists of a part where the coefficients are gathered into matrix form, the parallel matrix transpose, and a final part where the coefficients are put back according to the mapping used.

```
MODULE ParallelFFT3D(tag,a,b)
IF (tag == "INV") THEN
   CALL MLTFFT1D(a)
   CALL ParallelTranspose("INV",b,a)
   CALL MLTFFT2D(b)
ELSE
   CALL MLTFFT2D(b)
   CALL ParallelTranspose("FW",b,a)
   CALL MLTFFT1D(a)
END IF
```

All other parts of the program use the same patterns for the parallelization as the ones shown in this section.

3.9.4 Limitations

Two types of limitations can be encountered when trying to run a parallel code on a computer. Increasing the number of processors working on a problem will no longer lead to a faster calculation or the memory available is not sufficient to perform a calculation, independently on the number of processors available. The first type of

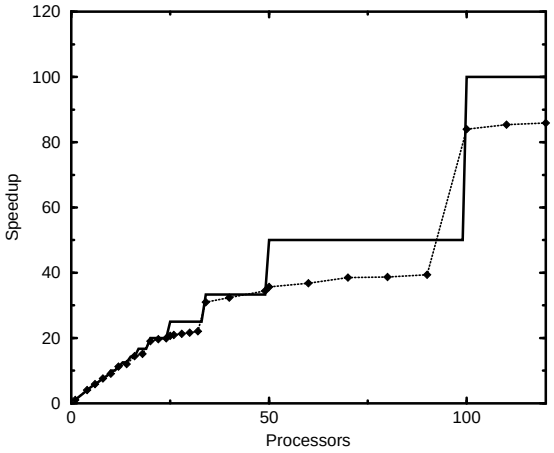


Figure 13. Maximal theoretical speedup for a calculation with a real space grid of dimension 100 (solid line). Effective speedup for a 32 water molecule system with an energy cutoff of 70 Rydberg and a real space grid of dimension 100 (dotted line with diamonds)

limitation is related to bad load-balancing or the computation becomes dominated by the non-scaling part of the communication routines. Load–balancing problems in the `CPMD` code are almost exclusively due to the distribution of the real space arrays. Only the x coordinate is distributed. There are typically of the order of 100 grid points in each direction. Figure 13 shows the maximal theoretical speedup for a calculation with a real space grid of dimension 100. The steps are due to the load–balancing problems initiated by the granularity of the problem (the dimension is an integer value). No further speedup can be achieved once 100 processors are reached. The second curve in Fig. 13 shows actual calculations of the full `CPMD` code. It is clearly shown that the load balancing problem in the Fourier transforms affects the performance of this special example. Where this steps appear and how severe the performance losses are depends of course on the system under consideration.

To overcome this limitation a method based on processor groups has been implemented into the code. For the two most important routines where the real space grid load–balancing problem appears, the calculation of the charge density and the application of the local potential, a second level of parallelism is introduced. The processors are arranged into a two-dimensional grid and groups are build according to the row and column indices. Each processor is a member of its column group (`colgrp`) and its row group (`rowgrp`). In a first step a data exchange in the column group assures that all the data needed to perform Fourier transforms within the row groups are available. Then each row group performs the Fourier transforms independently and in the end another data exchange in the column groups rebuilds the original data distribution. This scheme (shown in the pseudo code for the density calculation) needs roughly double the amount of communication. Advantages

are the improved load–balancing for the Fourier transforms and the bigger data packages in the matrix transposes. The number of plane waves in the row groups ($N_{\rm PW}^{pr}$) is calculated as the sum over all local plane waves in the corresponding column groups.

```
MODULE Density
rho(1:N_x^pr,1:N_y,1:N_z) = 0
FOR i=1:N_b,2*Pc
   CALL ParallelTranspose(c(:,i),colgrp)
   scr1(1:N_x,1:N_pencil,r^PW) = 0
   FOR j=1:N_PW^pr
      scr1(ipg(1,i),mapxy(ipg(2,i),ipg(3,i))) = &
            & c(j,i) + I * c(j,i+1)
      scr1(img(1,i),mapxy(img(2,i),img(3,i))) = &
            & CONJG[c(j,i) + I * c(j,i+1)]
   END
   CALL ParallelFFT3D("INV",scr1,scr2,rowgrp)
   rho(1:N_x^pr,1:N_y,1:N_z) = rho(1:N_x^pr,1:N_y,1:N_z) + &
      & REAL[scr2(1:N_x^p,1:N_y,1:N_z)]**2 + &
      & IMAG[scr2(1:N_x^p,1:N_y,1:N_z)]**2
END
CALL GlobalSum(rho,colgrp)
```

The use of two task groups in the example shown in Fig. 13 leads to an increase of speedup for 256 processors from 120 to 184 on a Cray T3E/600 computer.

The effect of the non-scalability of the global communication used in `CPMD` is shown in Fig. 14. This example shows the percentage of time used in the global communication routines (global sums and broadcasts) and the time spend in the parallel Fourier transforms for a system of 64 silicon atoms with a energy cutoff of 12 Rydberg. It can clearly be seen that the global sums and broadcasts do not scale and therefore become more important the more processors are used. The Fourier transforms on the other hand scale nicely for this range of processors. Where the communication becomes dominant depends on the size of the system and the performance ratio of communication to cpu.

Finally, the memory available on each processor may become a bottleneck for large computations. The replicated data approach for some arrays adapted in the implementation of the code poses limits on the system size that can be processed on a given type of computer. In the outline given in this chapter there are two types of arrays that scale quadratically in system size that a replicated. The overlap matrix of the projectors with the wavefunctions (`fnl`) and the overlap matrices of the wavefunctions themselves (`smat`). The `fnl` matrix is involved in two types of calculations where the parallel loop goes either over the bands or the projectors. To avoid communication, two copies of the array are kept on each processor. Each copy holds the data needed in one of the distribution patterns. This scheme needs only a small adaptation of the code described above.

The distribution of the overlap matrices (`smat`) causes some more problems. In

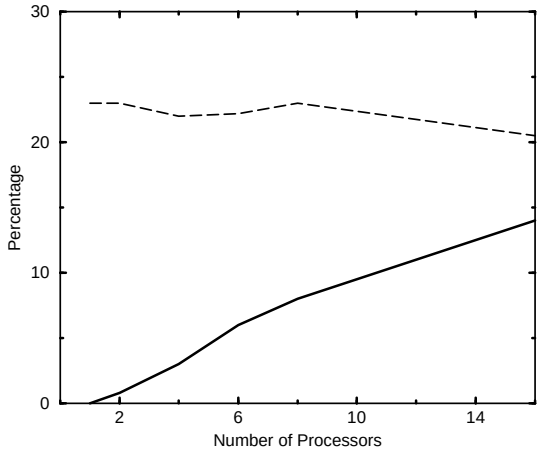


Figure 14. Percentage of total CPU time spend in global communication routines (solid line) and in Fourier transform routines (dashed line) for a system of 64 silicon atoms on a Cray T3E/600 computer.

addition to the adaptation of the overlap routine, also the matrix multiply routines needed for the orthogonalization step have to be done in parallel. Although there are libraries for these tasks available the complexity of the code is considerably increased.

3.9.5 Summary

Efficient parallel algorithms for the plane wave–pseudopotential density functional theory method exist. Implementations of these algorithms are available and were used in most of the large scale applications presented at the end of this paper (Sect. 5). Depending on the size of the problem, excellent speedups can be achieved even on computers with several hundreds of processors. The limitations presented in the last paragraph are of importance for high–end applications. Together with the extensions presented, existing plane wave codes are well suited also for the next generation of supercomputers.

4 Advanced Techniques: Beyond ...

4.1 Introduction

The discussion up to this point revolved essentially around the "basic" *ab initio* molecular dynamics methodologies. This means in particular that *classical* nuclei evolve in the electronic *ground state* in the *microcanonical* ensemble. This combination allows already a multitude of applications, but many circumstances exist where the underlying approximations are unsatisfactory. Among these cases are

situations where

- it is necessary to keep temperature and /or pressure constant (such as during journeys in phase diagrams or in the investigation of solid–state phase transitions),
- there is a sufficient population of excited electronic states (such as in materials with a small or vanishing electronic gap) or dynamical motion occurs in a single excited states (such as after photoexcitation events),
- light nuclei are involved in crucial steps of a process (such as in studies of proton transfer or muonium impurities).

In the following subsections techniques are introduced which transcede these limitations. Thus, the realm of *ab initio* molecular dynamics is considerably increased beyond the basic setup as discussed in general terms in Sect. 2 and concerning its implementation in Sect. 3. The presented "advanced techniques" are selected because they are available in the current version of the `CPMD` package [142], but their implementation is not discussed in detail here.

4.2 Beyond Microcanonics

4.2.1 Introduction

In the framework of statistical mechanics all ensembles can be formally obtained from the microcanonical or NVE ensemble – where particle number, volume and energy are the external thermodynamic control variables – by suitable Laplace transforms of its partition function; note that V is used for volume when it comes to labeling the various ensembles in Sect. 4 and its subsections. Thermodynamically this corresponds to Legendre transforms of the associated thermodynamic potentials where intensive and extensive conjugate variables are interchanged. In thermodynamics, this task is achieved by a "sufficiently weak" coupling of the original system to an appropriate infinitely large bath or reservoir via a link that establishes thermodynamic equilibrium. The same basic idea is instrumental in generating distribution functions of such ensembles by computer simulation [98,250]. Here, two important special cases are discussed: thermostats and barostats, which are used to impose temperature instead of energy and / or pressure instead of volume as external control parameters [12,445,270,585,217].

4.2.2 Imposing Temperature: Thermostats

In the limit of ergodic sampling the ensemble created by standard molecular dynamics is the microcanonical or NVE ensemble where in addition the total momentum is conserved [12,270,217]. Thus, the temperature is not a control variable in the Newtonian approach to molecular dynamics and whence it cannot be preselected and fixed. But it is evident that also within molecular dynamics the possibility to control the average temperature (as obtained from the average kinetic energy of the nuclei and the energy equipartition theorem) is welcome for physical reasons. A *deterministic* algorithm of achieving temperature control in the spirit of extended

system dynamics [14] by a sort of dynamical friction mechanism was devised by Nosé and Hoover [442,443,444,307], see e.g. Refs. [12,445,270,585,217] for reviews of this well–established technique. Thereby, the canonical or NVT ensemble is generated in the case of ergodic dynamics.

As discussed in depth in Sect. 2.4, the Car–Parrinello approach to *ab initio* molecular dynamics works due to a dynamical separation between the physical and fictitious temperatures of the nuclear and electronic subsystems, respectively. This separability and thus the associated metastability condition breaks down if the electronic excitation gap becomes comparable to the thermal energy or smaller, that is in particular for metallic systems. In order to satisfy nevertheless adiabaticity in the sense of Car and Parrinello it was proposed to couple separate thermostats [583] to the classical fields that stem from the electronic degrees of freedom [74,204]. Finally, the (long–term) stability of the molecular dynamics propagation can be increased due to the same mechanism, which enables one to increase the time step that still allows for adiabatic time evolution [638]. Note that these technical reasons to include additional thermostats are by construction absent from any Born–Oppenheimer molecular dynamics scheme.

It is well–known that the standard Nosé–Hoover thermostat method suffers from non–ergodicity problems for certain classes of Hamiltonians, such as the harmonic oscillator [307]. A closely related technique, the so–called Nosé–Hoover–chain thermostat [388], cures that problem and assures ergodic sampling of phase space even for the pathological harmonic oscillator. This is achieved by thermostatting the original thermostat by another thermostat, which in turn is thermostatted and so on. In addition to restoring ergodicity even with only a few thermostats in the chain, this technique is found to be much more efficient in imposing the desired temperature.

Nosé–Hoover–chain thermostatted Car–Parrinello molecular dynamics was introduced in Ref. [638]. The underlying equations of motion read

$$M_I \ddot{\mathbf{R}}_I = -\nabla_I E^{\mathrm{KS}} - M_I \dot{\xi}_1 \dot{\mathbf{R}}_I \tag{268}$$

$$\begin{aligned}
Q_1^{\mathrm{n}} \ddot{\xi}_1 &= \left[\sum_I M_I \dot{\mathbf{R}}_I^2 - g k_{\mathrm{B}} T \right] - Q_1^{\mathrm{n}} \dot{\xi}_1 \dot{\xi}_2 \\
Q_k^{\mathrm{n}} \ddot{\xi}_k &= \left[Q_{k-1}^{\mathrm{n}} \dot{\xi}_{k-1}^2 - k_{\mathrm{B}} T \right] - Q_k^{\mathrm{n}} \dot{\xi}_k \dot{\xi}_{k+1} \left(1 - \delta_{kK}\right) \qquad \text{where } k = 2, \ldots, K
\end{aligned}$$

for the nuclear part and

$$\ddot{\mu\phi}_i = -H_{\mathrm{e}}^{\mathrm{KS}} \phi_i + \sum_{ij} \Lambda_{ij} \phi_j - \mu \dot{\eta}_1 \dot{\phi}_i \tag{269}$$

$$\begin{aligned}
Q_1^{\mathrm{e}} \ddot{\eta}_1 &= 2 \left[\sum_i^{\mathrm{occ}} \mu \left\langle \phi_i \middle| \phi_i \right\rangle - T_{\mathrm{e}}^0 \right] - Q_1^{\mathrm{e}} \dot{\eta}_1 \dot{\eta}_2 \\
Q_l^{\mathrm{e}} \ddot{\eta}_l &= \left[Q_{l-1}^{\mathrm{e}} \dot{\eta}_{l-1}^2 - \frac{1}{\beta_{\mathrm{e}}} \right] - Q_l^{\mathrm{e}} \dot{\eta}_l \dot{\eta}_{l+1} \left(1 - \delta_{lL}\right) \qquad \text{where } l = 2, \ldots, L
\end{aligned}$$

for the electronic contribution. These equations are written down in density functional language (see Eq. (75) and Eq. (81) for the definitions of E^{KS} and $H_{\mathrm{e}}^{\mathrm{KS}}$,

respectively), but completely analogues expressions are operational if other electronic structure approaches are used instead. Using separate thermostatting baths $\{\xi_k\}$ and $\{\eta_l\}$, chains composed of K and L coupled thermostats are attached to the nuclear and electronic equations of motion, respectively.

By inspection of Eq. (268) it becomes intuitively clear how the thermostat works: $\dot{\xi}_1$ can be considered as a *dynamical* friction coefficient. The resulting "dissipative dynamics" leads to non–Hamiltonian flow, but the friction term can aquire positive or negative sign according to its equation of motion. This leads to damping or acceleration of the nuclei and thus to cooling or heating if the instantaneous kinetic energy of the nuclei is higher or lower than $k_{\mathrm{B}}T$ which is preset. As a result, this extended system dynamics can be shown to produce a canonical ensemble in the subspace of the nuclear coordinates and momenta. In spite of being non–Hamiltonian, Nosé–Hoover (–chain) dynamics is also distinguished by conserving an energy quantity of the extended system, see Eq. (272).

The desired average physical temperature is given by T and g denotes the number of dynamical degrees of freedom to which the nuclear thermostat chain is coupled (i.e. constraints imposed on the nuclei have to be subtracted). Similarly, T_{e}^0 is the desired fictitious kinetic energy of the electrons and $1/\beta_{\mathrm{e}}$ is the associated temperature. In principle, β_{e} should be chosen such that $1/\beta_{\mathrm{e}} = 2T_{\mathrm{e}}^0/N_{\mathrm{e}}$ where N_{e} is the number of dynamical degrees of freedom needed to parameterize the wavefunction minus the number of constraint conditions. It is found that this choice requires a very accurate integration of the resulting equations of motion (for instance by using a high–order Suzuki–Yoshida integrator, see Sect. VI.A in Ref. [638]). However, relevant quantities are rather insensitive to the particular value so that N_{e} can be replaced heuristically by N_{e}' which is the number of orbitals ϕ_i used to expand the wavefunction [638].

The choice of the "mass parameters" assigned to the thermostat degrees of freedom should be made such that the overlap of their power spectra and the ones the thermostatted subsystems is maximal [74,638]. The relations

$$Q_1^{\mathrm{n}} = \frac{g k_{\mathrm{B}} T}{\omega_{\mathrm{n}}^2} \quad , \qquad Q_k^{\mathrm{n}} = \frac{k_{\mathrm{B}} T}{\omega_{\mathrm{n}}^2} \tag{270}$$

$$Q_1^{\mathrm{e}} = \frac{2T_{\mathrm{e}}^0}{\omega_{\mathrm{e}}^2} \quad , \qquad Q_l^{\mathrm{e}} = \frac{1}{\beta_{\mathrm{e}} \omega_{\mathrm{e}}^2} \tag{271}$$

assures this if ω_{n} is a typical phonon or vibrational frequency of the nuclear subsystem (say of the order of 2000 to 4000 cm^{-1}) and ω_{e} is sufficiently large compared to the maximum frequency $\omega_{\mathrm{n}}^{\mathrm{max}}$ of the nuclear power spectrum (say 10 000 cm^{-1} or larger). The integration of these equations of motion is discussed in detail in Ref. [638] using the velocity Verlet / RATTLE algorithm.

In some instances, for example during equilibration runs, it is advantageous to go one step further and to actually couple one chain of Nosé–Hoover thermostats to every individual nuclear degree of freedom akin to what is done in path integral molecular dynamics simulations [637,644,646], see also Sect. 4.4. This so–called "massive thermostatting approach" is found to accelerate considerably the expensive equilibration periods within *ab initio* molecular dynamics, which is useful for both Car–Parrinello and Born–Oppenheimer dynamics.

In classical molecular dynamics two quantities are conserved during a simulation, the total energy and the total momentum. The same constants of motion apply to (exact) microcanonical Born–Oppenheimer molecular dynamics because the only *dynamical* variables are the nuclear positions and momenta as in classical molecular dynamics. In microcanonical Car–Parrinello molecular dynamics the total energy of the *extended* dynamical system composed of nuclear and electronic positions and momenta, that is $E_{\rm cons}$ as defined in Eq. (48), is also conserved, see e.g. Fig. 3 in Sect. 2.4. There is also a conserved energy quantity in the case of thermostatted molecular dynamics according to Eq. (268)–(269). Instead of Eq. (48) this constant of motion reads

$$\begin{aligned} E_{\rm cons}^{\rm NVT} = & \sum_{i}^{\rm occ} \mu \left\langle \dot{\phi}_i \middle| \dot{\phi}_i \right\rangle + \sum_{I} \frac{1}{2} M_I \dot{\mathbf{R}}_I^2 + E^{\rm KS}\left[\{\phi_i\},\{\mathbf{R}_I\}\right] \\ & + \sum_{l=1}^{L} \frac{1}{2} Q_l^{\rm e} \dot{\eta}_l^2 + \sum_{l=2}^{L} \frac{\eta_l}{\beta_{\rm e}} + 2T_{\rm e}^0 \eta_1 \\ & + \sum_{k=1}^{K} \frac{1}{2} Q_k^{\rm n} \dot{\xi}_k^2 + \sum_{k=2}^{K} k_{\rm B} T \xi_k + g k_{\rm B} T \xi_1 \end{aligned} \tag{272}$$

for Nosé–Hoover–chain thermostatted canonical Car–Parrinello molecular dynamics [638].

In microcanonical Car–Parrinello molecular dynamics the total nuclear momentum $\mathbf{P}_{\rm n}$ is no more a constant of motion as a result of the fictitious dynamics of the wavefunction; this quantity as well as other symmetries and associated invariants are discussed in Ref. [467]. However, a generalized linear momentum which embraces the electronic degrees of freedom

$$\mathbf{P}_{\rm CP} = \mathbf{P}_{\rm n} + \mathbf{P}_{\rm e} = \sum_{I} \mathbf{P}_I + \sum_{i}^{\rm occ} \mu \left\langle \dot{\phi}_i \middle| -\nabla_{\mathbf{r}} \middle| \phi_i \right\rangle + {\rm c.c.} \tag{273}$$

can be defined [467,436]; $\mathbf{P}_I = M_I \dot{\mathbf{R}}_I$. This quantity is a constant of motion in unthermostatted Car–Parrinello molecular dynamics due to an exact cancellation of the nuclear and electronic contributions [467,436]. As a result, the nuclear momentum $\mathbf{P}_{\rm n}$ fluctuates during such a run, but in practice $\mathbf{P}_{\rm n}$ is conserved *on the average* as shown in Fig. 1 of Ref. [436]. This is analogues to the behavior of the physical total energy $E_{\rm phys}$ Eq. (49), which fluctuates slightly due to the presence of the fictitious kinetic energy of the electrons $T_{\rm e}$ Eq. (51).

As recently outlined in detail it is clear that the coupling of more than one thermostat to a dynamical system, such as done in Eq. (268)–(269), destroys the conservation of momentum [436], i.e. $\mathbf{P}_{\rm CP}$ is no more an invariant. In unfavorable cases, in particular in small–gap or metallic regimes where there is a substantial coupling of the nuclear and electronic subsystems, momentum can be transferred to the nuclear subsystem such that $\mathbf{P}_{\rm n}$ grows in the course of a simulation. This problem can be cured by controlling the nuclear momentum (using e.g. scaling or constraint methods) so that the total nuclear momentum $\mathbf{P}_{\rm n}$ remains small [436].

4.2.3 Imposing Pressure: Barostats

Keeping the pressure constant is a desirable feature for many applications of molecular dynamics. The concept of barostats and thus constant–pressure molecular dynamics was introduced in the framework of extended system dynamics by Hans Andersen [14], see e.g. Refs. [12,270,585,217] for introductions. This method was devised to allow for isotropic fluctuations in the volume of the supercell. A powerful extension consists in also allowing for changes of the *shape* of the supercell to occur as a result of applying external pressure [459,460,461,678], including the possibility of non–isotropic *external* stress [460]; the additional fictitious degrees of freedom in the Parrinello–Rahman approach [459,460,461] are the lattice vectors of the supercell, whereas the strain tensor is the dynamical variable in the Wentzcovitch approach [678]. These variable–cell approaches make it possible to study dynamically structural phase transitions in solids at finite temperatures. With the birth of *ab initio* molecular dynamics both approaches were combined starting out with isotropic volume fluctuations [94] *à la* Andersen [14] and followed by Born–Oppenheimer [681,682] and Car–Parrinello [201,202,55,56] variable–cell techniques.

The basic idea to allow for changes in the cell shape consists in constructing an extended Lagrangian where the primitive Bravais lattice vectors $\mathbf{a}_1$, $\mathbf{a}_2$ and $\mathbf{a}_3$ of the simulation cell are additional dynamical variables similar to the thermostat degree of freedom ξ, see Eq. (268). Using the 3×3 matrix $\mathbf{h} = [\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]$ (which fully defines the cell with volume Ω) the real–space position $\mathbf{R}_I$ of a particle in this original cell can be expressed as

$$\mathbf{R}_I = \mathbf{h}\mathbf{S}_I \tag{274}$$

where $\mathbf{S}_I$ is a scaled coordinate with components $\mathbf{S}_{I,u} \in [0,1]$ that defines the position of the Ith particle in a unit cube (i.e. $\Omega_{\text{unit}} = 1$) which is the scaled cell [459,460], see Sect. 3.1 for some definitions. The resulting metric tensor $\mathcal{G} = \mathbf{h}^{\text{t}}\mathbf{h}$ converts distances measured in scaled coordinates to distances as given by the original coordinates according to Eq. (106) and periodic boundary conditions are applied using Eq. (107).

In the case of *ab initio* molecular dynamics the orbitals have to be expressed suitably in the scaled coordinates $\mathbf{s} = \mathbf{h}^{-1}\mathbf{r}$. The normalized original orbitals $\phi_i(\mathbf{r})$ as defined in the unscaled cell $\mathbf{h}$ are transformed according to

$$\phi_i(\mathbf{r}) = \frac{1}{\sqrt{\Omega}}\, \phi_i\left(\mathbf{s}\right) \tag{275}$$

satisfying

$$\int_{\Omega} d\mathbf{r}\; \phi_i^{\star}(\mathbf{r})\phi_i(\mathbf{r}) = \int_{\Omega_{\text{unit}}} d\mathbf{s}\; \phi_i^{\star}(\mathbf{s})\phi_i(\mathbf{s}) \tag{276}$$

so that the resulting charge density is given by

$$n(\mathbf{r}) = \frac{1}{\Omega}\, n\left(\mathbf{s}\right) \quad . \tag{277}$$

in the scaled cell, i.e. the unit cube. Importantly, the scaled fields $\phi_i(\mathbf{s})$ and thus their charge density $n(\mathbf{s})$ do *not* depend on the dynamical variables associated to the cell degrees of freedom and thus can be varied independently from the cell; the

original unscaled fields $\phi_i(\mathbf{r})$ do depend on the cell variables $\mathbf{h}$ via the normalization by the cell volume $\Omega = \det \mathbf{h}$ as evidenced by Eq. (275).

After these preliminaries a variable–cell extended Lagrangian for *ab initio* molecular dynamics can be postulated [202,201,55]

$$\begin{aligned}\mathcal{L} = & \sum_i \mu \left\langle \dot{\phi}_i(\mathbf{s}) \middle| \dot{\phi}_i(\mathbf{s}) \right\rangle - E^{\mathrm{KS}}\left[\{\phi_i\}, \{\mathbf{h}\mathbf{S}_I\}\right] \\ & + \sum_{ij} \Lambda_{ij} \left(\langle \phi_i(\mathbf{s}) \, | \phi_j(\mathbf{s}) \rangle - \delta_{ij} \right) \\ & + \sum_I \frac{1}{2} M_I \left(\dot{\mathbf{S}}_I^{\mathrm{t}} \mathcal{G} \dot{\mathbf{S}}_I \right) + \frac{1}{2} W \, \mathrm{Tr} \, \dot{\mathbf{h}}^{\mathrm{t}} \dot{\mathbf{h}} - p \, \Omega \ , \end{aligned} \tag{278}$$

with additional nine dynamical degrees of freedom that are associated to the lattice vectors of the supercell $\mathbf{h}$. This constant–pressure Lagrangian reduces to the constant–volume Car–Parrinello Lagrangian, see e.g. Eq. (41) or Eq. (58), in the limit $\dot{\mathbf{h}} \to 0$ of a rigid cell (apart from a constant term $p\,\Omega$). Here, p defines the externally applied hydrostatic pressure, W defines the fictitious mass or inertia parameter that controls the time–scale of the motion of the cell $\mathbf{h}$ and the interaction energy E^{KS} is of the form that is defined in Eq. (75). In particular, this Lagrangian allows for symmetry–breaking fluctuations – which might be necessary to drive a solid–state phase transformation – to take place spontaneously. The resulting equations of motion read

$$M_I \ddot{\mathbf{S}}_{I,u} = - \sum_{v=1}^{3} \frac{\partial E^{\mathrm{KS}}}{\partial \mathbf{R}_{I,v}} \left(\mathbf{h}^{\mathrm{t}} \right)^{-1}_{vu} - M_I \sum_{v=1}^{3} \sum_{s=1}^{3} \mathcal{G}^{-1}_{uv} \dot{\mathcal{G}}_{vs} \dot{S}_{I,s} \tag{279}$$

$$\mu \ddot{\phi}_i(\mathbf{s}) = - \frac{\delta E^{\mathrm{KS}}}{\delta \phi_i^{\star}(\mathbf{s})} + \sum_j \Lambda_{ij} \phi_j(\mathbf{s}) \tag{280}$$

$$W \ddot{\mathbf{h}}_{uv} = \Omega \sum_{s=1}^{3} \left(\mathbf{\Pi}^{\mathrm{tot}}_{us} - p \, \delta_{us} \right) \left(\mathbf{h}^{\mathrm{t}} \right)^{-1}_{sv} \ , \tag{281}$$

where the total internal stress tensor

$$\mathbf{\Pi}^{\mathrm{tot}}_{us} = \frac{1}{\Omega} \sum_I M_I \left(\dot{\mathbf{S}}_I^{\mathrm{t}} \mathcal{G} \dot{\mathbf{S}}_I \right)_{us} + \Pi_{us} \tag{282}$$

is the sum of the thermal contribution due to nuclear motion at finite temperature and the electronic stress tensor [440,441] Π which is defined in Eq. (189) and the following equations, see Sect. 3.4.

Similar to the thermostat case discussed in the previous section one can recognize a sort of frictional feedback mechanism. The average internal pressure $\langle (1/3) \, \mathrm{Tr} \, \mathbf{\Pi}^{\mathrm{tot}} \rangle$ equals the externally applied pressure p as a result of maintaining dynamically a balance between $p\,\delta$ and the instantaneous internal stress $\mathbf{\Pi}^{\mathrm{tot}}$ by virtue of the friction coefficient $\propto \dot{\mathcal{G}}$ in Eq. (279). Ergodic trajectories obtained from solving the associated *ab initio* equations of motion Eq. (279)–(281) lead to a sampling according to the isobaric–isoenthalpic or NpH ensemble. However, the generated dynamics is fictitious similar to the constant–temperature case discussed

in the previous section. The isobaric–isothermal or NpT ensemble is obtained by combining barostats and thermostats, see Ref. [389] for a general formulation and Ref. [391] for reversible integration schemes.

An important practical issue in isobaric *ab initio* molecular dynamics simulations is related to problems caused by using a finite basis set, i.e. "incomplete–basis–set" or Pulay–type contributions to the stress, see also Sect. 2.5. Using a finite plane wave basis (together with a finite number of **k**–points) in the presence of a fluctuating cell [245,211] one can either fix the number of plane waves or fix the energy cutoff; see Eq. (122) for their relation according to a rule–of–thumb. A constant number of plane waves implies no Pulay stress but a decreasing precision of the calculation as the volume of the supercell increases, whence leading to a systematically biased (but smooth) equation of state. The constant cutoff procedure has better convergence properties towards the infinite–basis–set limit [245]. However, it produces in general unphysical discontinuities in the total energy and thus in the equation of state at volumes where the number of plane waves changes abruptly, see e.g. Fig. 5 in Ref. [211].

Computationally, the number of plane waves has to be fixed in the framework of Car–Parrinello variable–cell molecular dynamics [94,202,201,55], whereas the energy cutoff can easily be kept constant in Born–Oppenheimer approaches to variable–cell molecular dynamics [681,682]. Sticking to the Car–Parrinello technique a practical remedy [202,55] to this problem consists in modifying the electronic kinetic energy term Eq. (173) in a plane wave expansion Eq. (172) of the Kohn–Sham functional E^{KS} Eq. (75)

$$E_{\mathrm{kin}} = \sum_i f_i \sum_{\mathbf{q}} \frac{1}{2} \left|\mathbf{G}\right|^2 \left|c_i(\mathbf{q})\right|^2 \ , \tag{283}$$

where the unscaled $\mathbf{G}$ and scaled $\mathbf{q} = 2\pi\mathbf{g}$ reciprocal lattice vectors are interrelated via the cell $\mathbf{h}$ according to Eq. (111) (thus $\mathbf{Gr} = \mathbf{qs}$) and the cutoff Eq. (121) is defined as $(1/2)\ |\mathbf{G}|^2 \leq E_{\mathrm{cut}}$ for a fixed number of $\mathbf{q}$–vectors, see Sect. 3.1. The modified kinetic energy at the Γ–point of the Brillouin zone associated to the supercell reads

$$\tilde{E}_{\mathrm{kin}} = \sum_i f_i \sum_{\mathbf{q}} \frac{1}{2} \left|\tilde{\mathbf{G}}\left(A, \sigma, E_{\mathrm{cut}}^{\mathrm{eff}}\right)\right|^2 \left|c_i(\mathbf{q})\right|^2 \tag{284}$$

$$\left|\tilde{\mathbf{G}}\left(A, \sigma, E_{\mathrm{cut}}^{\mathrm{eff}}\right)\right|^2 = |\mathbf{G}|^2 + A\left\{1 + \mathrm{erf}\left[\frac{\frac{1}{2}|\mathbf{G}|^2 - E_{\mathrm{cut}}^{\mathrm{eff}}}{\sigma}\right]\right\} \tag{285}$$

where A, σ and $E_{\mathrm{cut}}^{\mathrm{eff}}$ are positive definite constants and the number of scaled vectors $\mathbf{q}$, that is the number of plane waves, is strictly kept fixed.

In the limit of a vanishing smoothing ($A \to 0; \sigma \to \infty$) the constant number of plane wave result is recovered. In limit of a sharp step function ($A \to \infty; \sigma \to 0$) all plane waves with $(1/2)\ |\mathbf{G}|^2 \gg E_{\mathrm{cut}}^{\mathrm{eff}}$ have a negligible weight in $\tilde{E}_{\mathrm{kin}}$ and are thus effectively suppressed. This situation mimics a constant cutoff calculation at an "effective cutoff" of $\approx E_{\mathrm{cut}}^{\mathrm{eff}}$ within a constant number of plane wave scheme. For this trick to work note that $E_{\mathrm{cut}} \gg E_{\mathrm{cut}}^{\mathrm{eff}}$ has to be satisfied. In the case $A > 0$ the electronic stress tensor Π given by Eq. (189) features an additional term (due to

changes in the "effective basis set" as a result of variations of the supercell), which is related to the Pulay stress [219,660].

Finally, the strength of the smoothing $A > 0$ should be kept as modest as possible since the modification Eq. (284) of the kinetic energy leads to an increase of the highest frequency in the electronic power spectrum $\propto A$. This implies a decrease of the permissible molecular dynamics time step $\Delta t^{\max}$ according to Eq. (55). It is found that a suitably tuned set of the additional parameters $(A, \sigma, E_{\rm cut}^{\rm eff})$ leads to an efficiently converging constant–pressure scheme in conjunction with a fairly small number of plane waves [202,55]. Note that the cutoff was kept strictly constant in applications of the Born–Oppenheimer implementation [679] of variable–cell molecular dynamics [681,682], but the smoothing scheme presented here could be implemented in this case as well. An efficient method to correct for the discontinuities of *static* total energy calculations performed at constant cutoff was proposed in Ref. [211]. Evidently, the best way to deal with the incomplete–basis–set problem is to increase the cutoff such that the resulting artifacts become negligible on the physically relevant energy scale.

4.3 Beyond Ground States

4.3.1 Introduction

Extending *ab initio* molecular dynamics to a single non–interacting excited state is straightforward in the framework of wavefunction–based methods such as Hartree–Fock [365,254,191,379,281,284,316,293], generalized valence bond (GVB) [282,283,228,229,230], complete active space SCF (CASSCF) [566,567], or full configuration interaction (FCI) [372] approaches, see Sect. 2.7. However, these methods are computationally quite demanding – given present–day algorithms and hardware. Promising steps in the direction of including several excited states and non–adiabatic couplings are also made [385,386,387,71].

Density functional theory offers an alternative route to approximately solving electronic structure problems and recent approaches to excited–state properties within this framework look promising. In the following, two limiting and thus certainly idealistic situations will be considered, which are characterized by either

- many closely–spaced excited electronic states with a *broad thermal* distribution of fractional occupation numbers, or by

- a single electronic state that is *completely decoupled* from all other states.

The first situation is encountered for metallic systems with collective excitations or for materials at high temperatures compared to the Fermi temperature. It is noted in passing that associating fractional occupation numbers to one–particle orbitals is also one route to go beyond a single–determinant ansatz for constructing the charge density [458,168]. The second case applies for instance to large–gap molecular systems which complete a chemical reaction in a single excited state as a result of e.g. a vertical HOMO / LUMO or instantaneous one–particle / one–hole photoexcitation.

4.3.2 Many Excited States: Free Energy Functional

The free energy functional approach to excited–state molecular dynamics [5,7] is a mean–field approach similar in spirit to Ehrenfest molecular dynamics, see Sect. 2.2. The total wavefunction is first factorized into a nuclear and an electronic wavefunction Eq. (3) followed by taking the classical limit for the nuclear subsystem. Thus, classical nuclei move in the *average* field as obtained from averaging over all electronic states Eq. (25). A difference is that according to Ehrenfest molecular dynamics the electrons are propagated in real time and can perform non–adiabatic transitions by virtue of direct coupling terms $\propto \mathbf{d}^{kl}$ between all states Ψ_k subject to energy conservation, see Sect. 2.2 and in particular Eqs. (27)–(29). The average force or Ehrenfest force is obtained by weighting the different states k according to their diagonal density matrix elements (that is $\propto |c_k(t)|^2$ in Eq. (27)) whereas the coherent transitions are driven by the off–diagonal contributions (which are $\propto c_k^\star c_l$, see Eq. (27)).

In the free energy approach [5,7], the excited states are populated according to the Fermi–Dirac (finite–temperature equilibrium) distribution which is based on the assumption that the electrons "equilibrate" more rapidly than the timescale of the nuclear motion. This means that the set of electronic states evolves at a given temperature "isothermally" (rather than adiabatically) under the inclusion of *incoherent* electronic transitions at the nuclei move. Thus, instead of computing the force acting on the nuclei from the electronic ground–state energy it is obtained from the electronic *free* energy as defined in the canonical ensemble. By allowing such electronic transitions to occur the free energy approach transcends the usual Born–Oppenheimer approximation. However, the approximation of an instantaneous equilibration of the electronic subsystem implies that the electronic structure at a given nuclear configuration $\{\mathbf{R}_I\}$ is completely independent from previous configurations along a molecular dynamics trajectory. Due to this assumption the notion "free energy Born–Oppenheimer approximation" was coined in Ref. [101] in a similar context. Certain non–equilibrium situations can also be modeled within the free energy approach by starting off with an initial orbital occupation pattern that does not correspond to any temperature in its thermodynamic meaning, see e.g. Refs. [570,572,574] for such applications.

The free energy functional as defined in Ref. [5] is introduced most elegantly [7,9] by starting the discussion for the special case of *non*–interacting Fermions

$$H_\mathrm{s} = -\frac{1}{2}\nabla^2 - \sum_I \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}|} \tag{286}$$

in a *fixed* external potential due to a collection of nuclei at positions $\{\mathbf{R}_I\}$. The associated grand partition function and its thermodynamic potential ("grand free energy") are given by

$$\Xi_\mathrm{s}(\mu VT) = \det{}^2\left(1 + \exp\left[-\beta\left(H_\mathrm{s} - \mu\right)\right]\right) \tag{287}$$

$$\Omega_\mathrm{s}(\mu VT) = -k_\mathrm{B}T \ln \Xi_\mathrm{s}(\mu VT) \ , \tag{288}$$

where μ is the chemical potential acting on the electrons and the square of the

determinant stems from considering the spin–unpolarized special case only. This reduces to the well–known grand potential expression

$$\begin{aligned}\Omega_{\mathrm{s}}(\mu VT) &= -2k_{\mathrm{B}}T \ln\det\left(1+\exp\left[-\beta\left(H_{\mathrm{s}}-\mu\right)\right]\right) \\ &= -2k_{\mathrm{B}}T \sum_i \ln\left(1+\exp\left[-\beta\left(\epsilon_{\mathrm{s}}^{(i)}-\mu\right)\right]\right)\end{aligned} \tag{289}$$

for non–interacting spin–1/2 Fermions where $\{\epsilon_{\mathrm{s}}^{(i)}\}$ are the eigenvalues of a one–particle Hamiltonian such as Eq. (286); here the standard identity $\ln\det\mathbf{M} = \mathrm{Tr}\ln\mathbf{M}$ was invoked for positive definite $\mathbf{M}$.

According to thermodynamics the Helmholtz free energy $\mathcal{F}(NVT)$ associated to Eq. (288) can be obtained from an appropriate Legendre transformation of the grand free energy $\Omega(\mu VT)$

$$\mathcal{F}_{\mathrm{s}}(NVT) = \Omega_{\mathrm{s}}(\mu VT) + \mu N + \sum_{I<J} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} \tag{290}$$

by fixing the average number of electrons N and determining μ from the conventional thermodynamic condition

$$N = -\left(\frac{\partial\Omega}{\partial\mu}\right)_{VT} \quad . \tag{291}$$

In addition, the internuclear Coulomb interactions between the classical nuclei were included at this stage in Eq. (290). Thus, derivatives of the free energy Eq. (290) with respect to ionic positions $-\nabla_I\mathcal{F}_{\mathrm{s}}$ define forces on the nuclei that could be used in a (hypothetical) molecular dynamics scheme using non–interacting electrons.

The interactions between the electrons can be "switched on" by resorting to Kohn–Sham density functional theory and the concept of a non–interacting reference system. Thus, instead of using the simple one–particle Hamiltonian Eq. (286) the effective Kohn–Sham Hamiltonian Eq. (83) has to be utilized. As a result, the grand free energy Eq. (287) can be written as

$$\Omega^{\mathrm{KS}}(\mu VT) = -2k_{\mathrm{B}}T \ln\left[\det\left(1+\exp\left[-\beta\left(H^{\mathrm{KS}}-\mu\right)\right]\right)\right] \tag{292}$$

$$H^{\mathrm{KS}} = -\frac{1}{2}\nabla^2 - \sum_I \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}|} + V_{\mathrm{H}}(\mathbf{r}) + \frac{\delta\Omega_{\mathrm{xc}}[n]}{\delta n(\mathbf{r})} \tag{293}$$

$$H^{\mathrm{KS}}\phi_i = \epsilon_i\phi_i \tag{294}$$

where Ω_{xc} is the exchange–correlation functional at finite temperature. By virtue of Eq. (289) one can immediately see that Ω^{KS} is nothing else than the "Fermi–Dirac weighted sum" of the bare Kohn–Sham eigenvalues $\{\epsilon_i\}$. Whence, this term is the extension to finite temperatures of the "band–structure energy" (or of the "sum of orbital energies" in the analogues Hartree–Fock case [604,418]) contribution to the total electronic energy, see Eq. (86).

In order to obtain the correct total electronic free energy of the interacting electrons as defined in Eq. (86) the corresponding extra terms (properly generalized to finite temperatures) have to be included in Ω^{KS}. This finally allows one to write

down the generalization of the Helmholtz free energy of the interacting many–electron case

$$\mathcal{F}^{\mathrm{KS}}(NVT) = \Omega^{\mathrm{KS}}(\mu VT) + \mu \int d\mathbf{r}\ \ n(\mathbf{r}) + \sum_{I<J} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|}$$
$$-\frac{1}{2}\int d\mathbf{r}\ \ V_{\mathrm{H}}(\mathbf{r})\ n(\mathbf{r}) + \Omega_{\mathrm{xc}} - \int d\mathbf{r}\ \ \frac{\delta\Omega_{\mathrm{xc}}[n]}{\delta n(\mathbf{r})}\ n(\mathbf{r}) \qquad (295)$$

in the framework of a Kohn–Sham–like formulation. The corresponding one–particle density at the Γ–point is given by

$$n(\mathbf{r}) = \sum_i f_i(\beta)\ \ |\phi_i(\mathbf{r})|^2 \qquad (296)$$

$$f_i(\beta) = \left(1 + \exp\left[\beta\left(\epsilon_i - \mu\right)\right]\right)^{-1}\ \ , \qquad (297)$$

where the fractional occupation numbers $\{f_i\}$ are obtained from the Fermi–Dirac distribution at temperature T in terms of the Kohn–Sham eigenvalues $\{\epsilon_i\}$. Finally, *ab initio* forces can be obtained as usual from the nuclear gradient of $\mathcal{F}^{\mathrm{KS}}$, which makes molecular dynamics possible.

By construction, the total free energy Eq. (295) reduces to that of the non–interacting toy model Eq. (290) once the electron–electron interaction is switched off. Another useful limit is the ground–state limit $\beta \to \infty$ where the free energy $\mathcal{F}^{\mathrm{KS}}(NVT)$ yields the standard Kohn–Sham total energy expression E^{KS} as defined in Eq. (86) after invoking the appropriate limit $\Omega_{\mathrm{xc}} \to E_{\mathrm{xc}}$ as $T \to 0$. Most importantly, stability analysis [5,7] of Eq. (295) shows that this functional shares the same stationary point as the exact finite–temperature functional due to Mermin [424], see e.g. the textbooks [458,168] for introductions to density functional formalisms at finite temperatures. This implies that the self–consistent density, which defines the stationary point of $\mathcal{F}^{\mathrm{KS}}$, is identical to the exact one. This analysis reveals furthermore that, unfortunately, this stationary point is not an extremum but a saddle point so that no variational principle and, numerically speaking, no direct minimization algorithms can be applied. For the same reason a Car–Parrinello fictitious dynamics approach to molecular dynamics is not a straightforward option, whereas Born–Oppenheimer dynamics based on diagonalization can be used directly.

The band–structure energy term is evaluated in the `CPMD` package [142] by diagonalizing the Kohn–Sham Hamiltonian after a suitable "preconditioning" [5], see Sect. 3.6.2. Specifically, a second–order Trotter approximation is used

$$\mathrm{Tr}\ \exp\left[-\beta H^{\mathrm{KS}}\right] = \sum_i \exp\left[-\beta\epsilon_i\right] = \sum_i \rho_{ii}(\beta) \qquad (298)$$

$$= \mathrm{Tr}\ \left(\left\{\exp\left[-\frac{\Delta\tau}{2}\left(-\frac{1}{2}\nabla^2\right)\right]\exp\left[-\Delta\tau V^{\mathrm{KS}}[n]\right]\right.$$
$$\left.\exp\left[-\frac{\Delta\tau}{2}\left(-\frac{1}{2}\nabla^2\right)\right]\right\} + \mathcal{O}\left(\Delta\tau^3\right)\Bigg)^P \qquad (299)$$

$$\approx \sum_i \left\{\rho_{ii}(\Delta\tau)\right\}^P = \sum_i \left\{\exp\left[-\Delta\tau\epsilon_i\right]\right\}^P \qquad (300)$$

in order to compute first the diagonal elements $\rho_{ii}(\Delta\tau)$ of the "high–temperature" Boltzmann operator $\rho(\Delta\tau)$; here $\Delta\tau = \beta/P$ and P is the Trotter "time slice" as introduced in Sect. 4.4.2. To this end, the kinetic and potential energies can be conveniently evaluated in reciprocal and real space, respectively, by using the split–operator / FFT technique [183]. The Kohn–Sham eigenvalues ϵ_i are finally obtained from the density matrix via $\epsilon_i = -(1/\Delta\tau)\ln\rho_{ii}(\Delta\tau)$. They are used in order to compute the occupation numbers $\{f_i\}$, the density $n(\mathbf{r})$, the band–structure energy Ω^{KS}, and thus the free energy Eq. (295).

In practice a diagonalization / density–mixing scheme is employed in order to compute the self–consistent density $n(\mathbf{r})$. Grossly speaking a suitably constructed trial input density n_{in} (see e.g. the Appendix of Ref. [571] for such a method) is used in order to compute the potential $V^{\mathrm{KS}}[n_{\mathrm{in}}]$. Then the lowest–order approximant to the Boltzmann operator Eq. (300) is diagonalized using an iterative Lanczos–type method. This yields an output density n_{out} and the corresponding free energy $\mathcal{F}^{\mathrm{KS}}[n_{\mathrm{out}}]$. Finally, the densities are mixed and the former steps are iterated until a stationary solution n_{scf} of $\mathcal{F}^{\mathrm{KS}}[n_{\mathrm{scf}}]$ is achieved, see Sect. 3.6.4 for some details on such methods. Of course the most time–consuming part of the calculation is in the iterative diagonalization. In principle this is not required, and it should be possible to compute the output density directly from the Fermi–Dirac density matrix even in a linear scaling scheme [243], thus circumventing the explicit calculation of the Kohn–Sham eigenstates. However, to date efforts in this direction have failed, or given methods which are too slow to be useful [9].

As a method, molecular dynamics with the free energy functional is most appropriate to use when the excitation gap is either small, or in cases where the gap might close during a chemical transformation. In the latter case no instabilities are encountered with this approach, which is not true for ground–state *ab initio* molecular dynamics methods. The price to pay is the quite demanding iterative computation of well–converged forces. Besides allowing such applications with physically relevant excitations this method can also be straightforwardly combined with **k**–point sampling and applied to metals at "zero" temperature. In this case, the electronic "temperature" is only used as a smearing parameter of the Fermi edge by introducing fractional occupation numbers, which is known to improve greatly the convergence of these ground–state electronic structure calculations [220,232,185,676,680,343,260,344,414,243].

Finite–temperature expressions for the exchange–correlation functional Ω_{xc} are available in the literature. However, for most temperatures of interest the corrections to the ground–state expression are small and it seems justified to use one of the various well–established parameterizations of the exchange–correlation energy E_{xc} at zero temperature, see Sect. 2.7.

4.3.3 A Single Excited State: S_1–Dynamics

For large–gap systems with well separated electronic states it might be desirable to single out a particular state in order to allow the nuclei to move on the associated excited state potential energy surface. Approaches that rely on fractional occupation numbers such as ensemble density functional theories – including the

free energy functional discussed in the previous section – are difficult to adapt for cases where the symmetry and / or spin of the electronic state should be fixed [168]. An early approach in order to select a particular excited state was based on introducing a "quadratic restoring potential" which vanishes only at the eigenvalue of the particular state [417,111].

A method that combines Roothaan's symmetry–adapted wavefunctions with Kohn–Sham density functional theory was proposed in Ref. [214] and used to simulate a photoisomerization via molecular dynamics. Viewed from Kohn–Sham theory this approach consists in building up the spin density of an open–shell system based on a symmetry–adapted wavefunction that is constructed from spin–restricted determinants (the "microstates"). Viewed from the restricted open–shell Hartree–Fock theory *à la* Roothaan it amounts essentially to replacing Hartree–Fock exchange by an approximate exchange–correlation density functional. This procedure leads to an orbital–dependent density functional which was formulated explicitely for the first–excited singlet state S_1 in Ref. [214]. The relation of this approach to previous theories is discussed in some detail in Ref. [214]. In particular, the success of the closely–related Ziegler–Rauk–Baerends "sum methods" [704,150,600] was an important stimulus. More recently several papers [252,439,193,195,196] appeared that are similar in spirit to the method of Ref. [214]. The approach of Refs. [193,195,196] can be viewed as a generalization of the special case (S_1 state) worked out in Ref. [214] to arbitrary spin states. In addition, the generalized method [193,195,196] was derived within the framework of density functional theory, whereas the wavefunction perspective was the starting point in Ref. [214].

In the following, the method is outlined with the focus to perform molecular dynamics in the S_1 state. Promoting one electron from the HOMO to the LUMO in a closed–shell system with $2n$ electrons assigned to n doubly occupied orbitals (that is spin–restricted orbitals that have the same spatial part for both spin up α and spin down β electrons) leads to four different excited wavefunctions or determinants, see Fig. 15 for a sketch. Two states $|t_1\rangle$ and $|t_2\rangle$ are energetically degenerate triplets t whereas the two states $|m_1\rangle$ and $|m_2\rangle$ are not eigenfunctions of the total spin operator and thus degenerate mixed states m ("spin contamination"). Note in particular that the m states do not correspond – as is well known – to singlet states despite the suggestive occupation pattern in Fig. 15.

However, suitable Clebsch–Gordon projections of the mixed states $|m_1\rangle$ and $|m_2\rangle$ yield another triplet state $|t_3\rangle$ and the desired first excited singlet or S_1 state $|s_1\rangle$. Here, the ansatz [214] for the total energy of the S_1 state is given by

$$E_{S_1}\left[\{\phi_i\}\right] = 2E_m^{\mathrm{KS}}\left[\{\phi_i\}\right] - E_t^{\mathrm{KS}}\left[\{\phi_i\}\right] \tag{301}$$

where the energies of the mixed and triplet determinants

$$E_m^{\mathrm{KS}}[\{\phi_i\}] = T_{\mathrm{s}}[n] + \int d\mathbf{r}\;\; V_{\mathrm{ext}}(\mathbf{r})n(\mathbf{r}) + \frac{1}{2}\int d\mathbf{r}\;\; V_{\mathrm{H}}(\mathbf{r})n(\mathbf{r}) + E_{\mathrm{xc}}[n_m^\alpha, n_m^\beta] \tag{302}$$

$$E_t^{\mathrm{KS}}[\{\phi_i\}] = T_{\mathrm{s}}[n] + \int d\mathbf{r}\;\; V_{\mathrm{ext}}(\mathbf{r})n(\mathbf{r}) + \frac{1}{2}\int d\mathbf{r}\;\; V_{\mathrm{H}}(\mathbf{r})n(\mathbf{r}) + E_{\mathrm{xc}}[n_t^\alpha, n_t^\beta] \tag{303}$$

are expressed in terms of (restricted) Kohn–Sham spin–density functionals constructed from the set $\{\phi_i\}$, cf. Eq. (75). The associated S_1 wavefunction is given

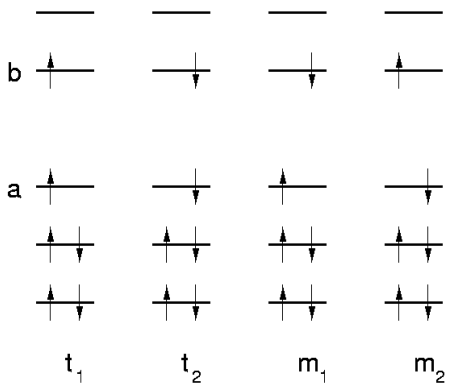


Figure 15. Four possible determinants $|t_1\rangle$, $|t_2\rangle$, $|m_1\rangle$ and $|m_2\rangle$ as a result of the promotion of a single electron from the HOMO to the LUMO of a closed shell system, see text for further details. Taken from Ref. [214].

by

$$|\,s_1\,[\{\phi_i\}]\rangle = \sqrt{2}\,|\,m\,[\{\phi_i\}]\rangle - |\,t\,[\{\phi_i\}]\rangle \tag{304}$$

where the "microstates" m and t are both constructed from the same set $\{\phi_i\}$ of $n+1$ spin–*restricted* orbitals. Using this particular set of orbitals the total density

$$n(\mathbf{r}) = n_m^\alpha(\mathbf{r}) + n_m^\beta(\mathbf{r}) = n_t^\alpha(\mathbf{r}) + n_t^\beta(\mathbf{r}) \tag{305}$$

is of course identical for both the m and t determinants whereas their spin densities clearly differ, see Fig. 16. Thus, the decisive difference between the m and t functionals Eq. (302) and Eq. (303), respectively, comes exclusively from the exchange–correlation functional $E_{\rm xc}$, whereas kinetic, external and Hartree energy are identical by construction. Note that this basic philosophy can be generalized to other spin–states by adapting suitably the microstates and the corresponding coefficients in Eq. (301) and Eq. (304).

Having defined a density functional for the first excited singlet state the corresponding Kohn–Sham equations are obtained by varying Eq. (301) using Eq. (302) and Eq. (303) subject to the orthonormality constraint $\sum_{i,j=1}^{n+1}\Lambda_{ij}(\langle\phi_i\mid\phi_j\rangle - \delta_{ij})$. Following this procedure the equation for the doubly occupied orbitals $i = 1,\ldots,n-1$ reads

$$\begin{aligned}
&\Big\{ -\frac{1}{2}\nabla^2 + V_{\rm H}(\mathbf{r}) + V_{\rm ext}(\mathbf{r}) \\
&+ V_{\rm xc}^\alpha[n_m^\alpha(\mathbf{r}), n_m^\beta(\mathbf{r})] + V_{\rm xc}^\beta[n_m^\alpha(\mathbf{r}), n_m^\beta(\mathbf{r})] \\
&- \frac{1}{2}V_{\rm xc}^\alpha[n_t^\alpha(\mathbf{r}), n_t^\beta(\mathbf{r})] - \frac{1}{2}V_{\rm xc}^\beta[n_t^\alpha(\mathbf{r}), n_t^\beta(\mathbf{r})]\Big\}\phi_i(\mathbf{r}) = \sum_{j=1}^{n+1}\Lambda_{ij}\phi_j(\mathbf{r})
\end{aligned} \tag{306}$$

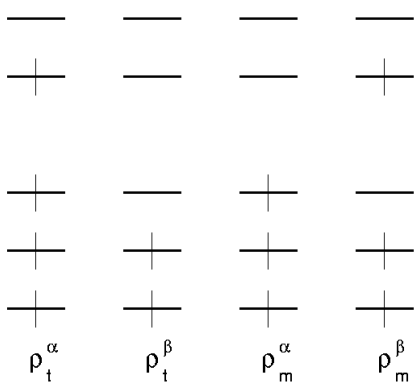


Figure 16. Four patterns of spin densities n_t^α, n_t^β, n_m^α, and n_m^β corresponding to the two spin–restricted determinants $|t\rangle$ and $|m\rangle$ sketched in Fig. 15, see text for further details. Taken from Ref. [214].

whereas

$$\Big\{ \frac{1}{2}\Big[-\frac{1}{2}\nabla^2 + V_{\rm H}(\mathbf{r}) + V_{\rm ext}(\mathbf{r})\Big] + V_{\rm xc}^\alpha[n_m^\alpha(\mathbf{r}), n_m^\beta(\mathbf{r})] - \frac{1}{2}V_{\rm xc}^\alpha[n_t^\alpha(\mathbf{r}), n_t^\beta(\mathbf{r})]\Big\}\phi_{\rm a}(\mathbf{r}) = \sum_{j=1}^{n+1}\Lambda_{aj}\phi_j(\mathbf{r}) \ , \quad (307)$$

and

$$\Big\{ \frac{1}{2}\Big[-\frac{1}{2}\nabla^2 + V_{\rm H}(\mathbf{r}) + V_{\rm ext}(\mathbf{r})\Big] + V_{\rm xc}^\beta[n_m^\alpha(\mathbf{r}), n_m^\beta(\mathbf{r})] - \frac{1}{2}V_{\rm xc}^\alpha[n_t^\alpha(\mathbf{r}), n_t^\beta(\mathbf{r})]\Big\}\phi_{\rm b}(\mathbf{r}) = \sum_{j=1}^{n+1}\Lambda_{bj}\phi_j(\mathbf{r}) \ . \quad (308)$$

are two *different* equations for the two singly–occupied open–shell orbitals a and b, respectively, see Fig. 15. Note that these Kohn–Sham–like equations feature an orbital–dependent exchange–correlation potential where $V_{\rm xc}^\alpha[n_m^\alpha, n_m^\beta] = \delta E_{\rm xc}[n_m^\alpha, n_m^\beta]/\delta n_m^\alpha$ and analogues definitions hold for the β and t cases.

The set of equations Eq. (306)–(308) could be solved by diagonalization of the corresponding "restricted open–shell Kohn–Sham Hamiltonian" or alternatively by direct minimization of the associated total energy functional. The algorithm proposed in Ref. [240], which allows to properly and efficiently minimize such orbital–dependent functionals including the orthonormality constraints, was implemented in the `CPMD` package [142]. Based on this minimization technique Born–Oppenheimer molecular dynamics simulations can be performed in the first excited singlet state.

The alternative Car–Parrinello formulation seems inconvenient because the singly and doubly occupied orbitals would have to be constrained not to mix.

4.4 Beyond Classical Nuclei

4.4.1 Introduction

Up to this point the nuclei were approximated as classical point particles as customarily done in standard molecular dynamics. There are, however, many situations where quantum dispersion broadening and tunneling effects play an important role and cannot be neglected if the simulation aims at being realistic – which is the generic goal of *ab initio* simulations. The *ab initio* path integral technique [395] and its extension to quasiclassical time evolution [411] is able to cope with such situations at finite temperatures. It is also implemented in the `CPMD` package [142]. The central idea is to quantize the nuclei using Feynman's path integrals and at the same time to include the electronic degrees of freedom akin to *ab initio* molecular dynamics – that is "on–the–fly". The main ingredients and approximations underlying the *ab initio* path integral approach [395,399,644,404] are

- the adiabatic separation of electrons and nuclei where the electrons are kept in their ground state without any coupling to electronically excited states (Born–Oppenheimer or "clamped–nuclei" approximation),
- using a particular approximate electronic structure theory in order to calculate the interactions,
- approximating the continuous path integral for the nuclei by a finite discretization (Trotter factorization) and neglecting the indistinguishability of identical nuclei (Boltzmann statistics), and
- using finite supercells with periodic boundary conditions and finite sampling times (finite–size and finite–time effects) as usual.

Thus, quantum effects such as zero–point motion and tunneling as well as thermal fluctuations are included at some preset temperature without further simplifications consisting e.g. in quasiclassical or quasiharmonic approximations, restricting the Hilbert space, or in artificially reducing the dimensionality of the problem.

4.4.2 Ab Initio Path Integrals: Statics

For the purpose of introducing *ab initio* path integrals [395] it is convenient to start directly with Feynman's formulation of quantum–statistical mechanics in terms of path integrals as opposed to Schrödinger's formulation in terms of wavefunctions which was used in Sect. 2.1 in order to derive *ab initio* molecular dynamics. For a general introduction to path integrals the reader is referred to standard textbooks [187,188,334], whereas their use in numerical simulations is discussed for instance in Refs. [233,126,542,120,124,646,407].

The derivation of the expressions for *ab initio* path integrals is based on assuming the non–relativistic standard Hamiltonian, see Eq. (2). The corresponding canonical partition function of a collection of interacting nuclei with positions $\mathbf{R} = \{\mathbf{R}_I\}$

and electrons $\mathbf{r} = \{\mathbf{r}_i\}$ can be written as a path integral

$$\mathcal{Z} = \oint' \mathcal{D}\mathbf{R} \oint' \mathcal{D}\mathbf{r} \exp\left[-\frac{1}{\hbar}\int_0^{\hbar\beta} d\tau\, \mathcal{L}_{\mathrm{E}}\left(\{\dot{\mathbf{R}}_I(\tau)\},\{\mathbf{R}_I(\tau)\};\{\dot{\mathbf{r}}_i(\tau)\},\{\mathbf{r}_i(\tau)\}\right)\right] \quad (309)$$

where

$$\begin{aligned}\mathcal{L}_{\mathrm{E}} &= T(\dot{\mathbf{R}}) + V(\mathbf{R}) + T(\dot{\mathbf{r}}) + V(\mathbf{r}) + V(\mathbf{R},\mathbf{r}) \\ &= \sum_I \frac{1}{2} M_I \left(\frac{d\mathbf{R}_I}{d\tau}\right)^2 + \sum_{I<J} \frac{\mathrm{e}^2 Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} \\ &\quad + \sum_i \frac{1}{2} m_{\mathrm{e}} \left(\frac{d\mathbf{r}_i}{d\tau}\right)^2 + \sum_{i<j} \frac{\mathrm{e}^2}{|\mathbf{r}_i - \mathbf{r}_j|} - \sum_{I,i} \frac{\mathrm{e}^2 Z_I}{|\mathbf{R}_I - \mathbf{r}_i|} \ , \end{aligned} \quad (310)$$

denotes the *Euclidean* Lagrangian. The primes in Eq. (309) indicate that the proper sums over all permutations corresponding to Bose–Einstein and/or Fermi–Dirac statistics have to be included. It is important to note that in Eq. (309) and (310) the positions $\{\mathbf{R}_I\}$ and $\{\mathbf{r}_i\}$ are not operators but simply *functions* of the imaginary time $\tau \in [0, \hbar\beta]$ which parameterizes fluctuations around the classical path. This implies that the dots denote here derivatives with respect to imaginary time τ as defined in Eq. (310). According to Eq. (309) exact quantum mechanics at finite temperature $T = 1/k_{\mathrm{B}}\beta$ is recovered if all closed paths $[\{\mathbf{R}_I\};\{\mathbf{r}_i\}]$ of "length" $\hbar\beta$ are summed up and weighted with the exponential of the Euclidean action measured in units of $\hbar$; atomic units will be used again from here on. The partial trace over the electronic subsystem can formally be written down exactly

$$\mathcal{Z} = \oint' \mathcal{D}\mathbf{R} \exp\left[-\int_0^{\beta} d\tau \left(T(\dot{\mathbf{R}}) + V(\mathbf{R})\right)\right] \mathcal{Z}\left[\mathbf{R}\right] \ , \quad (311)$$

with the aid of the influence functional [187,334]

$$\mathcal{Z}[\mathbf{R}] = \oint' \mathcal{D}\mathbf{r} \exp\left[-\int_0^{\beta} d\tau \left(T(\dot{\mathbf{r}}) + V(\mathbf{r}) + V(\mathbf{R},\mathbf{r})\right)\right] \ . \quad (312)$$

Note that $\mathcal{Z}[\mathbf{R}]$ is a complicated and unknown functional for a given nuclear path configuration $[\{\mathbf{R}_I\}]$. As a consequence the interactions between the nuclei become highly nonlocal in imaginary time due to memory effects.

In the standard Born–Oppenheimer or "clamped nuclei" approximation, see Ref. [340] for instance, the nuclei are frozen in some configuration and the complete electronic problem is solved for this single static configuration. In addition to the nondiagonal correction terms that are already neglected in the adiabatic approximation, the diagonal terms are now neglected as well. Thus the potential for the nuclear motion is simply defined as the bare electronic eigenvalues obtained from a series of fixed nuclear configurations.

In the statistical mechanics formulation of the problem Eq. (311)–(312) the Born–Oppenheimer approximation amounts to a "quenched average": at imaginary time τ the nuclei are frozen at a particular configuration $\mathbf{R}(\tau)$ and the electrons explore their configuration space subject only to that single configuration. This

implies that the electronic degrees of freedom at different imaginary times τ and τ' become completely decoupled. Thus, the influence functional $\mathcal{Z}[\mathbf{R}]$ has to be local in τ and becomes particularly simple; a discussion of adiabatic corrections in the path integral formulation can be found in Ref. [101]. For each τ the influence functional $\mathcal{Z}[\mathbf{R}]$ is given by the partition function of the electronic subsystem evaluated for the respective nuclear configuration $\mathbf{R}(\tau)$. Assuming that the temperature is small compared to the gap in the electronic spectrum only the electronic ground state with energy $E_0\left(\mathbf{R}(\tau)\right)$ (obtained from solving Eq. (20) without the internuclear Coulomb repulsion term) is populated. This electronic ground state dominance leads to the following simple expression

$$\mathcal{Z}[\mathbf{R}]_{\mathrm{BO}} = \exp\left[-\int_0^\beta d\tau E_0(\mathbf{R}(\tau))\right] \quad , \tag{313}$$

which yields the final result

$$\mathcal{Z}_{\mathrm{BO}} = \oint \mathcal{D}\mathbf{R} \exp\left[-\int_0^\beta d\tau \left(T(\dot{\mathbf{R}}) + V(\mathbf{R}) + E_0(\mathbf{R})\right)\right] \quad . \tag{314}$$

Here nuclear exchange is neglected by assuming that the nuclei are distinguishable so that they can be treated within Boltzmann statistics, which corresponds to the Hartree approximation for the *nuclear* density matrix. The presentation given here follows Ref. [399] and alternative derivations were given in Sect. 2.3 of Refs. [124] and in the appendix of Ref. [427]. There, a wavefunction basis instead of the position basis as in Eq. (312) was formally used in order to evaluate the influence functional due to the electrons.

The partition function Eq. (314) together with the Coulomb Hamiltonian Eq. (2) leads after applying the lowest–order Trotter factorization [334] to the following discretized expression

$$\begin{aligned}\mathcal{Z}_{\mathrm{BO}} = \lim_{P\to\infty} \prod_{s=1}^{P} \prod_{I=1}^{N} \left[\left(\frac{M_I P}{2\pi\beta}\right)^{3/2} \int d\mathbf{R}_I^{(s)}\right] \\ \times \exp\left[-\beta \sum_{s=1}^{P} \left\{\sum_{I=1}^{N} \frac{1}{2} M_I \omega_P^2 \left(\mathbf{R}_I^{(s)} - \mathbf{R}_I^{(s+1)}\right)^2 + \frac{1}{P} E_0\left(\{\mathbf{R}_I\}^{(s)}\right)\right\}\right]\end{aligned} \tag{315}$$

for the path integral with $\omega_P^2 = P/\beta^2$ Thus, the continuous parameter $\tau \in [0,\beta]$ is discretized using P so–called Trotter slices or "time slices" $s = 1,\ldots,P$ of "duration" $\Delta\tau = \beta/P$. The paths

$$\begin{aligned}\left\{\{\mathbf{R}_I\}^{(s)}\right\} &= \left(\{\mathbf{R}_I\}^{(1)};\ldots;\{\mathbf{R}_I\}^{(P)}\right) \\ &= \left(\mathbf{R}_1^{(1)},\ldots,\mathbf{R}_N^{(1)};\ldots;\mathbf{R}_1^{(P)},\ldots,\mathbf{R}_N^{(P)}\right)\end{aligned} \tag{316}$$

have to be closed due to the trace condition, i.e. they are periodic in imaginary time τ which implies $\mathbf{R}_I(0) \equiv \mathbf{R}_I(\beta)$ and thus $\mathbf{R}_I^{(P+1)} = \mathbf{R}_I^{(1)}$; the internuclear Coulomb repulsion $V(\mathbf{R})$ is now included in the definition of the total electronic energy E_0. Note that Eq. (315) is an *exact* reformulation of Eq. (314) in the limit of an infinitely fine discretization $P \to \infty$ of the paths.

The effective classical partition function Eq. (315) with a fixed discretization P is isomorphic to that for N polymers each comprised by P monomers [233,126,120]. Each quantum degree of freedom is found to be represented by a ring polymer or necklace. The intrapolymeric interactions stem from the kinetic energy $T(\dot{\mathbf{R}})$ and consist of harmonic nearest–neighbor couplings $\propto \omega_P$ along the closed chain. The interpolymeric interaction is given by the scaled potential $E_0^{(s)}/P$ which is only evaluated for configurations $\{\mathbf{R}_I\}^{(s)}$ at the *same* imaginary time slice s.

In order to evaluate operators based on an expression like Eq. (315) most numerical path integral schemes utilize Metropolis Monte Carlo sampling with the effective potential

$$V_{\text{eff}} = \sum_{s=1}^{P} \left\{ \sum_{I=1}^{N} \frac{1}{2} M_I \omega_P^2 \left(\mathbf{R}_I^{(s)} - \mathbf{R}_I^{(s+1)} \right)^2 + \frac{1}{P} E_0 \left(\{\mathbf{R}_I\}^{(s)} \right) \right\} \tag{317}$$

of the isomorphic classical system [233,126,542,120,124,646,407]. Molecular dynamics techniques were also proposed in order to sample configuration space, see Refs. [99,490,462,501,273] for pioneering work and Ref. [646] for an authoritative review. Formally a Lagrangian can be obtained from the expression Eq. (317) by extending it

$$\mathcal{L}_{\text{PIMD}} = \sum_{s=1}^{P} \left\{ \sum_{I=1}^{N} \left(\frac{1}{2M_I'} \mathbf{P}_I^{(s)} - \frac{1}{2} M_I \omega_P^2 \left(\mathbf{R}_I^{(s)} - \mathbf{R}_I^{(s+1)} \right)^2 \right) - \frac{1}{P} E_0 \left(\{\mathbf{R}_I\}^{(s)} \right) \right\} \tag{318}$$

with $N \times P$ fictitious momenta $\mathbf{P}_I^{(s)}$ and corresponding (unphysical) fictitious masses M_I'. At this stage the time dependence of positions and momenta and thus the time evolution in *phase space* as generated by Eq. (318) has no physical meaning. The sole use of "time" is to parameterize the deterministic dynamical exploration of *configuration space*. The trajectories of the positions in configuration space, can, however, be analyzed similar to the ones obtained from the stochastic dynamics that underlies the Monte Carlo method.

The crucial ingredient in *ab initio* [395,399,644,404] as opposed to standard [233,126,542,120,124,646,407] path integral simulations consists in computing the interactions E_0 "on–the–fly" like in *ab initio* molecular dynamics. In analogy to this case both the Car–Parrinello and Born–Oppenheimer approaches from Sects. 2.4 and 2.3, respectively, can be combined with any electronic structure method. The first implementation [395] was based on the Car–Parrinello / density functional combination from Sect. 2.4 which leads to the following extended Lagrangian

$$\begin{aligned} \mathcal{L}_{\text{AIPI}} = \frac{1}{P} \sum_{s=1}^{P} \Bigg\{ & \sum_i \mu \left\langle \dot{\phi}_i^{(s)} \middle| \dot{\phi}_i^{(s)} \right\rangle - E^{\text{KS}} \left[\{\phi_i\}^{(s)}, \{\mathbf{R}_I\}^{(s)} \right] \\ & + \sum_{ij} \Lambda_{ij}^{(s)} \left(\left\langle \phi_i^{(s)} \middle| \phi_j^{(s)} \right\rangle - \delta_{ij} \right) \Bigg\} \\ & + \sum_{s=1}^{P} \left\{ \sum_I \frac{1}{2} M_I' \left(\dot{\mathbf{R}}_I^{(s)} \right)^2 - \sum_{I=1}^{N} \frac{1}{2} M_I \omega_P^2 \left(\mathbf{R}_I^{(s)} - \mathbf{R}_I^{(s+1)} \right)^2 \right\} , \end{aligned} \tag{319}$$

where the interaction energy $E^{\mathrm{KS}}[\{\phi_i\}^{(s)}, \{\mathbf{R}_I\}^{(s)}]$ at time slice s is defined in Eq. (75); note that here and in the following the dots denote derivatives with respect to propagation time t and that $E_0^{\mathrm{KS}} = \min E^{\mathrm{KS}}$. The standard Car–Parrinello Lagrangian, see e.g. Eq. (41) or Eq. (58), is recovered in the limit $P = 1$ which corresponds to classical nuclei. Mixed classical / quantum systems can easily be treated by representing an arbitrary subset of the nuclei in Eq. (319) with only one time slice.

This simplest formulation of *ab initio* path integrals, however, is insufficient for the following reason: ergodicity of the trajectories and adiabaticity in the sense of Car–Parrinello simulations are not guaranteed. It is known since the very first molecular dynamics computer experiments that quasiharmonic systems (such as coupled stiff harmonic oscillators subject to weak anharmonicities, i.e. the famous Fermi–Pasta–Ulam chains) can easily lead to nonergodic behavior in the sampling of phase space [210]. Similarly "microcanonical" path integral molecular dynamics simulations might lead to an insufficient exploration of configuration space depending on the parameters [273]. The severity of this nonergodicity problem is governed by the stiffness of the harmonic intrachain coupling $\propto \omega_P$ and the anharmonicity of the overall potential surface $\propto E^{\mathrm{KS}}/P$ which establishes the coupling of the modes. For a better and better discretization P the harmonic energy term dominates according to $\sim P$ whereas the mode–mixing coupling decreases like $\sim 1/P$. This problem can be cured by attaching Nosé–Hoover chain thermostats [388], see also Sect. 4.2, to all path integral degrees of freedom [637,644].

The second issue is related to the separation of the power spectra associated to nuclear and electronic subsystems during Car–Parrinello *ab initio* molecular dynamics which is instrumental for maintaining adiabaticity, see Sect. 2.4. In *ab initio* molecular dynamics with classical nuclei the highest phonon or vibrational frequency $\omega_{\mathrm{n}}^{\max}$ is dictated by the physics of the system, see e.g. Fig. 2. This means in particular that an upper limit is given by stiff intramolecular vibrations which do not exceed $\omega_{\mathrm{n}}^{\max} \leq 5000$ cm^{-1} or 150 THz. In *ab initio* path integral simulations, on the contrary, $\omega_{\mathrm{n}}^{\max}$ is given by ω_P which actually diverges with increasing discretization as $\sim \sqrt{P}$. The simplest counteraction would be to compensate this artifact by decreasing the fictitious electron mass μ until the power spectra are again separated for a fixed value of P and thus ω_P. This, however, would lead to a prohibitively small time step because $\Delta t^{\max} \propto \sqrt{\mu}$. This dilemma can be solved by thermostatting the electronic degrees of freedom as well [395,399,644], see Sect. 4.2 for a related discussion in the context of metals.

Finally, it is known that diagonalizing the harmonic spring interaction in Eq. (319) leads to more efficient propagators [637,644]. One of these transformation and the resulting Nosé–Hoover chain thermostatted equations of motion will be outlined in the following section, see in particular Eqs. (331)–(337). In addition to keeping the average temperature fixed it is also possible to generate path trajectories in the isobaric–isothermal NpT ensemble [646,392]. Instead of using Car–Parrinello fictitious dynamics in order to evaluate the interaction energy in Eq. (318), which is implemented in the `CPMD` package [142], it is evident that also the Born–Oppenheimer approach from Sect. 2.3 or the free energy functional from Sect. 4.3 can be used. This route eliminates the adiabaticity problem and was taken

up e.g. in Refs. [132,37,596,597,428,429,333].

A final observation concerning parallel supercomputers might be useful, see also Sect. 3.9. It is evident from the Lagrangian Eq. (319) and the resulting equations of motion (e.g. Eqs. (331)–(337)) that most of the numerical workload comes from calculating the *ab initio* forces on the nuclei. Given a fixed path configuration Eq. (316) the P underlying electronic structure problems are independent from each other and can be solved without communication on P nodes of a distributed memory machine. Communication is only necessary to send the final result, essentially the forces, to a special node that computes the quantum kinetic contribution to the energy and integrates finally the equations of motions. It is even conceivable to distribute this task on different supercomputers, i.e. "meta–computing" is within reach for such calculations. Thus, the algorithm is "embarrassingly parallel" provided that the memory per node is sufficient to solve the complete Kohn–Sham problem at a given time slice. If this is not the case the electronic structure calculation itself has to be parallelized on another hierarchical level as outlined in Sect. 3.9.

4.4.3 Ab Initio Path Centroids: Dynamics

Initially the molecular dynamics approach to path integral simulations was invented merely as a trick in order to sample configuration space similar to the Monte Carlo method. This perception changed recently with the introduction of the so–called "centroid molecular dynamics" technique [102], see Refs. [103,104,105,665,505,506,507] for background information. In a nutshell it is found that the time evolution of the centers of mass or centroids

$$\mathbf{R}_I^c(t) = \frac{1}{P} \sum_{s'=1}^{P} \mathbf{R}_I^{(s')}(t) \tag{320}$$

of the closed Feynman paths that represent the quantum nuclei contains quasiclassical information about the true quantum dynamics. The centroid molecular dynamics approach can be shown to be exact for harmonic potentials and to have the correct classical limit. The path centroids move in an effective potential which is generated by all the other modes of the paths at the given temperature. This effective potential thus includes the effects of quantum fluctuations on the (quasiclassical) time evolution of the centroid degrees of freedom. Roughly speaking the trajectory of the path centroids can be regarded as a classical trajectory of the system, which is approximately "renormalized" due to quantum effects.

The original centroid molecular dynamics technique [102,103,104,105,665] relies on the use of model potentials as the standard time–independent path integral simulations. This limitation was overcome independently in Refs. [469,411] by combining *ab initio* path integrals with centroid molecular dynamics. The resulting technique, *ab initio* centroid molecular dynamics can be considered as a quasiclassical generalization of standard *ab initio* molecular dynamics. At the same time, it preserves the virtues of the *ab initio* path integral technique [395,399,644,404] to generate exact time–independent quantum equilibrium averages.

Here, the so–called adiabatic formulation [105,390,106] of *ab initio* centroid molecular dynamics [411] is discussed. In close analogy to *ab initio* molecular dynamics with classical nuclei also the effective centroid potential is generated "on–the–fly" as the centroids are propagated. This is achieved by singling out the centroid coordinates in terms of a normal mode transformation [138] and accelerating the dynamics of all non–centroid modes artificially by assigning appropriate fictitious masses. At the same time, the fictitious electron dynamics *à la* Car–Parrinello is kept in order to calculate efficiently the *ab initio* forces on *all* modes from the electronic structure. This makes it necessary to maintain two levels of adiabaticity in the course of simulations, see Sect. 2.1 of Ref. [411] for a theoretical analysis of that issue.

The partition function Eq. (315), formulated in the so-called "primitive" path variables $\{\mathbf{R}_I\}^{(s)}$, is first transformed [644,646] to a representation in terms of the normal modes $\{\mathbf{u}_I\}^{(s)}$, which diagonalize the harmonic nearest–neighbor harmonic coupling [138]. The transformation follows from the Fourier expansion of a cyclic path

$$\mathbf{R}_I^{(s)} = \sum_{s'=1}^{P} \mathbf{a}_I^{(s')} \exp\left[2\pi i (s-1)(s'-1)/P\right] \quad , \tag{321}$$

where the coefficients $\{\mathbf{a}_I\}^{(s)}$ are complex. The normal mode variables $\{\mathbf{u}_I\}^{(s)}$ are then given in terms of the expansion coefficients according to

$$\begin{aligned} \mathbf{u}_I^{(1)} &= \mathbf{a}_I^{(1)} \\ \mathbf{u}_I^{(P)} &= \mathbf{a}_I^{((P+2)/2)} \\ \mathbf{u}_I^{(2s-2)} &= \mathrm{Re}\ (\mathbf{a}_I^{(s)}) \\ \mathbf{u}_I^{(2s-1)} &= \mathrm{Im}\ (\mathbf{a}_I^{(s)}) \quad . \end{aligned} \tag{322}$$

Associated with the normal mode transformation is a set of normal mode frequencies $\{\lambda\}^{(s)}$ given by

$$\lambda^{(2s-1)} = \lambda^{(2s-2)} = 2P\left[1 - \cos\left(\frac{2\pi(s-1)}{P}\right)\right] \tag{323}$$

with $\lambda^{(1)} = 0$ and $\lambda^{(P)} = 4P$. Equation (321) is equivalent to direct diagonalization of the matrix

$$\mathbf{A}_{ss'} = 2\delta_{ss'} - \delta_{s,s'-1} - \delta_{s,s'+1} \tag{324}$$

with the path periodicity condition $\mathbf{A}_{s0} = \mathbf{A}_{sP}$ and $\mathbf{A}_{s,P+1} = \mathbf{A}_{s1}$ and subsequent use of the unitary transformation matrix $\mathbf{U}$ to transform from the "primitive" variables $\{\mathbf{R}_I\}^{(s)}$ to the normal mode variables $\{\mathbf{u}_I\}^{(s)}$

$$\begin{aligned} \mathbf{R}_I^{(s)} &= \sqrt{P} \sum_{s'=1}^{P} \mathbf{U}_{ss'}^{\dagger} \mathbf{u}_I^{(s')} \\ \mathbf{u}_I^{(s)} &= \frac{1}{\sqrt{P}} \sum_{s'=1}^{P} \mathbf{U}_{ss'} \mathbf{R}_I^{(s')} \quad . \end{aligned} \tag{325}$$

The eigenvalues of $\mathbf{A}$ when multiplied by P are precisely the normal mode frequencies $\{\lambda\}^{(s)}$. Since the transformation is unitary, its Jacobian is unity. Finally, it is convenient to define a set of normal mode masses

$$M_I^{(s)} = \lambda^{(s)} M_I \tag{326}$$

that vary along the imaginary time axis $s = 1, \ldots, P$, where $\lambda^{(1)} = 0$ for the centroid mode $\mathbf{u}_I^{(1)}$.

Based on these transformations the Lagrangian corresponding to the *ab initio* path integral expressed in normal modes is obtained [644]

$$\begin{aligned}\mathcal{L}_{\text{AIPI}} = \frac{1}{P}\sum_{s=1}^{P}\Bigg\{ & \sum_i \mu \left\langle \dot{\phi}_i^{(s)} \middle| \dot{\phi}_i^{(s)} \right\rangle - E\left[\{\phi_i\}^{(s)}, \left\{\mathbf{R}_I\left(\mathbf{u}_I^{(1)}, \ldots, \mathbf{u}_I^{(P)}\right)\right\}^{(s)}\right] \\ & + \sum_{ij} \Lambda_{ij}^{(s)} \left(\left\langle \phi_i^{(s)} \middle| \phi_j^{(s)} \right\rangle - \delta_{ij} \right) \Bigg\} \\ & + \sum_{s=1}^{P} \Bigg\{ \sum_{I=1}^{N} \frac{1}{2} M_I'^{(s)} \left(\dot{\mathbf{u}}_I^{(s)}\right)^2 - \sum_{I=1}^{N} \frac{1}{2} M_I^{(s)} \omega_P^2 \left(\mathbf{u}_I^{(s)}\right)^2 \Bigg\} \ , \end{aligned} \tag{327}$$

where the masses $M_I'^{(s)}$ will be defined later, see Eq. (338). As indicated, the electronic energy $E^{(s)}$ is always evaluated in practice in terms of the "primitive" path variables $\{\mathbf{R}_I\}^{(s)}$ in Cartesian space. The necessary transformation to switch forth and back between "primitive" and normal mode variables is easily performed as given by the relations Eq. (325).

The chief advantage of the normal mode representation Eq. (325) for the present purpose is that the lowest–order normal mode $\mathbf{u}_I^{(1)}$

$$\mathbf{u}_I^{(1)} = \mathbf{R}_I^{\text{c}} = \frac{1}{P}\sum_{s'=1}^{P} \mathbf{R}_I^{(s')} \tag{328}$$

turns out to be identical to the centroid $\mathbf{R}_I^{\text{c}}$ of the path that represents the Ith nucleus. The centroid force can also be obtained from the matrix $\mathbf{U}$ according to [644]

$$\frac{\partial E}{\partial \mathbf{u}_I^{(1)}} = \frac{1}{P}\sum_{s'=1}^{P} \frac{\partial E^{(s')}}{\partial \mathbf{R}_I^{(s')}} \tag{329}$$

since $\mathbf{U}_{1s} = \mathbf{U}_{s1}^{\dagger} = 1/\sqrt{P}$ and the remaining normal mode forces are given by

$$\frac{\partial E}{\partial \mathbf{u}_I^{(s)}} = \frac{1}{\sqrt{P}}\sum_{s'=1}^{P} \mathbf{U}_{ss'} \frac{\partial E^{(s')}}{\partial \mathbf{R}_I^{(s')}} \quad \text{for } s = 2, \ldots, P \tag{330}$$

in terms of the "primitive" forces $-\partial E^{(s)}/\partial \mathbf{R}_I^{(s)}$. Here, E on the left–hand–side with no superscript (s) refers to the average electronic energy $E = (1/P)\sum_{s=1}^{P} E^{(s)}$ from which the forces have to be derived. Thus, the force Eq. (329) acting on each *centroid* variable $\mathbf{u}_I^{(1)}$, $I = 1, \ldots N$, is exactly the force averaged over imaginary time $s = 1, \ldots, P$, i.e. the *centroid force* on the Ith nucleus as already given in

Eq. (2.21) of Ref. [644]. This is the desired relation which allows in centroid molecular dynamics the centroid forces to be simply obtained as the average force which acts on the lowest-order normal mode Eq. (328). The non–centroid normal modes $\mathbf{u}_I^{(s)}$, $s = 2, 3, \ldots, P$ of the paths establish the effective potential in which the centroid moves.

At this stage the equations of motion for adiabatic *ab initio* centroid molecular dynamics [411] can be obtained from the Euler–Lagrange equations. These equations of motion read

$$M_I'^{(1)} \ddot{\mathbf{u}}_I^{(1)} = -\frac{1}{P} \sum_{s=1}^{P} \frac{\partial E\left[\{\phi_i\}^{(s)}, \{\mathbf{R}_I\}^{(s)}\right]}{\partial \mathbf{R}_I^{(s)}} \tag{331}$$

$$\begin{aligned} M_I'^{(s)} \ddot{u}_{I,\alpha}^{(s)} = & -\frac{\partial}{\partial u_{I,\alpha}^{(s)}} \frac{1}{P} \sum_{s'=1}^{P} E\left[\{\phi_i\}^{(s')}, \left\{\mathbf{R}_I\left(\mathbf{u}_I^{(1)}, \ldots, \mathbf{u}_I^{(P)}\right)\right\}^{(s')}\right] \\ & -M_I^{(s)} \omega_P^2 u_{I,\alpha}^{(s)} - M_I'^{(s)} \dot{\xi}_{I,\alpha,1}^{(s)} \dot{u}_{I,\alpha}^{(s)} \quad , s = 2, \ldots, P \end{aligned} \tag{332}$$

$$\ddot{\mu\phi}_i^{(s)} = -\frac{\delta E\left[\{\phi_i\}^{(s)}, \{\mathbf{R}_I\}^{(s)}\right]}{\delta \phi_i^{\star(s)}} + \sum_j \Lambda_{ij}^{(s)} \phi_j^{(s)} - \mu\dot{\eta}_1^{(s)} \dot{\phi}_i^{(s)} \quad , s = 1, \ldots, P \tag{333}$$

where $u_{I,\alpha}^{(s)}$ denotes the Cartesian components of a given normal mode vector $\mathbf{u}_I^{(s)} = (u_{I,1}^{(s)}, u_{I,2}^{(s)}, u_{I,3}^{(s)})$. In the present scheme, independent Nosé–Hoover chain thermostats [388] of length K are coupled to all non–centroid mode degrees of freedom $s = 2, \ldots, P$

$$Q^{\mathrm{n}} \ddot{\xi}_{I,\alpha,1}^{(s)} = \left[M_I'^{(s)} \left(\dot{u}_{I,\alpha}^{(s)}\right)^2 - k_{\mathrm{B}}T\right] - Q^{\mathrm{n}} \dot{\xi}_{I,\alpha,1}^{(s)} \dot{\xi}_{I,\alpha,2}^{(s)} \tag{334}$$

$$Q^{\mathrm{n}} \ddot{\xi}_{I,\alpha,k}^{(s)} = \left[Q^{\mathrm{n}} \left(\dot{\xi}_{I,\alpha,k-1}^{(s)}\right)^2 - k_{\mathrm{B}}T\right] - Q^{\mathrm{n}} \dot{\xi}_{I,\alpha,k}^{(s)} \dot{\xi}_{I,\alpha,k+1}^{(s)} \left(1 - \delta_{kK}\right), k = 2, ..., K \tag{335}$$

and all orbitals at a given imaginary time slice s are thermostatted by one such thermostat chain of length L

$$Q_1^{\mathrm{e}} \ddot{\eta}_1^{(s)} = 2\left[\sum_i \mu \left\langle \phi_i^{(s)} \middle| \phi_i^{(s)} \right\rangle - T_{\mathrm{e}}^0\right] - Q_1^{\mathrm{e}} \dot{\eta}_1^{(s)} \dot{\eta}_2^{(s)} \tag{336}$$

$$Q_l^{\mathrm{e}} \ddot{\eta}_l^{(s)} = \left[Q_{l-1}^{\mathrm{e}} \left(\dot{\eta}_{l-1}^{(s)}\right)^2 - \frac{1}{\beta_{\mathrm{e}}}\right] - Q_l^{\mathrm{e}} \dot{\eta}_l^{(s)} \dot{\eta}_{l+1}^{(s)} \left(1 - \delta_{lL}\right) \quad , l = 2, \ldots, L \ ; \tag{337}$$

note that for standard *ab initio* path integral runs as discussed in the previous section the centroid mode should be thermostatted as well. The desired fictitious kinetic energy of the electronic subsystem T_{e}^0 can be determined based on a short equivalent classical Car–Parrinello run with $P = 1$ and using again the relation $1/\beta_{\mathrm{e}} = 2T_{\mathrm{e}}^0/6N_{\mathrm{e}}'$ where N_{e}' is the number of orbitals. The mass parameters $\{Q_l^{\mathrm{e}}\}$ associated to the orbital thermostats are the same as those defined in Eq. (271), whereas the single mass parameter Q^{n} for the nuclei is determined by the harmonic interaction and is given by $Q^{\mathrm{n}} = k_{\mathrm{B}}T/\omega_P^2 = \beta/P$. The characteristic thermostat

frequency of the electronic degrees of freedom ω_e should again lie above the frequency spectrum associated to the fictitious nuclear dynamics. These is the method that is implemented in the CPMD package [142].

An important issue for adiabatic *ab initio* centroid molecular dynamics [411] is how to establish the time–scale separation of the non–centroid modes compared to the centroid modes. This is guaranteed if the *fictitious* normal mode masses $M_I'^{(s)}$ are taken to be

$$\begin{aligned} M_I'^{(1)} &= M_I \\ M_I'^{(s)} &= \gamma\, M_I^{(s)} \quad , s = 2, \ldots, P \ , \end{aligned} \tag{338}$$

where M_I is the *physical* nuclear mass, $M_I^{(s)}$ are the normal mode masses Eq. (326), and γ is the "centroid adiabaticity parameter"; note that this corrects a misprint of the definition of $M_I'^{(s)}$ for $s \geq 2$ in Ref. [411]. By choosing $0 < \gamma \ll 1$, the required time–scale separation between the centroid and non–centroid modes can be controlled so that the motion of the non–centroid modes is artificially accelerated, see Sect. 3 in Ref. [411] for a systematic study of the γ–dependence. Thus, the centroids with associated physical masses move quasiclassically in real–time in the centroid effective potential, whereas the fast dynamics of all other nuclear modes $s > 1$ is fictitious and serves only to generate the centroid effective potential "on–the–fly". In this sense γ (or rather γM_I) is similar to μ, the electronic adiabaticity parameter in Car–Parrinello molecular dynamics.

4.4.4 Other Approaches

It is evident from the outset that the Born–Oppenheimer approach to generate the *ab initio* forces can be used as well as Car–Parrinello molecular dynamics in order to generate the *ab initio* forces on the quantum nuclei. This variation was utilized in a variety of investigations ranging from clusters to molecular solids [132,37,596,597,428,429,333]. Closely related to the *ab initio* path integral approach as discussed here is a method that is based on Monte Carlo sampling of the path integral [672]. It is similar in spirit and in its implementation to Born–Oppenheimer molecular dynamics sampling as long as only time–averaged static observables are calculated. A semiempirical ("CNDO" and "INDO") version of Born–Oppenheimer *ab initio* path integral simulations was also devised [656] and applied to study muonated organic molecules [656,657].

A non–self–consistent approach to *ab initio* path integral calculations was advocated and used in a series of publications devoted to study the interplay of nuclear quantum effects and electronic structure in unsaturated hydrocarbons like benzene [544,503,81,543,504]. According to this philosophy, an ensemble of nuclear path configurations Eq. (316) is first generated at finite temperature with the aid of a parameterized model potential (or using a tight–binding Hamiltonian [504]). In a second, independent step electronic structure calculations (using Pariser–Parr–Pople, Hubbard, or Hartree–Fock Hamiltonians) are performed for this fixed ensemble of discretized quantum paths. The crucial difference compared to the self–consistent approaches presented above is that the creation of the thermal ensemble and the

subsequent analysis of its electronic properties is performed using different Hamiltonians.

Several attempts to treat also the electrons in the path integral formulation – instead of using wavefunctions as in the *ab initio* path integral family – were published [606,119,488,449,450]. These approaches are exact in principle, i.e. non–adiabaticity and full electron–phonon coupling is included at finite temperatures. However, they suffer from severe stability problems [121] in the limit of degenerate electrons, i.e. at very low temperatures compared to the Fermi temperature, which is the temperature range of interest for typical problems in chemistry and materials science. Recent progress on computing electronic forces from path integral Monte Carlo simulations was also achieved [708].

More traditional approaches use a wavefunction representation for both the electrons in the ground state and for nuclear density matrix instead of path integrals. The advantage is that real–time evolution is obtained more naturally compared to path integral simulations. A review of such methods with the emphasis of computing the interactions "on–the–fly" is provided in Ref. [158]. An approximate wavefunction–based quantum dynamics method which includes several excited states and their couplings was also devised and used [385,386,387,45]. An alternative approach to approximate quantum dynamics consists in performing instanton or semiclassical *ab initio* dynamics [325,47]. Also the approximate vibrational self–consistent field approach to nuclear quantum dynamics was combined with "on–the–fly" MP2 electronic structure calculations [122].

5 Applications: From Materials Science to Biochemistry

5.1 Introduction

Ab initio molecular dynamics was called a "virtual matter laboratory" [234], a notion that is fully justified in view of its relationship to experiments performed in the real laboratory. Ideally, a system is prepared in some initial state and than evolves according to the basic laws of physics – without the need of experimental input. It is clear to every practitioner that this viewpoint is highly idealistic for more than one reason, but still this philosophy allows one to compute observables with predictive power and also implies a broad range of applicability.

It is evident from the number of papers dealing with *ab initio* molecular dynamics, see for instance Fig. 1, that a truly comprehensive survey of applications cannot be given. Instead, the strategy chosen is to provide the reader with a wealth of references that try cover the full scope of this approach – instead of discussing in depth the physics or chemistry of only a few specific applications. To this end the selection is based on a general literature search in order to suppress personal preferences as much as possible. In addition the emphasis lies on recent applications that could not be covered in earlier reviews. This implies that several older key reference papers on similar topics are in general missing.

The first application of Car–Parrinello molecular dynamics [108] dealt with silicon, one of the basic materials in semiconductor industry. Classic solid–state application of this technique focus on the properties of crystals, such as those of CuCl where anharmonicity and off–center displacements of the Cu along the (111) directions were found to be important to describe the crystal structure as a function of temperature and pressure [64]. Various properties of solid nitromethane [647], crystalline nitric acid trihydrate [602], solid benzene [420], stage–1 alkali–graphite intercalation compounds [286,287], and of the one-dimensional intercalation compound $2HgS{\bullet}SnBr_2$ [530] were determined based on first principles. The molecular solid HBr undergoes various phase transitions upon compression. The dynamical behavior of one of these phases, disordered HBr–I, could be clarified using *ab initio* molecular dynamics [313]. Structure, phase transitions and short–time dynamics of magnesium silicate perovskites were analyzed in terms of *ab initio* trajectories [670]. The A7 to simple cubic transformation in As was investigated using *ab initio* molecular dynamics at constant–pressure[568]. By applying external pressure the hydrogen sublattice was found to undergo amorphization in $Mg(OH)_2$ and $Ca(OH)_2$ a phenomenon that was interpreted in terms of frustration [511]. Properties of solid cubane C_8H_8 were obtained in constant pressure simulations and compared to experiment [514]. *Ab initio* simulations of the graphitization of flat and stepped diamond (111) surfaces uncovered that the transition temperature depends sensibly on the type of the surface [327].

Sliding of grain boundaries in aluminum as a typical ductile metal was generated and analyzed in terms of atomistic rearrangements [432]. Microfracture in a sample of amorphous silicon carbide was induced by uniaxial strain and found to induce Si segregation at the surface [226]. The early stages of nitride growth on cubic silicon carbide including wetting were modeled by depositing nitrogen atoms on the Si–terminated SiC(001) surface [225].

Classical proton diffusion in crystalline silicon at high temperatures was an early application to the dynamics of atoms in solids [93]. Using the *ab initio* path integral technique [395,399,644,404], see Sect. 4.4 the preferred sites of hydrogen and muonium impurities in crystalline silicon [428,429], or the proton positions in $HCl \bullet nH_2O$ crystalline hydrates [516] could be located. The radiation–induced formation of $H_2^\star$ defects in c–Si via vacancies and self–interstitials was simulated by *ab initio* molecular dynamics [178]. The classical diffusion of hydrogen in crystalline GaAs was followed in terms of diffusion paths [668] and photoassisted reactivation of H–passivated Si donors in GaAs was simulated based on first principles [430]. Oxygen diffusion in p–doped silicon can be enhanced by adding hydrogen to the material, an effect that could be rationalized by simulations [107]. *Ab initio* dynamics helped in quantifying the barrier for the diffusion of atomic oxygen in a model silica host [279]. The microscopic mechanism of the proton diffusion in protonic conductors, in particular Sc–doped $SrTiO_3$ and Y–doped $SrCeO_3$, is studied via *ab initio* molecular dynamics, where is it found that covalent OH–bonds are formed during the process [561]. Ionic diffusion in a ternary superionic conductor was obtained by *ab initio* dynamics [677]. Proton motion and isomerization pathways of a complex photochromic molecular

crystal composed of 2–(2,4–dinitrobenzyl)pyridine dyes was generated by *ab initio* methods [216].

Also materials properties of polymers are investigated in quite some detail. Early applications of semiempirical ZDO molecular dynamics [666] were devoted to defects in conducting polymers, in particular to solitons, polarons and alkali doping in polyacetylene [666,667] as well as to muonium implanted in *trans* and *cis* polyacetylene [200]. More recent are calculations of Young's modulus for crystalline polyethylene [271], soliton dynamics in positively charged polyacetylene chains [125], charge localization in doped polypyrroles [140], chain rupture of polyethylene chains under tensile load [533], the influence of a knot on the strength of a polymer strand [534], or ion diffusion in polyethylene oxide [456].

5.3 Surfaces, Interfaces, and Adsorbates

A host of studies focusing on atoms and in particular on molecules interacting with surfaces appeared over the years. Recent studies focussed for instance on C_2H_2, C_2H_4, and trimethylgallium adsorbates on the GaAs(001)–(2×4) surface [248], thiophene on the catalytically active $MoS_2(010)$ [512] or RuS_2 [580] surfaces, small molecules on a nitric acid monohydrate crystal surface [624], CO on Si(001) [314], small molecules on TiO_2 [554,41], sulfur on Si(100) at various coverages [707], and sulfuric acid adsorbed on $ZrO_2(101)$ and $ZrO_2(001)$ [269].

Specific to *ab initio* molecular dynamics is its capability to describe also chemisorption as well as dynamical processes on (and of) surfaces including surface reactions [500]. The *ab initio* calculations of surface phonons in semiconductor surfaces can be based on the frozen–phonon, linear–response or nowadays molecular dynamics approaches, see Ref. [218] for a discussion and comparison. A review on the structure and energetics of oxide surfaces including molecular processes occurring on such surfaces is provided in Ref. [235], whereas Ref. [256] concentrates on the interaction of hydrogen with clean and adsorbate covered metal and semiconductor surfaces.

Recent applications in surface science include the transition from surface vibrations to liquid–like diffusional dynamics of the Ge(111) surface [607], the diffusion of Si adatoms on a double–layer stepped Si(001) surface [330], the structure of chemisorbed acetylene on the Si(001)–(2×1) surface [423], chemisorption of quinizarin on α–Al_2O_3 [212,213], the diffusion of a single Ga adatom on the GaAs(100)–c(4×4) surface [367], homoepitaxial crystal growth on Si(001) and the low–temperature dynamics of Si(111)–(7×7) [595,611], dissociation of an H_2O molecule on MgO [358,359], dissociation of Cl_2 on GaAs(110) [380], chlorine adsorption and reactions on Si(100) [691], molecular motion of NH_3 on MgO [358], dynamics and reactions of hydrated α–alumina surfaces [289], molecular vs. dissociative adsorption of water layers on MgO(100) as a function of coverage [448], oxidation of CO on Pt(111) [8,705], the reaction HCl + HOCl → H_2O + Cl_2 as it occurs on an ice surface [373], or desorption of D_2 from Si(100) [255]. Thermal contraction, the formation of adatom-vacancy pairs, and finally premelting was observed in *ab initio* simulations of the Al(110) surface at temperatures up to 900 K [415] Early stages of the oxidation of a Mg(0001) surface by direct attack of molecular O_2 was dynamically simulated [96] including

the penetration of the oxidation layer into the bulk. Similarly, the growth of an oxide layer was generated on an Si(100) surface [653].

The water–Pd(100), water–O/Pd(100) and water–Si(111) interfaces were simulated based on *ab initio* molecular dynamics [336,655]. Water covering the surface of a microscopic model of muscovite mica is found to form a two–dimensional network of hydrogen bonds, called two–dimensional ice, on that surface [447]. The metal–organic junction of monolayers of Pd–porphyrin and perylene on Au(111) was analyzed using an *ab initio* approach [355]. An interesting possibility is to compute the tip–surface interactions in atomic force microscopy as e.g. done for a neutral silicon tip interacting with an InP(110) surface [619] or Si(111) [481,482].

5.4 Liquids and Solutions

Molecular liquids certainly belong to the classic realm of molecular dynamics simulations. Water was and still is a challenge [581] for both experiment and simulations due to the directional nature and the weakness of the hydrogen bonds which leads to delicate association phenomena. Pioneering *ab initio* simulations of pure water at ambient [352] and supercritical conditions [205] were reported only a few years ago. More recently, these gradient–corrected density functional theory–based simulations were extended into several directions [587,573,575,576,579,118]. In the mean time (minimal–basis) Hartree–Fock *ab initio* molecular dynamics [291] as well as more approximate schemes [455] were also applied to liquid water. Since chemical reactions often occur in aqueous media the solvation properties of water are of utmost importance so that the hydration of ions [403,620,621,377,502] and small molecules [353,354,433] was investigated. Similarly to water liquid HF is a strongly associated liquid which features short–lived hydrogen–bonded zig–zag chains [521]. Another associated liquid, methanol, was simulated at 300 K using an adaptive finite–element method [634] in conjunction with Born–Oppenheimer molecular dynamics [635]. In agreement with experimental evidence, the majority of the molecules is found to be engaged in short linear hydrogen–bonded chains with some branching points [635]. Partial reviews on the subject of *ab initio* simulations as applied to hydrogen–bonded liquids can be found in the literature [586,406,247].

The *ab initio* simulated solvation behavior of "unbound electrons" in liquid ammonia at 260 K was found to be consistent with the physical picture extracted from experiment [155,156]. Similarly, *ab initio* molecular dynamics of dilute [553,203] and concentrated [569] molten $K_x{\cdot}(KCl)_{1-x}$ mixtures were performed at 1300 K entering the metallic regime. The structure of liquid ammonia at 273 K was investigated with a variety of techniques so that limitations of using classical nuclei, simple point charge models, small systems, and various density functionals could be assessed [164].

Ab initio molecular dynamics is also an ideal tool to study other complex fluids with partial covalency, metallic fluids, and their transformations as a function of temperature, pressure, or concentration. The properties of water–free KF • nHF melts depend crucially on polyfluoride anions $H_mF_{m+1}^-$ and solvated K^+ cations. *Ab initio* simulations allow for a direct comparison of these complexes in the liquid, gaseous and crystalline phase [515]. The changes of the measured structure factor of liquid sulfur as a function of temperature can be rationalized on the atomistic

level by various chain and ring structures that can be statistically analyzed in *ab initio* molecular dynamics simulations [631]. Liquid $GeSe_2$ is characterized by strong chemical bonds that impose a structure beyond the usual very short distances due to network formation [416]. Zintl–alloys such as liquid NaSn [552] or KPb [556] have very interesting bonding properties that manifest themselves in strong temperature– and concentration dependences of their structure factors (including the appearance of the so–called first sharp diffraction peak [555]) or electric conductivities.

Metals are ideal systems to investigate the metal–insulator transition upon expansion of the liquid [346,63] or melting [689]. Liquid copper was simulated at 1500 K: structural and dynamical data were found to be in excellent agreement with experimental [464]. Transport coefficients of liquid metals (including in particular extreme conditions) can also be obtained from first principles molecular dynamics using the Green–Kubo formalism [571,592]. The microscopic mechanism of the semiconductor–metal transition in liquid As_2Se_3 could be rationalized in terms of a structural change as found in *ab initio* simulations performed as a function of temperature and pressure [563]. The III–V semiconductors, such as GaAs, assume metallic behavior when melted, whereas the II–VI semiconductor CdTe does not. The different conductivities could be traced back to pronounced structural dissimilarities of the two systems in the melt [236].

5.5 Glasses and Amorphous Systems

Related to the simulation of dynamically disordered fluid systems are investigations of amorphous or glassy materials. In view of the severe limitations on system size and time scale (and thus on correlation lengths and times) *ab initio* molecular dynamics can only provide fairly local information in this sense. Within these inherent constraints the microscopic structure of amorphous selenium [304] and tetrahedral amorphous carbon [384], the amorphization of silica [684], boron doping in amorphous Si:H [181] or in tetrahedral amorphous carbon [227], as well as the Raman spectrum [465] and dynamic structure factor [466] of quartz glass and their relation to short–range order could be studied.

The properties of supercooled CdTe were compared to the behavior in the liquid state in terms of its local structure [237]. Defects in amorphous $Si_{1-x}Ge_x$ alloys generated by *ab initio* annealing were found to explain ESR spectra of this system [329]. The infrared spectrum of a sample of amorphous silicon was obtained and found to be in quantitative agreement with experimental data [152]. The CO_2 insertion into a model of argon–bombarded porous SiO_2 was studied [508]. In particular the electronic properties of amorphous GaN were investigated using *ab initio* methods [601].

Larger systems and longer annealing times are accessible after introducing more approximations into the first principle treatment of the electronic structure that underlies *ab initio* molecular dynamics. Using such methods [551], a host of different amorphous carbon nitride samples with various stoichiometries and densities could be generated and characterized in terms of trends [675]. Similarly, the pressure–induced glass–to–crystal transition in condensed sodium was investigated [22] and two structural models of amorphous GaN obtained at different densities were examined

in terms of their electronic structure [601].

5.6 Matter at Extreme Conditions

A strong advantage of *ab initio* simulations is their predictive power also at extreme conditions, an area where molecular dynamics relying on fitted potential models might encounter severe difficulties. Thus, high pressures and / or high temperatures such as those in the earth's core, on other planets, or on stars can be easily achieved in the virtual laboratory. This opens up the possibility to study phase transformations and chemical reactions at such conditions [56]. Furthermore, conditions of geophysical and astrophysical interest can nowadays be produced in the real laboratory, using techniques based on diamond anvil cells, shock waves, or lasers. The limitations of these experimental approaches are, however, not so much related to generating the extreme conditions as one might expect, but rather to measuring observables.

In the virtual laboratory this information is accessible and the melting of diamond at high pressure [222], the phase transformation from the antiferromagnetic insulating δ–O_2 phase to a nonmagnetic metallic molecular ζ–O_2 phase [557], the phase diagram of carbon at high pressures and temperatures [261] as well as transformations of methane [13], carbon monoxide [54], molecular CO_2 [267,558], water ice [363,364,58,50,51,52], solid [305,337,65,66,333] and hot fluid [5] hydrogen, solid $Ar(H_2)_2$ [53] under pressure could be probed. Along similar lines properties of a liquid Fe–S mixture under earth's core conditions [11], the viscosity of liquid iron [690,592], the sound velocity of dense hydrogen at conditions on jupiter [6], the phase diagram of water and ammonia up to 7000 K and 300 GPa [118], the laser heating of silicon [570,572] and graphite [574] etc. were investigated at extreme state points. A review on *ab initio* simulations relevant to minerals at conditions found in the earth's mantle is provided in Ref. [683].

5.7 Clusters, Fullerenes, and Nanotubes

Investigations of clusters by *ab initio* molecular dynamics were among the first applications of this technique. Here, the feasibility to conduct finite–temperature simulations and in particular the possibility to search globally for minima turned out to be instrumental [302,31,303,550,517,519], see e.g. Refs. [16,321,32] for reviews. Such investigations focus more and more on clusters with varying composition [518,293,199,348,349,161]. Cluster melting is also accessible on an *ab initio* footing [84,531,525,526] and molecular clusters, complexes or cluster aggregates are actively investigated [612,645,613,70,596,597,133,701,524].

III–V semiconductor clusters embedded in sodalite show quantum confinement and size effects that can be rationalized by *ab initio* simulations [625,95]. Supported clusters such as Cu_n on an MgO(100) surface are found to diffuse by "rolling" and "twisting" motions with very small barriers [438]. The diffusion of protonated helium clusters in various sodalite cages was generated using *ab initio* dynamics [198]. Photo–induced structural changes in Se chains and rings were generated by a vertical HOMO $\rightarrow$ LUMO excitation and monitored by *ab initio* dynamics [306]. With the discovery and production of finite carbon assemblies *ab initio* investigations of the

properties of fullerenes [19,17,451], the growth process of nanotubes [127,62,72], or the electrical conductivity of nanowires [38,272] became of great interest.

5.8 Complex and Floppy Molecules

The determination of the structure of a RNA duplex including its hydration water [311], investigations of geometry and electronic structure of porphyrins and porphyrazines [356], and the simulation of a bacteriochlorophyll crystal [381] are some applications to large molecules. Similarly, the "carboplatin" complex [623]– a drug with large ligands – as well as the organometallic complex Alq(3) [148] – an electroluminescent material used in organic light–emitting diodes – were investigated with respect to structural, dynamical and electronic properties.

The organometallic compound $C_2H_2Li_2$ has an unexpected ground–state structure that was found by careful *ab initio* simulated annealing [521]. In addition, this complex shows at high temperatures intramolecular hydrogen migration that is mediated via a lithium hydride unit [521]. Ground–state fluxionality of protonated methane CH_5^+ [397,408] including some isotopomers [409] and of protonated acetylene $C_2H_3^+$ [400] was shown to be driven by quantum effects. The related dynamical exchange of atoms in these molecules can also be excited by thermal fluctuations [630,85,401]. In addition it was shown that CH_5^+ is three–center two–electron bonded and that this bonding topology does not qualitatively change in the presence of strong quantum motion [402]. The fluxional behavior of the protonated ethane molecular ion $C_2H_7^+$ was investigated by *ab initio* molecular dynamics as well [172].

The neutral and ionized SiH_5 and Si_2H_3 species display a rich dynamical behavior which was seen during *ab initio* molecular dynamics simulations [246]. The lithium pentamer Li_5 was found to perform pseudorotational motion on a time scale of picoseconds or faster at temperatures as low as 77 K [231]. Using *ab initio* instanton dynamics the inversion splitting of the NH_3, ND_3, and PH_3 molecules due to the umbrella mode was estimated [325]. Similarly, a semiclassical *ab initio* dynamics approach as used to compute the tunneling rate for intramolecular proton transfer in malonaldehyde [47]. *Ab initio* simulated annealing can be used to explore the potential energy landscape and to locate various minima, such as for instance done for protonated water clusters [673]. Molecular dynamics simulations of the trimethylaluminum $Al(CH_3)_3$ have been carried out in order to investigate the properties of the gas–phase dimer [29]. The structures and vibrational frequencies of tetrathiafulvalene in different oxidation states was probed by *ab initio* molecular dynamics [324]. Implanted muons in organic molecules (benzene, 3–quinolyl nitronyl nitroxide, para–pyridyl nitronyl nitroxide, phenyl nitronyl nitroxide and para–nitrophenyl nitronyl nitroxide) were investigated using approximate *ab initio* path integral simulations that include the strong quantum broadening of the muonium [656,657].

5.9 Chemical Reactions and Transformations

Early applications of *ab initio* molecular dynamics were devoted to reactive scattering in the gas phase such as $CH_2 + H_2 \rightarrow CH_4$ [669] or $H^- + CH_4 \rightarrow CH_4 + H^-$ [365]. The "on–the–fly" approach can be compared to classical trajectory cal-

culations on very accurate global potential energy surfaces. This was for instance done for the well–studied exothermic exchange reaction $F + H_2 \rightarrow HF + H$ in Ref. [565]. Other gas phase reactions studied were $Li(2p) + H_2 \rightarrow LiH(^1\Sigma) + H(^1S)$ in Ref. [387], $F + C_2H_4 \rightarrow C_2H_3F + H$ in Ref. [83], $2O_3 \rightarrow 3O_2$ in Ref. [170], $F^- + CH_3Cl \rightarrow CH_3F + Cl^-$ in Ref. [605], hydroxyl radical with nitrogen dioxide radical [165], formaldehyde radical anion with CH_3Cl in Ref. [700], the reduction of $OH^\bullet$ with 3-hexanone [215] or the hydrolysis (or solvolysis, S_N2 nucleophilic substitution) of methyl chloride with water [2,3]. Photoreactions of molecules slowly become accessible to *ab initio* dynamics, such as for instance the *cis–trans* photoisomerization in ethylene [46], excited–state dynamics in conjugated polymers [71], bond breaking in the S_8 ring [562], transformations of diradicales [195,196], or the $S_0 \rightarrow S_1$ photo isomerization of formaldimine [214].

In addition to allowing to study complex gas phase chemistry, *ab initio* molecular dynamics opened the way to simulate reactions in solution at finite temperatures. This allows liquid state chemistry to take place in the virtual laboratory where thermal fluctuations and solvation effects are included. Some applications out of this emerging field are the cationic polymerization of 1,2,5–trioxane [146,147], the initial steps of the dissociation of HCl in water [353,354], the formation of sulfuric acid by letting SO_3 react in liquid water [421] or the acid–catalyzed addition of water to formaldehyde [422].

Proton transfer is a process of broad interest and implications in many fields. Intramolecular proton transfer was studied recently in malonaldehyde [695,47], a Mannich base [182], and formic acid dimers [427]. Pioneering *ab initio* molecular dynamics simulations of proton and hydroxyl diffusion in liquid water were reported in the mid nineties [640,641,642]. Related to this problem is the auto–dissociation of pure water at ambient conditions [628,629]. Since recently it became possible to study proton motion including nuclear quantum effects [645,410,412] by using the *ab initio* path integral technique [395,399,644,404], see Sect. 4.4.

Ab initio molecular dynamics also allows chemical reactions to take place in solid phases, in particular if a constant pressure methodology is used [56], see Sect. 4.2. For instance solid state reactions such as pressure–induced transformations of methane [13] and carbon monoxide [54] or the polymerization [57] and amorphization [56] of acetylene were investigated.

5.10 Catalysis and Zeolites

The polymerization of olefines is an important class of chemical reactions that is operated on the industrial scale. In the light of such applications the detailed understanding of these reactions might lead to the design of novel catalysts. Driven by such stimulations several catalysts were investigated in detail such as metal alkyles [609], platinum–phospine complexes [141], or Grubbs' ruthenium–phosphine complexes [1], metallocenes [696]. In addition, elementary steps of various chemical processes were the focus of *ab initio* molecular dynamics simulations. Among those are chain branching and termination steps in polymerizations [696], ethylene metathesis [1], "living polymerization" of isoprene with ethyl lithium [522], Ziegler–Natta heterogenous polymerization of ethylene [79,80], Reppe carbonylation of Ni–

$CH{=}CH_2$ using $Cl(CO)_2$ [20], or Sakakura–Tanaka functionalization [382]. As in the real laboratory, side reactions can occur also in the virtual laboratory, such as e.g. the β–hydrogen elimination as an unpredicted reaction path [383]. A digression on using finite–temperature *ab initio* dynamics in homogeneous catalysis research can be found in Ref. [697].

Zeolites often serve as catalysists as well and are at the same time ideal candidates for finite–temperature *ab initio* simulations in view of their chemical complexity. A host of different studies [559,100,268,614,545,206,560,598,207,315,208,209,546] contributed greatly to the understanding of these materials and the processes occurring therein such as the initial stages of the methanol to gasoline conversion [599]. Heterogenous catalysts are often poisoned, which was for instance studied in the case of hydrogen dissociation on the Pd(100) surface in the presence adsorbed sulfur layers [257].

5.11 Biophysics and Biochemistry

Applications of *ab initio* molecular dynamics to molecules and processes of interest in life sciences begin to emerge [18,113]. Investigations related to these interests are investigations of the crystal structure of a hydrated RNA duplex (sodium guanylyl–3'–5'–cytidine nona–hydrate) [311], structure models for the cytochrom P450 enzyme family [547,548,549], nanotubular polypeptides [112], a synthetic biomimetic model of galactose oxidase [523], aspects of the process of vision in form of the 11–*cis* to *all–trans* isomerization in rhodopsin [67,68,474], interconversion pathways of the protonated β–ionone Schiff base [615], or of the binding properties of small molecules of physiological relevance such as O_2, CO or NO to iron–porphyrines and its complexes [527,528,529].

Proton transport throught water wires is an important biophysical process in the chemiosmotic theory for biochemical ATP production. Using the *ab initio* path integral technique [395,399,644,404] the properties of linear water wires with an excess proton were studied at room temperature [419]. Amino acids are important ingredients as they are the building blocks of polypeptides, which in turn form channels and pores for ion exchange. Motivated by their ubiquity, glycine and alanine as well as some of their oligopeptides and helical (periodic) polypeptides were studied in great detail [323].

5.12 Outlook

Ab initio molecular dynamics is by now not only a standard tool in academic research but also becomes increasingly attractive to industrial researchers. Analysis of data bases, see caption of Fig. 1 for details, uncovers that quite a few companies seem to be interested in this methodology. Researchers affiliated to Bayer, Corning, DSM, Dupont, Exxon, Ford, Hitachi, Hoechst, Kodak, NEC, Philips, Pirelli, Shell, Toyota, Xerox and others cite the Car–Parrinello paper Ref. [108] or use *ab initio* molecular dynamics in their work. This trend will certainly be enhanced by the availability of efficient and general *ab initio* molecular dynamics packages which are commercially available.

Acknowledgments

Our knowledge on *ab initio* molecular dynamics has slowly grown in a fruitful and longstanding collaboration with Michele Parrinello, initially at IBM Zurich Research Laboratory in Rüschlikon and later at the Max–Planck–Institut für Festkörperforschung in Stuttgart, which we gratefully acknowledge at this occasion. In addition, we profited enormously from pleasant cooperations with too many colleagues and friends to be named here.

References

1. O. M. Aagaard, R. J. Meier, and F. Buda, J. Am. Chem. Soc. **120**, 7174 (1998).
2. M. Aida, H. Yamataka, and M. Dupuis, Chem. Phys. Lett. **292**, 474 (1998).
3. M. Aida, H. Yamataka, and M. Dupuis, Theor. Chem. Acc. **102**, 262 (1999).
4. H. Akai and P.H. Dederichs, J. Phys. C **18**, 2455 (1985).
5. A. Alavi, J. Kohanoff, M. Parrinello, and D. Frenkel, Phys. Rev. Lett. **73**, 2599 (1994).
6. A. Alavi, M. Parrinello, and D. Frenkel, Science **269**, 1252 (1995).
7. A. Alavi, in *Monte Carlo and Molecular Dynamics of Condensed Matter Systems*, Chapt. 25, p. 648, eds. K. Binder and G. Ciccotti (Italien Physical Society SIF, Bologna, 1996).
8. A. Alavi, P. Hu, T. Deutsch, P. L. Silvestrelli, and J. Hutter, Phys. Rev. Lett. **80**, 3650 (1998).
9. A. Alavi, private communication.
10. M. Albrecht, A. Shukla, M. Dolg, P. Fulde, and H. Stoll, Chem. Phys. Lett. **285**, 174 (1998).
11. D. Alfe and M. J. Gillan, Phys. Rev. B **58**, 8248 (1998).
12. M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Clarendon Press, Oxford, 1987; reprinted 1990).
13. F. Ancilotto, G. L. Chiarotti, S. Scandolo, and E. Tosatti, Science **275**, 1288 (1997).
14. H. C. Andersen, J. Chem. Phys. **72**, 2384 (1980).
15. H. C. Andersen, J. Comput. Phys. **52**, 24 (1983).
16. W. Andreoni, Z. Phys. D **19**, 31 (1991).
17. W. Andreoni, Annu. Rev. Phys. Chem. **49**, 405 (1998).
18. W. Andreoni, Perspect. Drug Discovery Des. **9–11**, 161 (1998).
19. W. Andreoni and A. Curioni, Appl. Phys. A **A66**, 299 (1998).
20. F. De Angelis, N. Re, A. Sgamellotti, A. Selloni, J. Weber, and C. Floriani, Chem. Phys. Lett. **291**, 57 (1998).
21. J. F. Annett, Comput. Mat. Sci **4**, 23 (1995).
22. M. I. Aoki and K. Tsumuraya, Phys. Rev. B **56**, 2962 (1997).
23. T. A. Arias, M. C. Payne, and J. D. Joannopoulos, Phys. Rev. Lett. **69**, 1077 (1992).
24. T. A. Arias, M. C. Payne, and J. D. Joannopoulos, Phys. Rev. B **45**, 1538 (1992).

25. T. A. Arias, Rev. Mod. Phys. **71**, 267 (1999).
26. E. Artacho, D. Sánchez–Portal, P. Ordejón, A. Garcia, and J. M. Soler, phys. stat. sol. (b) **215**, 809 (1999).
27. N. W. Ashcroft and N. D. Mermin, *Solid State Physics* (Saunders College Publishing, Philadelphia, 1976).
28. G. B. Bachelet, D. R. Hamann, and M. Schlüter, Phys. Rev. B **26**, 4199 (1982).
29. S. Balasubramanian, C. J. Mundy, and M. L. Klein, J. Phys. Chem. B **102**, 10 136 (1998).
30. A. Baldereschi, Phys. Rev. B **7**, 5212 (1973).
31. P. Ballone, W. Andreoni, R. Car, and M. Parrinello, Phys. Rev. Lett. **60**, 271 (1988).
32. P. Ballone and W. Andreoni, in *Metal Clusters*, ed. W. Ekardt (Wiley, New York, 1999).
33. P. Bandyopadhyay, S. Ten–no, and S. Iwata, J. Phys. Chem. A **103**, 6442 (1999).
34. R. N. Barnett, U. Landman, and A. Nitzan, J. Chem. Phys. **89**, 2242 (1988).
35. R. N. Barnett, U. Landman, A. Nitzan, and G. Rajagopal, J. Chem. Phys. **94**, 608 (1991).
36. R. N. Barnett and U. Landman, Phys. Rev. B **48**, 2081 (1993).
37. R. N. Barnett, H.–P. Cheng, H. Hakkinen, and U. Landman, J. Phys. Chem. **99**, 7731 (1995).
38. R. N. Barnett and U. Landman, Nature (London) **387**, 788 (1997).
39. S. Baroni and P. Giannozzi, Europhys. Lett. **17**, 547 (1992).
40. U. von Barth and L. Hedin, J. Phys. C **5**, 1629 (1972).
41. S. P. Bates, M. J. Gillan, and G. Kresse, J. Phys. Chem. B **102**, 2017 (1998).
42. A. D. Becke, Phys. Rev. A **38**, 3098 (1988).
43. A. D. Becke, J. Chem. Phys. **98**, 1372 (1993).
44. A. D. Becke, J. Chem. Phys. **98**, 5648 (1993).
45. M. Ben–Nun and T. J. Martinez, J. Chem. Phys. **108**, 7244 (1998).
46. M. Ben–Nun and T. J. Martinez, Chem. Phys. Lett. **298**, 57 (1998).
47. M. Ben–Nun and T. J. Martinez, J. Phys. Chem. A **103**, 6055 (1999).
48. P. Bendt and A. Zunger, Phys. Rev. B **26**, 3114 (1982).
49. P. Bendt and A. Zunger, Phys. Rev. Lett. **50**, 1684 (1983).
50. M. Benoit, D. Marx, and M. Parrinello, Nature (London) **392**, 258 (1998); see also J. Teixeira, Nature (London) **392**, 232 (1998).
51. M. Benoit, D. Marx, and M. Parrinello, Comp. Mat. Sci. **10**, 88 (1998).
52. M. Benoit, D. Marx, and M. Parrinello, Solid State Ionics **125**, 23 (1999).
53. S. Bernard, P. Loubeyre, and G. Zerah, Europhys. Lett. **37**, 477 (1997).
54. S. Bernard, G. L. Chiarotti, S. Scandolo, and E. Tosatti, Phys. Rev. Lett. **81**, 2092 (1998).
55. M. Bernasconi, G. L. Chiarotti, P. Focher, S. Scandolo, E. Tosatti, and M. Parrinello, J. Phys. Chem. Solids **56**, 501 (1995).
56. M. Bernasconi, M. Benoit, M. Parrinello, G. L. Chiarotti, P. Focher, and E. Tosatti, Phys. Scr. **T66**, 98 (1996).
57. M. Bernasconi, G. L. Chiarotti, P. Focher, M. Parrinello, and E. Tosatti, Phys.

Rev. Lett. **78**, 2008 (1997).
58. M. Bernasconi, P. L. Silvestrelli, and M. Parrinello, Phys. Rev. Lett. **81**, 1235 (1998).
59. B. J. Berne, G. Ciccotti, and D. F. Coker (Eds.), *Classical and Quantum Dynamics in Condensed Phase Simulations* (World Scientific, Singapore, 1998).
60. J. Bernholc, N. O. Lipari, and S. T. Pantelides, Phys. Rev. B **21**, 3545 (1980).
61. J. Bernholc, J.–Y. Yi, and D. J. Sullivan, Faraday Discuss. Chem. Soc. **92**, 217 (1991).
62. J. Bernholc, C. Brabec, M. B. Nardelli, A. Maiti, C. Roland, and B. I. Yakobson, Appl. Phys. A **67**, 39 (1998).
63. S. R. Bickham, O. Pfaffenzeller, L. A. Collins, J. D. Kress, and D. Hohl, Phys. Rev. B **58**, R11 813 (1998).
64. S. R. Bickham, J. D. Kress, L. A. Collins, and R. Stumpf, Phys. Rev. Lett. **83**, 568 (1999).
65. S. Biermann, D. Hohl, and D. Marx, J. Low Temp. Phys. **110**, 97 (1998).
66. S. Biermann, D. Hohl, and D. Marx, Solid State Commun. **108**, 337 (1998).
67. A. Bifone, H. J. M. de Groot, and F. Buda, J. Phys. Chem. B **101**, 2954 (1997).
68. A. Bifone, H. J. M. de Groot, and F. Buda, Pure Appl. Chem. **69**, 2105 (1997).
69. K. Binder and G. Ciccotti (Eds.), *Monte Carlo and Molecular Dynamics of Condensed Matter Systems* (Italien Physical Society SIF, Bologna 1996).
70. G. Bischof, A. Silbernagel, K. Hermansson, and M. Probst, Int. J. Quantum Chem. **65**, 803 (1997).
71. E. R. Bittner and D. S. Kosov, J. Chem. Phys. **110**, 6645 (1999).
72. X. Blase, A. De Vita, J.–C. Charlier, and R. Car, Phys. Rev. Lett. **80**, 1666 (1998).
73. P. E. Blöchl, Phys. Rev. B **41**, 5414 (1990).
74. P. E. Blöchl and M. Parrinello, Phys. Rev. B **45**, 9413 (1992).
75. P. E. Blöchl, Phys. Rev. B **50**, 17 953 (1994).
76. P. E. Blöchl, J. Chem. Phys. **103**, 7422 (1995).
77. S. Blügel, Ph.D. Thesis (Rheinisch–Westfälische Technische Hochschule Aachen, Aachen, 1988).
78. M. Bockstedte, A. Kley, J. Neugebauer, and M. Scheffler, Comput. Phys. Comm. **107**, 187 (1997).
79. M. Boero, M. Parrinello, and K. Terakura, J. Am. Chem. Soc. **120**, 2746 (1998).
80. M. Boero, M. Parrinello, and K. Terakura, Surf. Sci. **438**, 1 (1999).
81. M. C. Böhm, R. Ramírez, and J. Schulte, Chem. Phys. **227**, 271 (1998).
82. K. Bolton, W. L. Hase, and G. H. Peslherbe, in *Modern Methods for Multidimensional Dynamics Computations in Chemistry*, ed. D. L. Thompson (World Scientific, Singapore, 1998).
83. K. Bolton, B. H. Schlegel, W. L. Hase, and K. Song, Phys. Chem. Chem. Phys. **1**, 999 (1999).
84. V. Bonačić-Koutecký, J. Jellinek, M. Wiechert, and P. Fantucci, J. Chem. Phys. **107**, 6321 (1997).

85. D. W. Boo, Z. F. Liu, A. G. Suits, J. S. Tse, and Y. T. Lee, Science **269**, 57 (1995).
86. F. A. Bornemann and C. Schütte, Numer. Math. **78**, 359 (1998).
87. F. A. Bornemann and C. Schütte, Numer. Math. **83**, 179 (1999).
88. S. F. Boys and F. Bernardi, Mol. Phys. **19**, 553 (1970).
89. E. L. Briggs, D. J. Sullivan, and J. Bernholc, Phys. Rev. B **52**, R5471 (1995).
90. K. D. Brommer, B. E. Larson, M. Needels, and J. D. Joannopoulos, Computers in Physics **7**, 350 (1993).
91. J. Broughton and F. Kahn, Phys. Rev. B **40**, 12 098 (1989).
92. C. G. Broyden, Math. Comput. **19**, 577 (1965).
93. F. Buda, G. L. Chiarotti, R. Car, and M. Parrinello, Phys. Rev. Lett. **63**, 294 (1989).
94. F. Buda, R. Car, and M. Parrinello, Phys. Rev. B **41**, 1680 (1990).
95. F. Buda and A. Fasolino, Phys. Rev. B **60**, 6131 (1999).
96. C. Bungaro, C. Noguera, P. Ballone, and W. Kress, Phys. Rev. Lett. **79**, 4433 (1997).
97. D.M. Bylander, L. Kleinman, and S. Lee, Phys. Rev. B **42**, 1394 (1990).
98. T. Cagin and J. R. Ray, Phys. Rev. A **37**, 247 (1988).
99. D. J. E. Callaway and A. Rahman, Phys. Rev. Lett. **49**, 613 (1982).
100. L. Campana, A. Selloni, J. Weber, and A. Goursot, J. Phys. Chem. B **101**, 9932 (1997).
101. J. Cao and B. J. Berne, J. Chem. Phys. **99**, 2902 (1993).
102. J. Cao and G. A. Voth, J. Chem. Phys. **99**, 10 070 (1993).
103. J. Cao and G. A. Voth, J. Chem. Phys. **100**, 5106 (1994).
104. J. Cao and G. A. Voth, J. Chem. Phys. **101**, 6157 (1994).
105. J. Cao and G. A. Voth, J. Chem. Phys. **101**, 6168 (1994).
106. J. Cao and G. J. Martyna, J. Chem. Phys. **104**, 2028 (1996).
107. R. B. Capaz, L. V. C. Assali, L. C. Kimerling, K. Cho, and J. D. Joannopoulos, Phys. Rev. B **59**, 4898 (1999).
108. R. Car and M. Parrinello, Phys. Rev. Lett. **55**, 2471 (1985).
109. R. Car and M. Parrinello, Solid State Commun. **62**, 403 (1987).
110. R. Car, M. Parrinello, and M. Payne, J. Phys.: Condens. Matter **3**, 9539 (1991).
111. R. Car, in *Monte Carlo and Molecular Dynamics of Condensed Matter Systems*, Chapt. 23, p. 601, eds. K. Binder and G. Ciccotti (Italien Physical Society SIF, Bologna, 1996).
112. P. Carloni, W. Andreoni, and M. Parrinello Phys. Rev. Lett. **79**, 761 (1997).
113. P. Carloni and F. Alber, Perspect. Drug Discovery Des. **9–11**, 169 (1998).
114. A. Caro, S. Ramos de Debiaggi, and M. Victoria, Phys. Rev. B **41**, 913 (1990).
115. E. A. Carter, G. Ciccotti, J. T. Hynes, and R. Kapral, Chem. Phys. Lett. **156**, 472 (1989).
116. `CASTEP`: Molecular Simulations Inc.; see Ref. [472].
117. C. Cavazzoni, Ph.D. Thesis: *Large Scale First–Principles Simulations of Water and Ammonia at High Pressure and Temperature* (Scuola Internazionale Superiore di Studi Avanzati (SISSA), Trieste, 1998).
118. C. Cavazzoni, G. L. Chiarotti, S. Scandolo, E. Tosatti, M. Bernasconi, and M.

Parrinello, Science **283**, 44 (1999).
119. D. M. Ceperley, Phys. Rev. Lett. **69**, 331 (1992).
120. D. M. Ceperley, Rev. Mod. Phys. **67**, 279 (1995); for errata and updates see `http://www.ncsa.uiuc.edu/Apps/CMP/papers/cep95a/cep95a.html`.
121. D. M. Ceperley in *Monte Carlo and Molecular Dynamics of Condensed Matter Systems*, Chapt. 16, p. 443, eds. K. Binder and G. Ciccotti (Italien Physical Society SIF, Bologna, 1996).
122. G. M. Chaban, J. O. Jung, and R. B. Gerber, J. Chem. Phys. **111**, 1823 (1999) and preprint.
123. D. J. Chadi and M. L. Cohen, Phys. Rev. B **8**, 5747 (1973).
124. Ch. Chakravarty, Int. Rev. Phys. Chem. **16**, 421 (1997).
125. B. Champagne, E. Deumens, Y. Öhrn, J. Chem. Phys. **107**, 5433 (1997).
126. D. Chandler in *Liquids, Freezing, and Glass Transition*, eds. J. P. Hansen, D. Levesque, and J. Zinn–Justin, (Elsevier, Amsterdam, 1991).
127. J.–C. Charlier, A. de Vita, X. Blase, and R. Car, Science **275**, 646 (1997).
128. S. Chawla and G. A. Voth, J. Chem. Phys. **108**, 4697 (1998).
129. J. R. Chelikowsky and S. G. Louie, Phys. Rev. B **29**, 3470 (1984).
130. J. R. Chelikowsky, N. Troullier, and Y. Saad, Phys. Rev. Lett. **72**, 1240 (1994).
131. J. R. Chelikowsky, N. Troullier, K. Wu, and Y. Saad, Phys. Rev. B **50**, 11 355 (1994).
132. H.–P. Cheng, R. N. Barnett, and U. Landman, Chem. Phys. Lett. **237**, 161 (1995).
133. H.–P. Cheng, J. Phys. Chem. A **102**, 6201 (1998).
134. K. Cho, T. A. Arias, J. D. Joannopoulos, and P. K. Lam, Phys. Rev. Lett. **71**, 1808 (1993).
135. G. Ciccotti, D. Frenkel, and I. R. McDonald, *Simulation of Liquids and Solids* (North–Holland, Amsterdam 1987).
136. G. Ciccotti and J. P. Ryckaert, Comput. Phys. Rep. **4**, 345 (1987).
137. L. J. Clarke, I. Štich, and M. C. Payne, Comput. Phys. Commun. **72**, 14 (1993).
138. R. D. Coalson, J. Chem. Phys. **85**, 926 (1986).
139. M. H. Cohen and V. Heine, Phys. Rev. **122**, 1821 (1961).
140. R. Colle and A. Curioni, J. Am. Chem. Soc. **120**, 4832 (1998).
141. B. B. Coussens, F. Buda, H. Oevering, and R. J. Meier, Organometallics **17**, 795 (1998).
142. `CPMD Version 3.3:` developed by J. Hutter, A. Alavi, T. Deutsch, M. Bernasconi, St. Goedecker, D. Marx, M. Tuckerman, and M. Parrinello, Max–Planck–Institut für Festkörperforschung and IBM Zurich Research Laboratory (1995–1999).
143. `CP-PAW:` P. E. Blöchl, IBM Zurich Research Laboratory; see Ref. [75].
144. C. Császár and P. Pulay, J. Mol. Struc. **114**, 31 (1984).
145. T. R. Cundari, M. T. Benson, M. L. Lutz, and S. O. Sommerer, in *Reviews in Computational Chemistry Vol. 8*, p. 145, eds. K. B. Lipkowitz and D. B. Boyd (VCH, New York, 1996).
146. A. Curioni, W. Andreoni, J. Hutter, H. Schiffer, and M. Parrinello, J. Am.

Chem. Soc. **116**, 11 251 (1994).
147. A. Curioni, M. Sprik, W. Andreoni, H. Schiffer, J. Hutter, and M. Parrinello, J. Am. Chem. Soc. **119**, 7218 (1997).
148. A. Curioni, M. Boero, and W. Andreoni, Chem. Phys. Lett. **294**, 263 (1998).
149. E. Dalgaard and P. Jorgensen, J. Chem. Phys. **69**, 3833 (1978).
150. C. Daul, Int. J. Quantum Chem. **52**, 867 (1994).
151. E. R. Davidson, J. Comput. Phys. **17**, 87 (1975).
152. A. Debernardi, M. Bernasconi, M. Cardona, and M. Parrinello, Appl. Phys. Lett. **71**, 2692 (1997).
153. P. H. Dederichs and R. Zeller, Phys. Rev. B **28**, 5462 (1983).
154. J. B. Delos, W. R. Thorson, and S. K. Knudson, Phys. Rev. A **6**, 709 (1972); see also following two papers.
155. Z. Deng, G. J. Martyna, and M. L. Klein, Phys. Rev. Lett. **68**, 2496 (1992).
156. Z. Deng, G. J. Martyna, and M. L. Klein, Phys. Rev. Lett. **71**, 267 (1993).
157. P. J. H. Denteneer and W. van Haeringen, J. Phys. C: Solid State Phys. **18**, 4127 (1985).
158. E. Deumens, A. Diz, R. Longo, and Y. Öhrn, Rev. Mod. Phys. **66**, 917 (1994).
159. A. Devenyi, K. Cho, T. A. Arias, and J. D. Joannopoulos, Phys. Rev. B **49**, 13 373 (1994).
160. B. K. Dey, A. Askar, and H. Rabitz, J. Chem. Phys. **109**, 8770 (1998).
161. A. Dhavale, D. G. Kanhere, C. Majumder, and G. P. Das, Eur. Phys. J. D **6**, 495 (1999).
162. P. A. M. Dirac, Proc. Cambridge Phil. Soc. **26**, 376 (1930).
163. P. A. M. Dirac, *The Principles of Quantum Mechanics* (Oxford University Press, Oxford, 1947; third edition); see in particular Chapt. V-§31–32.
164. M. Diraison, G. J. Martyna, and M. E. Tuckerman, J. Chem. Phys. **111**, 1096 (1999).
165. K. Doclo and U. Röthlisberger, Chem. Phys. Lett. **297**, 205 (1998).
166. M. Dolg and H. Stoll, in *Handbook on the Physics and Chemistry of Rare Earths Vol. 22*, p. 607, eds. K. A. Gschneidner Jr. and L. Eyring (Elsevier, Amsterdam, 1996).
167. J. Douady, Y. Ellinger, R. Subra, and B. Levy, J. Chem. Phys. **72**, 1452 (1980).
168. R. M. Dreizler and E. K. U. Gross, *Density–Functional Theory* (Springer, Berlin, 1990); see also the upcoming revised edition.
169. B. I. Dunlap, J. W. D. Connolly, and J. R. Sabin, J. Chem. Phys. **71**, 3396 (1979).
170. B. I. Dunlap, Int. J. Quantum Chem. **69**, 317 (1998).
171. B. I. Dunlap and R. W. Warren, Adv. Quantum Chem. **33**, 167 (1998).
172. A. L. L. East, Z. F. Liu, C. McCague, K. Cheng, and J. S. Tse, J. Phys. Chem. A **102**, 10 903 (1998).
173. J. W. Eastwood and D. R. K. Brownrigg, J. Comp. Phys. **32**, 24 (1979).
174. P. Ehrenfest, Z. Phys. **45**, 455 (1927).
175. E. Ermakova, J. Solca, H. Huber, and D. Marx, Chem. Phys. Lett. **246**, 204 (1995).
176. M. Ernzerhof and G. E. Scuseria, J. Chem. Phys. **110**, 5029 (1999).

177. R. Esser, P. Grassberger, J. Grotendorst, and M. Lewerenz (Eds.), *Molecular Dynamics on Parallel Computers* (World Scientific, Singapore, 2000).
178. S. K. Estreicher, J. L. Hastings, and P. A. Fedders, Phys. Rev. Lett. **82**, 815 (1999).
179. R. A. Evarestov and V. P. Smirnov, Phys. Stat. Sol. (b) **119**, 9 (1983).
180. M. Fähnle, C. Elsässer, and H. Krimmel, phys. stat. sol. (b) **191**, 9 (1995).
181. P. A. Fedders and D. A. Drabold, Phys. Rev. B **56**, 1864 (1997).
182. A. Fedorowicz, J. Mavri, P. Bala, and A. Koll, Chem. Phys. Lett. **289**, 457 (1998).
183. M. D. Feit, J. A. Fleck, Jr., and A. Steiger, J. Comput. Phys. **47**, 412 (1982).
184. P. Fernandez, A. Dal Corso, A. Baldereschi, and F. Mauri, Phys. Rev. B **55**, R1909 (1997).
185. G. W. Fernando, G.–X. Qian, M. Weinert, and J. W. Davenport, Phys. Rev. B **40**, 7985 (1989).
186. R. P. Feynman, Phys. Rev. **56**, 340 (1939); see in particular equation (2).
187. R. P. Feynman and A. R. Hibbs, *Quantum Mechanics and Path Integrals* (McGraw–Hill, New York, 1965).
188. R. P. Feynman, *Statistical Mechanics* (Addison–Wesley, Redwood City, 1972).
189. `fhi98md:` see Ref. [78].
190. M. J. Field, Chem. Phys. Lett. **172**, 83 (1990).
191. M. J. Field, J. Phys. Chem. **95**, 5104 (1991).
192. M. Filatov and W. Thiel, Mol. Phys. **91**, 847 (1997).
193. M. Filatov and S. Shaik, Chem. Phys. Lett. **288**, 689 (1998).
194. M. Filatov and W. Thiel, Phys. Rev. A **57**, 189 (1998).
195. M. Filatov and S. Shaik, J. Chem. Phys. **110**, 116 (1999).
196. M. Filatov and S. Shaik, Chem. Phys. Lett. **304**, 429 (1999).
197. A. Filippetti, D. Vanderbilt, W. Zhong, Y. Cai, and G.B. Bachelet, Phys. Rev. B **52**, 11793 (1995).
198. F. Filippone and F. A. Gianturco, J. Chem. Phys. **111**, 2761 (1999).
199. F. Finocchi and C. Noguera, Phys. Rev. B **57**, 14 646 (1998).
200. A. J. Fisher, W. Hayes, and F. L. Pratt, J. Phys.: Condens. Matter **3**, L9823 (1991).
201. P. Focher, G. L. Chiarotti, M. Bernasconi, E. Tosatti, and M. Parrinello, Europhys. Lett. **26**, 345 (1994).
202. P. Focher, Ph.D. Thesis: *First–principle studies of structural phase transformations* (Scuola Internazionale Superiore di Studi Avanzati (SISSA), Trieste, 1994).
203. E. S. Fois, A. Selloni, M. Parrinello, and R. Car, J. Phys. Chem. **92**, 3268 (1988).
204. E. S. Fois, J. I. Penman, and P. A. Madden, J. Chem. Phys. **98**, 6361 (1993).
205. E. S. Fois, M. Sprik, and M. Parrinello, Chem. Phys. Lett. **223**, 411 (1994).
206. E. Fois and A. Gamba, J. Phys. Chem. B **101**, 4487 (1997).
207. E. Fois, A. Gamba, and G. Tabacchi, J. Phys. Chem. B **102**, 3974 (1998).
208. E. Fois and A. Gamba, J. Phys. Chem. B **103**, 1794 (1999).
209. E. Fois, A. Gamba, and G. Tabacchi, Phys. Chem. Chem. Phys. **1**, 531 (1999).

210. J. Ford, Phys. Rep. **213**, 271 (1992).
211. G. P. Francis and M. C. Payne, J. Phys.: Condens. Matter **2**, 4395 (1990).
212. I. Frank, D. Marx, and M. Parrinello, J. Amer. Chem. Soc. **117**, 8037 (1995).
213. I. Frank, D. Marx, and M. Parrinello, J. Chem. Phys. **104**, 8143 (1996).
214. I. Frank, J. Hutter, D. Marx, and M. Parrinello, J. Chem. Phys. **108**, 4060 (1998).
215. I. Frank, M. Parrinello, and A. Klamt, J. Phys. Chem. A **102**, 3614 (1998).
216. I. Frank, D. Marx, and M. Parrinello, J. Phys. Chem. A **103**, 7341 (1999).
217. D. Frenkel and B. Smit, *Understanding Molecular Simulation – From Algorithms to Applications* (Academic Press, San Diego, 1996).
218. J. Fritsch and U. Schröder, Phys. Rep. **309**, 209 (1999).
219. S. Froyen and M. L. Cohen, J. Phys. C **19**, 2623 (1986).
220. C.–L. Fu and K.–M. Ho, Phys. Rev. B **28**, 5480 (1983).
221. M. Fuchs and M. Scheffler, Comput. Phys. Commun. **119**, 67 (1999).
222. G. Galli, R. M. Martin, R. Car, and M. Parrinello, Science **250**, 1547 (1990).
223. G. Galli and M. Parrinello, in *Computer Simulations in Materials Science*, p. 282, eds. M. Meyer and V. Pontikis (Kluwer, Dordrecht, 1991).
224. G. Galli and A. Pasquarello, in *Computer Simulation in Chemical Physics*, eds. M. P. Allen and D. J. Tildesley (Kluwer, Dordrecht, 1993).
225. G. Galli, A. Catellani, and F. Gygi, Phys. Rev. Lett. **83**, 2006 (1999).
226. G. Galli and F. Gygi, Phys. Rev. Lett. **82**, 3476 (1999).
227. A. Gambirasio and M. Bernasconi, Phys. Rev. B **60**, 12 007 (1999).
228. D. A. Gibson and E. A. Carter, J. Phys. Chem. **97**, 13 429 (1993).
229. D. A. Gibson, I. V. Ionova, and E. A. Carter, Chem. Phys. Lett. **240**, 261 (1995).
230. D. A. Gibson and E. A. Carter, Mol. Phys. **89**, 1265 (1996).
231. D. A. Gibson and E. A. Carter, Chem. Phys. Lett. **271**, 266 (1997).
232. M. J. Gillan, J. Phys.: Condens. Matter **1**, 689 (1989).
233. M. J. Gillan in *Computer Modelling of Fluids, Polymers, and Solids*, eds. C. R. A. Catlow, S. C. Parker, and M. P. Allen (Kluwer, Dordrecht, 1990).
234. M. J. Gillan, Contemp. Phys. **38**, 115 (1997).
235. M. J. Gillan, P. J. D. Lindan, L. N. Kantorovich, and S. P. Bates, Mineral. Mag. **62**, 669 (1998).
236. V. V. Godlevsky, J. J. Derby, and J. R. Chelikowsky, Phys. Rev. Lett. **81**, 4959 (1998).
237. V. V. Godlevsky, M. Jain, J. J. Derby, and J. R. Chelikowsky, Phys. Rev. B **60**, 8640 (1999).
238. S. Goedecker and K. Maschke, Phys. Rev. A **45**, 88 (1992).
239. S. Goedecker, M. Teter, and J. Hutter, Phys. Rev. B **54**, 1703 (1996).
240. S. Goedecker and C. J. Umrigar, Phys. Rev. A **55**, 1765 (1997).
241. S. Goedecker and O. V. Ivanov, Solid State Commun. **105**, 665 (1998).
242. S. Goedecker and O. V. Ivanov, Computers in Physics **12**, 548 (1998).
243. S. Goedecker, Rev. Mod. Phys. **71**, 1085 (1999).
244. H. Goldstein, *Klassische Mechanik* (Aula–Verlag, Wiesbaden, 1987).
245. P. Gomes Dacosta, O. H. Nielsen, and K. Kunc, J. Phys. C **19**, 3163 (1986).
246. X. G. Gong, D. Guenzburger, and E. B. Saitovitch, Chem. Phys. Lett. **275**,

392 (1997).
247. L. Gorb and J. Leszczynski, Comput. Chem. **3**, 179 (1999).
248. C. M. Goringe, L. J. Clark, M. H. Lee, M. C. Payne, I. Štich, J. A. White, M. J. Gillan, and A. P. Sutton, J. Phys. Chem. B **101**, 1498 (1997).
249. A. Görling, Phys. Rev. B **53**, 7024 (1996).
250. H. W. Graben and J. R. Ray, Mol. Phys. **80**, 1183 (1993).
251. T. Grabo, E. K. U. Gross, and M. Lüders, *Orbital Functionals in Density Functional Theory: The Optimized Effective Potential Method*; see `http://psi-k.dl.ac.uk/psi-k/highlights.html` and `ftp://ftp.dl.ac.uk/psik-network/Highlight_16.ps`.
252. J. Gräfenstein, E. Kraka, and D. Cremer, Chem. Phys. Lett. **288**, 593 (1998).
253. C. G. Gray and K. E. Gubbins, *Theory of Molecular Fluids Vol. 1* (Oxford, Clarendon, 1984).
254. J. C. Greer, R. Ahlrichs, and I. V. Hertel, Z. Phys. D **18**, 413 (1991); see in particular Sect. 3.
255. A. Gross, M. Bockstedte, and M. Scheffler, Phys. Rev. Lett. **79**, 701 (1997).
256. A. Gross, Surf. Sci. Rep. **32**, 291 (1998).
257. A. Gross, C.–M. Wei, and M. Scheffler, Surf. Sci. **416**, L1095 (1998).
258. E. K. U. Gross and W. Kohn, Adv. Quant. Chem. **21**, 255 (1990).
259. F. Grossmann, Comments At. Mol. Phys. **34**, 141 (1999).
260. M. P. Grumbach, D. Hohl, R. M. Martin, and R. Car, J. Phys.: Condens. Matter **6**, 1999 (1994).
261. M. P. Grumbach and R. M. Martin, Phys. Rev. B **54**, 15 730 (1996).
262. F. Gygi and A. Baldereschi, Phys. Rev. B **34**, 4405 (1986).
263. F. Gygi, Europhys. Lett. **19**, 617 (1992).
264. F. Gygi, Phys. Rev. B **48**, 11 692 (1993).
265. F. Gygi, Phys. Rev. B **51**, 11 190 (1995).
266. F. Gygi and G. Galli, Phys. Rev. B **52**, R2229 (1995).
267. F. Gygi, Comput. Mater. Sci. **10**, 63 (1998).
268. F. Haase, J. Sauer, and J. Hutter, Chem. Phys. Lett. **266**, 397 (1997).
269. F. Haase and J. Sauer, J. Am. Chem. Soc. **120**, 13 503 (1998).
270. R. Haberlandt, S. Fritzsche, G. Peinel, and K. Heinzinger, *Molekulardynamik – Grundlagen und Anwendungen* (Vieweg Verlag, Braunschweig, 1995).
271. J. C. L. Hageman, R. J. Meier, M. Heinemann, and R. A. de Groot, Macromolecules **30**, 5953 (1997).
272. H. Hakkinen, R. N. Barnett, and U. Landman, J. Phys. Chem. B **103**, 8814 (1999).
273. R. W. Hall and B. J. Berne, J. Chem. Phys. **81**, 3641 (1984).
274. D. R. Hamann, M. Schlüter, and C. Ciang, Phys. Rev. Lett. **43**, 1494 (1979).
275. D. R. Hamann, Phys. Rev. B **51**, 7337 (1995).
276. D. R. Hamann, Phys. Rev. B **51**, 9508 (1995).
277. D. R. Hamann, Phys. Rev. B **54**, 1568 (1996).
278. D. R. Hamann, Phys. Rev. B **56**, 14 979 (1997).
279. D. R. Hamann, Phys. Rev. Lett. **81**, 3447 (1998).
280. S. Hammes–Schiffer and H. C. Andersen, J. Chem. Phys. **99**, 523 (1993).
281. B. Hartke and E. A. Carter, Chem. Phys. Lett. **189**, 358 (1992).

282. B. Hartke and E. A. Carter, J. Chem. Phys. **97**, 6569 (1992).
283. B. Hartke and E. A. Carter, Chem. Phys. Lett. **216**, 324 (1993).
284. B. Hartke, D. A. Gibson, and E. A. Carter, Int. J. Quantum Chem. **45**, 59 (1993).
285. B. Hartke, private communication.
286. C. Hartwigsen, W. Witschel, and E. Spohr, Ber. Bunsenges. Phys. Chem. **101**, 859 (1997).
287. C. Hartwigsen, W. Witschel, and E. Spohr, Phys. Rev. B **55**, 4953 (1997).
288. C. Hartwigsen, S. Goedecker, and J. Hutter, Phys. Rev. B **58**, 3661 (1998).
289. K. C. Hass, W. F. Schneider, A. Curioni, and W. Andreoni, Science **282**, 265 (1998).
290. M. Head–Gordon and J. A. Pople, J. Phys. Chem. **92**, 3063 (1988).
291. F. Hedman and A. Laaksonen, Mol. Simul. **20**, 265 (1998).
292. W. J. Hehre, L. Radom, P. v. R. Schleyer, and J. A. Pople, *Ab Initio Molecular Orbital Theory* (John Wiley & Sons, New York, 1986).
293. A. Heidenreich and J. Sauer, Z. Phys. D **35**, 279 (1995).
294. V. Heine and D. Weaire, Solid St. Phys. **24**, 249 (1970).
295. H. Hellmann, Z. Phys. **85**, 180 (1933), see equation (3b); note that Hellmann already connects the "Hellmann–Feynman theorem" to first–order perturbation theory for cases where it does not hold rigorously, for instance when the wavefunction used to evaluate the expectation value is not the exact one.
296. H. Hellmann, J. Chem. Phys. **3**, 61 (1935).
297. H. Hellmann and W. Kassatotschkin, J. Chem. Phys. **4**, 324 (1936).
298. C. Herring, Phys. Rev. **57**, 1169 (1940).
299. K.–M. Ho, J. Ihm, and J. D. Joannopoulos, Phys. Rev. B **25**, 4260 (1982).
300. R. W. Hockney, Methods Comput. Phys. **9**, 136 (1970).
301. P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).
302. D. Hohl, R. O. Jones, R. Car, and M. Parrinello, Chem. Phys. Lett. **139**, 540 (1987).
303. D. Hohl, R. O. Jones, R. Car, and M. Parrinello, J. Chem. Phys. **89**, 6823 (1988).
304. D. Hohl and R. O. Jones, Phys. Rev. B **43**, 3856 (1991).
305. D. Hohl, V. Natoli, D. M. Ceperley, and R. M. Martin, Phys. Rev. Lett. **71**, 541 (1993).
306. K. Hoshino, F. Shimojo, and T. Nishida, J. Phys. Soc. Jap. **68**, 1907 (1999).
307. W. G. Hoover, Phys. Rev. A **31**, 1695 (1985).
308. J. Hutter, H. P. Lüthi, and M. Parrinello, Comput. Mat. Sci. **2**, 244 (1994).
309. J. Hutter, M. Parrinello, and S. Vogel, J. Chem. Phys. **101**, 3862 (1994).
310. J. Hutter, M. E. Tuckerman, and M. Parrinello, J. Chem. Phys. **102**, 859 (1995).
311. J. Hutter, P. Carloni, and M. Parrinello, J. Am. Chem. Soc. **118**, 8710 (1996).
312. J. Ihm, A. Zunger, and M. L. Cohen, J. Phys. C **12**, 4409 (1979).
313. T. Ikeda, M. Sprik, K. Terakura, and M. Parrinello, Phys. Rev. Lett. **81**, 4416 (1998).
314. Y. Imamura, N. Matsui, Y. Morikawa, M. Hada, T. Kubo, M. Nishijima, and H. Nakatsuji, Chem. Phys. Lett. **287** 131 (1998).

315. Y. Jeanvoine, J. G. Ángyán, G. Kresse, and J. Hafner, J. Phys. Chem. B **102**, 7307 (1998).
316. J. Jellinek, V. Bonačić–Koutecký, P. Fantucci, and M. Wiechert, J. Chem. Phys. **101**, 10 092 (1994).
317. J. Jellinek, S. Srinivas, and P. Fantucci, Chem. Phys. Lett. **288**, 705 (1998).
318. B. G. Johnson, P. M. W. Gill, and J. A. Pople, J. Chem. Phys. **98**, 5612 (1993).
319. D. D. Johnson, Phys. Rev. B **38**, 12087 (1988).
320. R. O. Jones and O. Gunnarsson, Rev. Mod. Phys. **61**, 689 (1989).
321. R. O. Jones, Angew. Chem. Int. Ed. Engl. **30**, 630 (1991).
322. Y.–M. Juan and E. Kaxiras, Phys. Rev. B **48**, 14 944 (1993).
323. R. Kaschner and D. Hohl, J. Phys. Chem. A **102**, 5111 (1998).
324. C. Katan, J. Phys. Chem. A **103**, 1407 (1999).
325. M. I. Katsnelson, M. van Schilfgaarde, V. P. Antropov, and B. N. Harmon, Phys. Rev. A **54**, 4802 (1996).
326. G. P. Kerker, J. Phys. C **13**, L189 (1980).
327. G. Kern and J. Hafner, Phys. Rev. B **58**, 13 167 (1998).
328. V. Keshari and Y. Ishikawa, Chem. Phys. Lett. **218**, 406 (1994).
329. E. Kim, Y. H. Lee, J. J. Lee, and Y. G. Hwang, Europhys. Lett. **40**, 147 (1997).
330. E. Kim, C. W. Oh, Y. H. Lee, Phys. Rev. Lett. **79**, 4621 (1997).
331. F. Kirchhoff, G. Kresse, and M. J. Gillan, Phys. Rev. B **57**, 10 482 (1998).
332. S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, Science **220**, 671 (1983).
333. H. Kitamura, S. Tsuneyuki, T. Ogitsu, and T. Miyake, *Quantum distribution of protons in solid molecular hydrogen under megabar pressure*, preprint.
334. H. Kleinert, *Path Integrals in Quantum Mechanics, Statistics and Polymer Physics* (World Scientific, Singapore, 1990).
335. L. Kleinman and D. M. Bylander, Phys. Rev. Lett. **48**, 1425 (1982).
336. A. Klesing, D. Labrenz, R. A. van Santen, J. Chem. Soc., Faraday Trans. **94**, 3229 (1998).
337. J. Kohanoff, S. Scandolo, G. L. Chiarotti, and E. Tosatti, Phys. Rev. Lett. **78**, 2783 (1997).
338. W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).
339. W. Kohn, Chem. Phys. Lett. **208**, 167 (1993).
340. W. Kołos, Adv. Quant. Chem. **5**, 99 (1970).
341. R. Kosloff, J. Phys. Chem. **92**, 2087 (1988).
342. R. Kosloff, Annu. Rev. Phys. Chem. **45**, 145 (1994).
343. G. Kresse and J. Hafner, Phys. Rev. B **47**, 558 (1993).
344. G. Kresse and J. Furthmüller, Phys. Rev. B **54**, 11 169 (1996).
345. G. Kresse and J. Furthmüller, Comput. Mat. Sci **6**, 15 (1996).
346. G. Kresse and J. Hafner, Phys. Rev. B **55**, 7539 (1997).
347. G. Kresse and D. Joubert, Phys. Rev. B **59**, 1758 (1999).
348. V. Kumar and V. Sundararajan, Phys. Rev. B **57**, 4939 (1998).
349. V. Kumar, Phys. Rev.B **60**, 2916 (1999).
350. W. Kutzelnigg, Mol. Phys. **90**, 909 (1997).
351. K. Laasonen, A. Pasquarello, R. Car, C. Lee, and D. Vanderbilt, Phys. Rev.

B **47**, 10 142 (1993).
352. K. Laasonen, M. Sprik, M. Parrinello, and R. Car, J. Chem. Phys. **99**, 9080 (1993).
353. K. Laasonen and M. L. Klein, J. Amer. Chem. Soc. **116**, 11 620 (1994).
354. K. Laasonen and M. L. Klein, J. Phys. Chem. **98**, 10 079 (1994).
355. D. Lamoen, P. Ballone, and M. Parrinello, Phys. Rev. B **54**, 5097 (1996).
356. D. Lamoen and M. Parrinello, Chem. Phys. Lett. **248**, 309 (1996).
357. C. Lanczos, J. Res. Nat. Bur. Stand. **45**, 255 (1950).
358. W. Langel and M. Parrinello, Phys. Rev. Lett. **73**, 504 (1994).
359. W. Langel and M. Parrinello, J. Chem. Phys. **103**, 3240 (1995).
360. W. Langel, Chem. Phys. Lett. **259**, 7 (1996).
361. G. Lauritsch and P.–G. Reinhard, Int. J. Mod. Phys. C **5**, 65 (1994).
362. C. Lee, W. Yang, and R. G. Parr, Phys. Rev. B **37**, 785 (1988).
363. C. Lee, D. Vanderbilt, K. Laasonen, R. Car, and M. Parrinello, Phys. Rev. Lett. **69**, 462 (1992).
364. C. Lee, D. Vanderbilt, K. Laasonen, R. Car, and M. Parrinello, Phys. Rev. B **47**, 4863 (1993).
365. C. Leforestier, J. Chem. Phys. **68**, 4406 (1978).
366. C. Leforestier, R. H. Bisseling, C. Cerjan, M. D. Feit, R. Friesner, A. Guldberg, A. Hammerich, G. Jolicard, W. Karrlein, H.–D. Meyer, N. Lipkin, O. Roncero, and R. Kosloff, J. Comp. Phys. **94**, 59 (1991).
367. J. G. LePage, M. Alouani, D. L. Dorsey, J. W. Wilkins, and P. E. Blöchl, Phys. Rev. B **58**, 1499 (1998).
368. I. N. Levine, *Quantum Chemistry* (Allyn and Bacon, Boston, 1983).
369. J.S. Lin, A. Qteish, M.C. Payne, and V. Heine, Phys. Rev. B **47**, 4174 (1993).
370. G. Lippert, J. Hutter, and M. Parrinello, Molec. Phys. **92**, 477 (1997).
371. G. Lippert, J. Hutter, and M. Parrinello, Theor. Chem. Acc. **103**, 124 (1999).
372. Z. Liu, L. E. Carter, and E. A. Carter, J. Phys. Chem. **99**, 4355 (1995).
373. Z. F. Liu, C. K. Siu, and J. S. Tse, Chem. Phys. Lett. **309**, 335 (1999).
374. S. G. Louie, S. Froyen, and M. L. Cohen, Phys. Rev. B **26**, 1738 (1982).
375. H. Löwen, P. A. Madden, and J.–P. Hansen, Phys. Rev. Lett. **68**, 1081 (1992); see also Ref. [376].
376. H. Löwen and I. D'amico, J. Phys.: Condens. Matter **9**, 8879 (1997).
377. M. Lubin and J. H. Weare, Mineral. Mag. **62A**, 915 (1998).
378. G. Makov and M. C. Payne, Phys. Rev. B **51**, 4014 (1995).
379. S. A. Maluendes and M. Dupuis, Int. J. Quantum Chem. **42**, 1327 (1992).
380. A. Marcellini, C. A. Pignedoli, M. Ferrario, and C. M. Bertoni, Surf. Sci. **402–404**, 47 (1998).
381. M. Marchi, J. Hutter, and M. Parrinello, J. Am. Chem. Soc. **118**, 7847 (1996).
382. P. Margl, T. Ziegler, and P. E. Blöchl, J. Am. Chem. Soc. **117**, 12 625 (1995).
383. P. M. Margl, T. K. Woo, P. E. Blöchl, and T. Ziegler, J. Am. Chem. Soc. **120**, 2174 (1998).
384. N. A. Marks, D. R. McKenzie, B. A. Pailthorpe, M. Barnasconi, and M. Parrinello, Phys. Rev. Lett. **76**, 768 (1996).
385. T. J. Martinez, M. Ben–Nun, and G. Ashkenazi, J. Chem. Phys. **104**, 2847 (1996).

386. T. J. Martinez, M. Ben–Nun, and R. D. Levine, J. Phys. Chem. **100**, 7884 (1996).
387. T. J. Martinez, Chem. Phys. Lett. **272**, 139 (1997).
388. G. J. Martyna, M. L. Klein, and M. Tuckerman, J. Chem. Phys. **97**, 2635 (1992).
389. G. J. Martyna, D. J. Tobias, and M. L. Klein, J. Chem. Phys. **101**, 4177 (1994).
390. G. J. Martyna, J. Chem. Phys. **104**, 2018 (1996).
391. G. J. Martyna, M. E. Tuckerman, D. J. Tobias, and M. L. Klein, Mol. Phys. **87**, 1117 (1996).
392. G. J. Martyna, A. Hughes, and M. E. Tuckerman, J. Chem. Phys. **110**, 3275 (1999).
393. G. J. Martyna and M. E. Tuckerman, J. Chem. Phys. **110**, 2810 (1999).
394. G. J. Martyna, K. Pihakari, and M. E. Tuckerman, to be published.
395. D. Marx and M. Parrinello, Z. Phys. B (Rapid Note) **95**, 143 (1994); a misprinted sign in the Lagrangian is correctly given in Eq. (319) of the present review.
396. D. Marx, J. Hutter, and M. Parrinello, Chem. Phys. Lett. **241**, 457 (1995).
397. D. Marx and M. Parrinello, Nature (London) **375**, 216 (1995).
398. D. Marx, E. Fois, and M. Parrinello, Int. J. Quantum Chem. **57**, 655 (1996).
399. D. Marx and M. Parrinello, J. Chem. Phys. **104**, 4077 (1996).
400. D. Marx and M. Parrinello, Science **271**, 179 (1996).
401. D. Marx and M. Parrinello, Z. Phys. D **41**, 253 (1997).
402. D. Marx and A. Savin, Angew. Chem. Int. Ed. Engl. **36**, 2077 (1997); Angew. Chem. **109**, 2168 (1997).
403. D. Marx, M. Sprik, and M. Parrinello, Chem. Phys. Lett. **272**, 360 (1997); see also Refs. [396,398].
404. D. Marx, in *Classical and Quantum Dynamics in Condensed Phase Simulations* Chapt. 15, eds. B. J. Berne, G. Ciccotti, und D. F. Coker (World Scientific, Singapore, 1998).
405. D. Marx, Nachr. Chem. Tech. Lab. **47**, 186 (1999).
406. D. Marx, in *New Approaches to Problems in Liquid State Theory* p. 439, eds. C. Caccamo, J.–P. Hansen, and G. Stell (Kluwer, Dordrecht, 1999).
407. D. Marx and M. H. Müser, J. Phys.: Condens. Matter **11**, R117 (1999).
408. D. Marx and M. Parrinello, Science **284**, 59 (1999); see also Ref. [408].
409. D. Marx and M. Parrinello, Science **286**, 1051 (1999); see `http://www.sciencemag.org/cgi/content/full/286/5442/1051a`.
410. D. Marx, M. E. Tuckerman, J. Hutter, and M. Parrinello, Nature (London) **397**, 601 (1999); see also J. T. Hynes, Nature (London) **397**, 565 (1999).
411. D. Marx, M. E. Tuckerman, and G. J. Martyna, Comput. Phys. Commun. **118**, 166 (1999); the misprinted definition of the fictitious normal mode masses given in Eq. (2.51) is corrected in Eq. (338) of the present review. However, the correct definition was implemented in the `CPMD` package [142] so that all data reported in the paper are unaffected.
412. D. Marx, M. E. Tuckerman, and M. Parrinello, *Solvated excess protons in water: Quantum effects on the hydration structure*, J. Phys.: Condens. Matter

(in press).
413. N. Marzari and D. Vanderbilt, Phys. Rev. B **56**, 12 847 (1997).
414. N. Marzari, D. Vanderbilt, and M. C. Payne, Phys. Rev. Lett. **79**, 1337 (1997).
415. N. Marzari, D. Vanderbilt, A. De Vita and M. C. Payne, Phys. Rev. Lett. **82**, 3296 (1999).
416. C. Massobrio, A. Pasquarello, and R. Car, Phys. Rev. Lett. **80**, 2342 (1998).
417. F. Mauri, G. Galli, and R. Car, Phys. Rev. B **47**, 9973 (1993).
418. R. McWeeny, *Methods of Molecular Quantum Mechanics* (Academic Press, London, 1992).
419. H. S. Mei, M. E. Tuckerman, D. E. Sagnella, and M. L. Klein, J. Phys. Chem. B **102**, 10 446 (1998).
420. E. J. Meijer and M. Sprik, J. Chem. Phys. **105**, 8684 (1996).
421. E. J. Meijer and M. Sprik, J. Phys. Chem. A **102**, 2893 (1998).
422. E. J. Meijer and M. Sprik, J. Am. Chem. Soc. **120**, 6345 (1998).
423. B. Meng, D. Maroudas, and W. H. Weinberg, Chem. Phys. Lett. **278**, 97 (1997).
424. N. D. Mermin, Phys. Rev. **137**, A1441 (1965).
425. A. Messiah, *Quantum Mechanics* (North–Holland Publishing Company, Amsterdam, 1964); see in particular Chapter VI.I.4 in Volume I.
426. H.–D. Meyer and W. H. Miller, J. Chem. Phys. **70**, 3214 (1979).
427. S. Miura, M. E. Tuckerman, and M. L. Klein, J. Chem. Phys. **109**, 5290 (1998).
428. T. Miyake, T. Ogitsu, and S. Tsuneyuki, Phys. Rev. Lett. **81**, 1873 (1998).
429. T. Miyake, T. Ogitsu, and S. Tsuneyuki, *First–principles study of the quantum states of muonium and hydrogen in crystalline silicon*, Phys. Rev. B (in press).
430. Y. Miyamoto, O. Sugino, and Y. Mochizuki, Appl. Phys. Lett. **75**, 2915 (1999).
431. N. A. Modine, G. Zumbach, and E. Kaxiras, Phys. Rev. B **55**, 10 289 (1997).
432. C. Molteni, N. Marzari, M. C. Payne, and V. Heine, Phys. Rev. Lett. **79**, 869 (1997).
433. C. Molteni and M. Parrinello, J. Am. Chem. Soc. **120**, 2168 (1998); see also Ref. [434].
434. C. Molteni and M. Parrinello, Chem. Phys. Lett. **275**, 409 (1998).
435. H. J. Monkhorst and J. D. Pack, Phys. Rev. B **13**, 5189 (1976).
436. T. Morishita and S. Nosé, Phys. Rev. B **59**, 15126 (1999).
437. J. J. Mortensen and M. Parrinello, to be published.
438. V. Musolino, A. Selloni, and R. Car, Phys. Rev. Lett. **83**, 3242 (1999).
439. Á. Nagy, Phys. Rev. A **57**, 1672 (1998).
440. O. H. Nielsen and R. M. Martin, Phys. Rev. Lett. **50**, 697 (1983).
441. O. H. Nielsen and R. M. Martin, Phys. Rev. B **32**, 3780 (1985); **32**, 3792 (1985).
442. S. Nosé and M. L. Klein, Mol. Phys. **50**, 1055 (1983).
443. S. Nosé, Mol. Phys. **52**, 255 (1984).
444. S. Nosé, J. Chem. Phys. **81**, 511 (1984).
445. S. Nosé, Prog. Theor. Phys. Suppl. **103**, 1 (1991).

446. `NWChem:` developed and distributed by Pacific Northwest National Laboratory, USA.
447. M. Odelius, M. Bernasconi, and M. Parrinello, Phys. Rev. Lett. **78**, 2855 (1997).
448. M. Odelius, Phys. Rev. Lett. **82**, 3919 (1999).
449. K.–d. Oh and P. A. Deymier, Phys. Rev. Lett. **81**, 3104 (1998).
450. K.–d. Oh and P. A. Deymier, Phys. Rev. B **58**, 7577 (1998).
451. T. Ohtsuki, K. Ohno, K. Shiga, Y. Kawazoe, Y. Maruyama, and K. Masumoto, Phys. Rev. Lett. **81**, 967 (1998).
452. P. Ordejon, D. Drabold, M Grumbach, and R.M. Martin, Phys. Rev. B **48**, 14646 (1993).
453. P. Ordejón, Comp. Mater. Sci. **12**, 157 (1998).
454. J. Ortega, J. P. Lewis, and O. F. Sankey, Phys. Rev. B **50**, 10 516 (1994).
455. J. Ortega, J. P. Lewis, and O. F. Sankey, J. Chem. Phys. **106**, 3696 (1997).
456. A. Palma, A. Pasquarello, G. Ciccotti, and R. Car, J. Chem. Phys. **108**, 9933 (1998).
457. M. Palummo, L. Reining, and P. Ballone, J. Phys. IV (Paris) **3:(C7)**, 1955 (1993).
458. R. G. Parr and W. Yang, *Density–Functional Theory of Atoms and Molecules* (Oxford University Press, Oxford, 1989).
459. M. Parrinello and A. Rahman, Phys. Rev. Lett. **45**, 1196 (1980).
460. M. Parrinello and A. Rahman, J. Appl. Phys. **52**, 7182 (1981).
461. M. Parrinello and A. Rahman, J. Chem. Phys. **76**, 2662 (1982).
462. M. Parrinello and A. Rahman, J. Chem. Phys. **80**, 860 (1984).
463. M. Parrinello, Solid State Commun. **102**, 107 (1997).
464. A. Pasquarello, K. Laasonen, R. Car, Ch. Lee, and D. Vanderbilt, Phys. Rev. Lett. **69**, 1982 (1992).
465. A. Pasquarello and R. Car, Phys. Rev. Lett. **80**, 5145 (1998).
466. A. Pasquarello, J. Sarnthein, and R. Car, Phys. Rev. B: **57**, 14 133 (1998).
467. G. Pastore, E. Smargiassi, and F. Buda, Phys. Rev. A **44**, 6334 (1991).
468. G. Pastore, in *Monte Carlo and Molecular Dynamics of Condensed Matter Systems*, Chapt. 24, p. 635, eds. K. Binder and G. Ciccotti (Italien Physical Society SIF, Bologna, 1996).
469. M. Pavese, D. R. Berard, and G. A. Voth, Chem. Phys. Lett. **300**, 93 (1999).
470. M. C. Payne, J. Phys.: Condens. Matter **1**, 2199 (1989).
471. M. C. Payne, J. D. Joannopoulos, D. C. Allan, M. P. Teter, and D. Vanderbilt, Phys. Rev. Lett. **56**, 2656 (1986).
472. M. C. Payne, M. P. Teter, D. C. Allan, T. A. Arias, and J. D. Joannopoulos, Rev. Mod. Phys. **64**, 1045 (1992).
473. M. Pearson, E. Smargiassi, and P. Madden, J. Phys. Condens. Matter **5**, 3221 (1993).
474. G. La Penna, F. Buda, A. Bifone, and H. J. M. de Groot, Chem. Phys. Lett. **294**, 447 (1998).
475. J. P. Perdew and A. Zunger, Phys. Rev. B **23**, 5048 (1981).
476. J. P. Perdew, Phys. Rev. B **33**, 8822 (1986); Erratum: Phys. Rev. B **34**, 7406 (1986).

477. J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh, and C. Fiolhais, Phys. Rev. B **46**, 6671 (1992); Erratum: Phys. Rev. B **48**, 4978 (1993).
478. J. P. Perdew and Y. Wang, Phys. Rev. B **45**, 13 244 (1992).
479. J. P. Perdew, K. Burke, M. Ernzerhof, Phys. Rev. Lett. **77**, 3865 (1996); Erratum: Phys. Rev. Lett. **78**, 1396 (1997); see Refs. [480,706] for revised versions of the "PBE" functional.
480. J. P. Perdew, K. Burke, M. Ernzerhof, Phys. Rev. Lett. **80**, 891 (1998).
481. R. Perez, M. C. Payne, I. Štich, and K. Terakura, Appl. Surf. Sci. **123/124**, 249 (1998).
482. R. Perez, I. Štich I, M. C. Payne MC, and K. Terakura, Appl. Surf. Sci. **140**, 320 (1999).
483. B. G. Pfrommer, J. Demmel, and H. Simon, J. Comput. Phys. **150**, 287 (1999).
484. J. C. Phillips, Phys. Rev. **112**, 685 (1958).
485. J. C. Phillips and L. Kleinman, Phys. Rev. **116**, 287 (1959).
486. Physics and Astronomy Classification Scheme (PACS); see `http://publish.aps.org/PACS.`
487. W. E. Pickett, Comput. Phys. Rep. **9**, 115 (1989).
488. C. Pierleoni, D. M. Ceperley, B. Bernu, and W. R. Magro, Phys. Rev. Lett. **73**, 2145 (1994).
489. W.T. Pollard and R.A. Friesner, J. Chem. Phys. **99**, 6742 (1993).
490. J. Polonyi and H. W. Wild, Phys. Rev. Lett. **51**, 2257 (1983).
491. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes – The Art of Scientific Computing*, Chapt. 9.6, (Cambridge University Press, Cambridge, 1992).
492. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes – The Art of Scientific Computing*, Chapt. 12.1, (Cambridge University Press, Cambridge, 1992).
493. E. I. Proynov, S. Sirois, and D. R. Salahub, Int. J. Quantum Chem. **64**, 427 (1997).
494. P. Pulay, Molec. Phys. **17**, 197 (1969).
495. P. Pulay, Chem. Phys. Lett. **73**, 393 (1980).
496. P. Pulay, Adv. Chem. Phys. **69**, 241 (1987).
497. A. Putrino and G. B. Bachelet, in *Advances in Computational Materials Science II*, eds. V. Fiorentini and F. Meloni (SIF, Bologna, 1998).
498. M. Qiu, X.–Y. Zhou, M. Jiang, P.–L. Cao, and Z. Zeng, Phys. Lett. A **245**, 430 (1998).
499. A. Qteish, Phys. Rev. B **52**, 14497 (1995).
500. M. R. Radeke and E. A. Carter, Annu. Rev. Phys. Chem. **48**, 243 (1997).
501. B. De Raedt, M. Sprik, and M. L. Klein, J. Chem. Phys. **80**, 5719 (1984).
502. L. M. Ramaniah, M. Bernasconi, and M. Parrinello, J. Chem. Phys. **111**, 1587 (1999).
503. R. Ramírez, J. Schulte, and M. C. Böhm, Chem. Phys. Lett. **275**, 377 (1997).
504. R. Ramírez, E. Hernandez, J. Schulte, and M. C. Böhm, Chem. Phys. Lett. **291**, 44 (1998).

505. R. Ramírez, T. López–Ciudad, and J. C. Noya, Phys. Rev. Lett. **81**, 3303 (1998).
506. R. Ramírez and T. López–Ciudad, J. Chem. Phys. **111**, 3339 (1999).
507. R. Ramírez and T. López–Ciudad, Phys. Rev. Lett. **83**, 4456 (1999).
508. G. Ranghino, A. Anselmino, L. Meda, C. Tonini, and G. F. Cerofolini, J. Phys. Chem. B **101**, 7723 (1997).
509. A. M. Rappe, K. M. Rabe, E. Kaxiras, and J. D. Joannopoulos, Phys. Rev. B **41**, 1227 (1990).
510. A. M. Rappe, J. D. Joannopoulos, and P. A. Bash, J. Am. Chem. Soc. **114**, 6466 (1992).
511. S. Raugei, P. L. Silvestrelli, and M. Parrinello, Phys. Rev. Lett. **83**, 2222 (1999).
512. P. Raybaud, J. Hafner, G. Kresse, and H. Toulhoat, Phys. Rev. Lett. **80**, 1481 (1998).
513. D. K. Remler and P. A. Madden, Molec. Phys. **70**, 921 (1990).
514. S. L. Richardson and J. L. Martins, Phys. Rev. B **58**, 15 307 (1998).
515. T. von Rosenvinge, M. Parrinello, and M. L. Klein, J. Chem. Phys. **107**, 8012 (1997).
516. T. von Rosenvinge, M. E. Tuckerman, and M. L. Klein, Faraday Discuss. **106**, 273 (1997).
517. U. Röthlisberger and W. Andreoni, J. Chem. Phys. **94**, 8129 (1991).
518. U. Röthlisberger and W. Andreoni, Chem. Phys. Lett. **198**, 478 (1992).
519. U. Röthlisberger, W. Andreoni, and M. Parrinello, Phys. Rev. Lett. **72**, 665 (1994).
520. U. Röthlisberger and M. L. Klein, J. Am. Chem. Soc. **117**, 42 (1995).
521. U. Röthlisberger and M. Parrinello, J. Chem. Phys. **106**, 4658 (1997).
522. U. Röthlisberger, M. Sprik, and M. L. Klein, J. Chem. Soc., Faraday Trans. **94**, 501 (1998).
523. U. Röthlisberger and P. Carloni, Int. J. Quantum Chem. **73**, 209 (1999).
524. R. Rousseau, G. Dietrich, S. Krückeberg, K. Lützenkirchen, D. Marx, L. Schweikhard, and C. Walther, Chem. Phys. Lett. **295**, 41 (1998).
525. R. Rousseau and D. Marx, Phys. Rev. Lett. **80**, 2574 (1998).
526. R. Rousseau and D. Marx, J. Chem. Phys. **111**, 5091 (1999).
527. C. Rovira and M. Parrinello, Int. J. Quantum Chem. **70**, 387 (1998).
528. C. Rovira, K. Kunc, J. Hutter, P. Ballone, and M. Parrinello, Int. J. Quantum Chem. **69**, 31 (1998).
529. C. Rovira and M. Parrinello, Chem. Eur. J. **5**, 250 (1999).
530. E. Ruiz and M. C. Payne, Chem. Eur. J. **4**, 2485 (1998).
531. A. Rytkonen, H. Hakkinen, and M. Manninen, Phys. Rev. Lett. **80**, 3940 (1998).
532. U. Saalmann and R. Schmidt, Z. Phys. D **38**, 153 (1996).
533. A. M. Saitta and M. L. Klein, J. Chem. Phys. **111**, 9434 (1999).
534. A. M. Saitta, P. D. Soper, E. Wasserman, and M. L. Klein, Nature (London) **399**, 46 (1999).
535. J. J. Sakurai, *Modern Quantum Mechanics* (Addison–Wesley Publishing Company, Redwood City, 1985); see in particular Chapter 2.4.

536. H. Sambe and R. H. Felton, J. Chem. Phys. **62**, 1122 (1975).
537. D. Sánchez–Portal, E. Artacho, and J. M. Soler, J. Phys.: Condens. Matter **8**, 3859 (1996).
538. E. Sandré and A. Pasturel, Mol. Simul. **20**, 63 (1997).
539. G. C. Schatz, Rev. Mod. Phys. **61**, 669 (1989).
540. F. Scheck, *Mechanik* (Springer Verlag, Berlin, 1988); see in particular Chapter 2.35.
541. R. Schinke, *Photodissociation Dynamics* (Cambridge University Press, Cambridge, 1995).
542. K. E. Schmidt and D. M. Ceperley, in *The Monte Carlo Method in Condensed Matter Physics*, p. 205, ed. K. Binder (Springer Verlag, Berlin, 1992).
543. J. Schulte, M. C. Böhm, and R. Ramírez, Mol. Phys. **93**, 801 (1998).
544. J. Schütt, M. C. Böhm, and R. Ramírez, Chem. Phys. Lett. **248**, 379 (1996).
545. K. Schwarz, E. Nusterer, P. Margl, and P. E. Blöchl, Int. J. Quantum Chem. **61**, 369 (1997).
546. K. Schwarz, E. Nusterer, and P. E. Blöchl, Catal. Today **50**, 501 (1999).
547. M. D. Segall, M. C. Payne, S. W. Ellis, G. T. Tucker, and R. N. Boyes, Eur. J. Drug Metab. Pharmacokinet. **22**, 283 (1997).
548. M. D. Segall, M. C. Payne, S. W. Ellis, G. T. Tucker, and R. N. Boyes, Phys. Rev. E **57**, 4618 (1998).
549. M. D. Segall, M. C. Payne, S. W. Ellis, G. T. Tucker, and P. J. Eddershaw, Xenobiotica **29**, 561 (1999).
550. G. Seifert and R. O. Jones, Z. Phys. D **20**, 77 (1991).
551. G. Seifert, D. Porezag, and Th. Frauenheim, Int. J. Quantum Chem. **58**, 185 (1996).
552. G. Seifert, R. Kaschner, M. Schöne, and G. Pastore, J. Phys.: Condens. Matter **10**, 1175 (1998).
553. A. Selloni, P. Carnevali, R. Car, and M. Parrinello, Phys. Rev. Lett. **59**, 823 (1987).
554. A. Selloni, A. Vittadini, M. Grätzel, Surf. Sci. **402–404**, 219 (1998).
555. Y. Senda, F. Shimojo, and K. Hoshino, J. Phys.: Condens. Matter **11**, 2199 (1999).
556. Y. Senda, F. Shimojo, and K. Hoshino, J. Phys.: Condens. Matter **11**, 5387 (1999).
557. S. Serra, G. Chiarotti, S. Scandolo, and E. Tosatti, Phys. Rev. Lett. **80**, 5160 (1998).
558. S. Serra, C. Cavazzoni, G. L. Chiarotti, S. Scandolo, and E. Tosatti, Science **284**, 788 (1999).
559. R. Shah, M. C. Payne, M.-H. Lee, and J. D. Gale, Science **271**, 1395 (1996).
560. R. Shah, M. C. Payne, and J. D. Gale, Int. J. Quantum Chem. **61**, 393 (1997).
561. F. Shimojo, K. Hoshino, and H. Okazaki, Solid State Ionics **113**, 319 (1998).
562. F. Shimojo, K. Hoshino, and Y. Zempo, J. Phys.: Condens. Matter **10**, L177 (1998).
563. F. Shimojo, S. Munejiri, K. Hoshino, and Y. Zempo, J. Phys.: Condens. Matter **11**, L153 (1999).
564. E. L. Shirley, D. C. Allan, R. M. Martin, and J. D. Joannopoulos, Phys. Rev.

B **40**, 3652 (1989).
565. A. J. R. da Silva, H.–Y. Cheng, D. A. Gibson, K. L. Sorge, Z. Liu, and E. A. Carter, Spectrochim. Acta A **53**, 1285 (1997).
566. A. J. R. da Silva, M. R. Radeke, and E. A. Carter, Surf. Sci. Lett. **381**, L628 (1997).
567. A. J. R. da Silva, J. W. Pang, E. A. Carter, and D. Neuhauser, J. Phys. Chem. A **102**, 881 (1998).
568. C. R. S. da Silva and R. W. Wentzcovitch, Comp. Mat. Sci. **8**, 219 (1997).
569. P. L. Silvestrelli, A. Alavi, M. Parrinello, and D. Frenkel, Europhys. Lett. **33**, 551 (1996).
570. P. L. Silvestrelli, A. Alavi, M. Parrinello, and D. Frenkel, Phys. Rev. Lett. **77**, 3149 (1996).
571. P. L. Silvestrelli, A. Alavi, and M. Parrinello, Phys. Rev. B **55**, 15 515 (1997).
572. P. L. Silvestrelli, A. Alavi, M. Parrinello, and D. Frenkel, Phys. Rev. B **56**, 3806 (1997).
573. P. L. Silvestrelli, M. Bernasconi, and M. Parrinello, Chem. Phys. Lett. **277**, 478 (1997).
574. P. L. Silvestrelli and M. Parrinello, J. Appl. Phys. **83**, 2478 (1998).
575. P. L. Silvestrelli and M. Parrinello, Phys. Rev. Lett. **82**, 3308 (1999); Erratum: Phys. Rev. Lett. **82**, 5415 (1999).
576. P. L. Silvestrelli and M. Parrinello, J. Chem. Phys. **111**, 3572 (1999).
577. F. Sim, A. St.–Amant, I. Papai, and D. R. Salahub, J. Am. Chem. Soc. **114**, 4391 (1992).
578. D. J. Singh, *Planewaves, Pseudopotentials and the* LAPW *Method* (Kluwer, Dordrecht, 1994).
579. L. D. Site, A. Alavi, and R. M. Lynden–Bell, Mol. Phys. **96**, 1683 (1999).
580. V. Smelyansky, J. Hafner, and G. Kresse, Phys. Rev. B **58**, R1782 (1998).
581. A. K. Soper, J. Phys.: Condens. Matter **9**, 2717 (1997).
582. M. Sprik and M. L. Klein, J. Chem. Phys. **89**, 1592 (1988); Erratum: J. Chem. Phys. **90**, 7614 (1989).
583. M. Sprik, J. Phys. Chem. **95**, 2283 (1991).
584. M. Sprik, in *Computer Simulation in Chemical Physics*, eds. M. P. Allen and D. J. Tildesley (Kluwer, Dordrecht, 1993).
585. M. Sprik, in *Monte Carlo and Molecular Dynamics of Condensed Matter Systems*, Chapt. 2, p. 43, eds. K. Binder and G. Ciccotti (Italien Physical Society SIF, Bologna, 1996).
586. M. Sprik, J. Phys.: Condens. Matter **8**, 9405 (1996).
587. M. Sprik, J. Hutter, and M. Parrinello, J. Chem. Phys. **105**, 1142 (1996).
588. M. Sprik, in *Classical and Quantum Dynamics in Condensed Phase Simulations*, Chapt. 13, eds. B. J. Berne, G. Ciccotti, and D. F. Coker (World Scientific, Singapore, 1998).
589. M. Sprik and G. Ciccotti, J. Chem. Phys. **109**, 7737 (1998).
590. M. Springborg (Ed.), *Density–Functional Methods in Chemistry and Materials Science*, (John Wiley & Sons, New York, 1997).
591. G. P. Srivastava and D. Weaire, Adv. Phys. **36**, 463 (1987).
592. R. Stadler, A. Alfe, G. Kresse, G. A. de Wijs, and M. J. Gillan, J. Non–Cryst.

Solids **250**–**252**, 82 (1999).
593. T. Starkloff and J.D. Joannopoulos, Phys. Rev. B **16**, 5212 (1977).
594. I. Štich, R. Car, M. Parrinello, and S. Baroni, Phys. Rev. B **39**, 4997 (1989).
595. I. Štich, Surf. Sci. **368**, 152 (1996).
596. I. Štich, D. Marx, M. Parrinello, and K. Terakura, Phys. Rev. Lett. **78**, 3669 (1997).
597. I. Štich, D. Marx, M. Parrinello, and K. Terakura, J. Chem. Phys. **107**, 9482 (1997).
598. I. Štich, J. D. Gale, K. Terakura, and M. C. Payne, Chem. Phys. Lett. **283**, 402 (1998).
599. I. Štich, J. D. Gale, K. Terakura, and M. C. Payne, J. Am. Chem. Soc. **121**, 3292 (1999).
600. A. C. Stückl, C. A. Daul, and H. U. Güdel, Int. J. Quantum Chem. **61**, 579 (1997).
601. P. Stumm and D. A. Drabold, Phys. Rev. Lett. **79**, 677 (1997).
602. D. M. Sullivan, K. Bagchi, M. E. Tuckerman, and M. L. Klein, J. Phys. Chem. A **103**, 8678 (1999).
603. W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson, J. Chem. Phys. **76**, 637 (1982).
604. A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry – Introduction to Advanced Electronic Structure Theory* (McGraw–Hill Publishing Company, New York, 1989).
605. H. Tachikawa and M. Igarashi, Chem. Phys. Lett. **303**, 81 (1999).
606. M. Takahashi and M. Imada, J. Phys. Soc. Japan **53**, 963 (1984); **53**, 3765 (1984); **53**, 3770 (1984).
607. N. Takeuchi, A. Selloni, and E. Tosatti, Phys. Rev. B **55**, 15 405 (1997).
608. S. Tanaka, J. Chem. Phys. **100**, 7416 (1994).
609. A. Tarazona, E. Koglin, F. Buda, B. B. Coussens, J. Renkema, S. van Heel, and R. J. Maier, J. Phys. Chem. B **101**, 4370 (1997).
610. F. Tassone, F. Mauri, and R. Car, Phys. Rev. B **50**, 10 561 (1994).
611. K. Terakura, T. Yamasaki, T. Uda, and I. Štich, Surf. Sci. **386**, 207 (1997).
612. V. Termath and J. Sauer, Chem. Phys. Lett. **255**, 187 (1996).
613. V. Termath and J. Sauer, Mol. Phys. **91**, 963 (1997).
614. V. Termath, F. Haase, J. Sauer, J. Hutter, and M. Parrinello, J. Am. Chem. Soc. **120**, 8512 (1998).
615. F. Terstegen, E. A. Carter, and V. Buss, Int. J. Quantum Chem. **75**, 141 (1999).
616. M. P. Teter, M. C. Payne, and D. C. Allen, Phys. Rev. B **40**, 12 255 (1989).
617. J. Theilhaber, Phys. Rev. B **46**, 12 990 (1992)
618. D. Thouless, J. Nucl. Phys. **21**, 225 (1960).
619. J. Tobik, I. Štich, R. Perez, and K. Terakura, Phys. Rev. B **60**, 11 639 (1999).
620. A. Tongraar, K. R. Liedl, and B. M. Rode, J. Phys. Chem. A **101**, 6299 (1997).
621. A. Tongraar, K. R. Liedl, and B. M. Rode, J. Phys. Chem. A **102**, 10 340 (1998).
622. W.C. Topp and J.J. Hopfield, Phys. Rev. B **7**, 1295 (1974).

623. E. Tornaghi, W. Andreoni, P. Carloni, J. Hutter, and M. Parrinello, Chem. Phys. Lett. **246**, 469 (1995).
624. G. Toth, J. Phys. Chem. A **101**, 8871 (1997).
625. A. Trave, F. Buda, and A. Fasolino, Phys. Rev. Lett. **77**, 5405 (1996).
626. N. Trouiller and J. L. Martins, Phys. Rev. B **43**, 1993 (1991).
627. N. Trouiller and J. L. Martins, Phys. Rev. B **43**, 8861 (1991).
628. B. L. Trout and M. Parrinello, Chem. Phys. Lett. **288**, 343 (1998).
629. B. L. Trout and M. Parrinello, J. Phys. Chem. B **103**, 7340 (1999).
630. J. S. Tse, D. D. Klug, and K. Laasonen, Phys. Rev. Lett. **74**, 876 (1995).
631. J. S. Tse and D. D. Klug, Phys. Rev. B **59**, 34 (1999).
632. E. Tsuchida and M. Tsukada, Solid State Commun. **94**, 5 (1995).
633. E. Tsuchida and M. Tsukada, Phys. Rev. B **52**, 5573 (1995).
634. E. Tsuchida and M. Tsukada, J. Phys. Soc. Japan **67**, 3844 (1998).
635. E. Tsuchida, Y. Kanada, and M. Tsukada, Chem. Phys. Lett. **311**, 236 (1999).
636. M. Tuckerman, B. J. Berne, and G. J. Martyna, J. Chem. Phys. **97**, 1990 (1992).
637. M. E. Tuckerman, B. J. Berne, G. J. Martyna, and M. L. Klein, J. Chem. Phys. **99**, 2796 (1993).
638. M. E. Tuckerman and M. Parrinello, J. Chem. Phys. **101**, 1302 (1994); see also Ref. [310].
639. M. E. Tuckerman and M. Parrinello, J. Chem. Phys. **101**, 1316 (1994).
640. M. Tuckerman, K. Laasonen, M. Sprik, and M. Parrinello, J. Phys.: Condens. Matter **6**, A93 (1994).
641. M. Tuckerman, K. Laasonen, M. Sprik, and M. Parrinello, J. Phys. Chem. **99**, 5749 (1995).
642. M. Tuckerman, K. Laasonen, M. Sprik, and M. Parrinello, J. Chem. Phys. **103**, 150 (1995).
643. M. E. Tuckerman, P. J. Ungar, T. von Rosenvinge, and M. L. Klein, J. Phys. Chem. **100**, 12 878 (1996).
644. M. E. Tuckerman, D. Marx, M. L. Klein, and M. Parrinello, J. Chem. Phys. **104**, 5579 (1996).
645. M. E. Tuckerman, D. Marx, M. L. Klein, and M. Parrinello, Science **275**, 817 (1997).
646. M. E. Tuckerman and A. Hughes, in *Classical and Quantum Dynamics in Condensed Phase Simulations*, Chapt. 14, p. 311, eds. B. J. Berne, G. Ciccotti, and D. F. Coker (World Scientific, Singapore, 1998).
647. M. E. Tuckerman and M. L. Klein, Chem. Phys. Lett. **283**, 147 (1998).
648. M. E. Tuckerman, private communication.
649. J. C. Tully, in *Modern Theoretical Chemistry: Dynamics of Molecular Collisions*, Part B, p. 217, ed. W. H. Miller (Plenum Press, New York, 1976).
650. J. C. Tully, in *Modern Methods for Multidimensional Dynamics Computations in Chemistry*, ed. D. L. Thompson (World Scientific, Singapore, 1998).
651. J. C. Tully, in *Classical and Quantum Dynamics in Condensed Phase Simulations*, Chapt. 21, p. 489, eds. B. J. Berne, G. Ciccotti, and D. F. Coker (World Scientific, Singapore, 1998).

652. C. J. Tymczak and X.–Q. Wang, Phys. Rev. Lett. **78**, 3654 (1997).
653. T. Uchiyama, T. Uda, and K. Terakura, Surf. Sci. **433–435**, 896 (1999).
654. K. Uehara, M. Ishitobi, T. Oda, and Y. Hiwatari, Mol. Simul. **18**, 385 (1997); Mol. Simul. **19**, 75 (1997).
655. C. P. Ursenbach, A. Calhoun, and G. A. Voth, J. Chem. Phys. **106**, 2811 (1997).
656. R. M. Valladares, A. J. Fisher, and W. Hayes, Chem. Phys. Lett. **242**, 1 (1995).
657. R. M. Valladares, A. J. Fisher, S. J. Blundell, and W. Hayes, J. Phys.: Condens. Matter **10**, 10 701 (1998).
658. D. Vanderbilt and S. G. Louie, Phys. Rev. B **30**, 6118 (1984).
659. D. Vanderbilt, Phys. Rev. B **32**, 8412 (1985).
660. D. Vanderbilt, Phys. Rev. Lett. **59**, 1456 (1987).
661. D. Vanderbilt, Phys. Rev. B **41**, 7892 (1990).
662. T. Van Voorhis and G. Scuseria, J. Chem. Phys. **109**, 400 (1998).
663. `VASP:` see Ref. [344].
664. L. Verlet, Phys. Rev. **159**, 98 (1967).
665. G. A. Voth, Adv. Chem. Phys. **93**, 135 (1996).
666. D. S. Wallace, A. M. Stoneham, W. Hayes, A. J. Fisher, and A. H. Harker, J. Phys.: Condens. Matter **3**, 3879 (1991).
667. D. S. Wallace, A. M. Stoneham, W. Hayes, A. J. Fisher, and A. Testa, J. Phys.: Condens. Matter **3**, 3905 (1991).
668. C. Wang and Q.–M. Zhang, Phys. Rev. B **59**, 4864 (1999).
669. I. S. Y. Wang and M. Karplus, J. Amer. Chem. Soc. **95**, 8160 (1973).
670. M. C. Warren, G. J. Ackland, B. B. Karki, and S. J. Clark, Mineral. Mag. **62**, 585 (1998).
671. A. Warshel and M. Karplus, Chem. Phys. Lett. **32**, 11 (1975).
672. R. O. Weht, J. Kohanoff, D. A. Estrin, and C. Chakravarty, J. Chem. Phys. **108**, 8848 (1998).
673. D. Wei and D. R. Salahub, J. Chem. Phys. **106**, 6086 (1997).
674. S. Wei and M. Y. Chou, Phys. Rev. Lett. **76**, 2650 (1996).
675. F. Weich, J. Widany, and Th. Frauenheim, Carbon **37**, 545 (1999).
676. M. Weinert and J. W. Davenport, Phys. Rev. B **45**, 13 709 (1992).
677. S. Wengert, R. Nesper, W. Andreoni, and M. Parrinello, Phys. Rev. Lett. **77**, 5083 (1996).
678. R. M. Wentzcovitch, Phys. Rev. B **44**, 2358 (1991).
679. R. M. Wentzcovitch and J. L. Martins, Solid State Commun. **78**, 831 (1991).
680. R. M. Wentzcovitch, J. L. Martins, and P. B. Allen, Phys. Rev. B **45**, 11 372 (1992).
681. R. M. Wentzcovitch, J. L. Martins, and G. D. Price, Phys. Rev. Lett. **70**, 3947 (1993).
682. R. M. Wentzcovitch, Phys. Rev. B **50**, 10 358 (1994).
683. R. M. Wentzcovitch and G. D. Price, Top. Mol. Organ. Eng. **15**, 39 (1997).
684. R. M. Wentzcovitch, C. da Silva, J. R. Chelikowsky, and N. Binggeli, Phys. Rev. Lett. **80**, 2149 (1998).
685. J. A. White and D. M. Bird, Phys. Rev. B **50**, 4954 (1994).

686. S. R. White, J. W. Wilkins, and M. P. Teter, Phys. Rev. B **39**, 5819 (1989).
687. J. Wiggs and H. Jónsson, Comput. Phys. Commun. **81**, 1 (1994).
688. J. Wiggs and H. Jónsson, Comput. Phys. Commun. **87**, 319 (1995).
689. G. A. de Wijs, G. Kresse, and M. J. Gillan, Phys. Rev. B **57**, 8223 (1998).
690. G. A. de Wijs, G. Kresse, L. Vocadlo, K. Dobson, D. Alfe, M. J. Gillan, and G. D. Price, Nature (London) **392**, 805 (1998).
691. G. A. de Wijs, A. De Vita, and A. Selloni, Phys. Rev. B **57**, 10 021 (1998).
692. A. Willetts and N. C. Handy, Chem. Phys. Lett. **227**, 194 (1994).
693. M. Wilson and P. M. Madden, J. Phys.: Condens. Matter **5**, 2687 (1993).
694. E. Wimmer, Science **269**, 1397 (1995).
695. K. Wolf, W. Mikenda, E. Nusterer, K. Schwarz, and C. Ulbricht, Chem. Eur. J. **4**, 1418 (1998).
696. T. K. Woo, P. M. Margl, T. Ziegler, and P. E. Blöchl, Organometallics **16**, 3454 (1997).
697. T. K. Woo, P. M. Margl, L. Deng, L. Cavallo, and T. Ziegler, Catal. Today **50**, 479 (1999).
698. D. M. Wood and A. Zunger, Phys. Rev. A **18**, 1343 (1985).
699. K. Yamaguchi and T. Mukoyama, J. Phys. B **29**, 4059 (1996).
700. H. Yamataka, M. Aida, and M. Dupuis, Chem. Phys. Lett. **300**, 583 (1999).
701. L. Ye and H.-P. Cheng, J. Chem. Phys. **108**, 2015 (1998).
702. J.–Y. Yi, D. J. Oh, J. Bernholc, and R. Car, Chem. Phys. Lett. **174**, 461 (1990).
703. M.T. Yin and M.L. Cohen, Phys. Rev. B **26**, 3259 (1982).
704. T. Ziegler, A. Rauk, and E. J. Baerends, Theor. Chim. Acta **43**, 261 (1977).
705. C. J. Zhang, P. J. Hu, and A. Alavi, J. Am. Chem. Soc. **121**, 7931 (1999).
706. Y. Zhang and W. Yang, Phys. Rev. Lett. **80**, 890 (1998).
707. Y.–J. Zhao, J. Ming, G. Lai, and P.–L. Cao, Phys. Lett. A **255**, 361 (1999).
708. F. Zong and D. M. Ceperley, Phys. Rev. E **58**, 5123 (1998).

RELATIVISTIC ELECTRONIC-STRUCTURE CALCULATIONS FOR ATOMS AND MOLECULES

MARKUS REIHER AND BERND A. HESS
Chair of Theoretical Chemistry
Friedrich–Alexander-Universität Erlangen–Nürnberg
Egerlandstr. 3, 91058 Erlangen
Germany
E-mail: {Markus.Reiher, Hess}@chemie.uni-erlangen.de

We give an account of theoretical methods for calculations on atoms and molecules in a relativistic quantum-mechanical framework. After a short introduction into the nature of relativistic effects, we describe fully numerical methods for 4-component atomic structure calculations. The account closes with a detailed discussion of approximate relativistic methods for the description of the electronic structure of molecules.

1 Qualitative description of relativistic effects

At the beginning of the last century, Albert Einstein discovered the special and general theory of relativity which turned out to be the key to a unified description of classical mechanics and electrodynamics. It turns out that in particular in the case of fast-moving particles, non-relativistic mechanics is an approximation to relativistic physics. In the framework of non-relativistic quantum mechanics, methods have been developed to accurately describe the electronic structure of light atoms and molecules consisting of light atoms. Most organic compounds belong to this category, but even in these cases there are subtle effects requiring a relativistic treatment, which are important for the interpretation of highly accurate experiments in spectroscopy. The so-called relativistic effects begin to play a major role in heavy atoms and their compounds. This is due to the fact that the relativistic effects on energies and other physical quantities increase with the fourth power of the nuclear charge Z.

At the first glance, relativistic quantum chemical calculations are much more expensive than their non-relativistic analogues are. This is due to the fact that any relativistic theory has to consider for every particle also the degrees of freedom for its charge-conjugated particle (the positron in the case of the electrons of an atom or a molecule) on equal footing. Since there is an additional doubling of the degrees of freedom because the spin of the electron plays a dynamical role in relativistic theories and, therefore, also has to be treated explicitly, the Dirac equation, which is the proper equation of motion for spin one-half particles like electrons, consists essentially of four coupled differential equations.

The field dealing with relativistic electronic-structure theory of atoms and molecules is often called Relativistic Quantum Chemistry. It has been developing rapidly in the last few decades, and meanwhile very good reviews are available, which provide a much more detailled discussion of relativistic effects than we shall be able to give [1,2,3,4,5]. An excellent account on the literature dealing with relativistic quantum chemistry has been provided by Pyykkö and can be found in [6,7]

and also online in the WWW [8].

1.1 Direct relativistic effects

Relativistic effects in atoms and molecules may be divided into kinematical effects, which do not cause a splitting of energy levels due to the spin degrees of freedom, and into effects of spin–orbit coupling. Kinematical effects are caused by electrons moving with high velocity in the vicinity of a (heavy) nucleus. This leads to contracted orbitals and, thus, to a contracted electron density distribution in a quantum-mechanical description. Connected with this contraction is a lowering of orbital energies and of the total energy (if compared in the energy scale of the non-relativistic energies). This direct influence of relativistic kinematics is termed as *direct relativistic effects*. It is mainly important for s and $p_{1/2}$ shells, since these have appreciable amplitude in the vicinity of the nucleus. Obviously, these effects show up in physical quantities like excitation energies, ionization energies, and electron affinities, as well as in chemical quantities like electronegativity.

1.2 Indirect relativistic effects

Orbitals with higher angular momentum have a node at the nucleus and therefore are hardly directly affected by direct kinematical relativistic effects. The modified shielding of the nuclear charge by the contracted core orbitals results, however, in an expansion mainly of the d and f orbitals. These effects are called *indirect relativistic effects*.

2 Fundamentals of relativistic quantum chemistry

The time-dependent Schrödinger equation is not Lorentz invariant, which becomes immediately clear if one observes that all differential operators for spatial coordinates represent second derivatives while the time coordinate occurs as a first derivative. But spatial and time coordinates must be treated, roughly speaking, in an equivalent way by a physical theory.

In 1928, Dirac found an equation for the free motion of an electron which fulfills the invariance demands:

$$-i\frac{\partial\Psi(\mathbf{r},t)}{\partial t} = \hat{H}_D\Psi(\mathbf{r},t), \tag{1}$$

with the Dirac Hamiltonian

$$\hat{H}_D = c\boldsymbol{\alpha}\cdot\hat{\mathbf{p}} + m_0c^2\beta \quad , \tag{2}$$

and the standard definition for the momentum operator

$$\hat{\mathbf{p}} = -i\hbar\nabla \ , \tag{3}$$

where c is the speed of light, m_0 the rest mass of the electron and in the standard representation of the 4×4 matrices α and β, the latter a diagonal matrix

$$\beta = \begin{pmatrix} \mathbf{1}_2 & \mathbf{0} \\ \mathbf{0} & -\mathbf{1}_2 \end{pmatrix} \ , \tag{4}$$

with

$$\mathbf{1}_2 = \begin{pmatrix} 1\,0 \\ 0\,1 \end{pmatrix} \quad , \quad \mathbf{0} = \begin{pmatrix} 0\,0 \\ 0\,0 \end{pmatrix} \quad . \tag{5}$$

and the 3-component vector $\boldsymbol{\alpha}$= $(\boldsymbol{\alpha}_x, \boldsymbol{\alpha}_y, \boldsymbol{\alpha}_z)$ is conveniently expressed by means of Pauli's spin matrices σ_s as

$$\boldsymbol{\alpha}_s = \begin{pmatrix} \mathbf{0} & \sigma_s \\ \sigma_s & \mathbf{0} \end{pmatrix} \ , \ s = x, y, z \ , \tag{6}$$

with the Pauli matrices

$$\sigma_x = \begin{pmatrix} 0\,1 \\ 1\,0 \end{pmatrix} \ , \ \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \ , \ \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad . \tag{7}$$

Dirac's equation is essentially a set of four coupled differential equations, and the wavefunction appears to be a 4-component spinor containing four functions. Dirac found that four is the lowest dimension possible for an equation for a spin one-half particle consistent with the relativistic invariance requirement.

In the Dirac equation, the electromagnetic field is introduced by means of external potentials. This is very much akin to the method of introducing potentials into non-relativistic equations by the method of "minimal coupling". A fully relativistic theory requires, however, that the degrees of freedom are quantized as well, a procedure carried out in quantum electrodynamics, and leading to deviations from the Coulomb interactions, which is only the first term of a series in the fine-structure constant $\alpha = e^2/\hbar c$. Thus, for the case of many interacting electrons, the so-called Dirac-Coulomb operator, which features the Coulomb repulsion between the electrons in addition to a Dirac operator for each particle, does *not* satisfy the relativistic invariance requirements, i. e., invariance with respect to Lorentz transformations. Thus, relativistic electronic structure calculations with four-component wave functions and the Dirac-Coulomb operator are *not* "fully" relativistic, as often claimed, but rather correct only to first order in the fine-structure constant. The so-called Breit operator introduces the next higher order in the electron–electron interaction operator.

3 Numerical 4-component calculations for atoms

Before we discuss the methods for the calculation of electronic structure for *molecules* in greater detail, we shall make some comments on *atoms*. Obviously, the same theory which can be used for the treatment of molecules also applies for atoms. But there are some methods which can be used in particular for atoms because of their spherical symmetry.

3.1 A short history of relativistic atomic structure calculations

Some important landmarks in the history of relativistic atomic structure calculations will be given here. We refer to the literature [9,10,11] for detailed discussions on this subject.

Numerical relativistic calculations of atoms with more than one electron started in 1935: Bertha Swirles [12] transferred the Hartree-Fock formalism to the Dirac equation. Due to the lack of computers at that time, only a few calculations could be carried out. The situation changed in the 1960's when Grant used the tensor algebra introduced by Racah (cf., e. g., [13,14]) for handling the analytic integration over all angular dependent terms and derived a general expression for the total electronic energy of a closed-shell atom in the central-field approximation. From this expression he deduced self-consistent field (SCF) equations for the determination of spinors [15,16,17].

In 1967, Kim [18] expressed the SCF equations within a basis set representation of the spinors. Desclaux [19] calculated highly accurate (fully numerical) spinor energies, total energies, and other expectation values for nearly all neutral atoms of the periodic table in the Dirac-Fock (DF) approximation, i. e., for closed shells and configuration averages. He also published a program for calculations on the multi-configuration (MC)DF level in 1975 [20]. In 1980 Grant *et al.* published their MCDF code [21,22] which was later reorganized into the GRASP package [23]. The numerical methods they used [24] are similar to those applied in Desclaux's code [25]. But there were (and still are) SCF convergence problems in some cases. Parpia *et al.* extended the code to facilitate large-scale computations through the use of dynamic memory allocation and improved convergence features (GRASP92 [26]). Additionally, there has been done work on relativistic basis set calculations for atoms during the last two decades (see for example [27,28,29]). Recently developed program packages for relativistic 4-component molecular electronic structure calculations can also be used for the calculation of the electronic structure of atoms (cf., e. g., [30,31,32,33,34,35]).

3.2 Reduction to equations dependent on the radial coordinate

When we assume that the mass of the atom's nucleus is infinitely large (as compared to the electron's mass) we describe the motion of the electrons in a central field potential. The spherical symmetry of this central field potential allows us to use the following ansatz for the 4-component spinor

$$\psi_p(\mathbf{r},\sigma) = \frac{1}{r}\begin{pmatrix} P_{n_p\kappa_p}(r)\ \Omega_{\kappa_p m_p}(\vartheta,\varphi,\sigma) \\ \mathrm{i}Q_{n_p\kappa_p}(r)\ \Omega_{-\kappa_p m_p}(\vartheta,\varphi,\sigma) \end{pmatrix} , \tag{8}$$

where $\Omega_{\kappa_p m_p}(\vartheta,\varphi,\sigma)$ are 2-component spherical spinors containing the vector coupling of angular momentum and spin (κ_i is the relativistic angular momentum quantum number). With this ansatz it is possible to treat all angular and spin dependent parts analytically. What remains is the calculation of the two radial functions $P_{n_p\kappa_p}(r)$ and $Q_{n_p\kappa_p}(r)$. Note that for this task coupled first-order differential equations have to be solved, e. g., the Dirac-Fock equations [17,36]

$$\begin{pmatrix} V_i^P(r)-\epsilon_i & A_i^\dagger(r) \\ A_i(r) & V_i^Q(r)-\epsilon_i \end{pmatrix}\begin{pmatrix} P_i(r) \\ Q_i(r) \end{pmatrix} = \begin{pmatrix} X_i^P(r) \\ X_i^Q(r) \end{pmatrix} , \tag{9}$$

in case of closed-shell atoms. Here we introduced the following functions

$$A_i(r) = c\left(\frac{\mathrm{d}}{\mathrm{d}r} + \frac{\kappa_i}{r}\right) \ , \tag{10}$$

$$V_i^Q(r) = V_i^P(r) - 2c^2 \ , \tag{11}$$

$$V_i^P(r) = V_{\mathrm{nuc}}(r) + \sum_j D_j U_{jj0}^C(r) - \sum_{\nu=0}(D_i/2)\,\Gamma_{ii\nu} U_{ii\nu}^C(r) \ , \tag{12}$$

$$X_i^R(r) = \sum_{j,j\neq i}\sum_{\nu}(D_j/2)\,\Gamma_{ij\nu} U_{ij\nu}^C(r) R_j(r) \ , \tag{13}$$

$$R_j(r) = P_j(r) \text{ or } R_j(r) = Q_j(r) \ ,$$

$$\Gamma_{ij\nu} = 2\begin{pmatrix} j_i & \nu & j_j \\ 1/2 & 0 & -1/2 \end{pmatrix}^2 \ . \tag{14}$$

The $\Gamma_{ij\nu}$ are the two-electron Coulomb-interaction structure factors (coupling coefficients) originating from the analytical treatment of spin and angular momentum. The potential functions $U_{ij\nu}^C(r)$ are defined as

$$U_{kl\nu}^C(r_1) = \int_0^\infty [P_k(r_2)P_l(r_2) + Q_k(r_2)Q_l(r_2)]\frac{r_<^\nu}{r_>^{\nu+1}}\,dr_2 \ , \tag{15}$$

where $r_{<,>}$ is the minimum and maximum, resp., of $\{r_1, r_2\}$. $D_i = 2|\kappa_i|$ denotes the occupation number of the i-th shell. The sums over ν result from the analytic treatment of angular and spin parts and run from $\nu_{\min} = |j_i - j_j|$ to $\nu_{\max} = j_i + j_j$ (constraints: $j_i + j_j + \nu$ has to be even if $\mathrm{sgn}(\kappa_i) \neq \mathrm{sgn}(\kappa_j)$ and $j_i + j_j + \nu$ has to be odd if $\mathrm{sgn}(\kappa_i) = \mathrm{sgn}(\kappa_j)$).

These SCF equations contain the r variable only and are, thus, one-dimensional. This makes them accessible for numerical solution methods which work particularly well for one-dimensional equations.

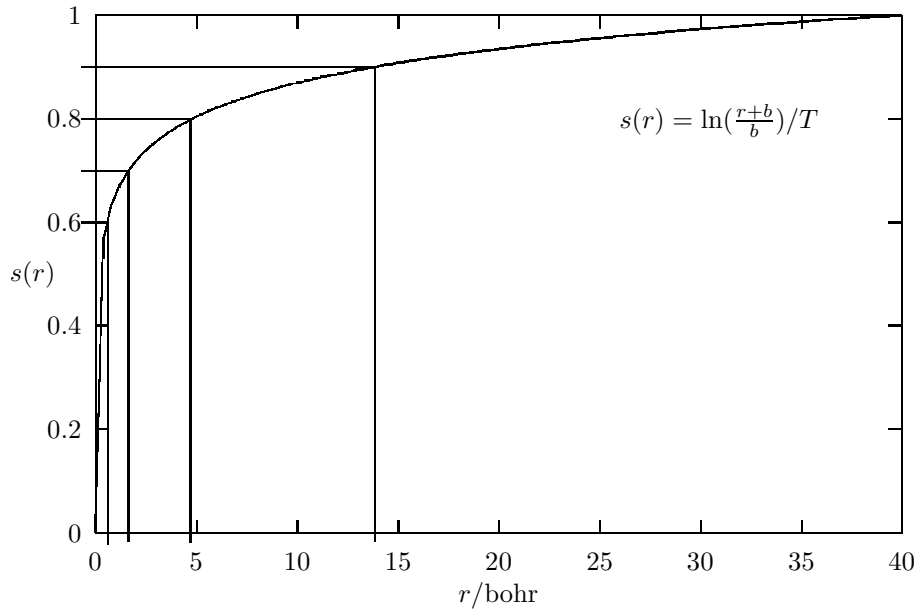
3.3 Numerical discretization and solution techniques

In atomic structure theory highly accurate calculations are possible. This high accuracy is guaranteed by the employment of fully numerical solution techniques which do not make use of basis sets such that a discussion of the size of a basis set is not necessary. Furthermore, the use of basis sets is more involved in relativistic electronic structure calculations as compared to non-relativistic analogues due to the requirement to maintain so-called kinetic balance. We shall come back to this issue later.

Finite-difference methods operating on a grid consisting of equidistant points ($\{x_i\}, x_i = ih + x_0$) are known to be the most accurate techniques available. Additionally, on an equidistant grid all discretized operators appear in a simple form. The uniform step size h allows us to use the Richardson extrapolation method [37,38] for the control of the numerical truncation error. Many methods are available for the discretization of differential equations on equidistant grids and for the integration (quadrature) of functions needed for the calculation of expectation values.

Since a discretization in the variable r is not efficient (too many grid points would lie at large distances instead near the origin where they are needed), a vari-

Figure 1. An example for a variable transformation function $r \mapsto s = s(r)$ of typical shape (logarithmic grid: b=0.001, $T = \ln[(r_{max} + b)/b]$, and r_{max}=40). The horizontal lines indicate the equidistant s-grid while the vertical lines demonstrate how the r-grid is then generated.



able transformation is introduced. In this new variable s the equidistant grid is created and all differential equations to be solved have to be transformed to this new variable s. Fig. 1 shows how such a transformation function must principally look like.

Simple discretization schemes use derivatives of Lagrangian interpolation polynomials that approximate the function known only at the grid points $\{x_i\}$. These schemes consist of tabulated numbers multiplied with the function's values at m contiguous grid points and are referred to as "m-point-formulae" by Bickley [39] (cf. [p. 914][40]). For an acceptable truncation error $O(h^t)$, $t = 4$ or higher, m is larger than t which leads to an extended amount of computation. Efficient discretization schemes are available for the discretization of the differential equation encountered in the course of a quantum chemical electronic structure calculation (cf., e. g., [41,42,43,44]).

Once a discretization scheme is chosen numerical solution methods like inverse iteration, shooting methods etc. can be applied to solve the discretized equations and obtain values for the radial functions at equidistant grid points. These are implemented in standard computer programs [23,26,45].

3.4 Description of the positive charge of the nucleus

The choice of a point-like atomic nucleus (PNC) limits the calculations to atoms with a nuclear charge number $Z \leq c$, i. e., $Z_{max} \approx 137$. The series expansion around

the origin for a non-singular electron-nucleus potential shows (see below) that this limit can be overcome using an atomic nucleus of finite size with a model function for the description of the distribution of the positive charge inside the nucleus (we will denote this choice as FNC for *finite nucleus case*). The FNC limit for the nuclear charge number Z is approximately 170 due to the one-particle energies of the electrons entering the negative continuum.

Theoretical nuclear physics does not provide a unique model function for the positive charge distribution derived from quantum chromodynamics. That is why there is a certain degree of arbitrariness in the choice of such functions.

The potential corresponding to a given model charge density distribution $\rho(u)$ is

$$-rV(r) = 4\pi \int_0^r \rho(u)u^2 du + 4\pi r \int_r^\infty \rho(u)u du \ . \tag{16}$$

Many model potentials $V(r)$ have been used but three became important in electronic structure calculations:

3.4.1 A piecewise defined model: Homogeneous charge density distribution

The uniformly or homogeneously charged sphere is a simple model for the finite size of the nucleus. The total nuclear charge $+Z$ is uniformly distributed over the nuclear volume $4/3\pi R^3$,

$$\rho_{\rm hom}(r) = \begin{cases} 3Z/(4\pi R^3) & ; r \le R \\ 0 & ; r > R \end{cases} \ , \tag{17}$$

where R denotes an empirically fixed sphere radius (the "size" of the nucleus). This charge density distribution leads to the homogeneous electron-nucleus potential

$$V_{\rm hom}(r) = \begin{cases} -\frac{Z}{2R}\left[3 - \frac{r^2}{R^2}\right] & ; r \le R \\ -Z/r & ; r > R \end{cases} \ , \tag{18}$$

provided that the charge density distribution is normalized to the total charge

$$4\pi \int_0^\infty \rho_{hom}(u)u^2 du = Z \ . \tag{19}$$

3.4.2 Continuously defined models: Gaussian and Fermi charge density distributions

The Gaussian charge density distribution

$$\rho_{\rm gauss}(r) = \rho_{\rm gauss,0} \exp\left[-r^2/R^2\right] \tag{20}$$

with $\rho_{\rm gauss,0}$ fixed through the normalization condition,

$$\rho_{\rm gauss,0} = \frac{Z}{4\pi} \frac{1}{\sqrt{\pi}R^3} \ , \tag{21}$$

leads to the following electron-nucleus potential

$$V_{\rm gauss}(r) = -\frac{Z}{r}{\rm erf}[r/R] \tag{22}$$

(erf[x] is the error function). The Fermi distribution,

$$\rho_{\rm fermi}(r) = \rho_{\rm fermi,0} f\left[1+\exp\left[(r-R)/(cR)\right]\right]^{-1} \ , \tag{23}$$

is another nuclear charge model often used in atomic structure calculations although the evaluation of the corresponding potential is quite difficult.

3.5 Drawbacks of the choice for the nuclear charge distribution

We suppose the radial functions to be analytic at the origin (the shell index i of the expansion coefficients $a_j^{(r)} = a_{j,i}^{(r)}$ and $b_j^{(r)} = b_{j,i}^{(r)}$ is skipped for the sake of brevity),

$$P_i(r) = r^{\alpha_i} \sum_{j=0}^{\infty} a_j^{(r)} r^j \ , \tag{24}$$

$$Q_i(r) = r^{\alpha_i} \sum_{j=0}^{\infty} b_j^{(r)} r^j \ , \tag{25}$$

and determine the first exponent of this series expansion α_i by solving the SCF equation for the ith shell using series expansions for the coefficient functions[a]:

$$V_i^P(r) = v_{-1}^{(r)}\, r^{-1} + v_0^{(r)} + O(r^k) \ , \tag{26}$$

$$X_i^{P,Q}(r) = x_{-1}^{(r)}\, r^{-1} + x_0^{(r)} + O(r^k) \ , \tag{27}$$

with

$$v_{-1}^{(r)} = \begin{cases} -Z \text{ in the PNC} \\ \ \ 0 \text{ in the FNCs} \end{cases} . \tag{28}$$

Here we assume (which might not be true in the general MCDF case) that the inhomogeneity vanishes at the origin, i. e., $x_{-1}^{(r)} = x_0^{(r)} = 0$. We obtain from the SCF equations for the coefficients of the $r^{\alpha_i - 1}$ term

$$(v_{-1}^{(r)}/c)\, a_0^{(r)} + (\kappa_i - \alpha_i) b_0^{(r)} = 0 \ , \tag{29}$$

$$(v_{-1}^{(r)}/c)\, b_0^{(r)} + (\kappa_i + \alpha_i) a_0^{(r)} = 0 \ , \tag{30}$$

which yields

$$\alpha_i = \sqrt{\kappa_i^2 - (v_{-1}^{(r)})^2/c^2} = \begin{cases} \sqrt{\kappa_i^2 - Z^2/c^2} & , \text{ PNC} \\ |\kappa_i| & , \text{ FNCs} \end{cases} . \tag{31}$$

For PNC the first exponent of the series expansion is **not** integral. This creates substantial drawbacks for the numerical methods used, which always require finite higher derivatives which become singular at the origin.

[a] The potential functions $U_{kl\nu}(r)$ can contribute only to the r^j terms (with $j \geq 0$).

3.6 Technical problems in relativistic electronic structure calculations for atoms

In general, technical problems usually remain in the group working on the implementation of some theory into a computer program. Since this winterschool is mainly devoted to the discussion of problems, which one gets involved into when implementing a formalism, some peculiarities are discussed in the following sections.

3.6.1 Transformation to orbitals which are regular at the origin

As was pointed out in the previous section, a singular electron-nucleus potential of Coulomb type yields non-analytic solution functions. To obtain regular functions for the PNC the operator identity

$$\frac{d}{dx} = x^{-\gamma_i} \frac{d}{dx} x^{\gamma_i} - \frac{\gamma_i}{x} \tag{32}$$

with

$$\gamma_i = |\kappa_i| - \alpha_i = |\kappa_i| - \sqrt{\kappa_i^2 - (Z/c)^2} \tag{33}$$

suggested by Biegler-König [46] can be used to replace the differential operator d/dr in the SCF equations. The power γ_i depends on the quantum number κ_i and is chosen such that it cancels out the real power α_i in the short-range series expansion and replaces it by the integral number $|\kappa_i|$. A generalized distance variable x is chosen in Eq. (32) to indicate that the identity may be introduced in the old variable r or in the new variable s.

Another possibility, which is implemented in the standard codes, is to use the Taylor expansions of the radial functions from the origin to the first (inner) grid point and to start with the discretization at this first grid point.

3.6.2 Extrapolation techniques for efficient algorithms

Finite difference methods allow the use of techniques which extrapolate to step size $h \to 0$ (i. e., the exact solution) and control the numerical truncation error [38,47,48]. This can be done for every numerically calculated quantity F if we assume an analytic behaviour of F,

$$F(h) = F(0) + Ah^t + O(h^{t+1}) \ , \tag{34}$$

where t is the order of the truncation error connected to the chosen numerical method. If F is known for three different step sizes the quantities $F(0)$ (the extrapolated result for step size zero), A, and t can be calculated. Since all numerical methods are usually employed with an order $t \geq 4$, this can be used to check this "theoretical order". Note that the extrapolated value $F(0)$ will be correct to order $t + 1$ and we gain only one or two figures in accuracy compared to the result calculated with the largest number of grid points.

Multigrid methods with control of the numerical truncation error are very useful for the solution of matrix equations since they start with a small number of grid points, use extrapolation techniques similar to Richardson's and reduce the step

size until the result is accurate enough. The method of Bulirsch and Stoer (cf. [p. 718-725][47] and [p. 288-324][49]), for instance, consists essentially of three ideas:

- The calculated values for a given step size h are analytic functions of h – which is, of course, fulfilled here.
- The analytic expression can always be approximated as a rational function, i. e., a quotient of two polynomials in h, which is the basis for the rational extrapolation [p. 104-107] [47].
- The applied numerical method must be of even order in all higher corrections in the truncation term in order to gain two orders at a time [p. 717][47].

The SCF equations are now solved for a given number of points. Then, mid-points are added to this starting grid and the equations are solved again. This process is repeated until the rational extrapolation leads to a sufficiently small degree of error. A 3-point formula [p. 914][40] without origin correction for the first derivative, which is only of second order in h, can be applied for the discretization. This is possible since the numerical error is controlled by rational extrapolation. Its advantage is that the discretization matrix in the equation to be solved in the SCF procedure is only penta-diagonal and its elements are easily computed. Additionally, the Bulirsch-Stoer method might also be used in cases where one would like to use a diagonal representation for the coefficient functions of the differential equations for reasons of numerical stability.

3.6.3 Corrections for the electron-electron interaction: Breit interaction

As mentioned above, the electron–electron interaction is known from QED only as a series expansion, which in addition depends on the gauge fixing employed for the electromagnetic field. The first relativistic correction to the two-electron Coulomb operator is the Breit operator,

$$B_\omega(1,2) = -\frac{\boldsymbol{\alpha}_1\boldsymbol{\alpha}_2\cos(\omega r_{12})}{r_{12}} + (\boldsymbol{\alpha}_1\nabla_1)(\boldsymbol{\alpha}_2\nabla_2)\frac{\cos(\omega r_{12})-1}{\omega^2 r_{12}} \ , \tag{35}$$

where $\boldsymbol{\alpha}_i$ are the standard Dirac matrices with respect to electron i. The electron-electron distance vector is denoted as $\mathbf{r}_{12}$ and its length is r_{12}. The frequency of the exchanged photon divided by the speed of light c is ω. The Breit operator may be written in its long wavelength limit as

$$B_0(1,2) = -\frac{\boldsymbol{\alpha}_1\boldsymbol{\alpha}_2}{r_{12}} - \frac{1}{2}(\boldsymbol{\alpha}_1\nabla_1)(\boldsymbol{\alpha}_2\nabla_2)r_{12} + O(\omega^2 r_{12}) \ , \tag{36}$$

where we used the Taylor series expansion for the cosine. This expression is equivalent to the original one given by Breit [50,51,52]

$$B_0(1,2) = -\frac{\boldsymbol{\alpha}_1\boldsymbol{\alpha}_2}{r_{12}} + \frac{1}{2r_{12}}\left(\boldsymbol{\alpha}_1\boldsymbol{\alpha}_2 - \frac{(\boldsymbol{\alpha}_1\mathbf{r}_{12})(\boldsymbol{\alpha}_2\mathbf{r}_{12})}{r_{12}^2}\right). \tag{37}$$

The Breit interaction becomes important for the calculation of fine structure splittings and for highly charged ions. Unfortunately, it turns out that the calculation of the matrix elements resulting from this operator may become cumbersome and time consuming even in the frequency-independent limit, $\omega \to 0$.

For historical reasons, the Breit operator was split into the Gaunt term (first term in the operator given in Eq. (37)) and the retardation term. While the Gaunt term leads to matrix elements which are comparable in complexity to those obtained for the Coulomb interaction, the retardation term turns out to be more difficult to compute [17].

In 1976, Grant and Pyper [53] gave a general formulation of the **total** Breit interaction. Unfortunately, the individual matrix elements for the frequency-independent Breit interaction turned out to be non-symmetric in the electronic radial coordinates, r_1 and r_2, while the operator in its original form is symmetric in the total electronic coordinates. Their expression for the two-electron matrix element involves integrals over the radial variable r_2 which have to be evaluated for every value of r_1 which can become very time consuming (however, there exist some tricks which reduce the effort).

The fully numerical MCDF package GRASP includes the Breit interaction only as a first-order perturbation correction (after the self-consistent field (SCF) iterations converged with the Coulomb interaction terms only). Only the Gaunt interaction has been routinely treated in numerical calculations in a self-consistent way thus far. Note that the basis set approach is not affected by these problems.

For a long time it has been thought that the Breit interaction must not be evaluated self-consistently and should be treated only as a first-order perturbation. DF calculations have shown that the self-consistent calculation of the Breit interaction does not lead to instabilities. Since in numerical DF and in MCDF calculations the negative continuum is excluded, the treatment of the Breit interaction self-consistently leads to results comparable to those from all order perturbation theory.

A reformulation of the frequency-independent Breit operator yielding relatively simple expressions for the matrix elements and an algorithm which uses the same routines as in Coulomb-only calculations has been derived [54]. Table 1 shows some results obtained with this algorithm for the light atoms helium and beryllium. Two points are important in this context. First, the Breit contribution to the total energy is relatively small and roughly speaking of the order of effects coming from choosing different electron-nucleus potentials or different values for the speed of light (the actual recommended value is $c = 137.0359895$, however, even in the recent literature the old value $c = 137.037$ is still in use). The small effect of the Breit correction implies that its fully numerical calculation is far from being trivial since very small numbers occur in iterative calculations.

3.6.4 Other relativistic corrections

The frequency-independent Breit interaction is by far not the only correction arising from quantum electrodynamics. Other corrections, like *self energy* or *vacuum polarization*, become also important. An exact calculation of electronic structure on a quantum electrodynamical basis is only possible for atoms with, say, one to three electrons (for a recent review see [56]).

The main problem for atomic structure calculations is that these correction terms cannot be easily included into the implemented framework since they cannot

Table 1. Relativistic and non-relativistic Dirac-Fock calculations on helium and beryllium. The results have been obtained with the fully numerical atomic structure program ADRIEN [55] (meaning of styles: *italics: relativistic effect*, **bold face: Breit effect**, `Typewriter: SCF effect`).

	type	this work[1]	ref. data	from
He	non-rel.		2.861679996	A
	DF-Coulomb	2.861*8133420*	2.861*813323*	B, GRASP
	Breit/pert.	2.861**7495464**	2.861**7495**	C, GRASP
	Breit/SCF	2.861**7495**455	2.861**749**0	D, basis sets
Be	non-rel.		14.573023169	A
	DF-Coulomb	14.57*58917109*	14.57*5891698*	B, GRASP
	Breit/pert.	14.575**1892247**	14.575**1895**	C, GRASP
	Breit/SCF	14.575**1892**612	14.575**1891**	D, basis sets

A J. Stiehler and J. Hinze, Universität Bielefeld, *unpublished* (1995) (numerical, point nucleus)
B K. Dyall and L. Visscher, *At. Data Nucl. Data Tables* (1997) **67**, 207 (numerical, "homogeneous" nucleus), and `http://theochem.chem.rug.nl/~luuk/FiniteNuclei/`
C F. A. Parpia *et al.*, *J. Phys. B* (1992) **25**, 1 (numerical, "homogeneous" nucleus)
D Y. Ishikawa *et al.*, *Phys. Rev. A* (1991) **43**, 3270 (basis sets, "homogeneous" nucleus, $c = 137.0370$)
[1] M. Reiher, presented at the 35th Symposium for Theoretical Chemistry, STC99, Freiberg (all calculations with 1000 grid points, "homogeneous" nucleus)

be reduced to simple radial-dependent potential functions (apart from the vacuum polarization for which the Uehling potential might be used).

3.6.5 Large CI calculations and generation of configuration state functions

While large configuration interaction (CI) calculations have become possible with the invention of the Davidson algorithm in the seventies of the last century [57,58], it has been adopted for relativistic atomic structure calculations twenty years later when Froese Fischer *et al.* introduced it into the GRASP package by Grant *et al.* [26] (for an application see, e. g., [59]).

Davidson's algorithm is designed for the calculation of a few eigenvectors of a large CI matrix by an iterative procedure. To set up this large $N \times N$ CI matrix it is necessary to construct the N-dimensional space spanned by the configuration state functions (CSFs) in jj coupling. This automatical construction of jj coupled CSFs is not trivial and routinely done in the framework of Racah algebra.

3.7 Outlook

Although atomic structure calculations have a very long history some techniques known in molecular quantum chemistry for a long time have never been implemented into existing fully numerical atomic structure programs. This is astonishing since these techniques are very promising. For example, the DIIS method by Pulay

[60,61,62] could cure many of the SCF convergence problems known in relativistic atomic structure calculations.

Another example is the recently developed Jacobi-Davidson method by van der Vorst *et al.* [63,64,65,66,67] which improves on the existing Davidson method for the determination of eigenvectors particular in CI problems.

4 Molecular Calculations

Whereas calculations on atoms are generally feasible with grid-based techniques, calculations on molecules commonly make use of expansion techniques. Very similar to non-relativistic wave functions, it is customary to expand molecular spinors in a basis of Gaussian functions. The most rigorous method for treating relativity in quantum chemical calculations then starts from the 4-component no-pair Dirac–Coulomb–Breit (DCB) Hamiltonian [68,69]. This Hamiltonian includes terms to second order in the fine-structure constant α, and is expected to be sufficiently accurate for neutral and weakly ionized states of even the heaviest atoms and molecules [70]. Calculations based on the DCB Hamiltonian may serve as reference for more approximate treatments of relativistic effects, such as perturbation theory, relativistic pseudo-potential techniques, and relativistic local density methods.

Four-component spinors obtained by solving single- or multiconfigurational Dirac–Fock equations provide a natural starting point for calculation of dynamic electron correlation effects, which are important in the quantitative determination of ground and excited states properties such as transition energies, electric and magnetic moments, or oscillator strengths. The coupled-cluster method has emerged as the most powerful tool for handling correlation effects in atomic and molecular systems (see [71] for a recent review). It includes correlation effects to high order and is size extensive, a property of particular importance for heavy systems, where relativistic effects are also most pronounced. Significant progress has been made recently in the development and application of the relativistic coupled-cluster method (RCC) to atoms. Some of these applications were done numerically [72,73] and others used basis sets, either local [74,75,76,77,78] or global [79,80,81,82,83,84].

Uzi Kaldor's group in Tel Aviv has recently developed and implemented a relativistic version of the multireference valence-universal Fock-space coupled-cluster method for atomic systems, using a discrete basis of four-component Gaussians spinors (G-spinors). The method starts from the DCB Hamiltonian and treats correlation by the coupled-cluster singles-and-doubles (CCSD) approximation, which includes single and double virtual excitations in a self-consistent manner, incorporating the effects of the Coulomb and Breit interactions to all orders in these excitations. Accurate ionization potentials, excitation energies and fine-structure splittings were obtained by this method for a variety of heavy and superheavy atomic systems [84,85,86,87,88,89,90,91,92]

Four-component correlation methods for molecules have become available only recently. They include relativistic configuration interaction (CI) [93], second order many-body perturbation theory (MBPT) [94], and relativistic coupled cluster (RCC) [95,96,97,98]. Both MBPT and RCC were used in a single-configuration version, suitable for limited classes of molecular states, with no degeneracy or near-degeneracy.

The development and implementation of accurate relativistic 4-component multireference molecular approaches is a challenging problem. We are aware of only one such method based on the relativistic 4-component multiconfigurational SCF approach [99]. Also by Kaldor's group, molecular single-reference four-component RCC code [98] has been developed.

Four-component methods usually involve complex spinors and place heavy demands on computational resources.

4.1 Two-Component Methods

While theoretically most appealing, the four-component methods are very expensive as concerns computational resources. This is, of course, because the charge-conjugated degrees of freedom are treated as dynamical variables and thus require their own basis set in the calculation. For technical reasons, the basis set for the small component tends to be even larger than the large component basis.

Since the charge-conjugated degrees of freedom are not excited at energies typical for the valence shell of neutral or mildly ionized atoms and molecules, it is desirable to integrate them out at the very beginning. This leads to a transformed Hamiltonian, operating on a two-component wave function for the electronic degree of freedom. Moreover, it is possible also for the Dirac equation to separate off spin-dependent terms rigorously [100], so that in many cases one can use a spin-averaged one-component wave function, calculated from a Hamiltonian transformed from the spin-free ("scalar relativistic") part of the Dirac equation. The transformed Hamiltonians are obtained by means of a unitary transformation that annihilates the coupling between the "electron-like" and the "positron-like" degrees of freedom. Their wave functions still have formally four components. Since, however, there is no coupling any more between the states of positive energy (the electrons) and the states of negative energy (the positrons), we now have the possibility to focus on the former and work with two-component wave functions only. While spin–orbit coupling is described in the "Dirac-like" (four component) representation by a purely algebraic structure (the Clifford algebra of the Dirac matrices), there is a "space part" of the spin–orbit coupling operator in the decoupled representation.

The decoupled representation is achieved by a unitary transformation

$$H^{\text{decoupled}} = U^\dagger D U = \begin{pmatrix} h_+ & 0 \\ 0 & h_- \end{pmatrix} \tag{38}$$

with

$$\begin{aligned} UU^\dagger &= 1 \\ U &= \begin{pmatrix} (1+X^\dagger X)^{-1/2} & -(1+X^\dagger X)^{-1/2} X^\dagger \\ X(1+XX^\dagger)^{-1/2} & (1+XX^\dagger)^{-1/2} \end{pmatrix} \end{aligned} \tag{39}$$

and D denoting a Dirac-type Hamiltonian. The operator X maintains the exact relationship between the large and the small components

$$\phi_S = X\phi_L \tag{40}$$

for any trial function for the small component ϕ_S and for the large component ϕ_L. The operator X is not known in general. If (ϕ_S, ϕ_L) is an exact eigenfunction (ψ_L, ψ_S) of the one-electron Dirac equation, it could in principle be determined by expressing the small component in terms of the large component by means of the coupled system of equations resulting from

$$\begin{aligned} c\boldsymbol{\sigma p}\,\psi_S + V\,\psi_L &= E\,\psi_L \\ c\boldsymbol{\sigma p}\,\psi_L - 2mc^2\,\psi_S + V\,\psi_S &= E\,\psi_S, \end{aligned} \tag{41}$$

using the expression for ψ_S from the lower equation

$$2mc^2\,\psi_S = \left(1 + \frac{E-V}{2mc^2}\right)^{-1} c\boldsymbol{\sigma p}\,\psi_L. \tag{42}$$

In the general case, X must fulfill the non-linear equation [101,102,103]

$$X = \frac{1}{2mc^2}\left(c\boldsymbol{\sigma p} - [X,V] - X(c\boldsymbol{\sigma p})X\right). \tag{43}$$

Obviously, the solution of this equation for X is as complex as the solution of the Dirac equation itself, and approximations have to be employed.

Since the transformed large component, now describing electron states only, should be normalized to one, the equation contains renormalization terms $(1 + X^\dagger X)^{-1/2}$ to take the change from the Dirac normalization prescription for any four-component wave function Φ

$$\begin{aligned} \langle\Phi|\Phi\rangle &= \langle\phi_L|\phi_L\rangle + \langle\phi_S|\phi_S\rangle \\ &= \langle\phi_L|\phi_L\rangle + \langle X\phi_L|X\phi_L\rangle \end{aligned} \tag{44}$$

into account. Unfortunately, closed-form solutions for Eq. (40) are known only for a restricted class of potentials [104]. A very important special case is, however, the free particle, defined by $V \equiv 0$. In this case, we find a closed-form solution

$$X^{V=0} = \left(mc^2 + \sqrt{m^2c^4 + p^2c^2}\right)^{-1} c\boldsymbol{\sigma p}. \tag{45}$$

This defines the *exact* Foldy-Wouthuysen transformation for the free particle. Note that the square root is not expanded here.

Early attempts to reduce the Dirac and Dirac–Coulomb–Breit Hamiltonian to the electronic degrees of freedom are characterized by expansions of the operators in powers of $(E-V)/mc^2$, the most popular of these methods being the Foldy–Wouthuysen transformation [105]. It has, however, been recognized early on that the resulting expressions are too singular beyond the first order of perturbation theory [106] and that, in particular, they cannot be used in a variational calculation. Variationally stable transformed Hamiltonians have appeared only recently in the literature [107,108,101]. Before introducing these methods in more detail, we shall briefly discuss the classical methods of reducing the Dirac equation to two components.

4.1.1 Elimination of the small component

The method of elimination starts from the Dirac equation in the split form Eq. (41). The expression for the small component obtained from the second of those equations is inserted into the first one, yielding

$$(V - E)\,\psi_L + \frac{1}{2mc^2}\,[\boldsymbol{\sigma p}\,\omega(\boldsymbol{r})\,\boldsymbol{\sigma p}]\,\psi_L = 0 \tag{46}$$

with

$$\omega(\boldsymbol{r}) = \left(1 - \frac{V - E}{2mc^2}\right)^{-1}. \tag{47}$$

This substitution leads to an equation for the large component only, and Eq. (47) has been used as the basis to formulate energy-dependent, non-hermitean operators [109,110,111,112,113,114,115,116]. If desired, the spin dependence can be isolated using

$$(\boldsymbol{\sigma u})(\boldsymbol{\sigma v}) = \boldsymbol{uv} + i\boldsymbol{\sigma}(\boldsymbol{u} \times \boldsymbol{v}). \tag{48}$$

The energy dependence is, however, undesirable, since orbital-dependent Hamiltonians and non-orthogonal orbitals result. The simplest way to arrive at a hermitean, energy-dependent operator is by expanding

$$\left(1 - \frac{V - E}{2mc^2}\right)^{-1} = \sum_{n=0}^{\infty} \left(\frac{V - E}{2mc^2}\right)^n. \tag{49}$$

Keeping only the lowest-order term, the non-relativistic Schrödinger equation is recovered. Low-order relativistic corrections can be extracted by keeping the next higher term and eliminating the energy dependence by means of systematic expansion in c^{-2}. This leads to the Pauli Hamiltonian

$$H_{\mathrm{Pauli}} = \frac{p^2}{2m} + V + \frac{1}{4m^2c^2}\left(\frac{-p^4}{2m} + \frac{1}{2}(\Delta V) + \boldsymbol{\sigma}(\boldsymbol{\nabla} V) \times \boldsymbol{p}\right), \tag{50}$$

where the so-called mass–velocity term $-p^4/8m^3c^2$, the Darwin term $\Delta V/4m^2c^2$, and the spin–orbit coupling term $\boldsymbol{\sigma}(\boldsymbol{\nabla} V) \times \boldsymbol{p}/4m^2c^2$ describe relativistic corrections to $O(c^{-2})$. Several problems are connected with this operator: The minus sign of the mass–velocity term yields a strongly attractive term for states with high momentum, and leads to variational collapse in unconstrained variation; the Darwin term degenerates to a highly singular Delta-Function term in the case of the potential of a point-like nucleus; the spin–orbit coupling term leads to variational collapse as well, since it is not bounded below. These problems cannot be remedied by going to higher orders [106]. In fact, the expansion in Eq. (49) is invalid for $V - E > 2mc^2$, and this condition occurs certainly in regions close to the nucleus. Operators based on simple expansions of Eq. (49) in c^{-2} are in general singular and cannot be used for variational calculations. The Pauli operator is therefore defined only for perturbation theory to lowest order. In practical calculations, its expectation values give satisfactory relativistic corrections to the energy up to the first and second transition metal row.

Making use of special features of the matrix representation of the Dirac equation, Dyall has recently worked out a Modified Elimination of the Small Component

[117,118]. His method takes the proper renormalization mentioned above into account. In particular this normalized variant of the Modified Elimination of the Small Component is free from the singularities which plague the classical elimination method. A very well-studied technique to arrive at regular expansions has been developed in the mid-eighties [101,102]. It is based on rewriting $\omega(\boldsymbol{r})$ in (47) and choosing a different expansion parameter. Writing $\omega(\boldsymbol{r})$ as

$$\omega(\boldsymbol{r}) = \frac{2mc^2}{2mc^2 - V}\left(1 + \frac{E}{2mc^2 - V}\right)^{-1} \tag{51}$$

and expanding the term in parentheses is the basis of the so-called *regular approximations*, which were developed by the Amsterdam group [119,120] to a workable method for electronic-structure calculations.

A truncation of the expansion (51) defines the Zero- and First-Order Regular Approximation (ZORA, FORA) [121]. A particular noteworthy feature of ZORA is that even in the zeroth order there is an efficient relativistic correction for the region close to the nucleus, where the main relativistic effects come from. Excellent agreement of orbital energies and other valence shell properties with the results from the Dirac equation is obtained in this zero-order approximation, in particular in the Scaled ZORA variant [122], which takes the renormalization to the transformed large component approximately into account, using

$$\frac{1}{\sqrt{1 + X^\dagger X}} \approx \frac{1}{\sqrt{1 + \langle \phi_L X^\dagger | X \phi_L \rangle}} \tag{52}$$

The analysis [123] shows that in regions of high potential the zero-order Hamiltonian reproduces relativistic energies up to an error of order $-E^2/c^2$. On the other hand, in regions where the potential is small, but the kinetic energy of the particle high, the ZORA Hamiltonian does not provide any relativistic correction.

The main disadvantage of the method is its dependence on the zero point of the electrostatic potential, i. e., gauge dependence. This occurs because the potential enters non-linearly (in the denominator of the operator for the energy), so that a constant shift of the potential does not lead to a constant shift in the energy. This deficiency can, however, be approximately remedied by suitable means [122,124].

4.1.2 Transformation to two components

An alternative to the elimination-type methods is the attempt to achieve the block diagonalization of the Dirac operator according to (38) directly. The time-honoured method is the Foldy–Wouthuysen transformation [105]. The idea is to identify "odd" and "even" operators in the split form of the Dirac equation, i. e., operators which couple the large and small component, and such which do not. Apart from the even term $(\beta - 1)$, we can identify the even operator $\mathcal{E} = V$ and the odd operator $\mathcal{O} = c\boldsymbol{\alpha}\boldsymbol{p}$, and find

$$[\mathcal{E}, \beta] = 0, \qquad \{\mathcal{O}, \beta\} = 0. \tag{53}$$

The braces denote the anticommutator $\{A, B\} = AB + BA$. We now look for a unitary matrix which removes the odd term. The Foldy–Wouthuysen transformation

uses the ansatz

$$\Phi_1 = \exp(iS_1)\Phi,$$
$$H_1 = H + i\,[S_1, H] + \ldots \tag{54}$$

The choice $S_1 = -i\beta\mathcal{O}/2m$ removes the odd term, but introduces new odd terms of higher order, which are in turn removed by iteration of the transformation : $\Phi_n = \exp(iS_n)\Phi_{n-1}$. At this point, the resulting operators are again expanded in a power series in c^{-1}. Up to second order, we obtain again the Pauli Hamiltonian, Eq. (50). While different expressions occur in higher orders, the problems with singular operators are essentially the same as in the case of the Elimination of the Small Component discussed above. Additional problems occur, since the wave functions obtained in the Foldy–Wouthuysen procedure are no longer analytic functions of c^{-1} in the neighborhood of $c^{-1} = 0$ [125,126], as is the case for the Dirac wave function [127]. This means that the non-relativistic limit is not well defined.

To obtain a valid limiting procedure for $c^{-1} \to 0$, the perturbation theory has to be formulated by considering the non-relativistic limit of the metric (essentially the normalization requirement) and that of the operator itself separately. Both for the metric and for the operator limiting procedures must be defined. This is most conveniently done by formulating the Dirac equation in terms of a scaled small component $c\psi_S$, and a regular perturbation formalism results [128,127,129,130,131,125,126]. In the more recent literature, this four-component method has been dubbed Direct Perturbation Theory. The second-order results are equivalent to the perturbative results of the Pauli operator in an infinite basis set. In contrast to the singular expansions which are traditionally employed to derive the Pauli operator, Direct Perturbation Theory gives workable and regular results also for higher orders.

Another possibility which has meanwhile proven of considerable practical value is to avoid expansion in reciprocal powers of c throughout, and rather expand in the coupling strengh $Z\alpha c\hbar$, if closed expressions cannot be obtained [107,108,132].

The Douglas–Kroll (DK) transformation defines a transformation of the external-field Dirac Hamiltonian to two-component form which leads, in contrast to the Foldy–Wouthuysen transformation, to operators which are bounded from below and can be used variationally, similar to the Regular Approximations discussed above. As in the FW transformation, it is not possible in the DK formalism to give the transformation in closed form. It is rather defined by a sequence of unitary transformations $U_0, U_1, \ldots$, the first of which is in fact a free-particle Foldy–Wouthuysen transformation defined by

$$U_0 = A(1 + \beta R), \qquad U_0^{-1} = (R\beta + 1)A, \tag{55}$$

with

$$A = \sqrt{\frac{E_p + mc^2}{2E_p}} \tag{56}$$

$$R = \frac{c\boldsymbol{\alpha}\,\boldsymbol{p}}{E_p + mc^2} \tag{57}$$

$$E_p = c\sqrt{p^2 + m^2c^2}. \tag{58}$$

Applying U_0 to D leads to

$$U_0 D U_0^{-1} = \beta E_p + \mathcal{E}_1 + \mathcal{O}_1 \equiv H_1 \tag{59}$$

with even and odd operators of first order, given by

$$\begin{aligned} \mathcal{E}_1 &= A(V + RVR)A \ , \\ \mathcal{O}_1 &= \beta A(RV - VR)A \ . \end{aligned} \tag{60}$$

The following unitary transformation – it turns out that only one more is required to decouple the upper and lower components to sufficient accuracy for chemical applications – is defined by the somewhat unusual parametrization

$$U_1 = \sqrt{1 + W_1^2} + W_1. \tag{61}$$

For any anti-hermitean operator W_1 with $W_1^\dagger = -W_1$, it is easily seen that U_1 is unitary. Performing the transformation through U_1 and expanding the square root in powers of W_1 leads to

$$\begin{aligned} U_1 H_1 U_1^{-1} = \beta E_p &- [\beta E_p, W_1] + \mathcal{E}_1 + \mathcal{O}_1 \\ &+ \frac{1}{2}\beta E_p W_1^2 + \frac{1}{2} W_1^2 \beta E_p - W_1 \beta E_p W_1 \\ &+ [W_1, \mathcal{O}_1] + [W_1, \mathcal{E}_1] + \ldots \end{aligned} \tag{62}$$

where the dots denote terms in higher than second order of W_1. The first-order odd term is now eliminated by equating

$$[\beta E_p, W_1] = \mathcal{O}_1 \tag{63}$$

and solving for W_1. We arrive at a momentum-space integral operator for W_1

$$W_1 \Phi(\boldsymbol{p}) = \int d^3p' \ W_1(\boldsymbol{p}, \boldsymbol{p'}) \Phi(\boldsymbol{p'}) \tag{64}$$

with a kernel

$$W_1(\boldsymbol{p}, \boldsymbol{p'}) = A(R - R')A' \frac{V(\boldsymbol{p}, \boldsymbol{p'})}{E_{p'} + E_p}, \tag{65}$$

where $V(\boldsymbol{p}, \boldsymbol{p'})$ denotes the Fourier transform of the external potential, and the primed quantities are to be expressed in terms of the variable $\boldsymbol{p'}$.

The final result is

$$H^{\text{decoupled}} \approx \beta E_p + \mathcal{E}_1 - \beta(W_1 E_p W_1 + \frac{1}{2}[{W_1}^2, E_p]), \tag{66}$$

where the approximation sign denotes equivalence up to second order in the external potential. Higher-order transformations may be devised by definitions similar to Eq. (61) in order to remove odd terms of higher order in a way similar to the method described above. The performance of the second-order operator was found satisfactory for chemical applications. At this point, a projection to the upper components may be made, with the result that the β matrix becomes the unit matrix, and the $\boldsymbol{\alpha}$ matrices are to be replaced by $\boldsymbol{\sigma}$ matrices.

The no-pair Hamiltonian $\hat{H}_+$ including all spin-dependent terms now operates on the two upper components only and is obtained as [133,134,5]

$$\hat{H}_+ = \sum_i E_p(i) + \sum_i V_{\rm eff}(i) + \frac{1}{2}\sum_{i\neq j} V_{\rm eff}(i,j)\ , \tag{67}$$

with

$$E_p(i) = \sqrt{p^2c^2 + m^2c^4},$$

$$V_{\rm eff}(i) = \sqrt{\frac{E_i + mc^2}{2E_i}}\,[V + (\boldsymbol{\sigma}_i\mathbf{P}_i)V(i)(\boldsymbol{\sigma}_i\mathbf{P}_i)]\,\sqrt{\frac{E_i + mc^2}{2E_i}},$$

$$\begin{aligned} V_{\rm eff}(i,j) = A_iA_j\Bigg[&\frac{1}{r_{ij}} + (\boldsymbol{\sigma}_i\mathbf{P}_i)\frac{1}{r_{ij}}(\boldsymbol{\sigma}_i\mathbf{P}_i) + (\boldsymbol{\sigma}_j\mathbf{P}_j)\frac{1}{r_{ij}}(\ \boldsymbol{\sigma}_j\mathbf{P}_j) \\ &+(\boldsymbol{\sigma}_i\mathbf{P}_i)(\boldsymbol{\sigma}_j\mathbf{P}_j)\frac{1}{r_{ij}}(\boldsymbol{\sigma}_i\mathbf{P}_i)(\boldsymbol{\sigma}_j\mathbf{P}_j) + \hat{B}_{ij}(\boldsymbol{\sigma}_i\mathbf{P}_i)(\boldsymbol{\sigma}_j\mathbf{P}_j) \\ &+(\boldsymbol{\sigma}_i\mathbf{P}_i)B_{ij}(\boldsymbol{\sigma}_j\mathbf{P}_j) + (\boldsymbol{\sigma}_j\mathbf{P}_j)B_{ij}(\boldsymbol{\sigma}_i\mathbf{P}_i) + (\boldsymbol{\sigma}_i\mathbf{P}_i)(\boldsymbol{\sigma}_j\mathbf{P}_j)B_{ij}\Bigg]A_iA_j, \end{aligned}$$

$$A_i = \sqrt{\frac{E_i + mc^2}{2E_i}},$$

$$\mathbf{P}_i = \frac{c\mathbf{p}_i}{E_i + mc^2},$$

$$\hat{B}_{ij} = -\frac{1}{2}\frac{1}{r_{ij}}\left[\boldsymbol{\sigma}_i\cdot\boldsymbol{\sigma}_j + (\ \boldsymbol{\sigma}_i\cdot\frac{\mathbf{r}_{ij}}{r_{ij}})(\boldsymbol{\sigma}_j\cdot\frac{\mathbf{r}_{ij}}{r_{ij}})\right].$$

Making repeated use of the Dirac relation

$$(\boldsymbol{\sigma}\mathbf{u})(\boldsymbol{\sigma}\mathbf{v}) = \mathbf{u}\mathbf{v} + i\boldsymbol{\sigma}(\mathbf{u}\times\mathbf{v}), \tag{68}$$

which is valid for operators $\mathbf{u}$ and $\mathbf{v}$ not containing $\boldsymbol{\sigma}$ matrices, terms linear in either one of the $\boldsymbol{\sigma}$ matrices are extracted. These terms constitute, per definition, the spin–orbit interaction part of the operator $\hat{H}_+$. There are, of course, spin-independent terms characteristic for relativistic kinematics, which constitute the above-mentioned "scalar relativistic" part of the operator, and terms with more than one $\boldsymbol{\sigma}$ matrix which contribute, e. g., to spin-spin coupling mechanisms. The spin–orbit part of the one-electron effective potential and the two-electron spin–orbit part resulting from the Coulomb interaction reduce to the spin–same–orbit interaction

$$\begin{aligned} \hat{H}_{\rm so}^{\rm same-orbit} = &\sum_i A_i i\boldsymbol{\sigma}_i(\mathbf{P}_iV(i)\times\mathbf{P}_i)A_i \\ &+\frac{1}{2}\sum_{i\neq j} A_iA_j\left[i\boldsymbol{\sigma}_i(\mathbf{P}_i\frac{1}{r_{ij}}\times\mathbf{P}_i) + i\boldsymbol{\sigma}_j(\mathbf{P}_j\frac{1}{r_{ij}}\times\mathbf{P}_j)\right]A_iA_j, \end{aligned} \tag{69}$$

and the extraction of the terms linear in $\boldsymbol{\sigma}_i$ and $\boldsymbol{\sigma}_j$ from the Breit contributions constitute the spin–other–orbit interaction [133]

$$\hat{H}_{\rm so}^{\rm other-orbit} = -\frac{1}{2}\sum_{i\neq j} A_i A_j \left[2i\boldsymbol{\sigma}_i(\mathbf{P}_i\frac{1}{r_{ij}}\times\mathbf{P}_j) + 2i\boldsymbol{\sigma}_j(\mathbf{P}_j\frac{1}{r_{ij}}\times\mathbf{P}_i)\right] A_i A_j. \quad (70)$$

Collecting terms, rearranging, and introducing explicitly the Coulomb potential of the nuclei, we obtain [5] a workable expression for a variationally stable spin–orbit operator

$$\begin{aligned}\hat{H}_{\rm so}^{+} = &\sum_i\sum_\alpha c^2 Z_\alpha \frac{A_i}{E_i+mc^2}\,\boldsymbol{\sigma}_i(\frac{\mathbf{r}_{i\alpha}}{r_{i\alpha}^3}\times\mathbf{p}_i)\,\frac{A_i}{E_i+mc^2}\\ &- c^2\sum_{i\neq j}\frac{A_i A_j}{E_i+mc^2}\,(\frac{\mathbf{r}_{ij}}{r_{ij}^3}\times\mathbf{p}_i)\cdot(\boldsymbol{\sigma}_i+2\boldsymbol{\sigma}_j)\,\frac{A_i A_j}{E_i+mc^2}. \end{aligned} \quad (71)$$

In the case of singlet ground states well separated from the rest of the spectrum, it is often convenient to use the spin–averaged approximation and treat the spin–orbit coupling operator in a second step, be it perturbatively or variationally in a spin–orbit configuration interaction procedure with two-component spinors. In most applications (see, however, [134,135]) the DK transformation of the external potential V is limited to its one-electron part while the two-electron terms are left in their Coulomb form. This leads to the most frequently used spin-averaged 1–component many-electron no-pair Hamiltonian:

$$H_+ = \sum_i E_p(i) + \sum_i V_{\rm eff}(i) + \sum_{i<j}\frac{1}{r_{ij}}, \quad (72)$$

where

$$\begin{aligned} V_{\rm eff}(i) = &-A_i[V(i) + \boldsymbol{P}_i V(i)\boldsymbol{P}_i]A_i\\ &- W_1(i)E_p(i)W_1(i) - \frac{1}{2}[(W_1(i))^2, E_p(i)]. \end{aligned} \quad (73)$$

Since the prefactors $(E_i+mc^2)^{-1}$ grow asymptotically (for $|p_i|\to\infty$, i. e., $r_i\to 0$) like $1/|\mathbf{p}_i|$, all contributions of momentum operators in the numerator (leading to the $1/r^3$ divergence in the case of the Breit–Pauli operator) are cancelled asymptotically, and only a Coulomb singularity remains. The Breit-Pauli operator may be recovered by reintroducing c explicitly, expanding $A_i(E_i+mc^2)^{-1}$ into powers of c^{-2}

$$\frac{A_i}{E_i+mc^2} = \frac{1}{2mc^2} - \frac{3p_i^2}{16m^3c^4} + \ldots \quad (74)$$

and keeping only the lowest-order term.

Douglas–Kroll-transformed Hamiltonians have been used in many quantum-chemical calculations on molecules, density-functional theory [136] including implementation of derivatives [137], and recently also for calculations of solids [138,139,140]. A numerical analysis of the energy values [108,141] and also perturbation theory [103] shows that the eigenvalues of the second-order Douglas–Kroll-transformed Hamiltonian for a single particle agrees with the results of the Dirac equation to order c^{-4}.

Note that this is the same order in which deviations in the matrix representation of the Dirac equation itself are expected [142,103].

The Douglas–Kroll transformation can be carried to higher orders, if desired [143]. In this way, arbitrary accuracy with respect to the eigenvalues of D can be achieved, and many applications of this method are reported in the literature [144,145,146,147].

The most elaborate of these methods based on the Douglas–Kroll transformation is again of the coupled-cluster type and has been published recently [148]. It possesses many of the essential features of the 4-component approach, including the description of spin–orbital and spin–spin interactions. The CdH molecule and its ions were chosen for the pilot application of two- and four-component Fock-Space RCC [148]. The calculated values were obtained in very good agreement with experiment. While the four-component method gives the best results, one- and two-component calculations include almost all the relativistic effects. Since they are much cheaper than four-component calculations, they offer a viable alternative for systems with heavy atoms.

5 Epilogue

In this account, we presented some of the recent devolpments in relativistic electronic structure theory for atoms and molecules. The field has seen a tremendous development during the last two decades. Especially approximate relativistic operators have been studied during this period. It took quite a long time until it was realized that a meaningful description of the structure of atoms and molecules containing heavy atoms can only be achieved on a relativistic basis.

References

1. K. S. Pitzer. *Acc. Chem. Res.*, **12(8)**:271–276, August 1979.
2. P. Pyykkö and J.-P. Desclaux. *Acc. Chem. Res.*, **12(8)**:276–281, August 1979.
3. P. Pyykkö. *Chem. Rev.*, **88**:563–594, 1988.
4. W. H. E. Schwarz, A. Rutkowski, and S. G. Wang. *Int. J. Quantum Chem.*, **57**:641–653, 1996.
5. B. A. Heß. *Ber. Bunsenges. Phys. Chem.*, **101**:1, 1997.
6. P. Pyykkö. *Relativistic Theory of Atoms and Molecules - A Bibliography 1916-1985*, volume **41** of *Lecture Notes in Chemistry*. Springer-Verlag, Berlin, 1986.
7. P. Pyykkö. *Relativistic Theory of Atoms and Molecules II - A Bibliography 1986-1992*, volume **60** of *Lecture Notes in Chemistry*. Springer-Verlag, Berlin, 1993.
8. P. Pyykkö. Database 'RTAM' (relativistic quantum chemistry database 1915-1998; `http://www.csc.fi/lul/rtam/rtamquery.html`), 1999.
9. A. Hibbert. *Rep. Prog. Phys.*, **38**:1217–1338, 1975.
10. W. R. Johnson and K. T. Cheng. *(in: Atomic Inner-Shell Physics)*, pages 3–30. Physics of Atoms and Molecules. Plenum Press, New York, London, 1985.
11. I. P. Grant. *Adv. At. Mol. Opt. Phys.*, **32**:169–186, 1994.
12. B. Swirles. *Proc. Roy. Soc. London A*, **152**:625–649, 1935.

13. B. R. Judd. *Operator Techniques in Atomic Spectroscopy*. Princeton Landmarks in Physics. Princeton University Press, Princeton NJ, 1998.
14. A. R. Edmonds. *Angular Momentum in Quantum Mechanics*. Princeton Landmarks in Physics. Princeton University Press, Princeton NJ, 1996.
15. I. P. Grant. *Proc. Roy. Soc. London A*, **262**:555–576, 1961.
16. I. P. Grant. *Proc. Phys. Soc.*, **86**:523–527, 1965.
17. I. P. Grant. *Adv. Phys.*, **19**:747–811, 1970.
18. Y.-K. Kim. *Phys. Rev.*, **154(1)**:17–39, February 1967.
19. J. P. Desclaux. *At. Data Nucl. Data Tables*, **12(4)**:311–406, 1973.
20. J. P. Desclaux. *Comp. Phys. Comm.*, **9**:31–45, 1975.
21. I. P. Grant, B. J. McKenzie, P. H. Norrington, D. F. Mayers, and N. C. Pyper. *Comp. Phys. Comm.*, **21**:207–231, 1980.
22. B. J. McKenzie, I. P. Grant, and P. H. Norrington. *Comp. Phys. Comm.*, **21**:233–246, 1980.
23. K. G. Dyall, I. P. Grant, C. T. Johnson, F. A. Parpia, and E. P. Plummer. *Comp. Phys. Comm.*, **55**:425–456, 1989.
24. K. G. Dyall. *GRASP - Users' Manual*. Oxford, March 1988.
25. J. P. Desclaux, D. F. Mayers, and F. O'Brien. *J. Phys. B: Atom. Molec. Phys.*, **4**:631–642, 1971.
26. F. A. Parpia, C. F. Fischer, and I. P. Grant. *Comp. Phys. Comm.*, **94**:249, 1996.
27. H. M. Quiney, I. P. Grant, and S. Wilson. *J. Phys. B: At. Mol. Phys.*, **20**:1413–1422, 1987.
28. H. M. Quiney, I. P. Grant, and S. Wilson. *J. Phys. B: At. Mol. Opt. Phys.*, **23**:L271–L278, 1990.
29. F. A. Parpia and A. K. Mohanty. *Phys. Rev. A*, **46(7)**:3735–3745, October 1992.
30. K. G. Dyall, P. R. Taylor, K. Faegri. Jr., and H. Partridge. *J. Chem. Phys.*, **95(4)**:2583–2594, August 1991.
31. O. Visser, L. Visscher, P. J. Aerts, and W. C. Nieuwpoort. *Theor. Chim. Acta*, **81**:405–416, 1992.
32. K. G. Dyall. *(in: Relativistic and Electron Correlation Effects in Molecules and Solids)*, volume **318** of *NATO ASI Series; Series B: Physics*, pages 17–58. Plenum Press, New York, 1994.
33. W. C. Nieuwpoort, P. J. C. Aerts, and L. Visscher. *(in: Relativistic and Electron Correlation Effects in Molecules and Solids)*, volume **318** of *NATO ASI Series; Series B: Physics*, pages 59–70. Plenum Press, New York, 1994.
34. H. J. Aa. Jensen, K. G. Dyall, T. Saue, and . Jr. K. Fægri. *J. Chem. Phys.*, **104(11)**:4083–4097, March 1996.
35. T. Saue, K. Fægri, T. Helgaker, and O. Gropen. *Mol. Phys.*, **91**:937–950, 1997.
36. L. Szasz. *The Electronic Structure of Atoms*. John Wiley & Sons, New York, 1992.
37. L. F. Richardson. *Phil. Trans. Roy. Soc. London A*, **210**:307–357, 1910.
38. C.-E. Fröberg. *Numerical Mathematics - Theory and Computer Applications*. The Benjamin/Cummings Publishing Company, Menlo Park, 1. edition, 1985.

39. W. G. Bickley. *Mathematical Gazette*, **25**:19–27, 1941.
40. M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions*. Dover Publications, New York, 9. edition, 1972.
41. C. Froese Fischer. *The Hartree-Fock Method for Atoms*. John Wiley & Sons, New York, **1**. edition, 1977.
42. C. Froese Fischer, T. Brage, and P. Jönsson. *Computational Atomic Structure – An MCHF Approach*. Institute of Physics Publishing, Bristol, Philadelphia, 1997.
43. D. Andrae and J. Hinze. *Int. J. Quantum Chem.*, **63**:65–91, 1997.
44. D. Andrae, M. Reiher, and J. Hinze. *Int. J. Quantum Chem.*, in press 2000.
45. J. P. Desclaux. *(in: Methods and Techniques in Computational Chemistry: METECC-94)*, volume A – Small Systems, pages 253–274. STEF, Cagliari, 1993.
46. F. Biegler-König. Fakultät für Chemie, Universität Bielefeld, unpublished.
47. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in Fortran – The Art of Scientific Computing*. Cambridge University Press, Cambridge, **2**. edition, 1992.
48. C. Brezinski and M. Redivo. Zaglia. *Extrapolation Methods – Theory and Practice*, volume **2** of *Studies in Computational Mathematics*. North-Holland, 1991.
49. P. Deuflhard and A. Hohmann. *Numerische Mathematik – Eine algorithmisch orientierte Einführung*. Walter de Gruyter, Berlin, 1991.
50. G. Breit. *Phys. Rev.*, **34 (2. Ser.)(4)**:553–573, August 1929.
51. G. Breit. *Phys. Rev.*, **36(3)**:383–397, August 1930.
52. G. Breit. *Phys. Rev.*, **39**:616–624, February 1932.
53. I. P. Grant and N. C. Pyper. *J. Phys. B: Atom. Molec. Phys.*, **9(5)**:761–774, 1976.
54. M. Reiher and J. Hinze. *J. Phys. B*, **32**:5489–5505, December 1999.
55. M. Reiher. PhD thesis, Fakultät für Chemie, Universität Bielefeld, 1998.
56. P. J. Mohr, G. Plunien, and G. Soff. *Phys. Rep.*, **293**:227–369, 1998.
57. E. R. Davidson. *J. Comp. Phys.*, **17**:87–94, 1975.
58. C. W. Murray, S. C. Racine, and E. R. Davidson. *J. Comp. Phys.*, **103**:382–389, 1992.
59. J. Bieroń, P. Jönsson, and C. Froese Fischer. *Phys. Rev. A*, **53(4)**:2181–2188, April 1996.
60. P. Pulay. *Chem. Phys. Lett.*, **73(2)**:393–398, July 19.
61. P. Pulay. *J. Comp. Chem.*, **3**:556–560, 1982.
62. T. H. Fischer and J. Almlöf. *J. Phys. Chem.*, **96**:9768–9774, 1992.
63. H. J. J. van Dam, J. H. van Lenthe, G. L. G. Sleijpen, and H. A. van der Vorst. *Int. J. Quantum Chem.*, **17**:267, 1996.
64. H. A. van der Vorst and G. L. G. Sleijpen. *A parallelizable and fast algorithm for very large generalized eigenproblems (in: Applied Parallel Computing)*, volume **1184** of *Lecture Notes in Computer Science*, pages 686–696. Springer-Verlag, Berlin, 1996.
65. H. A. Van der Vorst and T. F. Chan. *Linear System Solvers: Sparse Iterative Methods (in: Parallel Numerical Algorithms)*, volume **4** of *ICASE/LaRC In-*

terdisc. Ser. in Science and Engineering, pages 167–202. Kluwer Academic, Dordrecht, 1997.
66. G. H. Golub and H. A. van der Vorst. *Closer to the solution: Iterative linear solvers (in: The State of the Art in Numerical Analysis)*, pages 63–92. Clarendon Press, Oxford, 1997.
67. H. A. van der Vorst and G. H. Golub. *150 Years old and still alive: eigenproblems (in: The State of the Art in Numerical Analysis)*, pages 93–119. Clarendon Press, Oxford, 1997.
68. J. Sucher. *Phys. Rev. A*, **22**:348–362, 1980.
69. W. Buchmüller and K. Dietz. *Z. Phys. C*, **5**:45, 1980.
70. I. Lindgren. *Many-Body Methods in Quantum Chemistry, edited by U. Kaldor*, page 293. Springer-Verlag, 1989.
71. R. J. Bartlett. *Modern Electronic Structure Theory, edited by D. R. Yarkony*, page 1047. World Scientific, Singapore, 1995.
72. E. Lindroth. *Phys. Rev. A*, **37**:316, 1988.
73. A.-M. Mårtensson-Pendrill. *Numerical Determination of the Electronic Structure of Atoms, Diatomic and Polyatomic Molecule, edited by M. Defranceschi and J. Delhalle*, page 131. Kluwer, 1989.
74. S. A. Blundell, W. R. Johnson, Z. W. Liu, and J. Sapirstein. *Phys. Rev. A*, **39**:3768, 1989.
75. S. A. Blundell, W. R. Johnson, Z. Liu, and J. Sapirstein. *Phys. Rev. A*, **40**:2233, 1989.
76. Z. W. Liu and H. P. Kelly. *Phys. Rev. A*, **43**:3305, 1991.
77. S. A. Blundell, W. R. Johnson, and J. Sapirstein. *Phys. Rev. Lett.*, **65**:1411, 1990.
78. S. A. Blundell, W. R. Johnson, and J. Sapirstein. *Phys. Rev. A*, **43**:3407, 1991.
79. H. Sekino and R. J. Bartlett. *Int. J. Quantum Chem.*, **24**:241, 1990.
80. A.-M. Mårtensson-Pendrill, S. A. Alexander, L. Adamowicz, N. Oliphant, J. Olsen, P. Öster, H. M. Quiney, S. Salomonson, and D. Sundholm. *Phys. Rev. A*, **43**:3355, 1991.
81. E. Ilyabaev and U. Kaldor. *Chem. Phys. Lett.*, **194**:95, 1992.
82. E. Ilyabaev and U. Kaldor. *Phys. Rev. A*, **47**:137, 1993.
83. E. Ilyabaev and U. Kaldor. *J. Chem. Phys.*, **98**:7126, 1992.
84. E. Eliav, U. Kaldor, and Y. Ishikawa. *Chem. Phys. Lett.*, **222**:82, 1994.
85. E. Eliav, U. Kaldor, and Y. Ishikawa. *Phys. Rev. A*, **50**:1121, 1994.
86. E. Eliav, U. Kaldor, and Y. Ishikawa. *Phys. Rev. A*, **49**:1724, 1994.
87. E. Eliav, U. Kaldor, and Y. Ishikawa. *Int. J. Quantum Chem. Symp.*, **28**:205, 1994.
88. E. Eliav, U. Kaldor, P. Schwerdtfeger, B. A. Heß, and Y. Ishikawa. *Phys. Rev. Lett.*, **73**:3203–3206, 1994.
89. E. Eliav, U. Kaldor, and Y. Ishikawa. *Phys. Rev. Lett.*, **74**:1079, 1995.
90. E. Eliav, U. Kaldor, and Y. Ishikawa. *Phys. Rev.*, **51**:225, 1995.
91. E. Eliav, U. Kaldor, and Y. Ishikawa. *Phys. Rev. Lett.*, **77**:5350, 1996.
92. E. Eliav, U. Kaldor, Y. Ishikawa, M. Seth, and P. Pyykkö. *Phys. Rev. A*, **53**:3926–3933, 1996.

93. L. Visscher, T. Saue, W. C. Nieuwpoort, K. Fægri, and O. Gropen. *J. Chem. Phys.*, **99**:6704–6715, 1993.
94. K. G. Dyall. *Chem. Phys. Lett.*, **224**:186–194, 1994.
95. L. Visscher, K. G. Dyall, and T. J. Lee. *Int. J. Quantum Chem. Symp.*, **29**:411, 1995.
96. L. Visscher, K. G. Dyall, and T. J. Lee. *J. Chem. Phys.*, **105**:8769, 1996.
97. L. Visscher and K. G. Dyall. *J. Chem. Phys.*, **104**:9040, 1996.
98. E. Eliav and U. Kaldor. *Chem. Phys. Lett.*, **248**:405, 1996.
99. J. J .Å. Jensen, K. G. Dyall, T. Saue, and K. Fægri. Jr. *J. Chem. Phys.*, **104**:4083–4097, 1996.
100. K. G. Dyall. *J. Chem. Phys.*, **100**:2118–2127, 1994.
101. Ch. Chang, M. Pélissier, and Ph. Durand. *Phys. Scr.*, **34**:394–404, 1986.
102. J. L. Heully, I. Lindgren, E. Lindroth, S. Lundquist, and A. M. Mårtensson-Pendrill. *J. Phys. B*, **19**:2799–2815, 1986.
103. W. Kutzelnigg. *Chem. Phys.*, **224**:203–222, 1997.
104. A. G. Nikitin. *J. Phys. A*, **31**:3297–3300, 1998.
105. L. L. Foldy and S. A. Wouthuysen. *Phys. Rev.*, **78**:29–36, 1950.
106. J. D. Morrison and R. E. Moss. *Mol. Phys.*, **41**:491–507, 1980.
107. M. Douglas and N. M. Kroll. *Ann. Phys. (N.Y.)*, **82**:89–155, 1974.
108. B. A. Heß. *Phys. Rev. A*, **33**:3742–3748, 1986.
109. P. O. Löwdin. *J. Molec. Spectr.*, **14**:131–144, 1964.
110. R. D. Cowan and D. C. Griffin. *J. Opt. Soc. Am.*, **66**:1010–1014, 1976.
111. J. H. Wood and A. M. Boring. *Phys. Rev. B*, **18**:2701–2711, 1978.
112. J. C. Barthelat, M. Pélissier, and Ph. Durand. *Phys. Rev. A*, **21**:1773–1785, 1980.
113. J. Karwowski and J. Kobus. *Chem. Phys.*, **55**:361–369, 1981.
114. J. Karwowski and M. Szulkin. *J. Phys. B*, **15**:1915–1925, 1982.
115. J. Karwowski and J. Kobus. *Int. J. Quantum Chem.*, **28**:741–756, 1985.
116. J. Wood, I. P. Grant, and S. J. Wilson. *J. Phys. B*, **18**:3027–3041, 1985.
117. K. G. Dyall. *J .Chem. Phys.*, **106**:9618–9626, 1997.
118. K. G. Dyall. *J .Chem. Phys.*, **109**:4201–4208, 1998.
119. E. van Lenthe, R. van Leeuwen, E. J. Baerends, and J. G. Snijders. *New Challenges in Computational Quantum Chemistry, edited by R. Broer, P. J. C. Aerts and P. S. Bagus*, pages 93–111. 1995.
120. E. van Lenthe, R. van Leeuwen, E. J. Baerends, and J. G. Snijders. *Int. J. Quantum Chem.*, **57**:281–293, 1996.
121. E. van Lenthe, E. J. Baerends, and J. G. Snijders. *J. Chem. Phys.*, **99**:4597–4610, 1993.
122. E. van Lenthe, E. J. Baerends, and J. G. Snijders. *J. Chem. Phys.*, **101**:9783–9792, 1994.
123. R. van Leeuwen, E. van Lenthe, E. J. Baerends, and J. G. Snijders. *J. Chem. Phys.*, **101**:1271–1281, 1994.
124. C. van Wüllen. *J. Chem. Phys*, **109**:392–400, 1998.
125. W. Kutzelnigg. *Z. Phys. D*, **11**:15–28, 1989.
126. W. Kutzelnigg. *Z. Phys. D*, **15**:27–50, 1989.
127. E. C. Titchmarsh. *Proc. Roy. Soc. (London)*, **A266**:33–46, 1962.

128. G. L. Sewell. *Proc. Camb. Phil. Soc.*, **45**:631–637, 1949.
129. A. Rutkowski. *J. Phys. B*, **19**:149–158, 1986.
130. A. Rutkowski. *J. Phys. B*, **19**:3431–3442, 1986.
131. A. Rutkowski. *J. Phys. B*, **19**:3443–3455, 1986.
132. G. Jansen and B. A. Heß. *Phys. Rev. A*, **39**:6016–6017, 1989.
133. R. Samzow and B. A. Heß. *Chem. Phys. Lett.*, **184**:491–495, 1991.
134. R. Samzow, B. A. Heß, and G. Jansen. *J. Chem. Phys.*, **96**:1227–1231, 1992.
135. C-Y. Park and J. E. Almlöf. *Chem. Phys. Lett.*, **231**:269–276, 1994.
136. O. D. Häberlen and N. Rösch. *Chem. Phys. Lett.*, **199**:491–496, 1992.
137. V. A. Nasluzov and N. Rösch. *Chem. Phys.*, **210**:413–425, 1996.
138. N. J. M. Geipel and B. A. Heß. *Chem. Phys. Lett.*, **273**:62–70, 1997.
139. J. C. Boettger. *Phys. Rev. B*, **57**:8743–8746, 1998.
140. G. M. Fehrenbach and G. Schmidt. *Phys. Rev. B*, **55**:6666–6669, 1997.
141. K. Molzberger and W. H. E. Schwarz. *Theor. Chim. Acta*, **94**:213–222, 1996.
142. R. E. Stanton and S. Havriliak. *J. Chem. Phys.*, **81**:1910–1918, 1984.
143. M. Barysz, A. J. Sadlej, and J. G. Snijders. *Int. J. Quantum Chem.*, **65**:225–239, 1997.
144. U. Kaldor and B. A. Heß. *Chem. Phys. Lett.*, **230**:1–7, 1994.
145. M. Seth, P. Schwerdtfeger, M. Dolg, K. Fægri, B. A. Heß, and U. Kaldor. *Chem. Phys. Lett*, **250**:461–465, 1996.
146. C. L. Collins, K. D. Dyall, and H. F. Schaefer. III. *J. Chem. Phys.*, **102**:2024, 1995.
147. V. Kellö, A. J. Sadlej, and B. A. Heß. *J. Chem. Phys.*, **105**:1995–2003, 1996.
148. E. Eliav, U. Kaldor, and B. A. Heß. *J. Chem. Phys.*, **108**:3409–3415, 1998.

EFFECTIVE CORE POTENTIALS

MICHAEL DOLG
Institut für Physikalische und Theoretische Chemie,
Wegelerstr. 12, 53115 Bonn
Germany
E-mail: dolg@thch.uni-bonn.de

After a brief review of relativistic effects on the electronic structure of atoms and molecules the basic ideas of the relativistic ab initio effective core potential method are outlined. The underlying approximations as well as the differences between the two commonly used versions of the approach, i.e., model potentials and pseudopotentials, are discussed. The article then focusses on the adjustment of atomic shape-consistent and energy-consistent pseudopotentials, as well as on corresponding core polarization potentials. Finally, the results of some calibration calculations for the homonuclear dimers of the halogen atoms are presented.

1 Introduction

The present manuscript discusses the two branches of effective core potential (ECP) approaches, i.e., the model potential (MP) and the pseudopotential (PP) techniques. The main focus is on those ECP schemes which proved to be successful in atomic and molecular relativistic electronic structure calculations during the past decade, and moreover, due to the authors own history, the presentation is somewhat biased towards the discussion of energy-consistent ab initio pseudopotentials. It is neither intended to give a complete overview over all effective core potential approaches developed since the pioneering work of Hellmann and Gombas around 1935, nor to cover all schemes currently on the market. In particular techniques developed especially for density functional theory and/or plane wave based computational approaches have been left out. A number of reviews on effective core potentials has been published during the last three decades and the reader is referred to them for more detailed information [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20].

1.1 Relativistic effects

Accurate ab initio electronic structure calculations for systems with heavy elements require the inclusion of relativistic effects, cf., e.g., the extensive bibliographies of relativistic calculations collected by Pyykkö [21,22,23]. Although this fact is nowadays generally acknowledged and the discussion of relativistic effects begins to be included in (quantum) chemical textbooks, a very brief and incomplete outline of relativistic effects will be given here in order to make more plausible why even for systems with second row elements a relativistic effective core potential study may be more accurate than a nonrelativistic all-electron investigation. Several excellent review articles focussing on relativistic effects exist [24,25,26,27,28,29,30,31,32,33,34,35,36,37,38].

For hydrogen and hydrogen-like ions with a point nucleus of charge Z the rela-

tivistic Schrödinger equation, i.e., the Dirac equation, is analytically solvable:

$$E_{n\kappa} = \pm c^2[1 + \left(\frac{Z/c}{n - |\kappa| + \sqrt{\kappa^2 - (Z/c)^2}}\right)^2]^{-1/2} - c^2 \ . \qquad (1)$$

Here c denotes the velocity of light ($c \approx 137.0359895$ a.u.). The relativistic quantum number κ is defined in terms of the quantum numbers of orbital and total angular momentum, l and j, as

$$\kappa = \mp(j + 1/2) \qquad \text{for} \qquad j = l \pm 1/2 \ . \qquad (2)$$

It is observed that in contrast to the nonrelativistic case two sets of solutions exist, which are separated by $\approx 2c^2$. This is due to the fact that the Dirac equation is not only a wave equation valid for an electron, but rather for spin-1/2 particles as electrons and positrons. The solutions near the zero of energy are called electronic states and essentially correspond to the nonrelativistic solutions, whereas those near $-2c^2$ are called positronic states. The wavefunction turns out to be a four-component vector (four-spinor), the two upper components (upper bispinor) being large for the electronic states, the two lower ones (lower bispinor) being large for the positronic states (charge degrees of freedom in the wavefunction). Since the focus in relativistic quantum chemistry is on electrons, it is common to use the terms *large components* and *small components* for the upper and lower components, respectively. The odd and even components may be related to spin up and down, respectively, of the particle (spin degrees of freedom). A Taylor expansion of Eq. 1 shows for the electronic states that the nonrelativistic energy increases as Z^2 and the relativistic corrections to it as Z^4. However, since the prefactor of the relativistic energy contributions contains $1/c^2$ the corrections are expected to become chemically important only for heavy nuclei. The substitution of the nonrelativistic Hamiltonian by a relativistic one leads to the so-called *direct relativistic effects*, i.e., a stabilization and a contraction of the hydrogenic functions. It is further observed that not all states with the same main quantum number n are degenerate as it is the case for the nonrelativistic solutions. In particular, states with the same nl are split into two subsets for $l > 0$ (spin-orbit splitting).

The total nonrelativistic Hartree-Fock energy of the rare gas atoms He, Ne, Ar, Kr, Xe and Rn is approximately proportional to $Z^{2.37}$, the correlation corrections (as estimated from local density functional calculations including a self-interaction correction) to $Z^{1.16}$ and the relativistic corrections (as estimated from quasirelativistic Wood-Boring calculations) to $Z^{4.34}$ (Fig. 1). Focussing on the one-electron functions rather than the total energy one observes a stabilization and contraction for valence s and p shells, but a destabilization and expansion for valence d and f shells (Figs. 2, 3). Besides the direct relativistic effects causing the stabilization and contraction as well as the splitting of the p, d, f, ... shells, so-called *indirect relativistic effects* or relativistic self-consistent field effects are present. The contraction of the inner shells causes a more efficient screening of the nuclear charge for the outer shells, thus leading to a decreased effective nuclear charge and an expansion and destabilization. Direct and indirect effects act on all shells, but direct effects dominate for s and p valence shells, whereas indirect effects dominate for d and f valence shells. Relativistic effects on orbitals have direct consequences

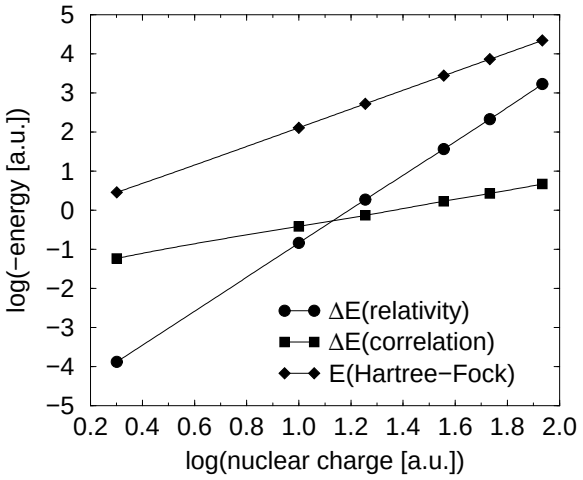


Figure 1. Total nonrelativistic Hartree-Fock energy, relativistic corrections (estimated from Wood-Boring calculations) and correlation contributions (estimated from correlation energy density functional calculations) for rare gas atoms.

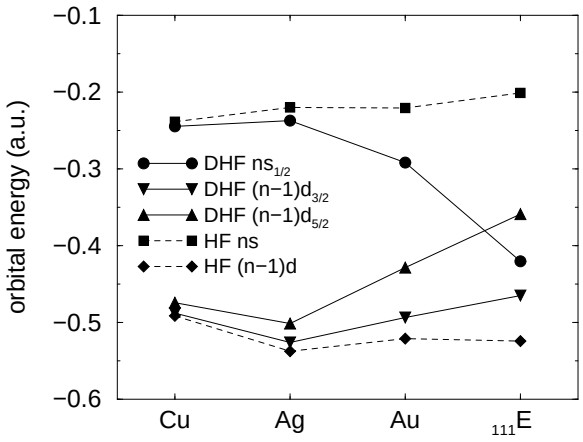


Figure 2. Nonrelativistic Hartree-Fock (HF) and relativistic Dirac-Hartree-Fock (DHF) orbital energies for the valence shells of the coinage metals (n = 4, 5, 6, 7 for Cu, Ag, Au and Eka-Au, respectively).

on quantum mechanical observables, e.g., the ionization potentials of the coinage metals are enhanced due to the relativistic stabilization of the valence s shell (Fig. 4). Clearly, since the energy and shape of valence orbitals is affected by relativistic effects, these are also important for chemical bonding. Quite often relativity leads to a bond length contraction, e.g., for the coinage metal hydrides (Fig. 5). In some rare cases, e.g., for some lanthanide or actinide systems [36], slight bond

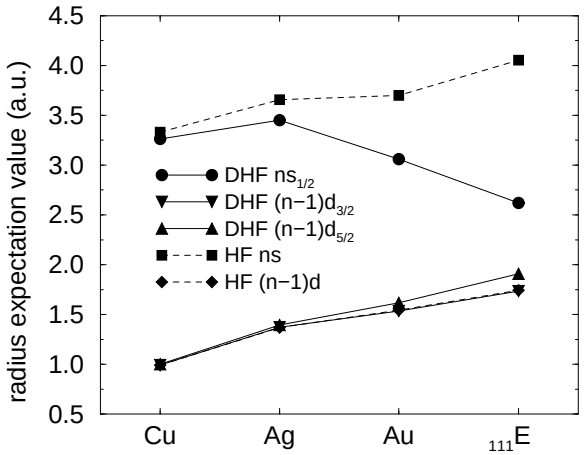


Figure 3. Nonrelativistic Hartree-Fock (HF) and relativistic Dirac-Hartree-Fock (DHF) orbital radius expectation values $< r >$ for the valence shells of the coinage metals (n = 4, 5, 6, 7 for Cu, Ag, Au and Eka-Au, respectively.

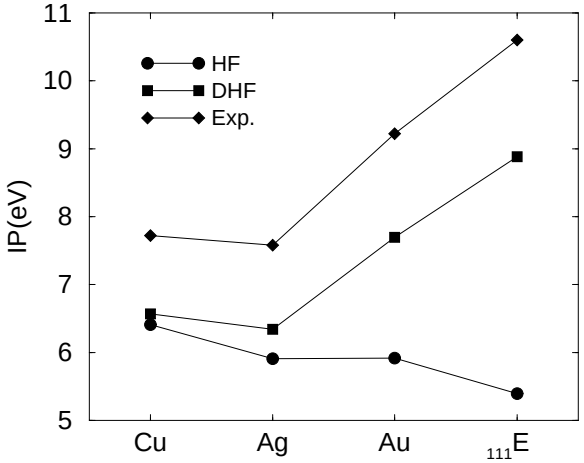


Figure 4. Nonrelativistic Hartree-Fock (HF), relativistic Dirac-Hartree-Fock (DHF) and experimental (Exp.) ionization potentials of the coinage metals. The experimental result for Eka-Au actually corresponds to the result of a high level correlated relativistic calculation (Eliav *et al.*, Phys. Rev. Lett. **73**, 3203 (1994)).

length expansions are found. One may relate the relativistic bond length changes to contractions or expansions of the valence orbitals mainly involved in bonding, but alternative explanations are also valid [27,33]. Besides bond lengths also binding energies and vibrational constants are influenced by relativistic effects. In simple cases bond stabilization or destabilization may be estimated on the basis of atomic data, e.g., for a mainly ionic A^+B^- system the relativistic effects in the ionization

potential of A and the electron affinity of B roughly determine the relativistic effect on the binding energy. Spin-orbit coupling lowers the energy of atoms with open p, d, and/or f shells. In molecules the lowering of the energy is typically much smaller due to the usually smaller number of unpaired electrons and the lower symmetry of the system. This often leads to a net destabilization of the bond by spin-orbit effects. In special cases, e.g., for the essentially van der Waals bonded dimer Hg_2, spin-orbit effects can also increase the binding energy [39].

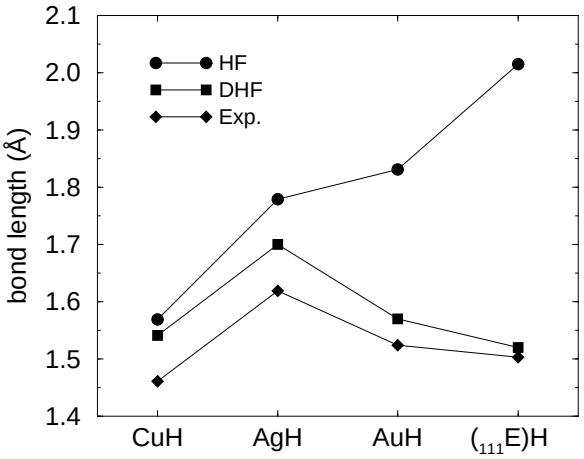


Figure 5. Nonrelativistic Hartree-Fock (HF), relativistic Dirac-Hartree-Fock (DHF) and experimental (Exp.) bond lengths of the coinage metal hydrides. The experimental result for the Eka-Au hydride actually corresponds to the result of a high level correlated relativistic calculation (Seth *et al.*, Chem. Phys. Lett. **250**, 461 (1996)).

1.2 Computational savings

More familiar than relativistic effects is to the general chemist the idea that only the valence electrons of an atom determine, at least qualitatively, its chemical behavior. The effective core potential approach is based on this experience and tries to provide a valence-only Hamiltonian which models in actual calculations for valence properties of atoms and molecules as accurately as possible the corresponding all-electron results. The main motivation to develop such schemes was initially the reduction of the computational effort, when only the chemically relevant subset of electrons is treated explicitly. Today, with a far advanced computer technology at hand and significantly improved algorithms implemented in quantum chemical program packages, the main advantage of effective core potentials is the ease with which relativistic effects can be included in the calculations.

2 All-electron Hamiltonian

Relativistic all-electron approaches are discussed here in brief for two reasons: on one hand relativistic ab initio effective core potentials are derived from (atomic) all-electron relativistic calculations, on the other hand they are often calibrated in atomic and molecular calculations against the results from all-electron relativistic calculations.

Starting point of the following considerations is a general configuration space Hamiltonian for n electrons and N nuclei, where we assume the Born-Oppenheimer approximation to hold and neglect external fields.

$$\mathcal{H} = \sum_{i}^{n} h(i) + \sum_{i<j}^{n} g(i,j) + \sum_{\lambda<\mu}^{N} \frac{Z_\lambda Z_\mu}{r_{\lambda\mu}} \ . \tag{3}$$

The indices i and j denote electrons, λ and μ nuclei. Z_λ is the charge of the nucleus λ. For the one- and two-particle operators h and g various expressions can be inserted (e.g., relativistic, quasirelativistic or nonrelativistic; all-electron or valence-only). The basic goal of quantum chemical methods is usually the approximate solution of the time-independent Schrödinger equation for a specific Hamiltonian, the system being in the state I, i.e.,

$$\mathcal{H}\Psi_I = E_I \Psi_I \ . \tag{4}$$

The most accurate electronic structure calculations nowadays applicable for atoms, molecules and also solids are based on the Dirac (D) one-particle Hamiltonian

$$h_D(i) = c\vec{\boldsymbol{\alpha}}_i \vec{p}_i + (\boldsymbol{\beta}_i - \mathbf{I}_4)c^2 + \sum_\lambda V_\lambda(r_{i\lambda}) \ , \tag{5}$$

which is correct to all orders of the fine-structure constant $\alpha = 1/c$. In these equations $\mathbf{I}_4$ denotes the 4×4 unit matrix, and $\vec{p}_i = -i\vec{\nabla}_i$ is the momentum operator for the i-th electron. $\vec{\boldsymbol{\alpha}}_i$ is a three-component vector whose elements together with $\boldsymbol{\beta}_i$ are the 4×4 Dirac matrices

$$\boldsymbol{\beta} = \begin{pmatrix} \mathbf{I}_2 & 0 \\ 0 & -\mathbf{I}_2 \end{pmatrix} \qquad \text{and} \qquad \vec{\boldsymbol{\alpha}} = \begin{pmatrix} \vec{0} & \vec{\boldsymbol{\sigma}} \\ \vec{\boldsymbol{\sigma}} & \vec{0} \end{pmatrix} \ , \tag{6}$$

which can be expressed in terms of the three-component vector of the 2×2 Pauli matrices $\vec{\boldsymbol{\sigma}}$,

$$\boldsymbol{\sigma}_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \ , \qquad \boldsymbol{\sigma}_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \ , \qquad \boldsymbol{\sigma}_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \ , \tag{7}$$

and the 2×2 unit matrix $\mathbf{I}_2$. The rest energy c^2 of the electron was subtracted from Eq. 5 in order to achieve a better compatibility to the nonrelativistic case, i.e., as in Eq. 1 the zero of energy corresponds to a free electron without kinetic energy. $V_\lambda(r_{i\lambda})$ denotes the electrostatic potential generated by the λ-th nucleus at the position of the i-th electron

$$V_\lambda(r_{i\lambda}) = -\frac{Z_\lambda}{r_{i\lambda}} \ . \tag{8}$$

In some cases a finite nucleus is used, e.g., a Gaussian-type charge distribution

$$\rho_\lambda(r) = \rho_\lambda^0 \; exp(-\eta_\lambda r^2) \qquad \text{with} \qquad 4\pi \int_0^\infty dr \; r^2 \rho_\lambda(r) = Z_\lambda \; . \tag{9}$$

The parameter η_λ can be determined from the nuclear radius R_λ, which is itself derived from the nuclear mass according to

$$\eta_\lambda = 3/(2R_\lambda^2) \qquad \text{with} \qquad R_\lambda = 2.2677 \times 10^{-5} M_\lambda^{1/3} a_0 \; . \tag{10}$$

Other charge distributions, e.g., a finite hard sphere or a Fermi-type nuclear model, are also used. The coupling of the upper and lower components of the wavefunction via $\vec{\boldsymbol{\alpha}}_i \vec{p}_i$ requires either kinetically balanced basis sets or the imposal of appropriate boundary conditions in order to avoid the so-called *finite basis set disease*.

The two-particle terms used in such calculations are either the nonrelativistic electrostatic Coulomb (C) interaction (yielding the Dirac-Coulomb (DC) Hamiltonian correct to $\mathrm{O}(\alpha^0)$)

$$g_C(i,j) = \frac{1}{r_{ij}} \; , \tag{11}$$

or in addition the magnetic Gaunt (G) interaction (yielding the Dirac-Coulomb-Gaunt (DCG) Hamiltonian correct to $\mathrm{O}(\alpha^0)$)

$$g_{CG}(i,j) = \frac{1}{r_{ij}} - \frac{\vec{\boldsymbol{\alpha}}_i \vec{\boldsymbol{\alpha}}_j}{r_{ij}} \; , \tag{12}$$

or in addition the retardation of the interaction due to the finite velocity of light, as it is accounted for in the frequency-independent Breit (B) interaction (yielding the Dirac-Coulomb-Breit (DCB) Hamiltonian correct to $\mathrm{O}(\alpha^2)$)

$$g_{CB}(i,j) = \frac{1}{r_{ij}} - \frac{1}{2r_{ij}} [\vec{\boldsymbol{\alpha}}_i \vec{\boldsymbol{\alpha}}_j + \frac{(\vec{\boldsymbol{\alpha}}_i \vec{r}_{ij})(\vec{\boldsymbol{\alpha}}_j \vec{r}_{ij})}{r_{ij}^2}] \; . \tag{13}$$

For further details the reader is referred to, e.g., a review article by Kutzelnigg [28]. The Gaunt- and Breit-interaction is often not treated variationally but rather by first-order perturbation theory after a variational treatment of the Dirac-Coulomb-Hamiltonian. The contribution of higher-order corrections such as the vaccuum polarization or self-energy of the electron can be derived from quantum electrodynamics (QED), but are usually neglected due to their negligible impact on chemical properties.

In principle problems of relativistic electronic structure calculations arise from the fact that the Dirac-Hamiltonian is not bounded from below and an energy-variation without additional precautions could lead to a *variational collapse* of the desired electronic solution into the positronic states. In addition, at the many-electron level an infinite number of unbound states with one electron in the positive and one in the negative continuuum are degenerate with the desired bound solution. A mixing-in of these unphysical states is possible without changing the energy and might lead to the so-called *continuum dissolution* or Brown-Ravenhall disease. Both problems are avoided if the Hamiltonian is, at least formally, projected onto the electronic states by means of suitable operators $\mathcal{P}_+$ (no-pair Hamiltonian):

$$\mathcal{H}_{np} = \mathcal{P}_+ \mathcal{H} \mathcal{P}_+ \; . \tag{14}$$

The Douglas-Kroll transformation [40] of the Dirac-Coulomb Hamiltonian in its implementation by Heß [41,42,43,44,45] leads to one of the currently most successful and popular forms of a relativistic no-pair Hamiltonian. The one-electron terms of the Douglas-Kroll-Heß (DKH) Hamiltonian have the form

$$h_{DKH}(i) = E_i - A_i[V(i) + \mathbf{R}_i V(i)\mathbf{R}_i]A_i - W_1(i)E_i W_1(i) - \frac{1}{2}\{W_1(i)^2, E_i\} \quad (15)$$

$$\text{with} \qquad E_i = E_{p_i} = c\sqrt{\vec{p}_i{}^2 + c^2}\ , \quad A_i = A_{p_i} = \sqrt{\frac{E_i + c^2}{2E_i}}\ , \quad \mathbf{R}_i = \frac{c\vec{\boldsymbol{\sigma}}_i\vec{p}_i}{E_i + c^2}\ ,$$

and $\{\}$ denoting an anticommutator. $W_1(i)$ is an integral operator with the kernel

$$W_1(\vec{p}, \vec{p}\,') = A_p(\mathbf{R}_p - \mathbf{R}_{p'})A_{p'}\frac{V(\vec{p}, \vec{p}\,')}{E_p + E_{p'}}\ , \quad (16)$$

where $V(\vec{p}, \vec{p}\,')$ is the Fourier transform of the external Potential $V(i)$. The two-electron terms

$$g_{DKH}(i,j) = A_i A_j[\frac{1}{r_{ij}} + \mathbf{R}_i\frac{1}{r_{ij}}\mathbf{R}_i + \mathbf{R}_j\frac{1}{r_{ij}}\mathbf{R}_j + \mathbf{R}_i\mathbf{R}_j\frac{1}{r_{ij}}\mathbf{R}_i\mathbf{R}_j]A_i A_j \quad (17)$$

increase the computational cost significantly, but have only small effects on the results and are therefore usually neglected, i.e., the unmodified Coulomb interaction is used.

A straightforward elimination of the small components from the Dirac equation leads to the two-component Wood-Boring (WB) equation [46], which exactly yields the (electronic) eigenvalues of the Dirac Hamiltonian upon iterating the energy-dependent Hamiltonian

$$h_{WB}(i) = \frac{1}{2}(\vec{\boldsymbol{\sigma}}_i\vec{p}_i)(1 + \frac{E_i - V(i)}{2c^2})^{-1}(\vec{\boldsymbol{\sigma}}_i\vec{p}_i) + \sum_\lambda V_\lambda(r_{i\lambda})\ . \quad (18)$$

Due to the energy-dependence of the Hamiltonian the Wood-Boring approach leads to nonorthogonal orbitals and has been mainly used in atomic finite difference calculations as an alternative to the more involved Dirac-Hartree-Fock calculations. The relation

$$(\vec{\boldsymbol{\sigma}}_j\vec{p}_j)f(r_j)(\vec{\boldsymbol{\sigma}}_j\vec{p}_j) = \vec{p}_j f(r_j)\vec{p}_j + i\vec{\boldsymbol{\sigma}}_j[(\vec{p}_j f(r_j)) \times \vec{p}_j] \quad (19)$$

allows the partitioning of spin-independent and spin-dependent parts and therefore the derivation of a scalar-relativistic DKH or WB Hamiltonian. This is also obtained by formally replacing $\vec{\boldsymbol{\sigma}}_i\vec{p}_i$ by $\vec{p}_i$ in Eqs. 15 and 18.

The WB approach was used to generate both model potentials as well as pseudopotentials. The DKH method was applied together with model potentials and to provide molecular all-electron results for calibration studies with valence-only schemes (cf. below).

3 Valence-only Hamiltonian

A significant reduction of the computational effort in quantum chemical investigations can be achieved by restriction of the actual calculations to the valence electron system and the implicit inclusion of the influence of the chemically inert atomic cores by means of suitable parametrized effective (core) potentials. This approach is in line with the chemists view that mainly the valence electrons of an element determine its chemical behavior, cf., e.g., the periodic table of elements. From a quantum mechanical point of view the partitioning of a many-electron system into subsystems is not possible, since electrons as elementary particles are indistinguishable. However, in the framework of effective one-particle approximations like Hartree-Fock or Dirac-Hartree-Fock theory a definition of core and valence orbitals/shells is possible either on the basis of energetic (orbital energies) or spatial (shape, radial maxima or expectation values of orbitals) arguments. If the core shells of a system are determined for one bonding situation, e.g., the free atoms, and then transferred to other bonding situations, e.g., the molecule, one speeks of the frozen-core or frozen-orbital approximation. This approach is underlying all valence-only schemes (cf., however, section 6). It is important to realize, however, that the chemists qualitative view of partitioning core and valence shells is usually not suitable for quantitative calculations, e.g., treatment of Ti ($[_{18}\mathrm{Ar}]\ 3d^2\ 4s^2\ ^3F_2$ ground state) or Ce ($[_{54}\mathrm{Xe}]\ 4f^1\ 5d^1\ 6s^2\ ^1G_4$ ground state) as a four valence electron systems leads to poor or even disastrous results [36], whereas it works very well for C ($[_2\mathrm{He}]\ 2s^2\ 2p^2\ ^3P_0$ ground state). The reason is the presence of partially occupied valence shells which have the same or even lower main quantum number as the fully occupied core shells. Although based on orbital energies the separation between core and valence shell may be reasonable, it is poor from a spatial point of view: the Ti 3d shell has its maximum density close to the one of the 3s and 3p shells, the Ce 4f shell has its maximum density even closer to the nucleus than the 5s and 5p shells. A change in the valence electron configuration in these compact valence orbitals, e.g., when looking at an excited atomic state or when forming a chemical bond, leads to too large changes of the shielding of the nuclear charge for the most diffuse core orbitals and consequently to a breakdown of the frozen core approximation. The most reliable effective core potentials have a separation of core and valence shells according to the main quantum number, e.g., 3s and 3p for Ti and 4s, 4p, 4d, 5s and 5p for Ce have to be included in the valence shell.

In effective core potential theory an effective model Hamiltonian approximation for $\mathcal{H}_{np}$ is seeked, which only acts on the states formed by the valence electrons:

$$\mathcal{H}_v = \sum_i^{n_v} h_v(i) + \sum_{i<j}^{n_v} g_v(i,j) + V_{cc} + V_{cpp}\ . \tag{20}$$

The subscripts c and v denote core and valence, respectively. h_v and g_v stand for effective one- and two-electron operators, V_{cc} represents the repulsion between all cores and nuclei of the system, and V_{cpp} is a core polarization potential (CPP). n_v

denotes the number of valence electrons treated explicitly in the calculations

$$n_v = n - \sum_{\lambda}^{N} (Z_\lambda - Q_\lambda) \ . \tag{21}$$

Here Q_λ denotes the charge of the core λ. Several choices exist for the formulation of such a valence-only model Hamiltonian, i.e., four-, two- or one-component approaches and explicit or implicit treatment of relativity. Since a reasonable compromise between accuracy and efficiency is desired, the standard effective core potential schemes use the implicit treatment of relativity (i.e., a nonrelativistic kinetic energy operator and inclusion of relativistic effects via parametrization of the effective core potential) and a one-component (scalar-quasirelativistic) or a two-component (quasirelativistic) treatment. Moreover, one may decide to keep the radial nodal structure of the (atomic) valence orbitals unchanged (model potentials, MP), or formally apply a pseudoorbital transformation to have the energetically lowest (atomic) valence orbital of each lj or l quantum number without radial nodes (pseudopotentials, PP).

Scalar-quasirelativistic and quasirelativistic effective core potentials use a formally nonrelativistic model Hamiltonian

$$h_v(i) = -\frac{1}{2}\Delta_i + V_{cv}(i) \qquad \text{and} \qquad g_v(i,j) = \frac{1}{r_{ij}} \qquad . \tag{22}$$

Relativistic contributions result only from the parametrization of the effective core potential V_{cv}, which describes the interaction of a valence electron with all nuclei and cores present in the system. The molecular pseudopotential is assumed to be a superposition of atomic pseudopotentials, with the Coulomb attraction between point charges as the leading term

$$V_{cv}(i) = \sum_{\lambda}^{N} (-\frac{Q_\lambda}{r_{\lambda i}} + \Delta V_{cv}^{\lambda}(\vec{r}_{\lambda i})) + ... \qquad . \tag{23}$$

For the interaction between nuclei and cores the point charge approximation also is the first term

$$V_{cc} = \sum_{\lambda<\mu}^{N} (\frac{Q_\lambda Q_\mu}{r_{\lambda\mu}} + \Delta V_{cc}^{\lambda\mu}(r_{\lambda\mu})) + ... \qquad . \tag{24}$$

It is hoped that a suitable parametrization of ΔV_{cv}^{λ} and $\Delta V_{cc}^{\lambda\mu}$ is able to compensate for all errors resulting from the simplifications of the original valence Hamiltonian.

3.1 Model Potentials

The most straightforward approach to come to an effective core potential is to use the Fock operator $\mathcal{F}_v$ of a valence orbital φ_a^V and to simplify the effective one particle potential

$$-\sum_{\lambda}^{N} \frac{Z_\lambda}{r_{\lambda i}} + \sum_{c} (2J_c(i) - K_c(i)) = \sum_{\lambda}^{N} (-\frac{Q_\lambda}{r_{\lambda i}} + \Delta V_{cv}^{\lambda}(\vec{r}_{\lambda i})) \ , \tag{25}$$

where the first sum is over all nuclei λ with charge Z_λ and the second over all core orbitals c. J_c and K_c denote the usual Coulomb and exchange operators. A first approximation is the assumption of non-overlapping cores, so that the second sum on the lhs can also be regarded as a superposition of one-center terms. A second approximation follows from the goal that relativistic effects should be treated implicitly. Therefore not only V_{cv} is approximated but also an additive relativistic correction term V_{rel}. In order to obtain the relevant atomic potentials $V_{rel}^\lambda + V_{cv}^\lambda$ a two- or one-component quasirelativistic atomic all-electron calculation is performed. The most widely used variant of the method are the ab initio model potentials (AIMP) of Seijo, Barandiarán and coworkers [48,49,50,51,52,53], where the quasirelativistic Hamiltonian proposed by Wood and Boring (WB) [46] for density functional calculations is used in the framework of Hartree-Fock theory according to the scheme outlined by Cowan and Griffin (CG) [47]. The WB and CG approaches correspond essentially to the use of an energy-dependent one-particle Hamiltonian, which results from the elimination of the small components from the Dirac-equation, within the Hartree-Fock scheme, disregarding any resulting non-orthogonality between orbitals of equal lj.

The AIMP method in its present form starts from a quasirelativistic all-electron Hartree-Fock calculation for the atom under consideration in a suitable electronic state and approximates the operators on the lhs of Eq. 25 for an atomic core λ as described in the following.

The long-range local Coulombic (C) part is spherical and is represented by a linear combination of Gaussians with prefactors $1/r$, i.e., a local radial model potential

$$-\frac{Z_\lambda - Q_\lambda}{r_{\lambda i}} + 2\sum_{c\in\lambda} J_c^\lambda(i) = \frac{1}{r_{\lambda i}} \sum_k C_k^\lambda e^{-\alpha_k^\lambda r_{\lambda i}^2} = \Delta V_C^\lambda(i) \ . \tag{26}$$

The exponents α_k^λ and coefficients C_k^λ are adjusted to the all-electron potential in a least-squares sense under the constraint that $\sum_k C_k^\lambda = Z_\lambda - Q_\lambda$ in order to enforce the correct asymptotic behavior of the model potential. Since the evaluation of integrals over such a local potential is not costly, any desired accuracy can be easily achieved by using a sufficiently long expansion. The nonlocal exchange (X) part is substituted by its spectral representation in the space defined by a set of functions χ_p^λ centered on core λ

$$-\sum_{c\in\lambda} K_c^\lambda(i) = \sum_{p,q} |\chi_p^\lambda(i) > A_{pq}^\lambda < \chi_q^\lambda(i)| = \Delta V_X^\lambda(i) \ . \tag{27}$$

It should be noted that this model potential operator yields the same one-center integrals as the true core exchange operator as long as the basis functions can be represented by the set of the χ_p^λ. Two- and three-center integrals are approximated. Since, in contrast to the Coulomb part, the exchange part is short ranged, a moderate number of functions χ_p^λ is needed and the one-center approximation is expected to be very good, at least for not too large cores. In practical applications the basis used in the spectral representation is chosen to be identical to the primitive functions of the valence basis set used for the atom under consideration and the A_{pq}^λ are calculated during the input processing of each AIMP calculation.

With the Coulomb and exchange parts of the model potential discussed so far the core-like solutions of the valence Fock equation still would fall below the desired valence-like solutions. In order to prevent the valence-orbitals to collapse into the core during a variational treatment and to retain a Aufbau principle for the valence electron system, the core-orbitals are shifted to higher energies by means of a shift operator

$$P^{\lambda}(i) = \sum_{c \in \lambda} (D_c^{\lambda}) |\varphi_c^{\lambda}(i) >< \varphi_c^{\lambda}(i)| \ . \tag{28}$$

Here the φ_c^{λ} denote the core orbitals localized on core λ. For practical calculations they are represented by a sufficiently large (all-electron) basis set. In principle only $D_c^{\lambda} \rightarrow \infty$ would effect a strict orthogonality between core and valence orbitals, however the more or less arbitrary choice $D_c^{\lambda} = -2\epsilon_c^{\lambda}$ is usually made due to numerical reasons. With this choice there is not strict orthogonality between core and valence orbitals, but the resulting errors are expected to be small.

The approach which has been described so far is the nonrelativistic AIMP method. It should be noted that for the derivation of the model potential

$$\Delta V_{cv,av}^{\lambda}(i) = \Delta V_C^{\lambda}(i) + \Delta V_X^{\lambda}(i) + P^{\lambda}(i) \tag{29}$$

no valence properties, e.g., valence orbitals or valence orbital energies, have been used in the nonrelativistic AIMP approach. The scalar-quasirelativistic and quasirelativistic extensions of the AIMP approach are called CG-AIMP (one-component) and WB-AIMP (two-component), respectively.

For an one-electron atom in the central field approximation one obtains from Eq. 18 the following radial equation:

$$\left(\mathcal{H}_S + \mathcal{H}_{MV} + \mathcal{H}_D + \mathcal{H}_{SO}\right) P_{n\kappa}(r) = \epsilon_{n\kappa} P_{n\kappa}(r) \ . \tag{30}$$

The nonrelativistic Schrödinger Hamiltonian

$$h_S(i) \ = \ -\frac{1}{2}\frac{d^2}{dr^2} + \frac{l(l+1)}{2r^2} + V(r) \tag{31}$$

is augmented by three energy-dependent relativistic terms, i.e., a mass-velocity (MV), a Darwin (D) and a spin-orbit (SO) term

$$\mathcal{H}_{MV} = -\frac{\alpha^2}{2}\left[\epsilon_{n\kappa} - V(r)\right]^2 \ , \qquad \mathcal{H}_D = -\frac{\alpha^2}{4}\frac{dV}{dr} B_{n\kappa}\left(\frac{d}{dr} - \frac{1}{r}\right), \tag{32}$$

$$\mathcal{H}_{SO} = -\frac{\alpha^2}{4}\frac{dV}{dr} B_{n\kappa}\frac{\kappa+1}{r}, \qquad B_{n\kappa} = \left(1 + \frac{\alpha^2}{2}\left[\epsilon_{n\kappa} - V(r)\right]\right)^{-1} \ .$$

In the many-electron case the correct nonlocal Hartree-Fock potential is used in Eq. 30, but a local approximation to it in Eqs. 32. In the CG-AIMP approach the mass-velocity and Darwin operators are cast together with the exchange terms into their spectral representation Eq. 27. The valence orbital energies $\epsilon_{n\kappa}$ are kept fixed during the extraction process and are also used for any semi-core orbitals of the same κ, which are included in the AIMP valence space. A similar strategy is followed in order to deal with the first derivative of the valence orbital in the Darwin term. It should be noted, however, that due to the use of relativistic core orbitals

and core orbital energies relativistic contributions are also present in the Coulomb and shift terms of the AIMP. The WB-AIMP method adds to this a representation of the spin-orbit operator in the form

$$\Delta V^{\lambda}_{cv,so}(i) = \sum_{l}(\sum_{k} \frac{B^{\lambda}_{lk}}{r^{2}_{\lambda i}} e^{-\beta^{\lambda}_{lk} r^{2}_{\lambda i}}) \mathcal{P}^{\lambda}_{l}(i) \vec{l}_{\lambda i} \vec{s}_{i} \mathcal{P}^{\lambda}_{l}(i) \ , \tag{33}$$

where $\vec{l}_{\lambda i} = \vec{r}_{\lambda i} \times \vec{p}_{i}$ and $\vec{s}_{i}$ denote the operators of orbital angular momentum and spin, respectively, and $\mathcal{P}^{\lambda}_{l}$ is the projection operator onto the subspace of angular quantum number l with respect to core λ. The coefficients B^{λ}_{lk} and exponents β^{λ}_{lk} are determined by means of a least-squares fit to the radial components of the Wood-Boring spin-orbit term. We note here in passing, that the Hamiltonian proposed by Cowan and Griffin is not identical to the spin-orbit averaged form of the Hamiltonian proposed by Wood and Boring [90]. The one-component Cowan-Griffin equation is identical to the Wood-Boring equation for $l = 0$, but it yields the eigenvalues of the Klein-Gordon equation (valid for a spin-0 particle) for $l > 0$. The reason is that in addition to the spin-orbit term the Darwin term was also neglected for $l > 0$ by Cowan and Griffin. The CG-AIMP approach, however, uses in fact the properly spin-averaged Wood-Boring Hamiltonian and not the Cowan-Griffin Hamiltonian. Ab initio model potential parameters and corresponding basis sets are available on the internet under `http://www.qui.uam.es/Data/AIMPLibs.html` . Since the model potential approach yields valence orbitals which have the same nodal structure as the all-electron orbitals, it is possible to combine the approach with an explicit treatment of relativistic effects in the valence shell, e.g., in the framework of the DKH no-pair Hamiltonian [54,55]. Corresponding ab initio model potential parameters are available on the internet under `http://www.thch.uni-bonn.de/tc/TCB.download.html` .

3.2 Pseudopotentials

The pseudopotential method was first developed by Hellmann [56] and Gombás [57] around 1935. The quantum mechanical foundations of the method have been investigated later by Fényes and Szépfalusy in the framework of Hartree and Hartree-Fock theory, respectively. The approach became more popular after the work of Preuss [58] for molecules and Phillips and Kleinman [59] for solids. Many of the approximations underlying the method were discussed extensively in the literature, e.g., cf. papers of Weeks *et al.*[1] and Dixon and Robertson [3]. However, since the modern pseudopotentials used today have little in common with the formulas one obtains by a strict derivation of the theory, only a rough derivation in the framework of nonrelativistic Hartree-Fock theory is presented in the following.

The space of orthonormal orbitals of a system with a single valence electron outside a closed shell core may be partitioned into a subspace for the doubly occupied core orbitals φ_c and a subspace for the singly occupied valence orbital φ_v. For the moment the space of the unoccupied virtual orbitals is not considered. The Fock equation for the valence orbital φ_v

$$\mathcal{F}_{v}\varphi_{v} = \epsilon_{v}\varphi_{v} + \sum_{c \neq v} \epsilon_{vc}\varphi_{c} \tag{34}$$

($\mathcal{F}_v$ denotes the Fock operator) can be transformed by application of $1 - P_c$ from the left into a pseudo eigenvalue equation

$$(1 - P_c)\mathcal{F}_v\varphi_v = \epsilon_v\varphi_v \ , \tag{35}$$

with the projector P_c on the subspace of the core orbitals

$$P_c = \sum_c |\varphi_c >< \varphi_c| \ . \tag{36}$$

Reductions in the basis set used to represent the valence orbital φ_v can be only achieved if by admixture of core orbitals φ_c the radial nodes are eliminated and the shape of the resulting pseudo (p) valence orbital φ_p is as smooth as possible in the core region (pseudoorbital transformation)

$$\varphi_p = N_p(\varphi_v + \sum_{c \neq v} \omega_c\varphi_c) \ . \tag{37}$$

N_p denotes a normalization factor depending on the coefficients ω_c. The original valence orbital with the full nodal structure in terms of the pseudo valence orbital with the simplified nodal structure

$$\varphi_v = (N_p)^{-1}(1 - P_c)\varphi_p \tag{38}$$

may be inserted into the pseudo eigenvalue problem Eq. 35

$$(1 - P_c)\mathcal{F}_v(1 - P_c)\varphi_p = \epsilon_v(1 - P_c)\varphi_p \ . \tag{39}$$

Using the so-called generalized Phillips-Kleinman pseudopotential [1]

$$V^{GPK} = -P_c\mathcal{F}_v - \mathcal{F}_vP_c + P_c\mathcal{F}_vP_c + \epsilon_vP_c \tag{40}$$

one recovers again a pseudo eigenvalue problem for the pseudo valence orbital

$$(\mathcal{F}_v + V^{GPK})\varphi_p = \epsilon_v\varphi_p \ . \tag{41}$$

If one assumes the core orbitals φ_c to be also eigenfunctions of the Fock operator $\mathcal{F}_v$, i.e., $[\mathcal{F}_v, P_c] = 0$, and uses the idempotency of the projection operator $P_c = P_c^n$ ($n \leq 1$), one recovers a simplified pseudo eigenvalue problem

$$(\mathcal{F}_v + V^{PK})\varphi_p = \epsilon_v\varphi_p \tag{42}$$

containing the so-called Phillips-Kleinman pseudopotential [59]

$$V^{PK} = \sum_{c \neq v}(\epsilon_v - \epsilon_c)|\varphi_c >< \varphi_c| \ . \tag{43}$$

The transition from a single valence electron to n_v valence electron requires formally in Eq. 39 the following substitutions:

$$\mathcal{F}_v \ \longmapsto \ \sum_i^{n_v} \mathcal{F}_v(i) + \sum_{i<j}^{n_v} g(i,j) \ , \qquad \epsilon_v \ \longmapsto \ E^v \ , \tag{44}$$

$$(1 - P_c) \ \longmapsto \ \prod_i^{n_v}(1 - P_c(i)) \ , \qquad \varphi_p \ \longmapsto \ \Psi^p \ .$$

The sum of effective one-particle operators $\mathcal{F}_v(i)$ has to be augmented by the interelectronic interaction terms $g(i,j)$ between the valence electrons. The valence model Hamiltonian $\mathcal{H}_v$ then reads

$$\mathcal{H}_v = \prod_k^{n_v}(1-P_c(k))[\sum_i^{n_v}\mathcal{F}_v(i) + \sum_{i<j}^{n_v} g(i,j)]\prod_l^{n_v}(1-P_c(l)) \qquad (45)$$
$$+ E_v[\sum_i^{n_v} P_c(i) - \sum_{i<j}^{n_v} P_c(i)P_c(j) - + \ ...] \ + \ V_{cc} \ .$$

This form of valence model Hamiltonian is essentially useless for practical calculations, since it contains complicated many-electron operators due to the introduction of products of projection operators. In addition the use of such a Hamiltonian would not bring about any computational savings with respect to an all-electron treatment, since the derivation given so far essentially consists of a rewriting of the Fock equation for a valence orbital in a different form. Reductions in the computational effort can be only achieved by elimination of the core electron system and simulation of its influence on the valence electrons by introducing a suitable model Hamiltonian:

$$h_v(i) \ = \ [(1-P_c(i))\mathcal{F}_v(i)(1-P_c(i)) + E_v P_c(i)] \ \longmapsto \ -\frac{1}{2}\Delta_i + V_{cv}(i) \ , \qquad (46)$$

$$g_v(i,j) \ = \ [(1-P_c(i))(1-P_c(j))g(i,j)(1-P_c(i))(1-P_c(j))] \ \longmapsto \ \frac{1}{r_{ij}} \ .$$

4 Analytical form of pseudopotentials

The simplest and historically the first choice is the local ansatz for ΔV_{cv}^{λ} in Eq. 22, however, such an ansatz is too inaccurate and therefore has soon been replaced by a so-called semilocal form. In case of quasirelativistic pseudopotentials, i.e., when spin-orbit coupling is included, the semilocal ansatz in two-component form may be written as

$$\Delta V_{cv}^{\lambda}(\vec{r}_{\lambda i}) = \sum_{l=0}^{L-1}\sum_{j=|l-1/2|}^{l+1/2}(V_{lj}^{\lambda}(r_{\lambda i}) - V_L^{\lambda}(r_{\lambda i}))P_{lj}^{\lambda}(i) + V_L^{\lambda}(r_{\lambda i}) \ . \qquad (47)$$

P_{lj}^{λ} denotes a projection operator on spinor spherical harmonics centered at the core λ

$$P_{lj}^{\lambda}(i) = P_{l,l\pm1/2}^{\lambda}(i) = P_{\kappa}^{\lambda}(i) = \sum_{m_j=-j}^{j} \mid \lambda l j m_j(i)\rangle\langle\lambda l j m_j(i) \mid \ . \qquad (48)$$

For scalar-quasirelativistic calculations, i.e., when spin-orbit coupling is neglected, a one-component form may be obtained by averaging over the spin

$$\Delta V_{cv,av}^{\lambda}(\vec{r}_{\lambda i}) = \sum_{l=0}^{L-1}(V_l^{\lambda}(r_{\lambda i}) - V_L^{\lambda}(r_{\lambda i}))P_l^{\lambda}(i) + V_L^{\lambda}(r_{\lambda i}) \ . \qquad (49)$$

The projection operator P_l^λ refers now to the spherical harmonics centered at the core λ

$$P_l^\lambda(i) = \sum_{m_l=-l}^{l} \mid \lambda l m_l(i) >< \lambda l m_l(i) \mid \ . \tag{50}$$

A spin-orbit operator may be defined

$$\Delta V_{cv,so}^\lambda(\vec{r}_{\lambda i}) = \sum_{l=1}^{L-1} \frac{\Delta V_l^\lambda(r_{\lambda i})}{2l+1}[l P_{l,l+1/2}^\lambda(i) - (l+1)P_{l,l-1/2}^\lambda(i)] \tag{51}$$

which contains essentially the difference between the two-component pseudopotentials

$$\Delta V_l^\lambda(r_{\lambda i}) = V_{l,l+1/2}^\lambda(r_{\lambda i}) - V_{l,l-1/2}^\lambda(r_{\lambda i}) \ . \tag{52}$$

For practical calculations it is advantageous to separate space and spin

$$\Delta V_{cv,so}^\lambda(\vec{r}_{\lambda i}) = \sum_{l=1}^{L-1} \frac{2\Delta V_l^\lambda(r_{\lambda i})}{2l+1} P_l^\lambda(i) \vec{l}_{\lambda i} \vec{s}_i P_l^\lambda(i) \ . \tag{53}$$

The potentials V_{lj}^λ and V_l^λ ($l = 0$ to $l = L$) respectively ΔV_l^λ ($l = 1$ to $l = L$ -1) are represented as a linear combination of Gaussians multiplied by powers of the electron-core distance:

$$V_{lj}^\lambda(r_{\lambda i}) = \sum_k A_{ljk}^\lambda \ r_{\lambda i}^{n_{ljk}^\lambda} \ exp(-a_{ljk}^\lambda r_{\lambda i}^2) \ , \tag{54}$$

$$[\Delta]V_l^\lambda(r_{\lambda i}) = \sum_k [\Delta]A_{lk}^\lambda \ r_{\lambda i}^{n_{lk}^\lambda} \ exp(-a_{lk}^\lambda r_{\lambda i}^2) \ . \tag{55}$$

The necessary one-electron integrals over cartesian Gaussians have been presented, e.g., by McMurchie and Davidson [60] or by Pitzer and Winter [61,62]. Alternatively, making use of the operator identity

$$\sum_{m_l=-l}^{+l} |lm_l > V_l(r) < lm_l| = \sum_{i,j} |\chi_i^{(l)} > A_{ij}^{(l)} < \chi_j^{(l)}| \ . \tag{56}$$

a nonlocal representation in a (nearly) complete auxiliary basis set can be used instead [63]. Once the constants $A_{ij}^{(l)}$ have been determined the integral evaluation is reduced to overlap integrals (between the auxiliary basis and the actual molecular basis sets) and therefore the derivatives with respect to the nuclear coordinates needed in geometry optimizations become much easier to evaluate.

In case of large cores a correction to the point charge repulsion model in Eq. 24 is needed. A Born-Mayer type ansatz proved to be quite successful

$$\Delta V_{cc}^{\lambda\mu}(r_{\lambda\mu}) = B_{\lambda\mu} exp(-b_{\lambda\mu} r_{\lambda\mu}) \ . \tag{57}$$

For a core-nucleus repulsion the parameters $B_{\lambda\mu}$ and $b_{\lambda\mu}$ can be obtained directly from the electrostatic potential of the atomic core electron system, for a core-core repulsion the deviation from the point charge model has to be determined

by Hartree-Fock or Dirac-Hartree-Fock calculations for the interaction between the frozen cores.

Relativistic pseudopotentials to be used in four-component Dirac-Hartree-Fock calculations can also be successfully generated and used [64], however, the advantage of obtaining accurate results at a low computational cost is certainly lost within this scheme.

5 Adjustment of pseudopotentials

5.1 Shape-consistent pseudopotentials

The origin of shape-consistent pseudopotentials [65,66] lies in the insight that the admixture of only core orbitals to valence orbitals in order to remove the radial nodes leads to too contracted pseudo valence orbitals and finally as a consequence to poor molecular results, e.g., to too short bond distances. It has been recognized about 20 years ago that it is indispensable to have the same shape of the pseudo valence orbital and the original valence orbital in the spatial valence region, where chemical bonding occurs. Formally this requires also an admixture of virtual orbitals in Eq. 37. Since these are usually not obtained in finite difference atomic calculations, another approach was developed. Starting point is an atomic all-electron calculation at the nonrelativistic, scalar-relativistic or quasirelativistic Hartree-Fock or the Dirac-Hartree-Fock level. In the latter case the small components are discarded and the large components of the energetically lowest valence shell of each quantum number lj are considered as valence orbitals after renormalization. To generate the pseudo valence orbitals $\varphi_{p,lj}$ the original valence orbitals $\varphi_{v,lj}$ are kept unchanged outside a certain matching radius r_c separating the spatial core and valence regions (shape-consistency; exactly achieved only for the reference state), whereas inside the matching radius the nodal structure is discarded and replaced by a smooth and in the interval [0,r_c] nodeless polynomial expansion $f_{lj}(r)$:

$$\varphi_{v,lj}(r) \rightarrow \varphi_{p,lj}(r) = \begin{cases} \varphi_{v,lj}(r) \text{ for } r \geq r_c \\ f_{lj}(r) \text{ for } r < r_c \end{cases} \tag{58}$$

The free parameters in f_{lj} are determined by normalization and continuity conditions, e.g., matching of f_{lj} and $\varphi_{v,lj}$ as well as their derivatives at r_c. The choice of r_c as well as the choice of f_{lj} is in certain limits arbitrary and a matter of experience.

Having a nodeless and smooth pseudo valence orbital $\varphi_{p,lj}$ and the corresponding orbital energy $\epsilon_{v,lj}$ at hand, the corresponding radial Fock equation

$$(-\frac{1}{2}\frac{d^2}{dr^2} + \frac{l(l+1)}{2r^2} + V_{lj}^{PP}(r) + W_{p,lj}[\{\varphi_{p',l'j'}\}])\varphi_{p,lj}(r) = \epsilon_{v,lj}\varphi_{p,lj}(r) \tag{59}$$

can be solved pointwise for the unknown pseudopotential V_{lj}^{PP} for each combination lj of interest. In Eq. 59 the term $W_{p,lj}$ stands for an effective valence Coulomb and exchange potential for $\varphi_{p,lj}$. Relativistic effects enter the potentials implicitly via the value of the orbital energy $\epsilon_{v,lj}$ and the shape of the pseudo valence orbital outside the matching radius. The resulting potentials V_{lj}^{PP} are tabulated on a grid

and are usually fitted to a linear combination of Gaussian functions according to

$$V^{PP} = -\frac{Q}{r} + \sum_{lj}(\sum_{k} A_{lj,k} r^{n_{lj,k}-2} e^{-\alpha_{lj,k} r^2})\mathcal{P}_{lj} \ . \tag{60}$$

Shape-consistent pseudopotentials including spin-orbit operators based on Dirac-Hartree-Fock calculations using the Dirac-Coulomb Hamiltonian have been generated by Christiansen, Ermler and coworkers [67,68,69,70,71,72,73,74,75,76]. The potentials and corresponding valence basis sets are available on the internet under `http://www.clarkson.edu/~pac/reps.html` . A similar set for main group and transition elements based on scalar-relativistic Cowan-Griffin all-electron calculations was published by Hay and Wadt [77,78,79,80,81]. Another almost complete set of pseudopotentials has been published by Stevens and coworkers [82,83,84].

5.2 Energy-consistent pseudopotentials

Energy-consistent ab initio pseudopotentials developed from energy-adjusted semiempirical pseudopotentials, i.e., potentials which were fitted to reproduce the experimental atomic spectrum. Due to the problems to account accurately for valence correlation effects, such semiempirical energy-adjustment could only be performed successfully for one-valence electron systems. The results for alkaline and alkaline-earth systems were quite good, however, due to the limited validity of the frozen-core approximation when going from a highly charged one-valence electron ion to a neutral atom or nearly neutral ion, it essentially failed for other elements, especially transition metals. However, the idea to fit exclusively to quantum mechanical observables like total valence energies (note that these may be written as sums of ionization potentials and excitation energies) instead of to rely on quantities like orbitals and orbital energies only meaningful in an approximate one-particle picture is very appealing and the approach regained attention in the ab initio framework [85].

In the most recent version of the energy-consistent pseudopotential approach the reference data is derived from finite-difference all-electron multi-configuration Dirac-Hartree-Fock calculations based on the Dirac-Coulomb or Dirac-Coulomb-Breit Hamiltonian [86]. These calculations are performed for a multitude of electronic configurations/states/levels I of the neutral atom and the low-charged ions. The total valence energies E_I^{AE} derived from these calculations define the pseudopotential parameters for a given ansatz in a least-squares sense. A corresponding set of finite-difference valence-only calculations (especially the same coupling scheme and correlation treatment has to be applied) is performed to generate the total valence energies E_I^{PP}, and the parameters are varied in such a way that the sum of weighted squared errors in the total valence energies becomes a minimum, i.e.,

$$\sum_{I}(w_I[E_I^{PP} - E_I^{AE}]^2) := min \ . \tag{61}$$

In principle this formalism can be used to generate one-, two- and also four-component pseudopotentials at any desired level of relativity (nonrelativistic Schrödinger, or relativistic Wood-Boring, Douglas-Kroll-Hess, Dirac-Coulomb or

Dirac-Coulomb-Breit Hamiltonian; implicit or explicit treatment of relativity in the valence shell) and electron correlation (single- or multi-configurational wavefunctions, e.g., the use of an intermediate coupling scheme is possible). The pseudo valence orbitals usually agree very well with the all-electron orbitals in the valence region, cf., e.g., Fig. 6. Parameters of energy-consistent ab initio pseudopo-

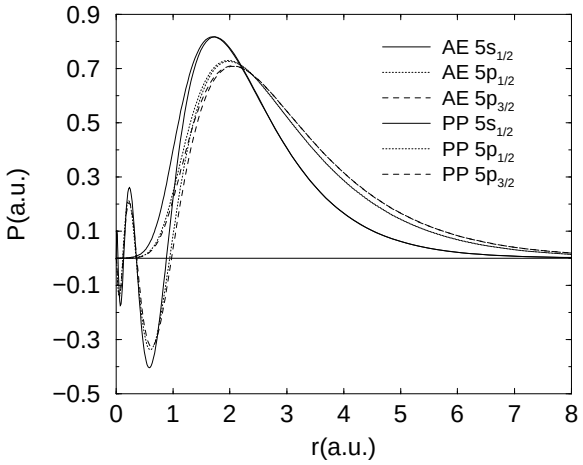


Figure 6. Valence spinors of the iodine atom in the $[_{46}\mathrm{Pd}]$ $5s^2$ $5p^5$ ground state configuration from average-level all-electron (AE) multiconfiguration Dirac-Hartree-Fock calculations and corresponding valence-only calculations using a relativistic energy-consistent 7-valence-electron pseudopotential (PP).

tentials and corresponding valence basis sets have been presented for almost all elements of the periodic table by Dolg, Preuss, Schwerdtfeger, Stoll and coworkers [85,86,87,88,89,90,91,92,93,94,95,96,97]. They are also available on the internet under `http://www.theochem.uni-stuttgart.de` . Since the functional form of energy-consistent pseudopotentials is identical to the one of shape-consistent pseudopotentials, both types of pseudopotentials can be used in standard quantum chemical program packages (COLUMBUS, GAUSSIAN, GAMESS, MOLPRO, TURBOMOLE, ...) as well as polymer or solid state codes using Gaussian basis sets (CRYSTAL, WANNIER, ...).

5.3 Limitations of accuracy

Effective core potentials are usually derived for atomic systems at the finite difference level and used in subsequent molecular calculations using finite basis sets. They are designed to model the more accurate all-electron calculations at low cost, but without significant loss of accuracy. Unfortunately the correct relativistic all-electron Hamiltonian for a many-electron system is not known and the various pseudopotentials merely model the existing approximate formulations. For most cases of chemical interest, e.g., geometries and binding energies, it usually does not

matter which particular Hamiltonian model is used, i.e., typically errors due to the finite basis set expansion or the limited correlation treatment are much larger than the small differences between the various all-electron models. For very accurate cal-

Table 1. Relative average energy of a configuration of Hg from all-electron (AE) multi-configuration Dirac-Hartree-Fock (DHF) average level calculations using the Dirac-Coulomb (DC) Hamiltonian with a finite nucleus with Fermi charge distribution (fn) or a point nucleus (pn). Contributions from the frequency-dependent Breit (B) interaction (frequency of the exchanged photon 10^3 cm^{-1}) and estimated contributions from quantum electrodynamics (QED, i.e., self-interaction and vacuum polarization) were evaluated in first-order perturbation theory. Errors of energy-consistent pseudopotentials (PP) with 20 valence electrons and different numbers of adjustable parameters with respect to the AE DHF(DC,pn)+B+QED data. All values in cm^{-1}.

configuration		AE, DHF (DC)+B+QED		contribution		error	
		fn	pn	B	QED	PP[a]	PP[b]
Hg	$6s^2$	0	0	0.0	0.0	0.0	0.0
	$6s^1\ 6p^1$	35632.3	35674.4	-52.5	-18.7	1.3	0.0
Hg^+	$6s^1$	68842.1	68885.1	-98.6	-11.6	-0.1	0.0
	$7s^1$	154127.4	154206.2	-220.6	-42.4	-0.4	0.0
	$8s^1$	178127.5	178215.5	-238.4	-41.7	1.1	0.1
	$9s^1$	188751.0	188843.2	-244.1	-40.6	1.6	-0.1
	$6p^1$	122036.8	122128.9	-154.2	-41.8	0.6	0.0
	$7p^1$	167514.3	167609.2	-224.1	-40.3	-3.3	0.0
	$8p^1$	183808.0	183903.6	-238.5	-40.0	-0.8	0.0
	$9p^1$	191697.2	191793.1	-244.0	-39.6	0.6	0.0
Hg^{++}		206962.2	207058.4	-249.8	-39.5	2.6	0.0

[a] energy-consistent pseudopotential with 26 adjustable parameters.

[b] energy-consistent pseudopotential with 54 adjustable parameters.

culations of excitation energies, ionization potentials and electron affinities, or for a detailed investigation of errors inherent in the effective core potential approach, however, such differences might become important. Tables 1 and 2 demonstrate that for very special cases like Hg, with a closed $5d^{10}$-shell in all electronic states considered, a small-core energy-consistent pseudopotential using a semilocal ansatz reaches an accuracy of 10 cm^{-1}, which is well below the effects of the nuclear model, the Breit interaction or higher-order quantum electrodynamical contributions. We also note that differences between results obtained with a frequency-dependent Breit term and the corresponding low-frequency limit amount to up to 10 cm^{-1}. Moreover, the quantum electrodynamic corrections listed in tables 1 and 2 might change by up to 20 cm^{-1} when more recent methods of their estimation are applied [98,99]. Therefore, it is important to state exactly which relativistic all-electron model the effective core potential simulates and, when comparing effective core potentials of different origins, to separate differences in the underlying all-electron approach from errors in the potential itself, e.g., due to the size of the core, the method of adjustment or the form of the valence model Hamiltonian. In this context we want to point out that the seemingly large errors for energy-adjusted pseudopotentials

reported by Mosyagin *et al.* [100,101] are mainly due to the invalid comparison of Wood-Boring-energy-adjusted and Dirac-Fock-orbital-adjusted pseudopotentials to all-electron Dirac-Fock data, i.e., differences in the all-electron model are considered to be pseudopotential errors.

Note that in the above example of Hg the average energy of a configuration (table 1) and the fine-structure (table 2) of one-valence electron states is more accurately represented than the fine-structure of the $6s^1\ 6p^1$ configuration. The small errors in the latter case are a consequence of the pseudoorbital transformation and the overestimation of the 6s-6p exchange integral with pseudo-valence spinors. This error could be reduced further upon using a smaller core, but the efficiency of the approach would be sacrificed. It is also obvious from the compiled data that the accuracy of the valence model Hamiltonian is also a question of the number of adjustable parameters. Claims that such very high accuracy as demonstrated here can only be achieved by adding nonlocal terms for outer core orbitals to the usual semilocal terms [100,101] appear to be invalid, at least for energy-consistent pseudopotentials. Moreover, additional nonlocal terms obviously do not improve the performance for atomic states with a $5d^9$ occupation or in molecular calculations (cf., e.g., tables III and XVII in Mosyagin *et al.* [100]).

Table 2. As table 1, but for fine-structure splittings. All values in cm^{-1}.

configuration		splitting	AE,DHF (DC)+B+QED		contribution		error	
			fn	pn	B	QED	PP[a]	PP[b]
Hg	$6s^16p^1$	$^3P_1$-$^3P_0$	1987.7	1988.6	-25.5	0.9	-14.7	3.0
		$^3P_2$-$^3P_0$	6082.6	6084.8	-96.8	2.9	-28.3	-3.5
		$^1P_1$-$^3P_0$	22994.4	22982.3	-72.4	2.2	-12.4	-9.4
Hg^+	$6p^1$	$^2P_{3/2}$-$^2P_{1/2}$	7765.3	7768.8	-132.8	4.8	-14.8	-0.1
	$7p^1$	$^2P_{3/2}$-$^2P_{1/2}$	2136.8	2137.9	-29.0	1.1	-1.7	0.2
	$8p^1$	$^2P_{3/2}$-$^2P_{1/2}$	939.4	939.9	-12.1	0.4	-4.6	-0.3
	$9p^1$	$^2P_{3/2}$-$^2P_{1/2}$	498.7	498.9	-6.2	0.2	-3.5	0.0

[a] energy-consistent pseudopotential with 26 adjustable parameters.

[b] energy-consistent pseudopotential with 54 adjustable parameters.

6 Core Polarization Potentials

The frozen-core approximation is underlying the effective core potential schemes. One may ask if it is possible to account for static (polarization of the core at the Hartree-Fock level) and dynamic (core-valence correlation) polarization of the cores in a both efficient and accurate way. The core polarization potential (CPP) approach originally developed by Meyer and coworkers [102] for all-electron calculations and adapted by the Stuttgart group [103] for pseudopotential calculations proved to be quite successful in the past. The core polarization potential is written in the

following form

$$V_{cpp} = -\frac{1}{2}\sum_{\lambda}\alpha_\lambda \vec{f_\lambda}^{\,2} + \sum_{\lambda,i} V^\lambda(i) \ . \tag{62}$$

Here α_λ denotes the dipole polarizability of the core λ and $\vec{f_\lambda}$ is the electric field at core λ generated by all other cores and nuclei as well as all valence electrons. Since the validity of the underlying multipole expansion breaks down for small distances from the core λ, the field has to be multiplied by a cut-off function:

$$\vec{f_\lambda} = -\sum_i \frac{\vec{r}_{i\lambda}}{r_{i\lambda}^3}(1-exp(-\delta_e^\lambda r_{i\lambda}^2))^{n_e} + \sum_{\mu\neq\lambda} Q_\mu \frac{\vec{r}_{\mu\lambda}}{r_{\mu\lambda}^3}(1-exp(-\delta_c^\lambda r_{\mu\lambda}^2))^{n_c} \ . \tag{63}$$

The necessary integrals over cartesian Gaussian functions have been presented by Schwerdtfeger and Silberbach [104]. In those cases where ns and np valence orbitals are present together with (n-1)d and (n-2)f valence orbitals, e.g., for Cs, it proved to be more accurate to augment the core polarization potential by a short-range local potential [64]

$$V^\lambda(i) = C^\lambda exp(-\gamma^\lambda r_{i\lambda}^2) \ . \tag{64}$$

A l-dependent cut-off function in Eq. 63 might even be more accurate [105].

7 Calibration Studies

Calibration studies, especially on molecules, are very important for effective core potential methods. Excellent results in atomic calculations are a necessary prerequisite for successful molecular calculations, but provide no guarantee for them. Therefore, effective core potentials should be systematically tested on atoms and small molecules before using them in larger systems. This is especially necessary for cases where a large core is used for economical reasons.

A number of such molecular calibration studies has been performed in the past for energy-consistent pseudopotentials [106,107,108,109,110,111]. Comparison is made to experimental data and/or all-electron results. Some care has to be taken before drawing final conclusions on the quality of pseudopotentials. Usually all molecular calculations are performed using finite basis sets, both at the one-electron and the many-electron level. The truncation of these basis sets leads to errors both at the all-electron and at the pseudopotential level. Most of the time it is relatively easy to generate basis sets of nearly the same quality at the one-particle level, e.g., by augmenting the standard all-electron and pseudopotential basis sets (which of course have to be of the same quality for the valence shells) by the same polarization and correlation functions. It is recommended, however, to approach the basis set limit, at least up to a given angular quantum number, as closely as possible. At the many-particle level it is sometimes more difficult to come to directly comparable basis sets. As an example imagine a large-core pseudopotential augmented by a core polarization potential, which both accounts for static and dynamic core polarization. Static core polarization occurs in the all-electron calculations automatically at the self-consistent field level and can be accounted for in the frozen-core case by single excitations out of the spherical core. Comparing the results of such all-electron

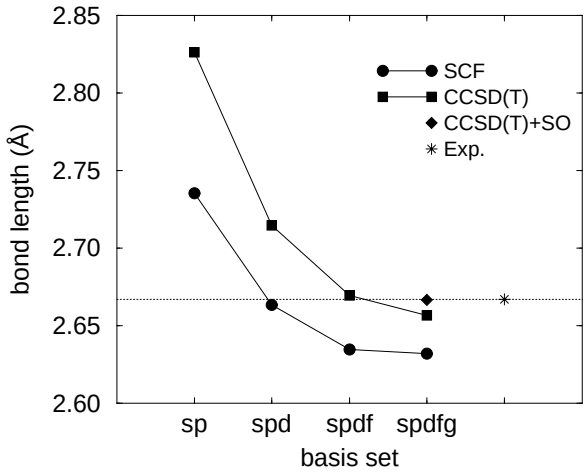


Figure 7. Bond length of the iodine dimer I_2 depending on the basis set. A subset of a 25s21p14d4f3g basis set and a relativistic energy-consistent 7-valence electron pseudopotential augmented by a core polarization potential is used. The experimental value is indicated by a vertical dashed line. The scalar-quasirelativistic pseudopotential calculations at the Hartree-Fock self-consistent field (SCF) and coupled-cluster with single, double and perturbative triple excitations (CCSD(T)) level of theory use an uncontracted (25s21p14d4f3g) Gaussian type basis set. Spin-orbit corrections (+SO) were derived from limited two-component configuration interaction calculations using the quasirelativistic pseudopotential and a contracted [3s3p1d1f] valence basis set of polarized triple-zeta quality.

calculations to pseudopotential Hartree-Fock calculations including the core polarization potential is not entirely correct, however, since the latter also accounts for core-valence correlation. This effect can be modelled in all-electron calculations by single excitations out of the spherical core and simultaneous single excitations in the valence shell. Thus, comparing the results of an all-electron calculation where all electrons are correlated to a correlated pseudopotential calculation with a core polarization potential is also not entirely correct, since the former calculation also accounts for core-core correlation effects. In addition, attention has to be paid with respect to the relativistic contributions taken into account in the Hamiltonian, i.e., the relativistic scheme used in the all-electron reference calculations should not be different, e.g., more approximate, from the scheme used to obtain the reference data in the pseudopotential generation. As an example for a calibration study the results for the iodine dimer I_2 in its ground state are compared to Hartree-Fock and coupled-cluster calculations using a relativistic energy-consistent 7-valence electron pseudopotential together with an uncontracted (25s21p14d4f3g) basis set. The pseudopotential was augmented by a core polarization potential. The results for the bond length, binding energy and vibrational frequency in dependence on the highest angular quantum number used in the basis set are given in Figs. 7 to 9. It is clear from these graphs that the experimental values are only approached for a large basis set and after inclusion of spin-orbit effects. It should be mentioned here that the core polarization potential also makes significant contributions, e.g., -0.03

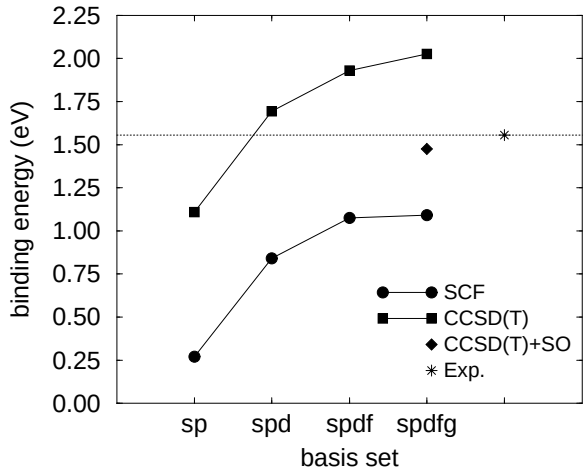


Figure 8. As figure 7, but for the binding energy.

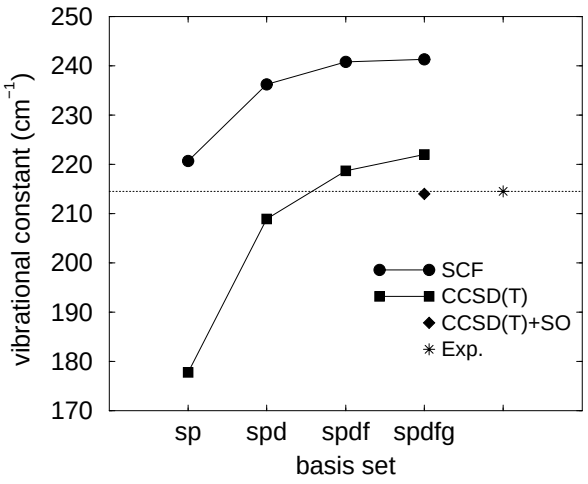


Figure 9. As figure 7, but for the vibrational constant.

Å, +0.09 eV and +3 cm^{-1} for the bond length, binding energy and vibrational constant, respectively, at the Hartree-Fock level. The performance of energy-consistent quasirelativistic 7-valence electron pseudopotentials for all halogen elements has been investigated in a study of the monohydrides and homonuclear dimers [112]. Special attention was also paid to the accuracy of valence correlation energies obtained with pseudo valence orbitals [113,114]. Some of the results for the dimers is presented in tables 3 to 5. The applied basis sets were uncontracted all-electron basis sets: (15s9p5d4f3g) for fluorine, (21s13p5d4f3g) for chlorine, (22s17p11d4f3g)

Table 3. Bond lengths R_e (Å) of the homonuclear halogen dimers from all-electron (AE) Douglas-Kroll-Heß (DKH) and valence-only pseudopotential (PP) Hartree-Fock self-consistent field (SCF) calculations. Core-valence correlation and valence correlation are accounted for by a core polarization potential (CPP) and a coupled-cluster treatment with singles, doubles and perturbative triples (CCSD(T)) including spin-orbit corrections (+SO). For the basis sets cf. the text.

	F_2	Cl_2	Br_2	I_2	At_2
AE DKH,SCF	1.327	1.975	2.273	2.671	2.843
PP,SCF	1.324	1.964	2.266	2.669	2.861
PP+CPP,SCF	1.323	1.958	2.252	2.639	2.822
PP+CPP,CCSD(T)+SO	1.409	1.982	2.281	2.668	2.979
Exp.	1.412	1.988	2.281	2.666	

Table 4. As table 3, but for vibrational constants ω_e (cm^{-1}).

	F_2	Cl_2	Br_2	I_2	At_2
AE DKH,SCF	1267[a]	615	354	232	169
PP,SCF	1271	619	356	238	168
PP+CPP,SCF	1273	622	359	241	172
PP+CPP,CCSD(T)+SO	927	561	324	215	117
Exp.	917	560	325	215	

[a] nonrelativistic result.

Table 5. As table 3, but for binding energies D_e (eV).

	F_2	Cl_2	Br_2	I_2	At_2
AE DKH,SCF	-1.07	1.23	1.01	0.92	0.81
PP,SCF	-1.03	1.26	1.04	0.95	0.79
PP+CPP,SCF	-1.03	1.27	1.08	1.04	0.95
PP+CPP,CCSD(T)+SO	1.66	2.44	1.95	1.57	0.80
Exp.	1.66	2.51	1.99	1.56	

for bromine, (25s20p14d4f3g) for iodine, and (27s22p19d13f3g) for astatine. At the HF level the calibration for the scalar-quasirelativistic pseudopotentials was against all-electron calculations using the Douglas-Kroll-Heß Hamiltonian, whereas at the CCSD(T) level including spin-orbit corrections from limited two-component CI calculations the calibration was with respect to experimental data.

A typical example demonstrating that pseudopotentials account for the major relativistic effects quite accurately and the largest errors in practical calculations are actually due to finite basis sets and too limited correlation treatments is provided by a series of theoretical investigations of gold monofluoride AuF (cf. table 6). All calculations used the same scalar-relativistic energy-consistent 19-valence-electron pseudopotential for Au [115], but the quality of the valence-only calculations was systematically increased during the years. Originally it was believed that AuF is not a stable molecule, until its existence was first predicted theoretically [115,116]

Table 6. Bond length R_e (Å), binding energy D_e (eV) and vibrational constant ω_e (cm^{-1}) of gold monofluoride AuF in the $^1\Sigma^+$ ground state. All theoretical results have been obtained with a 19-valence-electron energy-consistent pseudopotential adjusted to multi-configuration Dirac-Hartree-Fock reference data [115] using different basis sets and valence correlation methods.

method	basis	year	R_e	D_e	ω_e
CISD+SCC(LD)[115]	A	1989	1.978	2.24	509
CEPA-1[115]	A	1989	1.991	2.52	488
Exp.[117]		1992			560
QCISD(T)[116]	B	1994	1.939	3.09	539
Exp.[118]		1994		3.2	
CCSD(T)[119]	C	1997	1.909	3.29	573
MRCI+SCC(S)[119]	C	1997	1.916	3.14	562
MRACPF[119]	C	1997	1.916	3.20	560

Basis sets: A: Au (8s6p5d1f)/[7s3p4d1f], F (13s8p1d)/[7s3p1d];
B: Au (10s8p7d1f)/[9s5p6d1f], F (15s10p2d1f)/[9s7p2d1f];
C: Au (10s8p7d4f2g)/[9s5p6d4f2g], F (13s7p4d3f2g)/[6s5p4d3f2g] (aug-cc-p-vqz).

and later also proven experimentally [117,118]. The most recent calculations [119] are in excellent agreement with the available experimental data.

8 A few hints for practical calculations

Some of the simple hints for practical applications of effective core potentials given in the following may appear to be trivial or superfluous for some of the readers, but experience during the last years showed that they may be welcome by the more application-oriented ones who are less familiar with the methods.

Effective core potentials are usually a good and safe choice when properties related to the valence electron system are to be investigated. It should always be remembered, however, that the size of the core not only determines the computational effort, but it also influences the accuracy of the results. Small-core and medium-core potentials are usually safe to use, whereas the range of large-core potentials is much more limited. In the latter case it might be important to include a core-core and/or core-nucleus repulsion correction as well as a core polarization potential. It is not a wise decision to simply neglect these terms, e.g., because the CPP is not implemented in GAUSSIAN yet.

When using an effective core potential for the first time always do an atomic test calculation first, e.g., for the ionization potential or electron affinity, in order to check the correctness of your input and/or the programs library data. Especially in pseudopotential calculations well-known sources of input errors are the $1/r^n$ prefactors used in some parametrizations or the presence/absence of a local potential. It is recommended to use the valence basis set coming with the effective core potential, possibly augmented by additional diffuse and polarizations functions. Especially in case of pseudopotentials, where the detailed innermost shape of the pseudoorbitals is essentially arbitrary, it is not recommended to use (contracted) all-electron basis

sets or valence basis sets from other potentials, since significant basis set superposition errors may result. However, the added diffuse and polarization functions may safely be taken from all-electron or other effective core potential basis sets.

When comparing to other all-electron or valence-only calculations use basis sets and correlation treatments of the same quality and make sure that relativistic effects are included at similar levels. Note that in all-electron calculations basis set superposition errors tend to be larger than in valence-only calculations.

Acknowledgments

The author is grateful to H. Stoll (Stuttgart) for more than 15 years of cooperation on the field of pseudopotentials. Financial support of the Deutsche Forschungsgemeinschaft and the Fonds der Chemischen Industrie is also acknowledged.

References

1. J.D. Weeks, A. Hazi, and S.A. Rice, On the use of pseudopotentials in the quantum theory of atoms and molecules, Adv. Quant. Chem. **16**, 283 (1969).
2. J.N. Bardsley, Pseudopotentials in atomic and molecular physics, Case studies in atomic physics **4**, 299 (1974).
3. R.N. Dixon and I.L.Robertson, The use of pseudopotentials in molecular calculations, Spec. Period. Rep., Theor. Chem., The Chemical Society, London **3**, 100 (1978).
4. A. Hibbert, Model potentials in atomic structure theory, Adv. Atom. Molec. Phys. **18**, 309 (1982).
5. K.S. Pitzer, Relativistic calculations of dissociation energies and related properties, Int. J. Quant. Chem. **25**, 131 (1984).
6. L.R. Kahn, Electronic structure of molecules using one-component wave functions and relativistic effective core potentials, Int. J. Quant. Chem. **25**, 149 (1984).
7. M. Krauss and W.J. Stevens, Effective potentials in molecular quantum chemistry, Ann. Rev. Phys. Chem. **35**, 357 (1984).
8. P.A. Christiansen, W.C. Ermler, and K.S. Pitzer, Relativistic effects in chemical systems, Ann. Rev. Phys. Chem. **36**, 407 (1985).
9. W.C. Ermler, R.B. Ross, and P A. Christiansen, Spin-orbit coupling and other relativistic effects in atoms and molecules, Adv. Quantum Chem. **19**, 139 (1988).
10. C. Laughlin and G.A. Victor, Model-potential methods, Adv. At. Mol. Phys. **25**, 163 (1988).
11. O. Gropen, The relativistic effective core potential method, in: Methods in Computational Chemistry, ed. S. Wilson, Plenum, New York **2**, 109 (1988).
12. W.E. Pickett Pseudopotential methods in condensed matter applications, Comput. Phys. Rep. **9**, 115 (1989).
13. S. Huzinaga, Effective Hamiltonian method for molecules, J. Mol. Struct. (Theochem) **234**, 51 (1991).
14. J.R. Chelikowsky and M.L. Cohen, Ab initio pseudopotentials and the struc-

tural properties of semiconductors, in: Handbook on Semiconductors, ed. P.T. Landsberg, Elsevier, Amsterdam **1**, 59 (1992).
15. S. Huzinaga, Concept of active electrons in chemistry, Can. J. Chem. **73**, 619 (1995).
16. G. Frenking, I. Antes, M. Böhme, S. Dapprich, A.W. Ehlers, V. Jonas, A. Neuhaus, M. Otto, R. Stegmann, A. Veldkamp, and S.F. Vyboishchikov, Pseudopotential calculations of transition metal compounds - scope and limitations, Rev. Comp. Chem. **8**, 63 (1996).
17. T.R. Cundari, M.T. Benson, M.L. Lutz, and S.O. Sommerer, Effective core potential approaches to the chemistry of the heavier elements, Rev. Comp. Chem. **8**, 145 (1996).
18. K. Balasubramanian, Relativistic effective core potential techniques for molecules containing very heavy atoms, in: Encyclopedia of Computational Chemistry, eds. P.v.R. Schleyer *et al.*, Wiley, Chichester , 2471 (1998).
19. P. Pyykkö and H. Stoll, Relativistic Pseudopotential Calculations, 1993 - June 1999, in: R.S.C. Spec. Period. Rep., Chemical Modelling, Applications and Theory (2000), in press.
20. L. Seijo and Z. Barandiarán, The ab initio model potential method: a common strategy for effective core potential and embedded cluster calculations, in: Computational Chemistry: Reviews of Current Trends, ed. J. Leszczynski, World Scientific, Singapore, **4**, (in press).
21. P. Pyykkö, Relativistic Theory of Atoms and Molecules. A Bibliography 1916 – 1985, in: Lecture Notes in Chemistry, Springer, Berlin **41**, (1986).
22. P. Pyykkö, Relativistic Theory of Atoms and Molecules II. A Bibliography 1986 – 1992, in: Lecture Notes in Chemistry, Springer, Berlin **60**, (1993).
23. P. Pyykkö, Relativistic Theory of Atoms and Molecules III. A Bibliography 1993 – ..., in: `http://www.csc.fi/lul/rtam` (1999).
24. K.S. Pitzer, Relativistic effects on chemical properties, Acc. Chem. Res. **12**, 271 (1979).
25. P. Pyykkö, J.-P. Desclaux, Relativity and the periodic system of elements, Acc. Chem. Res. **12**, 276 (1979).
26. P. Pyykkö, Relativistic quantum chemistry, Adv. Quant. Chem. **11**, 353 (1978).
27. W.H.E. Schwarz, Relativistic calculations of molecules. Relativity and bond lengths, Phys. Scr. **36**, 403 (1987).
28. W. Kutzelnigg, The relativistic many body problem in molecular theory, Phys. Scr. **36**, 416 (1987).
29. K. Balasubramanian and K.S. Pitzer, Relativistic quantum theory, in: Ab initio Methods in Quantum Chemistry, ed. K.P. Lawley, Wiley, New York **1**, (1987).
30. P. Pyykkö, Relativistic effects in structural chemistry, Chem. Rev. **88**, 563 (1988).
31. G. Malli, Ab-initio relativistic quantum chemistry, in: Molecules in Physics, Chemistry and Biology, ed. J. Maruani, Kluwer, Dordrecht **2**, 85 (1988).
32. K. Balasubramanian, Relativity and chemical bonding, J. Phys. Chem. **93**, 6585 (1989).

33. W.H.E. Schwarz, Fundamentals of relativistic effects in chemistry, in: Theoretical Models of Chemical Bonding, part 2, The Concept of the Chemical Bond, Springer, Berlin , 593 (1990).
34. B.A. Heß, C.M. Marian, and S. Peyerimhoff, Ab initio calculation of spin-orbit effects in molecules including electron correlation, in: Advanced Series in Physical Chemistry: Modern Electronic Structure Theory, ed. D.R. Yarkony, World Scientific **2**, 152 (1995).
35. J. Almlöf and O. Gropen, Relativistic effects in chemistry, in: Reviews in Computational Chemistry, eds. K.B. Lipkowitz and B.D. Boyd, VCH Publishers, New York **8**, (1996).
36. M. Dolg and H. Stoll, Electronic structure calculations for molecules containing lanthanide atoms, in: Handbook on the Physics and Chemistry of Rare Earths, eds. K.A. Gschneidner, Jr., and L. Eyring, Elsevier, Amsterdam **22**, 607 (1996).
37. B.A. Heß, Relativistic effects in heavy-element chemistry, Ber. Bunsenges. **101**, 1 (1997).
38. B.A. Heß, Relativistic theory and applications, in: The Encyclopedia of Computational Chemistry, eds. P.v.R. Schleyer *et al.*, Wiley, Chichester , 2499 (1998).
39. M. Dolg and H.-J. Flad, Ground state properties of Hg_2. I. A pseudopotential configuration interaction study. J. Phys. Chem. **100**, 6147 (1996).
40. M. Douglas and N.M. Kroll, Quantum electrodynamical corrections to the fine structure of Helium, Ann. Phys. **82**, 89 (1974).
41. B.A. Heß, Applicability of the no-pair equation with free-particle projection operators to atomic and molecular structure calculations, Phys. Rev. A **32**, 756 (1985).
42. B.A. Heß, Relativistic electronic-structure calculations employing a two-component no-pair formalism with external field projection operators, Phys. Rev. A **33**, 3742 (1986).
43. G. Jansen and B.A. Heß, Revision of the Douglas-Kroll transformation, Phys. Rev. A **39**, 6016 (1989).
44. R. Samzow and B.A. Heß, Spin-orbit effects in the Br atom in the framework of the no-pair theory, Chem. Phys. Lett. **184**, 491 (1991).
45. R. Samzow, B.A. Heß, and G. Jansen, The two-electron terms of the no-pair Hamiltonian, J. Chem. Phys. **96**, 1227 (1991).
46. J.H. Wood and A.M. Boring, Improved Pauli Hamiltonian for local-potential problems, Phys. Rev. B **18**, 2701 (1978).
47. R.D. Cowan and D.C. Griffin, Approximate relativistic corrections to atomic radial wave functions, J. Opt. Soc. Am. **66**, 1010 (1976).
48. Z. Barandiarán and L. Seijo, The ab initio model potential method. Cowan-Griffin relativistic core potentials and valence basis sets from Li (Z=3) to La (Z=57), Can. J. Chem. **70**, 409 (1992).
49. Z. Barandiarán and L. Seijo, Quasirelativistic ab initio model potential calculations on the group IV hydrides (XH2, XH4; X=Si,Ge,Sn,Pb) and oxides (XO; X=Ge,Sn,Pb), J. Chem. Phys. **101**, 4049 (1994).
50. L. Seijo, Relativistic ab initio model potential calculations including spin-orbit

effects through the Wood-Boring Hamiltonian, J. Chem. Phys. **102**, 8078 (1995).
51. M. Casarubios and L. Seijo, The ab initio model potential method. Relativistic Wood-Boring valence spin-orbit potentials and spin-orbit-corrected basis sets from B(Z=5) to Ba(Z=56), J. Mol. Struct. (THEOCHEM) **426**, 59 (1998).
52. M. Casarubios and L. Seijo, The ab initio model potential method: Third-series transition metal elements, J. Chem. Phys. **110**, 784 (1999).
53. S. Díaz-Megías and L. Seijo, Wood-Boring ab initio model potential relativistic treatment of Ce and CeO, Chem. Phys. Lett. **299**, 613 (1999).
54. F. Rakowitz, C.M. Marian, L. Seijo, and U. Wahlgren, Spin-free relativistic no-pair ab initio core model potentials and valence basis sets for the transition metal elements Sc to Hg. Part I. J. Chem. Phys. **110**, 3678 (1999).
55. F. Rakowitz, C.M. Marian, and L. Seijo, Spin-free relativistic no-pair ab initio core model potentials and valence basis sets for the transition metal elements Sc to Hg. Part II. J. Chem. Phys. **111**, 10436 (1999).
56. H. Hellmann, A new approximation method in the problem of many electrons, J. Chem. Phys. **3**, 61 (1935).
57. P. Gombás, Über die metallische Bindung, Z. Phys. **94**, 473 (1935).
58. H. Preuss, Untersuchungen zum kombinierten Näherungsverfahren, Z. Naturf. **10a**, 365 (1955).
59. J.C. Phillips and L. Kleinman, A new method for calculating wavefunctions in crystals and molecules, Phys. Rev. **116**, 287 (1959).
60. L.E. McMurchie and E.R. Davidson, Calculation of integrals over ab initio pseudopotentials, J. Comput. Chem. **4**, 289 (1981).
61. R.M. Pitzer and N.W. Winter, Electronic structure methods for heavy-atom molecules, J. Phys. Chem. **92**, 3061 (1988).
62. R.M. Pitzer and N.W. Winter, Spin-orbit (core) and core potential integrals, Int. J. Quant. Chem. **40**, 773 (1991).
63. M. Pélissier, N. Komiha, and J.P. Daudey, One-center expansion for pseudopotential matrix elements, J. Comput. Chem. **9**, 298 (1988).
64. M. Dolg, Fully relativistic pseudopotentials for alkaline atoms: Dirac-Hartree-Fock and configuration interaction calculations of alkaline monohydrides. Theor. Chim. Acta **93**, 131 (1996).
65. P. Durand and J.C. Barthelat, A theoretical method to determine atomic pseudopotentials for electronic structure calculations of molecules and solids, Theor. Chim. Acta **38**, 283 (1975).
66. P.A. Christiansen, Y.S. Lee, and K.S. Pitzer, Improved ab initio effective core potentials for molecular calculations, J. Chem. Phys. **71**, 4445 (1979).
67. L.F. Pacios and P.A. Christiansen, Ab initio relativistic effective potentials with spin-orbit operators. I. Li through Ar, J. Chem. Phys. **82**, 2664 (1985).
68. M.M. Hurley, L.F. Pacios, P.A. Christiansen, R.B. Ross, and W.C. Ermler, Ab initio relativistic effective potentials with spin-orbit operators: II. K through Kr, J. Chem. Phys. **84**, 6840 (1986).
69. L.A. LaJohn, P.A. Christiansen, R.B. Ross, T. Atashroo, and W.C. Ermler, Ab initio relativistic effective potentials with spin-orbit operators. III. Rb through Xe, J. Chem. Phys. **87**, 2812 (1987).

70. R.B. Ross, J.M. Powers, T. Atashroo, W.C. Ermler, L.A. LaJohn, and P.A. Christiansen, Ab initio relativistic effective potentials with spin-orbit operators. IV. Cs through Rn, J. Chem. Phys. **93**, 6654 (1990).
71. W.C. Ermler, R.B. Ross, and P.A. Christiansen, Ab initio relativistic effective potentials with spin-orbit operators. VI. Fr through Pu, Int. J. Quant. Chem. **40**, 829 (1991).
72. N.M. Wallace, J.-P. Blaudeau, and R.M. Pitzer, Optimized Gaussian basis sets for use with relativistic effective (core) potentials: Li-Ar, Int. J. Quant. Chem. **40**, 789 (1991).
73. R.B. Ross, S. Gayen, and W.C. Ermler, Ab initio relativistic effective potentials with spin-orbit operators. V. Ce through Lu. J. Chem. Phys. **100**, 8145 (1994).
74. C.S. Nash, B.E. Bursten, and W.C. Ermler, Ab initio relativistic potentials with spin-orbit operators. VII. Am through element 118. J. Chem. Phys. **106**, 5133 (1997).
75. S.A. Wildman, G.A. DiLabio, and P.A. Christiansen, Accurate relativistic effective potentials for the sixth-row main group elements, J. Chem. Phys. **107**, 9975 (1997).
76. J.-P. Blaudeau and L.A. Curtiss, Optimized Gaussian basis sets for use with relativistic effective (core) potentials: K, Ca, Ga-Kr, Int. J. Quant. Chem. **61**, 943 (1997).
77. P.J. Hay, Ab initio studies of excited states of polyatomic molecules including spin-orbit and multiplet effects: The electronic states of UF_6, J. Chem. Phys. **79**, 5469 (1983).
78. P.J. Hay and W.R. Wadt, Ab initio effective core potentials for molecular calculations. Potentials for the transition metal atoms Sc to Hg, J. Chem. Phys. **82**, 270 (1985).
79. W.R. Wadt and P.J. Hay, Ab initio effective core potentials for molecular calculations. Potentials for main group elements Na to Bi, J. Chem. Phys. **82**, 284 (1985).
80. P.J. Hay and W.R. Wadt, Ab initio effective core potentials for molecular calculations. Potentials for K to Au including the outermost core orbitals, J. Chem. Phys. **82**, 299 (1985).
81. P.J. Hay and R.L. Martin, Theoretical studies of the structures and vibrational frequencies of actinide compounds using relativistic effective core potentials with Hartree-Fock and density functional methods: UF_6, NpF_6, and PuF_6, J. Chem. Phys. **109**, 3875 (1998).
82. W.J. Stevens, H. Basch, and M. Krauss, Compact effective potentials and efficient shared-exponent basis sets for the first- and second-row atoms, J. Chem. Phys. **81**, 6026 (1984).
83. W.J. Stevens, M. Krauss, H. Basch, and P.G. Jasien, Relativistic compact effective potentials and efficient, shared-exponent basis sets for the third-, fourth-, and fifth-row atoms, Can. J. Chem. **70**, 612 (1992).
84. T.R. Cundari and W.J. Stevens, Effective core potential methods for the lanthanides, J. Chem. Phys. **98**, 5555 (1993).
85. M. Dolg, U. Wedig, H. Stoll, and H. Preuss, Energy-adjusted ab initio pseu-

dopotentials for the first row transition elements, J. Chem. Phys. **86**, 866 (1987).
86. M. Dolg, H. Stoll, H. Preuss, and R.M. Pitzer, Relativistic and correlation effects for element 105 (Hahnium, Ha). A comparative study of M and MO (M=Nb, Ta, Ha) using energy-adjusted ab initio pseudopotentials. J. Phys. Chem. **97**, 5852 (1993).
87. M. Dolg, H. Stoll and H. Preuss, Energy-adjusted ab initio pseudopotentials for the rare earth elements, J. Chem. Phys. **90**, 1730 (1989).
88. M. Dolg, H. Stoll, A. Savin, and H. Preuss, Energy-adjusted pseudopotentials for the rare earth elements, Theor. Chim. Acta **75**, 173 (1989).
89. D. Andrae, U. Häußermann, M. Dolg, H. Stoll and H. Preuß, Energy-adjusted ab initio pseudopotentials for the second and third row transition elements, Theor. Chim. Acta **77**, 123 (1990).
90. W. Küchle, M. Dolg, H. Stoll, and H. Preuss, Ab initio pseudopotentials for Hg through Rn. I. Parameter sets and atomic calculations, Mol. Phys. **74**, 1245 (1991).
91. M. Dolg, H. Stoll, and H. Preuss, A combination of quasirelativistic pesudopotential and ligand field calculations for lanthanoid compounds, Theor. Chim. Acta **85**, 441 (1993).
92. U. Häussermann, M. Dolg, H. Stoll, H. Preuss, P. Schwerdtfeger, and R.M. Pitzer, Accuracy of energy-adjusted quasirelativistic ab initio pseudopotentials: all-electron and pseudopotential benchmark calculations for Hg, HgH and their cations. Mol. Phys. **78**, 1211 (1993).
93. A. Bergner, M. Dolg, W. Küchle, H. Stoll, and H. Preuss, Ab initio energy-adjusted pseudopotentials for elements of groups 13-17, Mol. Phys. **80**, 1431 (1993).
94. W. Küchle, M. Dolg, H. Stoll, and H. Preuss, Energy-adjusted pseudopotentials for the actinides. Parameter sets and test calculations for thorium and thorium monoxide, J. Chem. Phys. **100**, 7535 (1994).
95. M. Seth, P. Schwerdtfeger, and M. Dolg, The chemistry of the superheavy elements. I. Pseudopotentials for 111 and 112 and relativistic coupled cluster calculations for (112)H^+, (112)F_2, and (112)F_4, J. Chem. Phys. **106**, 3623 (1997).
96. B. Metz, M. Schweizer, H. Stoll, M. Dolg, and W. Liu, A small-core multiconfiguration Dirac-Hartree-Fock-adjusted pseudopotential for Tl. Application to TlX (X = F, Cl, Br, I), Theor. Chem. Acc. (2000), in press.
97. B. Metz, H. Stoll, and M. Dolg, A small-core MCDHF-adjusted pseudopotentials for post-d main group elements. Application to PbH and PbO, J. Chem. Phys. (2000), in press.
98. P. Pyykkö, M. Tokman, and L.N. Labzowski, Estimated valence-level Lamb shifts for group 1 and group 11 metal atoms, Phys. Rev. A **57**, R689 (1998).
99. L. Labzowski, I. Goidenko, M. Tokman, and P. Pyykkö, Calculated self-energy contributions for an ns valence electron using the multiple-commutator method, Phys. Rev. A **59**, 2707 (1999).
100. N.S. Mosyagin, A.V. Titov, and Z. Latajka, Generalized relativistic effective core potential: Gaussian expansion of potentials and pseudospinors for atoms

Hg through Rn, Int. J. Quant. Chem. **63**, 1107 (1997).
101. N.S. Mosyagin, E. Eliav, A.V. Titov, and U. Kaldor, Comparison of relativistic effective core potential and all-electron Dirac-Coulomb calculations of mercury transition energies by the relativistic coupled-cluster method. J. Phys. B **33**, 667 (2000).
102. W. Müller, J. Flesch, and W. Meyer, Treatment of intershell correlation effects in ab initio calculations by use of core polarization potentials. Method and application to alkali and alkaline earth atoms, J. Chem. Phys. **80**, 3297 (1984).
103. P. Fuentealba, H. Preuss, H. Stoll, and L.v. Szentpály, A proper account of core-polarization with pseudopotentials: single valence-electron alkali compounds, Chem. Phys. Lett. **89**, 418 (1982).
104. P. Schwerdtfeger and H. Silberbach, Multicenter integrals over long-range operators using Cartesian Gaussian functions, Phys. Rev. A **37**, 2834 (1988), erratum: ibidem, **42**, 665 (1990).
105. M. Foucrault, P. Millie, and J.P. Daudey, Nonperturbative method for core-valence correlation in pseudopotential calculations: Application to the Rb_2 and Cs_2 molecules, J. Chem. Phys. **96**, 1257 (1992).
106. U. Steinbrenner, A. Bergner, M. Dolg, and H. Stoll, On the transferability of energy adjusted pseudopotentials: a calibration study for XH_4 (X=C, Si, Ge, Sn, Pb). Mol. Phys. **82**, 3 (1994).
107. A. Nicklass and H. Stoll, On the importance of core polarization in heavy post-d elements: a pseudopotential calibration study for X_2H_6 (X = Si, Ge, Sn, Pb), Mol. Phys. **86**, 317 (1995).
108. P. Schwerdtfeger, T. Fischer, M. Dolg, G. Igel-Mann, A. Nicklass, H. Stoll, and A. Haaland, The accuracy of the pseudopotential approximation. I. Analysis of the spectroscopic constants for the electronic ground states of InCl and $InCl_3$ using various three valence electron pseudopotentials for indium, J. Chem. Phys. **102**, 2050 (1995).
109. T. Leininger, A. Nicklass, W. Küchle, H. Stoll, M. Dolg, and A. Bergner, The accuracy of the pseudopotential approximation: Non-frozen-core effects for spectroscopic constants of alkali fluorides XF (X = K, Rb, Cs), Chem. Phys. Lett. **255**, 274 (1996).
110. T. Leininger, A. Nicklass, H. Stoll, M. Dolg, and P. Schwerdtfeger, The accuracy of the pseudopotential approximation. II. A comparison of various core sizes for indium pseudopotentials in calculations for spectroscopic constants of InH, InF, and InCl, J. Chem. Phys. **105**, 1052 (1996).
111. W. Küchle, M. Dolg, and H. Stoll, Ab initio study of the lanthanide and actinide contraction, J. Phys. Chem. A **101**, 7128 (1997).
112. M. Dolg, Accuracy of energy-adjusted quasirelativistic pseudopotentials: a calibration study of XH and X_2 (X=F,Cl,Br,I,At). Mol. Phys. **88**, 1645 (1996).
113. M. Dolg, On the accuracy of valence correlation energies in pseudopotential calculations, J. Chem. Phys. **104**, 4061 (1996).
114. M. Dolg, Valence correlation energies from pseudopotential calculations, Chem. Phys. Lett. **250**, 75 (1996).

115. P. Schwerdtfeger, M. Dolg, W.H.E. Schwarz, G.A. Bowmaker, P.D.W. Boyd, Relativistic effects in gold chemistry. I. Diatomic gold compounds, J. Chem. Phys. **91**, 1762 (1989).
116. P. Schwerdtfeger, J.S. McFeaters, R.L. Stephens, M.J. Liddell, M. Dolg, B.A. Heß, Can AuF be synthesized ? A theoretical study using relativistic configuration interaction and plasma modeling techniques, Chem. Phys. Lett. **218**, 362 (1994).
117. K.L. Saenger and C.P. Sun, Yellow emission bands produced during gold etching in O_2-CF_4 rf glow-discharge plasmas: Evidence for gas-phase AuF, Phys. Rev. A **46**, 670 (1992).
118. D. Schröder, J. Hru*s*ák, I.C. Tornieporth-Oetting, T.M. Klapötke, H. Schwarz, Neutral gold(I) fluoride does exist, Angew. Chemie **106**, 223 (1994); Angew. Chem. Int. Ed. Engl. **33**, 212 (1994).
119. M. Dolg, Quasirelativistische und relativistische energiekonsistente Pseudopotentiale für quantentheoretische Untersuchungen der Chemie schwerer Elemente, Habilitationsschrift, Universität Stuttgart (1997), available on the internet at `http://www.mpipks-dresden.mpg/~dolg/habi.html` .

MOLECULAR PROPERTIES

JÜRGEN GAUSS
Institut für Physikalische Chemie
Universität Mainz
55099 Mainz
Germany
E-mail: gauss@slater.chemie.uni-mainz.de

An introduction is given into the quantum chemical calculation of molecular properties with special emphasis on analytic derivative theory, magnetic properties, and frequency-dependent properties.

1 Introduction

In order to relate results from quantum chemical calculations to experiment, it is essential to compute quantities that are directly available from measurements. Clearly, energies and wavefunctions obtained from the solution of the (electronic) Schrödinger equation are not sufficient for this purpose, and it is necessary to compute further quantities that characterize the atomic or molecular system of interest. In particular, theoretical predictions of structure, spectroscopic quantities, as well as properties such as, for example, dipole moment, polarizability, etc. are important to establish the connection to experiment and to initiate a fruitful interplay between theory and experiment.

It is quite obvious that the routine and efficient computation of the various atomic and molecular properties requires techniques which go beyond the "simple" solution of the Schrödinger equation. Besides theoretical aspects, computational efficiency is – as always in quantum chemistry – the most important issue that needs to be addressed.

Considering a molecule in a given electronic state, quantities of interest are:

a) energy differences, i.e., reaction energies, atomization energies, dissociation energies, energy differences between various isomers or conformers, etc.

b) molecular properties specific for a given electronic state. Examples include the equilibrium structure, dipole moment, polarizability, vibrational frequencies, magnetazibility, NMR chemical shifts, etc.

c) properties that characterize transitions between different electronic states. Examples are here electronic excitation energies, one- and two-photon transition strengths, radiative life times, ionization potentials, electron affinities, etc.

Properties of type a involve energy information at *different* points on the Born-Oppenheimer potential energy surface. For a dissociation energy, for example, the energy of the molecule as well as the energies of the fragments are needed. Properties of type b require information for one electronic state at a *single* point on the potential surface, while properties of type c involve information for *different* electronic states.

The given classification of molecular properties is of computational relevance, as, for example, properties of type a just require energy calculations for different points on the potential energy surface and in this respect are rather straightforward to calculate. Computation of properties of type b and c, however, is more involved and requires techniques such as analytic derivative and/or response theory.

In this lecture, a thorough discussion of the quantum chemical calculations for properties of type b will be given. Special emphasis will be put on analytic derivative theory (section 2), problems inherent to the calculation of magnetic properties (section 3) as well as frequency-dependent properties (section 4).

2 Molecular properties as analytical derivatives

2.1 General discussion

Properties of type b describe the "response" of the molecular system to an external perturbation. Let us consider as an example a molecule in an external electrical field ε. If we treat the field as a weak perturbation, a Taylor expansion around the the "field-free" case is a good description and yields for the energy

$$E(\varepsilon) = E(\varepsilon = 0) + \left.\frac{dE}{d\varepsilon}\right|_{\varepsilon=0} \varepsilon + \frac{1}{2}\left.\frac{d^2E}{d\varepsilon^2}\right|_{\varepsilon=0} \varepsilon^2 + \dots \qquad (1)$$

The first-order term in Eq. (1), i.e., the term *linear* in ε, involves the *first derivative* (gradient) of the energy with respect to ε, the second-order term, i.e., the term *quadratic* in ε, the corresponding *second derivative*, etc.

From Eq. (1), it is clear that derivatives of the energy play a key role in describing the response of a molecule to an external perturbation. However, to identify these derivatives with the molecular properties of interest, it is essential to consider also the physical aspect of the interaction with the external field. Doing that, it becomes clear that the first-order interaction with an external electric field involves the molecular dipole moment μ, the second-order interaction the molecular polarizability α, etc.[1] We can therefore make the following identifications

$$\begin{array}{lcll} \text{dipole moment } (\mu) & \hat{=} & -\left.\frac{dE}{d\varepsilon}\right|_{\varepsilon=0} & \text{(first derivative)} \\ \text{polarizability } (\alpha) & \hat{=} & -\left.\frac{d^2E}{d\varepsilon^2}\right|_{\varepsilon=0} & \text{(second derivative)} \\ \text{first hyperpolarizability } (\beta) & \hat{=} & -\left.\frac{d^3E}{d\varepsilon^3}\right|_{\varepsilon=0} & \text{(third derivative)} \\ & \dots & & \end{array}$$

and thus already obtain *computational expressions* for these particular molecular properties: they can be determined as the corresponding derivatives of the energy with respect to the components of the external field.

One might ask why a property such as the dipole moment is not just calculated as a simple expectation value, as it should be possible according to the postulates of quantum mechanics[2]. Indeed, the Hellmann-Feynman theorem[3] states the identity of the derivative and expectation value expression for first-order properties:

$$\frac{dE}{dx} = \langle\Psi|\frac{\partial H}{\partial x}|\Psi\rangle \qquad (\text{Hellmann} - \text{Feynman Theorem}). \qquad (2)$$

However, it should be noted that the Hellmann-Feynman theorem does not necessarily hold for approximate wavefunctions[a] with which we are generally dealing in quantum chemistry. In such cases, it has been shown that derivative expressions are usually the preferred choice [4].

The derivative ansatz can be generalized to other properties and in this way allows computation of a large variety of atomic and molecular properties. Table 1 gives an overview. The list includes the electrical properties which have been already discussed. Most important for chemical applications, however, are probably properties that are connected to geometrical derivatives. First derivatives with respect to nuclear displacements define the forces on nuclei which are essential for the location of stationary points on the Born-Oppenheimer potential surface and, thus, for the determination of equilibrium and transition state structures[5]. Second derivatives with respect to nuclear coordinates allow a characterization of the stationary points as minima, transition states, etc., but in addition enable a qualitative characterization of vibrational spectra within the harmonic-oscillator approximation. More quantitative treatments of vibrational spectra are possible via higher derivatives (cubic and quartic force constants; see, for example, the discussion given in Ref. 6).

Another important class are magnetic properties. For chemists, certainly NMR chemical shifts and the corresponding spin-spin coupling constants are of greatest interest. However, theory also allows calculation of magnetazibilities, spin-rotation constants, rotational g-tensors (see the book by W.H. Flygare[7] for a detailed account on these properties) as well as parameters that can be obtained from ESR spectroscopy (see the book by W. Weltner[8] for an introduction). A few examples of quantum chemical property calculations will be given later.

2.2 Numerical versus analytical differentiation

In principle, derivatives of the energy can be computed in a rather straightforward manner using finite-differentiation techniques, e.g., the gradient can be obtained via

$$\frac{dE}{dx} \approx \frac{E(\Delta x) - E(-\Delta x)}{2\Delta x} \tag{3}$$

with Δx as an appropriate chosen step size. The main advantage of such a *numerical differentiation* scheme is that it just requires the calculation of energies (though in the presence of the perturbation) and, thus, is rather easily implemented. As a consequence, finite-differentiation techniques have been and are still often used for the calculation of electric properties (so-called "finite-field" calculations). The disadvantages of the numerical differentiation scheme, however, are

a) the limited accuracy (a problem in particular for the computation of higher derivatives),

b) the high computational cost, as numerical differentiation requires for each derivative two additional energy calculations.

[a] It should be noted that the Hellmann-Feynman theorem also holds for a few special cases such as, for example, Hartree-Fock theory in the complete basis set limit.

Table 1. List of molecular properties which can be computed as derivatives of energies

Derivative	Observable
$\frac{dE}{d\varepsilon_i}$	dipole moment; in a similar manner also multipole moments, electric field gradients, etc.
$\frac{d^2E}{d\varepsilon_\alpha d\varepsilon_\beta}$	polarizability
$\frac{d^3E}{d\varepsilon_\alpha d\varepsilon_\beta d\varepsilon_\beta}$	(first) hyperpolarizability
$\frac{dE}{dx_i}$	forces on nuclei; stationary points on potential energy surfaces, equilibrium and transition state structures
$\frac{d^2E}{dx_i dx_j}$	harmonic force constants; harmonic vibrational frequencies
$\frac{d^3E}{dx_i dx_j dx_k}$	cubic force constants; vibrational corrections to distances and rotational constants
$\frac{d^4E}{dx_i dx_j dx_k dx_l}$	quartic force constants; anharmonic corrections to vibrational frequencies
$\frac{d^2E}{dx_i d\varepsilon_\alpha}$	dipole derivatives; infrared intensities within the harmonic approximation
$\frac{d^3E}{dx_i d\varepsilon_\alpha d\varepsilon_\beta}$	polarizability derivative; Raman intensities
$\frac{d^2E}{dB_\alpha dB_\beta}$	magnetazibility
$\frac{d^2E}{dm_{Kj} dB_\alpha}$	nuclear magnetic shielding tensor; relative NMR shifts
$\frac{d^2E}{dI_{Ki} dI_{Lj}}$	indirect spin-spin coupling constant
$\frac{d^2E}{dB_\alpha dJ_\beta}$	rotational g-tensor; rotational spectra in magnetic field
$\frac{d^2E}{dI_{Ki} dB_\alpha}$	nuclear spin-rotation tensor; fine structure in rotational spectra
$\frac{dE}{dm_{Kj}}$	spin density; hyperfine interaction constants
$\frac{d^2E}{dS_i dB_\alpha}$	electronic g-tensor

The latter issue is of particular concern if one is interested in the forces on the N nuclei of a molecule within a geometry optimization. The numerical evaluation of

gradients requires in this case $2*3N$ the cost of the corresponding energy calculation. It is obvious that in this way routine determination of geometries is impossible for larger molecules (see section 2.7).

Further disadvantages of the numerical differentiation scheme are

a) that there is no straightforward extension to the computation of frequency-dependent properties (see section 4) and

b) that handling of magnetic properties is less straightforward (see section 3), as the computation of the latter requires the capability of dealing with complex wavefunction parameters. This capability is in most cases not available.

The alternative to *numerical* differentiation is *analytic* differentiation. This means that first an analytic expression for the corresponding derivative is deduced and then implemented within a computer code for the actual computation of the corresponding property. As we will see later, the use of analytic derivative techniques solves the mentioned problems and, thus, is clearly the preferred choice for the computation of properties.

However, it should be noted that application of analytic derivative techniques is not as straightforward and often requires a complicated computer implementation. The latter often requires substantial programming efforts as well as theoretical work for the derivation of the appropriate derivative expressions. Nevertheless, as the implementation needs in principle to be carried out only once, this cannot be considered a major disadvantage. Analytic derivatives have been in the mean time implemented for most of the standard quantum chemical approaches. For some of the more advanced quantum chemical techniques, however, the task of programming analytic derivatives can become so demanding that the corresponding implementations are still missing[b]. In a similar way, lack of analytic schemes for the calculation of higher derivatives often necessitates the latter to be calculated within mixed analytic-numerical schemes in which lower analytic derivatives are numerically differentiated. A prominent example is here the computation of cubic and quartic force constants which often are obtained by numerical differentiation of analytically evaluated quadratic force constants[9].

2.3 Analytic derivatives: general discussion

We start our discussion of analytic derivatives with a description of the general structure of derivative theory. Naively, one would expect that suitable expressions for the derivatives of the energy can be obtained by simple differentiation of the energy expression with respect to the corresponding parameter(s). While this is in principle true, such an approach, however, does not necessarily lead to computationally efficient expressions. The main problem appears to be the implicit dependence of the energy on the perturbation through the wavefunction parameters.

For our general discussion, it is important to analyze first in which way energy and wavefunction depend on a given external perturbation. For both, we have an *explicit* dependence on the perturbation through either some set of fixed parameters

[b] Examples are here CCSDT and CCSDTQ

(e.g., the basis functions which are given at the beginning of a calculation) or the Hamiltonian and an *implicit* dependence through the wavefunction parameters which are determined by some set of equations. Examples for the latter are the molecular orbital (MO) coefficients, the configuration-interaction (CI) coefficients, as well as the coupled-cluster (CC) amplitudes. As long as we are not specifying these parameters further, we will denote them collectively by c.

The energy can be then written in the following convenient form

$$E = E(x, c(x)) \tag{4}$$

and the equations for the wavefunction parameters c in the form

$$g(x, c(x)) = 0 \tag{5}$$

with the function g indicating the structure of the equations for c. Note that Eqs. (4) and (5) are rather general and that various quantum chemical methods just differ in the definition of the set c as well as in the explicit expressions for E and g.

Differentiation of the general energy expression given in Eq. (4) then yields[c]

$$\frac{dE}{dx} = \left(\frac{\partial E}{\partial x}\right) + \left(\frac{\partial E}{\partial c}\right)\left(\frac{\partial c}{\partial x}\right). \tag{6}$$

The first term includes the *explicit* dependence on the perturbation through the Hamiltonian as well as the fixed set of parameters, i.e., the basis functions. This contribution is usually denoted as the *integral derivative* contribution and is easily handled (see section 2.6). The second term is more problematic, as it includes the derivatives of the coefficients c. Contrary to the first term, a straightforward computation is here not possible, as the derivatives $\partial c/\partial x$ are unknown. At a first sight, one might think that these derivatives need to be determined by solving additional equations obtained by differentiation of Eq. (5):

$$\frac{dg}{dx} = h(x, c, \frac{\partial c}{\partial x}) = 0. \tag{7}$$

However, as the determination of c via Eq. (5) usually is the computationally most expensive step of a quantum chemical calculation, solution of Eq. (7) for the perturbed c's would render the analytic scheme rather expensive. The cost would be similar to those of the numerical scheme.

For the following, it is necessary to distinguish between parameters c determined via the variation principle (variational parameters) and those not determined via the variation principle (non-variational parameters). Examples for the first type of parameters are the MO coefficients in Hartree-Fock (HF) theory or the CI coefficients in CI calculations, while CC amplitudes are examples for the second type.

For the variational coefficients, the following condition holds

$$\frac{dE}{dc} = \left(\frac{\partial E}{\partial c}\right) = 0 \tag{8}$$

[c]Note that all derivatives are taken here and in the following at the point $x = 0$.

as the variation principle requests minimization of the energy with respect to c. Thus, it is clear that in Eq. (6), i.e., the general expression for the energy gradient, the second term does not contribute and the whole expression simplifies to

$$\frac{dE}{dx} = \left(\frac{\partial E}{\partial x}\right). \tag{9}$$

There is no need to determine the perturbed coefficients $\partial c/\partial x$.

The situation is more complicated for the non-variational parameters, as here the corresponding derivatives $\partial E/\partial c$ do not vanish. Nevertheless, it is possible to eliminate the derivatives of c from the gradient expression. The most elegant way to demonstrate this uses an energy functional $\tilde{E}$ constructed from the energy expression (Eq. (4)) by augmenting it with the equations that determine c (Eq. (5)) multiplied by Lagrangian multipliers λ:

$$\tilde{E}(x, c(x), \lambda(x)) = E(x, c(x)) + \lambda(x) g(x, c(x)). \tag{10}$$

This functional provides the same energy as Eq. (4), as long as Eq. (5) is fulfilled. Furthermore, the energy functional can be made stationary with respect to λ and c by requesting that

$$\left(\frac{d\tilde{E}}{d\lambda}\right) = 0 \tag{11}$$

and

$$\left(\frac{d\tilde{E}}{dc}\right) = 0. \tag{12}$$

The first of the two equations is identical to Eq. (5) for the parameters c, while the second equation allows determination of the Lagrangian multipliers λ. Solution of this additional set is not needed for the determination of the energy but is required to make $\tilde{E}$ stationary with respect to λ.

With Eqs. (11) and (12), the derivative of $\tilde{E}$ with respect to x takes the following rather simple form

$$\frac{d\tilde{E}}{dx} = \left(\frac{\partial \tilde{E}}{\partial x}\right) \tag{13}$$

$$= \left(\frac{\partial E}{\partial x}\right) + \lambda \left(\frac{\partial g}{\partial x}\right) \tag{14}$$

and, as the value of $\tilde{E}$ is identical to E, Eq. (13) also represents the desired gradient expression for E. It can be thus concluded that like for the variational coefficients (see Eq. (9)) there is no need to compute derivatives for the non-variational coefficients c. However, unlike for the variational case, for each non-variational parameter c an additional, though perturbation-independent equation (Eq. (12)) needs to be solved.

Expressions for higher derivatives can be obtained by differentiating the gradient expressions (Eqs. (9) and (13), respectively) with respect to further perturbations.

Derivation of computationally efficient expressions might require some rearrangements, but there are rules which can simplify the derivation. These rules explicitly state that for the coefficients c the knowledge of the nth derivative is sufficient for the calculation of the $(2n+1)$th derivative of the energy ($2n+1$ rule) and for the Lagrangian multipliers λ knowledge of the nth derivative is sufficient to compute the $(2n+2)$th derivative of the energy ($2n+2$ rule). According to these two rules, none of the derivatives is required for the gradient (see discussion above). Knowledge of the derivatives of c suffices for second derivatives, while third derivatives require derivatives of c and for non-variational parameters also of λ. The derivatives of c and λ are determined by solving equations that are obtained by differentiating Eq. (11) and Eq. (12) with respect to the external perturbation, respectively. We illustrate the application of the two rules by some examples.

For the variational case, the expression for the second derivative of the energy takes the following form

$$\frac{d^2E}{dxdy} = \left(\frac{\partial^2 E}{\partial x \partial y}\right) + \left(\frac{\partial^2 E}{\partial x \partial c}\right)\frac{\partial c}{\partial y} + \left(\frac{\partial^2 E}{\partial y \partial c}\right)\frac{\partial c}{\partial x} + \left(\frac{\partial^2 E}{\partial c \partial c}\right)\frac{\partial c}{\partial x}\frac{\partial c}{\partial y}. \tag{15}$$

The contribution due to $\partial^2 c/\partial x \partial y$ vanishes here, as the corresponding prefactor $\partial E/\partial c$ is zero due to Eq. (8).

For non-variational approaches, we obtain for the second derivative of the energy

$$\frac{d^2E}{dxdy} = \left(\frac{\partial^2 \tilde{E}}{\partial x \partial y}\right) + \left(\frac{\partial^2 \tilde{E}}{\partial x \partial c}\right)\frac{\partial c}{\partial y} + \left(\frac{\partial^2 \tilde{E}}{\partial y \partial c}\right)\frac{\partial c}{\partial x} + \left(\frac{\partial^2 \tilde{E}}{\partial c \partial c}\right)\frac{\partial c}{\partial x}\frac{\partial c}{\partial y}. \tag{16}$$

The contributions due to $\partial^2 c/\partial x \partial y$ and $\partial^2 \lambda/\partial x \partial y$ vanish because of Eqs. (11) and (12), i.e., due to the stationarity of the functional $\tilde{E}$ with respect to c and λ. The contributions due to $\partial\lambda/\partial x$ and $\partial\lambda/\partial y$ vanish, because

$$\frac{d}{dx}\left(\frac{\partial \tilde{E}}{\partial \lambda}\right) = \frac{dg}{dx} = 0. \tag{17}$$

The prefactors for the terms involving the first derivatives of λ are just the derivatives of the left-hand side of Eq. (5). As Eq. (5) holds for all values of x and y, the corresponding total derivatives of $g(x,c)$ must vanish.

As another example, we give the expression for the third derivatives of the

energy in the non-variational case.

$$\begin{aligned}
\frac{d^3E}{dxdydz} = &\left(\frac{\partial^3\tilde{E}}{\partial x\partial y\partial z}\right) \\
&+\left(\frac{\partial^3\tilde{E}}{\partial x\partial y\partial c}\right)\frac{\partial c}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial x\partial z\partial c}\right)\frac{\partial c}{\partial y}+\left(\frac{\partial^3\tilde{E}}{\partial y\partial z\partial c}\right)\frac{\partial c}{\partial x} \\
&+\left(\frac{\partial^3\tilde{E}}{\partial x\partial c\partial c}\right)\frac{\partial c}{\partial y}\frac{\partial c}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial y\partial c\partial c}\right)\frac{\partial c}{\partial x}\frac{\partial c}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial z\partial c\partial c}\right)\frac{\partial c}{\partial x}\frac{\partial c}{\partial y} \\
&+\left(\frac{\partial^3\tilde{E}}{\partial c\partial c\partial c}\right)\frac{\partial c}{\partial x}\frac{\partial c}{\partial y}\frac{\partial c}{\partial z} \\
&+\left(\frac{\partial^3\tilde{E}}{\partial x\partial y\partial\lambda}\right)\frac{\partial\lambda}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial x\partial z\partial\lambda}\right)\frac{\partial\lambda}{\partial y}+\left(\frac{\partial^3\tilde{E}}{\partial y\partial z\partial\lambda}\right)\frac{\partial\lambda}{\partial x} \\
&+\left(\frac{\partial^3\tilde{E}}{\partial x\partial c\partial\lambda}\right)\frac{\partial c}{\partial y}\frac{\partial\lambda}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial y\partial c\partial\lambda}\right)\frac{\partial c}{\partial x}\frac{\partial\lambda}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial z\partial c\partial\lambda}\right)\frac{\partial c}{\partial x}\frac{\partial\lambda}{\partial y} \\
&+\left(\frac{\partial^3\tilde{E}}{\partial x\partial c\partial\lambda}\right)\frac{\partial c}{\partial z}\frac{\partial\lambda}{\partial y}+\left(\frac{\partial^3\tilde{E}}{\partial y\partial c\partial\lambda}\right)\frac{\partial c}{\partial z}\frac{\partial\lambda}{\partial x}+\left(\frac{\partial^3\tilde{E}}{\partial z\partial c\partial\lambda}\right)\frac{\partial c}{\partial y}\frac{\partial\lambda}{\partial x} \\
&+\left(\frac{\partial^3\tilde{E}}{\partial c\partial c\partial\lambda}\right)\frac{\partial c}{\partial x}\frac{\partial c}{\partial y}\frac{\partial\lambda}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial c\partial c\partial\lambda}\right)\frac{\partial c}{\partial x}\frac{\partial\lambda}{\partial y}\frac{\partial c}{\partial z}+\left(\frac{\partial^3\tilde{E}}{\partial c\partial c\partial\lambda}\right)\frac{\partial\lambda}{\partial x}\frac{\partial c}{\partial y}\frac{\partial c}{\partial z}
\end{aligned} \tag{18}$$

Note that Eq. (18) contains no contribution due to higher derivatives of c and λ. Again it can be shown that the corresponding prefactors are identical to zero.

The general approach to analytic derivatives presented here has been first formulated by Helgaker and Jørgensen[10]. Though the corresponding computationally efficient equations can be also obtained in different ways, for example, by using the interchange theorem of perturbation theory[11], the *Lagrangian approach* to analytic derivatives appears to be the simplest as well as the most elegant way to derive computationally efficient expressions. This is also demonstrated in the next section where a few specific examples for analytic energy derivatives are given.

2.4 Analytic derivatives: Specific examples

As first example, we discuss first derivatives (gradients) for standard closed-shell Hartree-Fock self-consistent-field (HF-SCF) calculations. Within the atomic-orbital (AO) notation (Greek indices $\mu, \nu, ...$ are used here and in the following to denote AOs), the following energy expression is obtained for this case

$$E = \sum_{\mu\nu} P_{\mu\nu} h_{\mu\nu} + \frac{1}{2}\sum_{\mu\nu\rho\sigma} P_{\mu\nu} P_{\sigma\rho}(\langle\mu\sigma|\nu\rho\rangle - \frac{1}{2}\langle\mu\sigma|\rho\nu\rangle) \tag{19}$$

with $P_{\mu\nu}$ as the closed-shell AO density matrix

$$P_{\mu\nu} = 2\sum_i c^*_{\mu i} c_{\nu i}, \tag{20}$$

$h_{\mu\nu}$ as the one-electron integrals, and $\langle\mu\sigma|\nu\rho\rangle$ as the two-electron integrals in Dirac notation. Indices $i, j, ...$ denote in Eq. (20) as well as in the following occupied molecular orbitals, while indices $a, b, ...$ are reserved for virtual orbitals and indices $p, q, ...$ are used for generic orbitals that are either occupied or unoccupied.

The variational wavefunction parameters are the MO coefficients $c_{\mu i}$. As the orthonormality of the MOs needs to be considered as additional constraint, it is most convenient to start with the following energy functional

$$\begin{aligned}\tilde{E} = &\sum_{\mu\nu} P_{\mu\nu} h_{\mu\nu} + \frac{1}{2}\sum_{\mu\nu\rho\sigma} P_{\mu\nu} P_{\sigma\rho}(\langle\mu\sigma|\nu\rho\rangle - \frac{1}{2}\langle\mu\sigma|\rho\nu\rangle) \\ &-2\sum_{ij}\varepsilon_{ij}(\sum_{\mu\nu} c^*_{\mu i} S_{\mu\nu} c_{\nu j} - \delta_{ij})\end{aligned} \tag{21}$$

instead of Eq. (19). The Lagrangian multipliers ε_{ij} can be shown to form a Hermitian matrix. It can be furthermore demonstrated that this matrix can be chosen diagonal which leads to the usual case of canonical HF orbitals (see, for example, Ref. 12). The diagonal element $\varepsilon_i \equiv \varepsilon_{ii}$ can be interpreted as orbital energies (Koopmans' theorem).

Based on the discussion in the previous section, the following gradient expression can be derived[13]

$$\begin{aligned}\frac{dE}{dx} = &\sum_{\mu\nu} P_{\mu\nu}\frac{\partial h_{\mu\nu}}{\partial x} + \frac{1}{2}\sum_{\mu\nu\sigma\rho} P_{\mu\nu} P_{\sigma\rho}(\frac{\partial\langle\mu\sigma|\nu\rho\rangle}{\partial x} - \frac{1}{2}\frac{\partial\langle\mu\sigma|\rho\nu\rangle}{\partial x}) \\ &-\sum_{\mu\nu} W_{\mu\nu}\frac{\partial S_{\mu\nu}}{\partial x}\end{aligned} \tag{22}$$

with the energy-weighted density matrix defined by

$$W_{\mu\nu} = 2\sum_i c^*_{\mu i}\varepsilon_i c_{\nu i}. \tag{23}$$

From Eq. (22), it is clear that evaluation of HF-SCF gradients consists of a series of contractions of density matrices with the corresponding integral derivatives. For perturbations of the one-electron type, the expression in Eq. (22) takes a particularly simple form, as only the first term needs to be considered:

$$\frac{dE}{dx} = \sum_{\mu\nu} P_{\mu\nu}\frac{\partial h_{\mu\nu}}{\partial x} \quad (\mathrm{one-electron\ perturbation}). \tag{24}$$

For geometrical derivatives, however, the two-electron contribution needs to be considered due to the perturbation dependence of the basis functions. The latter are usually centered at the nuclear positions and thus change with a displacement of the nuclear coordinates. Calculation of the integral derivatives is nowadays a routine task and efficient algorithms have been devised[14]. It should be also noted

that calculation of HF-SCF gradients does not require storage of integral derivatives which would be a serious bottleneck.

For second derivatives of the energy, the following expression is obtained at the HF-SCF level[15,16]

$$\begin{aligned}\frac{d^2E}{dxdy} &= \sum_{\mu\nu} P_{\mu\nu}\frac{\partial^2 h_{\mu\nu}}{\partial x\partial y} + \frac{1}{2}\sum_{\mu\nu\sigma\rho} P_{\mu\nu}P_{\sigma\rho}(\frac{\partial^2\langle\mu\sigma|\nu\rho\rangle}{\partial x\partial y} - \frac{1}{2}\frac{\partial^2\langle\mu\sigma|\rho\nu\rangle}{\partial x\partial y}) \\ &- \sum_{\mu\nu} W_{\mu\nu}\frac{\partial^2 S_{\mu\nu}}{\partial x\partial y} + \sum_{\mu\nu}\frac{\partial P_{\mu\nu}}{\partial y}(\frac{\partial h_{\mu\nu}}{\partial x} + \sum_{\sigma\rho} P_{\sigma\rho}(\frac{\partial\langle\mu\sigma|\nu\rho\rangle}{\partial x} - \frac{1}{2}\frac{\partial\langle\mu\sigma|\rho\nu\rangle}{\partial x})) \\ &- \sum_{\mu\nu}\frac{\partial W_{\mu\nu}}{\partial y}\frac{\partial S_{\mu\nu}}{\partial x} \end{aligned} \tag{25}$$

with

$$\frac{\partial P_{\mu\nu}}{\partial x} = 2\sum_i\{\frac{\partial c^*_{\mu i}}{\partial x}c_{\nu i} + c^*_{\mu i}\frac{\partial c_{\nu i}}{\partial x}\} \tag{26}$$

$$\frac{\partial W_{\mu\nu}}{\partial x} = 2\sum_i\{\frac{\partial c^*_{\mu i}}{\partial x}\epsilon_i c_{\nu i} + c^*_{\mu i}\epsilon_i\frac{\partial c_{\nu i}}{\partial x}\} + \sum_{ij} c^*_{\mu i}\frac{\partial\epsilon_{ji}}{\partial x}c_{\nu j}. \tag{27}$$

As a consequence of the $(2n+1)$ rule, the expression given in Eq (25) contains only first derivatives of the MO coefficients. The latter are determined through equations that are obtained by differentiating the HF equations

$$\sum_\nu f_{\mu\nu}c_{\nu i} = \sum_\nu S_{\mu\nu}c_{\nu i}\epsilon_i \tag{28}$$

with $f_{\mu\nu}$ as the Fock matrix

$$f_{\mu\nu} = h_{\mu\nu} + \sum_{\sigma\rho} P_{\sigma\rho}(\langle\mu\sigma|\nu\rho\rangle - \frac{1}{2}\langle\mu\sigma|\rho\nu\rangle) \tag{29}$$

or more conveniently by differentiating the equivalent condition

$$\sum_{\mu\nu} c^*_{\mu a}f_{\mu\nu}c_{\nu i} = 0 \tag{30}$$

which is also known as Brillouin's theorem. It is common to parametrize for this purpose the MO coefficient derivatives in the following way

$$\frac{\partial c_{\mu i}}{\partial x} = \sum_p c_{\mu p}U^x_{pi} \tag{31}$$

with the coupled-perturbed HF (CPHF) coefficients U^x_{pi} as the parameters to be determined. The CPHF equations[15,16] obtained from differentiating Eq. (28) or Eq. (30) take then (for real perturbations) the following form[d].

$$\sum_e\sum_m (4\langle ae|im\rangle - \langle ae|mi\rangle - \langle am|ei\rangle + \delta_{ae}\delta_{im}(\varepsilon_a - \varepsilon_i)) = B^x_{ai} \tag{32}$$

[d]Note that only the coefficients U^x_{ai} are determined through the CPHF equations, while the coefficients U^x_{ij} can be chosen in any way, as long as the differentiated orthonormality condition

$$U^x_{qp}{}^* + S^x_{pq} + U^x_{pq} = 0$$

with

$$B^x_{ai} = -\sum_{\mu\nu} c^*_{\mu a} c_{\nu i} \left(\frac{\partial h_{\mu\nu}}{\partial x} + \sum_{\sigma\rho} P_{\sigma\rho} \left(\frac{\partial \langle \mu\sigma|\nu\rho\rangle}{\partial x} - \frac{1}{2}\frac{\partial \langle \mu\sigma|\rho\nu\rangle}{\partial x}\right)\right) + \sum_{mn} S^x_{mn} \left(2\langle am|in\rangle - \langle am|ni\rangle\right) + S^x_{ai}\epsilon_i \tag{33}$$

with

$$S^x_{pq} = \sum_{\mu\nu} c^x_{\mu p} \frac{\partial S_{\mu\nu}}{\partial x} c_{\nu q}. \tag{34}$$

The CPHF equations are linear and can be solved using iterative algorithms[16].

As another example, first derivatives of the energy in coupled-cluster (CC) theory[17] are discussed. To simplify the discussion, we ignore the dependence of the CC energy on the molecular orbitals and consider only the CC amplitudes as wavefunction parameters. The general CC energy expression is then given by

$$E = \langle 0| \exp(-T) H \exp(T) |0\rangle \tag{35}$$

with T as the cluster operator and $|0\rangle$ denoting the reference determinant, usually provided by the HF wavefunction. The amplitudes in CC theory are determined through equations obtained by projecting the Schrödinger equation on the excitation manifold. Denoting with Φ_p the determinants within this excitation manifold, these equations, usually referred to as coupled-cluster equations, are given by

$$0 = \langle \Phi_p| \exp(-T) H \exp(T) |0\rangle. \tag{36}$$

For the derivatives of the CC gradient expression, we introduce the energy functional

$$\tilde{E} = \langle 0| \exp(-T) H \exp(T) |0\rangle + \sum_p \lambda_p \langle \Phi_p| \exp(-T) H \exp(T) |0\rangle \tag{37}$$

with λ_p as the appropriate Lagrangian multipliers. Eq. (37) is often written in the following short form

$$\tilde{E} = \langle 0| (1 + \Lambda) \exp(-T) H \exp(T) |0\rangle \tag{38}$$

where we have introduced the de-excitation operator Λ by

$$\langle 0|\Lambda = \sum_p \lambda_p \langle \Phi_p|. \tag{39}$$

For the CC energy gradient, we then obtain in agreement with the $(2n+1)$ and $(2n+2)$ rules

$$\frac{dE}{dx} = \langle 0| (1 + \Lambda) \exp(-T) \frac{\partial H}{\partial x} \exp(T) |0\rangle. \tag{40}$$

is fulfilled with the derivatives S^x_{pq} defined as in Eq. (34). The most common choice is

$$U^x_{ij} = -\frac{1}{2} S^x_{ij}.$$

Perturbed T and Λ thus are not required for the evaluation of CC gradients[18]. The λ amplitudes required in Eq. (40) are obtained as solutions to the following equations

$$0 = \langle 0|(1+\Lambda)[\exp(-T)H\exp(T) - E]|\Phi_p\rangle. \tag{41}$$

which are usually referred to as Λ equations.

The actual CC gradient expression is somewhat more involved due to the additional dependence of the CC energy on the MO coefficients. Though the MO coefficients are determined via the variational principle for the HF wavefunction, they are non-variational within CC theory. The reason is that the HF and not the CC energy is made stationary with respect to orbital rotations. Accordingly, the following energy functional represents a suitable starting point

$$\begin{aligned}\tilde{E} = \langle 0|(1+\Lambda)\exp(-T)H\exp(T)|0\rangle + \sum_a \sum_i Z_{ai} \sum_{\mu\nu} c^*_{\mu a} f_{\mu\nu} c_{\nu i} \\ + \sum_{pq} I_{pq} (\sum_{\mu\nu} c^*_{\mu p} S_{\mu\nu} c_{\nu q} - \delta_{pq}),\end{aligned} \tag{42}$$

where we augment the energy functional given in Eq. (37) by the HF condition (Eq. (30), Z_{ai} is the corresponding Lagrangian multiplier) and by the additional orthonormality constraint with I_{pq} as undetermined multiplier.

Differentiating this energy functional yields the complete gradient expression

$$\begin{aligned}\frac{d\tilde{E}}{dx} = \langle 0|(1+\Lambda)\exp(-T)\frac{\partial H}{\partial x}\exp(T)|0\rangle + \sum_a \sum_i Z_{ai} \sum_{\mu\nu} c^*_{\mu a} [\frac{\partial h_{\mu\nu}}{\partial x} \\ + \sum_{\sigma\rho} P_{\sigma\rho} (\frac{\partial \langle \mu\sigma|\nu\rho\rangle}{\partial x} - \frac{1}{2}\frac{\partial \langle \mu\sigma|\rho\nu\rangle}{\partial x})] c_{\nu i} + \sum_{pq} I_{pq} \sum_{\mu\nu} c^*_{\mu p} \frac{\partial S_{\mu\nu}}{\partial x} c_{\nu q},\end{aligned} \tag{43}$$

which is usually cast in the following general form[19,20]

$$\frac{dE}{dx} = \sum_{\mu\nu} D_{\mu\nu} \frac{\partial h_{\mu\nu}}{\partial x} + \sum_{\mu\nu\sigma\rho} \Gamma_{\mu\nu\sigma\rho} \frac{\partial \langle \mu\nu|\sigma\rho\rangle}{\partial x} + \sum_{\mu\nu} I_{\mu\nu} \frac{\partial S_{\mu\nu}}{\partial x} \tag{44}$$

with $D_{\mu\nu}$ as an effective one-particle density matrix, $\Gamma_{\mu\nu\sigma\rho}$ as the two-particle density matrix and $I_{\mu\nu}$ as a generalized energy-weighted density matrix. The form given in Eq. (44) for the gradient is rather general and has turned out very convenient for actual computations. Differences between the various quantum chemical approaches just exist in the definition of the corresponding density matrices.

A comment is necessary concerning the additional Lagrangian multipliers introduced in Eq. (43). Z_{ai} leads to an *orbital relaxation* contribution to the one-particle density matrix. It actually is the quantity that has been first introduced by Handy and Schaefer as Z-vector[21]. I_{pq} turns out to be the MO representation of the energy-weighted density matrix $I_{\mu\nu}$ in Eq. (44). Explicit expressions for $D_{\mu\nu}, \Gamma_{\mu\nu\sigma\rho}$, and $I_{\mu\nu}$ for the various CC models have been given in the literature[22–29] and will not be repeated here.

Finally, some basic aspects of the calculation of CC second derivatives are discussed. To emphasize the important aspects, we solely focus on the CC amplitudes

and ignore the MO coefficient contributions. A corresponding complete discussion can be found in the literature[30,31].

Differentiation of Eq. (37) with respect to x and y yields[31]

$$\begin{aligned}\frac{d^2E}{dxdy} = &\langle 0|(1+\Lambda)\exp(-T)\frac{\partial^2 H}{\partial x \partial y}\exp(T)|0\rangle \\ &+\langle 0|(1+\Lambda)[\exp(-T)\frac{\partial H}{\partial x}\exp(T),\frac{dT}{dy}]|0\rangle + \\ &+\langle 0|(1+\Lambda)[\exp(-T)\frac{\partial H}{\partial y}\exp(T),\frac{dT}{dx}]|0\rangle + \\ &+\langle 0|(1+\Lambda)[[\exp(-T)H\exp(T),\frac{dT}{dx}],\frac{dT}{dy}]|0\rangle. \end{aligned} \tag{45}$$

As stated by the $(2n+1)$ and $(2n+2)$ rules, the expression given in Eq. (45) contains only first derivatives of the cluster operator (and thus of the CC amplitudes) and no derivative contribution of the Λ operator. The required perturbed amplitudes are determined by solving the first-order CC equations obtained by differentiating the corresponding unperturbed equations given in Eq. (36) with respect to x or y:

$$\langle \Phi_p|\exp(-T)\frac{\partial H}{\partial x}\exp(T)|0\rangle + \langle \Phi_p|[\exp(-T)H\exp(T),\frac{\partial T}{\partial x}]|0\rangle = 0. \tag{46}$$

For some cases, it might be advantageous to rearrange Eq. (45) and to use instead the following expression[31]

$$\begin{aligned}\frac{d^2E}{dxdy} = &\langle 0|(1+\Lambda)\exp(-T)\frac{\partial^2 H}{\partial x \partial y}\exp(T)|0\rangle \\ &+\langle 0|(1+\Lambda)[\exp(-T)\frac{\partial H}{\partial x}\exp(T),\frac{dT}{dy}]|0\rangle + \\ &+\langle 0|\frac{d\Lambda}{dy}\exp(-T)\frac{\partial H}{\partial x}\exp(T)|0\rangle. \end{aligned} \tag{47}$$

In Eq. (47), derivatives of the Lagrangian multipliers appear (at a first sight contrary to the $(2n+2)$ rule), but one should realize that derivative amplitudes in Eq. (47) are only needed for one of the two perturbations, namely y. If the two perturbations belong to different classes, this might be of advantage, as the total number of equations that need to be solved can be lower when using Eq. (47) instead of Eq. (45). The most prominent example is the computation of NMR shielding constants, where with the second, so-called *asymmetric* expression, only 6 perturbed equations for all components of the magnetic field need to be solved, while use of the *symmetric* expression (Eq. (45)) requires solution of a total of $(3N_{atoms}+3)$ equations.

This example shows that different expressions for a derivative are possible. It depends on the circumstances, in particular on the property of interest, which of the deduced expressions is preferred and computationally more efficient.[31]

2.5 Advantages of analytic derivatives

Analytic derivatives are the preferred choice (if available for a quantum chemical approach) for the following reasons:

a) first of all, analytic derivatives generally provide higher accuracy for the calculated derivatives, as they are not affected by rounding errors (too small step sizes in the numerical differentiation) or problems due to contamination by higher derivatives (too large step size in the numerical differentiation). This aspect is of particular importance for higher derivatives, but of lesser concern for first derivatives;

b) the cost of analytic gradients is independent of the number of perturbations. Considering expressions Eq. (22) or Eq. (44), it is clear that the perturbation dependence only appears through the integral derivative contribution. As can be shown, computation of integral derivatives can be carried out with cost independent of the number of perturbations. The same computational advantage is not necessarily as pronounced as for higher derivatives. For example, analytic computation of CC second derivatives scales with the number of perturbations[31]. The same dependence is seen for the calculation of second derivatives based on numerical differentiation of analytically evaluated gradients;

c) magnetic properties can be easily treated, as all quantities in the analytic approach are either purely imaginary or real and, thus, can be handled using real arithmetic. The only complication arises due to the fact that for magnetic properties some matrices are antisymmetric instead of symmetric.

d) extension to frequency-dependent properties is possible in the framework of response theory.

2.6 Availability of analytic derivatives

The use of analytic derivatives in quantum chemistry started in 1969 with the classic work of Pulay[13] on HF gradients[e]. As soon as their applicability and efficiency for geometry optimizations and force constant calculations[33] had been realized, a lot of effort was devoted to theory as well as implementation of analytic derivatives. Important contributions to analytic derivatives were made by many groups; noteworthy probably in particular those of the groups of Pople, Schaefer, Handy as well as Bartlett. Landmarks in the development of analytic derivative techniques have been the implementation of second derivatives at the HF level and MP2 gradients in 1979 by Pople *et al.*[16], the formulation and implementation of MP2 second derivatives by Handy and Bartlett *et al.*[34–36] in 1985, and the implementation of CCSD gradients by Schaefer *et al.*[22] in 1987.

A more general important contribution to derivative theory has been the introduction of the Z-vector method by Handy and Schaefer[21] which had a large impact on the theory of gradients for correlated quantum chemical approaches. Though computation of integral derivatives is nowadays considered a routine issue, it should not be forgotten that the availability of integral derivative packages has been an important prerequisite for all of the advances mentioned above.

[e]It might be of interest to note that papers on analytic derivatives (though within a one-center basis set expansion) have been published as early as 1958 by Bratoz[32].

Table 2. Implementation of analytic geometrical derivatives within standard quantum chemical approaches

First derivatives		Ref.
HF	Pulay (1969)	13
DFT	Ziegler *et al.* (1988), Salahub *et al.* (1989),	37,38
	Delley (1991)	39
MCSCF	Kato and Morokuma (1979),	40
	Goddard, Handy, Schaefer (1979)	41
MP2	Pople *et al.*(1979)	16
MP3	Bartlett *et al.* (1985)	42
MP4(SDQ)	Gauss and Cremer (1987)	43
MP4	Gauss and Cremer (1988),	44
	Bartlett *et al.* (1988)	45
CID, CISD	Schaefer *et al.*, Pople *et al.* (1980)	46,47
CPF	Rice, Lee, Handy (1980)	48
QCISD	Gauss and Cremer (1988)	49
QCISD(T)	Gauss and Cremer (1989)	50
CCSD	Schaefer *et al.* (1987)	22
CCSD(T)	Scuseria(1990), Lee and Rendell (1990)	26,27
CCSDT-n	Gauss and Stanton (2000)	29
MR-CI, MR-ACPF	Shepard *et al.* (1991)	51
Second derivatives		
HF	Pople *et al.* (1979)	16
DFT	Handy *et al.* (1993), Johnson, Frisch (1994)	52,53
MCSCF	Schaefer, Handy *et al.* (1984)	54
MP2	Handy *et al.* (1985), Bartlett *et al.* (1986)	34,35,36
MP3, MP4	Gauss and Stanton (1997)	31
CISD	Schaefer *et al.* (1983)	55
CCSD	Koch, Jørgensen, Schaefer *et al.* (1990)	56
CCSD(T)	Gauss and Stanton (1997)	31
CCSDT-n	Gauss and Stanton (2000)	29
Third derivatives		
HF	Schaefer *et al.* (1984)	57

Table 2 gives an overview about the available analytic derivative techniques within the *standard* quantum chemical approaches. The list of references given there, however, must remain incomplete, as it is impossible to mention all papers in the area of analytic derivatives. Often, initial implementations were followed up by extensions to open-shell systems or in case of correlated approaches to other type of reference functions. In addition, there have been many papers considering algorithmic advances. For example, a number of papers have been published reporting direct or semi-direct implementations of MP2 gradients[58−60], thus significantly extending the range of the applicability of MP2 gradients.

With the increasing popularity of density functional theory (DFT) in quantum

chemistry, interest very soon focused on analytic DFT derivatives and implementation of first and second derivatives were reported[37−39,52,53]. While analytic derivatives are no more complicated for DFT than for HF-SCF, it should be noted that differentiation of the exchange-correlation contribution can pose some problems due to the usually employed numerical integration schemes[61].

2.7 Application of analytic derivatives: examples for geometrical derivatives

A few illustrative examples for the application of analytic (geometrical) derivatives will be given in the following. Fig. 1 shows a dimer of a hexabenzocoronene derivative (HBC-*t*Bu) which has been synthesized by Müllen *et al.*[62] The structure has been investigated by X-ray analysis[62] as well as by solid-state NMR spectroscopy in the group of Spiess[62]. These experimental studies have been supplemented by quantum chemical calculations[63]. We do not discuss the chemical aspects of these calculations rather focus on their computational aspects. The dimer of HBC-tBu consists of 264 atoms and the corresponding calculations using the 3-21G basis involved 1452 basis functions and 786 degrees of freedom in the geometry optimization.

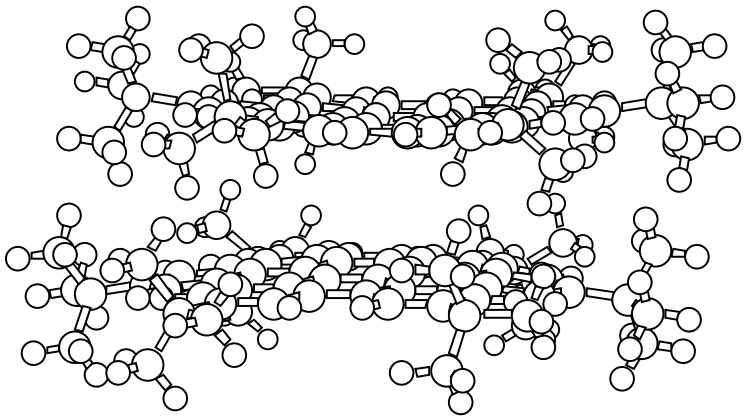


Fig. 1: Optimized structure of HBC-*t*Bu dimer

Using linear-scaling gradient techniques[64] the energy calculation (all calculations have been carried out in C_1 symmetry) required 4 hours and the corresponding gradient required 2.3 hours on a DEC PW 433au workstation. This clearly demonstrates that geometry optimizations for rather large systems are nowadays feasible. The important prerequisite is that the computational costs do not scale with the number of degrees of freedom.

As a second example, we discuss a quantum chemical study of the cubic and quartic force field of propadienylidene (C_3H_2) using analytic derivative techniques at the CC level[28,31]. Propadienylidene is the smallest stable cumulene carbene. It has been investigated using various spectroscopic techniques and attracted a lot of interest as *interstellar* molecule[65]. Computation of the cubic and quartic force

Table 3. Comparison of computed harmonic and fundamental frequencies for C_3H_2 with the experimental data from Ref. 66. All computational results have been obtained at the CCSD(T)/cc-pVTZ level.

	Harmonic Frequency (cm^{-1})	Infrared Intensity (km mol^{-1})	Fundamental Frequency (cm^{-1})	Experimental Fundamental (cm^{-1})
a_1 symmetry				
ν_1	3123	5	2997	3050–3060
ν_2	1998	250	1956	1952–1963
ν_3	1495	10	1458	1447–1449
ν_4	1119	2	1111	—
b_1 symmetry				
ν_5	217	3	211	—
b_2 symmetry				
ν_6	3212	0	3069	—
ν_7	1052	3	1034	—
ν_8	1020	19	996	999–1005
ν_9	275	114	287	—

fields were needed for a rigorous assignment of the matrix IR spectrum measured by Maier *et al.*[66] and for a theoretical determination of vibrational corrections to the rotational constants of C_3H_2. The latter are needed to deduce an equilibrium structure (r_e structure) for propadienylidene from the experimental vibrationally averaged rotational constants[67]. The calculation of the anharmonic force field was carried out by numerical differentiation of analytically evaluated force constants along the normal coordinates (for a detailed description, see Ref. 9). In this way, computation of anharmonic force fields is a rather *routine* matter, while the same cannot be stated for anharmonic force field calculations based on a numerical differentiation of energies. The computation of the semi-diagonal quartic force field at the CCSD(T)/cc-pVTZ level required about 45 days CPU time on a DEC PW 433au workstation. A single energy calculation requires less than half an hour, a gradient calculation about one hour and a complete second derivative calculations about 1.5 days.

The results from our calculations for C_3H_2 are summarized in Fig. 2 and Table 3. Fig. 2 compares the equilibrium geometry obtained from a least-squares fit to the experimental rotational constants corrected for vibrational effects (computed at the CCSD(T)/cc-pCVTZ level) with a structure determined at the CCSD(T)/cc-pCVQZ level (for further details, see Ref. 68). The agreement is excellent and differences amount to less than 0.001 Å. Table 3 compares the computed fundamental frequencies with those obtained in the matrix IR study[66]. Again the agreement is good, except that the calculation sheds some doubt on the assignment of the band at 3050-3060 cm^{-1} to the symmetric CH stretching mode. According to the calculations, an assigment of this band to the antisymmetric CH stretching mode

seems to be more plausible.

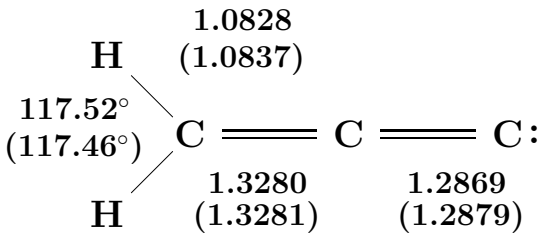


Fig. 2: Equilibrium structure (distances in Å, angles in degrees) of propadienylidene as obtained from an analysis of experimental rotational constants corrected with vibrational corrections obtained from a CCSD(T)/cc-pCVTZ anharmonic force field. Geometrical parameters obtained from a CCSD(T)/cc-pCVQZ optimization are given in parentheses.

3 Magnetic properties

Among the magnetic properties of a molecule, the parameters which characterize the NMR spectrum of a molecule are of particular interest to chemists. The quantum chemical calculation of NMR chemical shifts and also of indirect spin-spin coupling constants is of great importance, as the assignment of experimental NMR spectra is not straightforward. There is no simple relationship between chemical shifts and coupling constants on one hand and the structural parameters of interest on the other hand.

As the calculation of magnetic properties poses a few special problems, their computation is discussed in some detail in this section. Nevertheless, analytic derivative theory (as discussed in the previous sections) is the main prerequisite for the efficient computation of magnetic properties, as magnetazibilities are given as second derivatives of the energy with respect to the magnetic field, NMR chemical shifts as the corresponding mixed derivatives with respect to field and nuclear magnetic moments, and indirect spin-spin coupling constants as second derivative with respect to the corresponding nuclear spins (compare Table 1). In the following, the focus will be on the additional difficulties in the calculation of these properties. Our main focus is on NMR chemical shifts, but a few remarks are also given on the calculation of indirect spin-spin coupling constants.

3.1 Qualitative discussion

Before we discuss the actual computational aspects, it is appropriate to give a qualitative picture of the chemical shielding effect. Consider for this purpose a nucleus K with spin $\mathbf{I}_K$ in an external magnetic field $\mathbf{B}$. The spin is associated with a magnetic moment $\mathbf{m}_K$ according to

$$\mathbf{m}_K = \gamma_K \mathbf{I}_K \tag{48}$$

with γ_K as the gyromagnetic ratio. The magnetic moment $\mathbf{m}_K$ directly interacts with the external field leading to an energy correction of the form

$$\Delta E = -\mathbf{m}_K \cdot \mathbf{B}. \tag{49}$$

If this were the only contribution, NMR spectroscopy would not be able to distinguish between nuclei in different chemical environments.

The phenomenon of chemical shifts can only be understood if the electronic motion is explicitly considered. In particular, one has to take into account that the external magnetic field induces an electronic current[f] and that according to Biot-Savart's law[69] each current generates a magnetic field. As the induced field is proportional to the current and the current proportional to the external field, the induced magnetic field is in first order proportional to the external field. At the position of the nucleus K, the induced field can thus be written as

$$\mathbf{B}_{ind} = -\boldsymbol{\sigma}_K \mathbf{B} \tag{50}$$

with $\boldsymbol{\sigma}_K$ as the chemical shielding tensor. Note that $\boldsymbol{\sigma}_K$ is a tensor, as $\mathbf{B}_{ind}$ is not necessarily parallel or antiparallel to $\mathbf{B}$.

The total magnetic field at the nucleus K is given as the sum of external and internal field

$$\mathbf{B}_{local} = \mathbf{B} + \mathbf{B}_{ind} \tag{51}$$

and the energy correction and the splitting of the energy levels in the presence of an external magnetic field $\mathbf{B}$ is then

$$\Delta E = -\mathbf{m}_K \cdot \mathbf{B}_{local} \tag{52}$$

$$-\mathbf{m}_K(1 - \boldsymbol{\sigma}_K)\mathbf{B}. \tag{53}$$

The latter expression is identical to the shielding term in the effective Hamiltonian H_{NMR} often used in NMR spectroscopy[70]. The shielding term is supplemented in H_{NMR} by a second term which describes the coupling of the spins and thus is responsible for the multiplet structure in NMR spectra[70].

While the effective NMR Hamiltonian is often used to simulate spectra or to deduce the relevant NMR parameters from experimental spectra, the quantum chemical task is different. There, it is the goal to compute the relevant NMR parameters, i.e., the shielding tensors and the spin-spin coupling constants, for a given molecule without any further information.

To complete the introductory discussion, it should be noted that in most cases isotropic shieldings

$$\sigma_K = \frac{1}{3} Tr(\boldsymbol{\sigma}_K) \tag{54}$$

as well as relative shifts

$$\delta = \sigma_{ref} - \sigma_K \tag{55}$$

are reported. The relative shifts δ are given with respect to the shielding σ_{ref} of a reference compound (for example, TMS in case of ^{1}H and ^{13}C). Furthermore, as the chemical shielding is a rather small effect (which, however, can be very precisely measured), the dimensionless shielding constants are usually given in ppm.

[f] Note there is no electronic current in the field-free case.

3.2 Shieldings as second derivatives of the energy

Following the basic physical explanation of the chemical shielding effect, it is obvious that the shielding tensor is an electronic property that should be available through quantum chemical calculations. From Eq. (53), it is clear that the shielding tensor is a second-order response property with magnetic field and nuclear magnetic moment as the corresponding perturbations. Accordingly, the shielding tensor can be *computationally* obtained as the following second derivative of the electronic energy

$$\sigma_{ij}^{K} = \left(\frac{d^2 E}{dB_j dm_{Ki}} \right)_{\mathbf{B},\mathbf{m}_K=0}, \tag{56}$$

or in other words, the energy correction given in Eq. (53) can be considered as the corresponding *quadratic* term in a Taylor expansion of the molecular energy with respect to $\mathbf{B}$ and $\mathbf{m}_K$.

$$E(\mathbf{B},\mathbf{m}_K) = E(0) + \sum_j \sum_i \left(\frac{d^2 E}{dB_j dm_{Ki}} \right)_{\mathbf{B},\mathbf{m}_K=0} B_j \; m_{Ki} + \ldots \tag{57}$$

3.3 Molecular Hamiltonian in the presence of a magnetic field

For all property calculations, it is essential to specify first the corresponding perturbed Hamiltonian. While this is a rather straightforward task for most properties, some difficulties arise in case of the magnetic field.

In the presence of an external magnetic field, the momentum operator $\mathbf{p}$ in the electronic Hamiltonian[g]

$$H = \frac{\mathbf{p}^2}{2m} + V(\mathbf{r}) \tag{58}$$

needs to be replaced by its mechanical counterpart

$$\mathbf{p} \longrightarrow \boldsymbol{\pi} = \mathbf{p} + \frac{e}{c}\mathbf{A}(\mathbf{r}). \tag{59}$$

In Eq. (59), e denotes the elementary charge (the charge of the electron is $-e$), c the speed of light, and $\mathbf{A}$ the vector potential, which together with the scalar potential $\phi(\mathbf{r})$ represents the fundamental quantity for a theoretical description of electromagnetic field[71]. Note that the Gaussian unit system[72] is used throughout.

The magnetic field is uniquely determined via

$$\mathbf{B} = \boldsymbol{\nabla} \times \mathbf{A} \tag{60}$$

from the vector potential $\mathbf{A}$. It is obvious from Eq. (60) that the vector potential suffices to specify the magnetic interactions in the Hamiltonian. Nevertheless, one should note that $\mathbf{B}$ and *not* the potential $\mathbf{A}$ is the observable quantity. We later return to this aspect.

[g] For the current discussion it is sufficient to consider a one-electron system, as generalization to many-electron systems is straightforward.

Carrying out the substitution given by Eq. (59), one obtains for the Hamiltonian of an electron in an external magnetic field

$$H = -\frac{(\mathbf{p} - \frac{e}{c}\mathbf{A})^2}{2m} + V(\mathbf{r}). \tag{61}$$

For our purpose, it is sufficient to consider static magnetic fields of the form

$$\mathbf{B} = \text{const} \tag{62}$$

which are adequately described by the vector potential

$$\mathbf{A} = \frac{1}{2}\mathbf{B} \times \mathbf{r}. \tag{63}$$

As the vector potential given in Eq. (63) satisfies the (Coulomb gauge) condition

$$\boldsymbol{\nabla} \cdot \mathbf{A} = 0, \tag{64}$$

we can rewrite the Hamiltonian in Eq. (61) as

$$H = \frac{\mathbf{p}^2}{2m} + \frac{e}{mc}\mathbf{A} \cdot \mathbf{p} + \frac{e^2}{2mc^2}\mathbf{A}^2 + V(\mathbf{r}). \tag{65}$$

Note that $\mathbf{p}$ and not $\boldsymbol{\pi}$ is the canonical conjugate momentum to the position $\mathbf{r}$, so that the operator $\mathbf{p}$ is given by $-i\hbar\boldsymbol{\nabla}$ in the position representation.

Insertion of the explicit form of the vector potential (Eq. (63)) into Eq. (65) then yields for the Hamiltonian

$$H = -\frac{\hbar^2}{2m}\boldsymbol{\nabla}^2 - \frac{ie\hbar}{2mc}\mathbf{B} \cdot (\mathbf{r} \times \boldsymbol{\nabla}) + \frac{e^2}{8mc^2}[(\mathbf{B} \cdot \mathbf{B})(\mathbf{r} \cdot \mathbf{r}) - (\mathbf{B} \cdot \mathbf{r})(\mathbf{B} \cdot \mathbf{r})] + V(\mathbf{r}) \tag{66}$$

and thus the following expression for the first and second derivative of the Hamiltonian with respect to B_i

$$\frac{\partial h}{\partial B_i} = -\frac{ie\hbar}{2mc}(\mathbf{r} \times \boldsymbol{\nabla})_i \tag{67}$$

$$\frac{\partial^2 h}{\partial B_i \partial B_j} = \frac{e^2}{4mc^2}[(\mathbf{r} \cdot \mathbf{r})\delta_{ij} - r_i r_j]. \tag{68}$$

Considering in addition the vector potential due to the magnetic moments

$$\mathbf{A}' = \frac{\mathbf{m}_K \times (\mathbf{r} - \mathbf{R}_K)}{|\mathbf{r} - \mathbf{R}_K|^3} \tag{69}$$

the corresponding derivative of H with respect to m_K is given by

$$\frac{\partial h}{\partial m_{Kj}} = -\frac{ie\hbar}{mc}\frac{[(\mathbf{r} - \mathbf{R}_K) \times \nabla]_j}{|\mathbf{r} - \mathbf{R}_K|^3} \tag{70}$$

and for the mixed derivative with respect to B_i and m_K the following expression is obtained

$$\frac{\partial^2 h}{\partial B_i \partial m_{Kj}} = \frac{e^2}{2mc^2}\frac{\mathbf{r} \cdot (\mathbf{r} - \mathbf{R}_K)\delta_{ij} - r_j(\mathbf{r} - \mathbf{R}_K)_i}{|\mathbf{r} - \mathbf{R}_K|^3}. \tag{71}$$

From Eqs. (67) and (70), it is clear that the matrix elements of $\partial h/\partial B_i$ and $\partial h/\partial m_N$ are imaginary if the usual real basis functions are employed. This explains why a finite-field calculation of magnetic properties requires the capability of dealing with complex wavefunction parameters. On the other hand, in case of an analytic calculation all matrix elements are evaluated in the *zero-field limit* and thus are either real or purely imaginary. Real arithmetic is then sufficient, as i can be factored out from all imaginary quantities. The only difficulty arises due to the fact that matrices which are symmetric for "real" perturbations

$$\langle\mu|\frac{\partial h}{\partial x}|\nu\rangle = \langle\nu|\frac{\partial h}{\partial x}|\mu\rangle \tag{72}$$

are antisymmetric for "imaginary" perturbations such as the magnetic field

$$\langle\mu|\frac{\partial h}{\partial B_i}|\nu\rangle = (\langle\nu|\frac{\partial h}{\partial B_i}|\mu\rangle)^* = -\langle\nu|\frac{\partial h}{\partial B_i}|\mu\rangle \tag{73}$$

3.4 Gauge-origin dependence in the calculation of magnetic properties

The problem of gauge-invariance (or gauge-origin independence) in the calculation of magnetic properties arises because of the use of the vector potential $\mathbf{A}$ for the description of the magnetic interactions in the Hamiltonian. While $\mathbf{A}$ uniquely defines the magnetic field via Eq. (60), there is no unique choice of $\mathbf{A}$ to a given magnetic field $\mathbf{B}$. The reason is the $\boldsymbol{\nabla}$ operator in Eq. (60) which allows to add the gradient of an arbitrary scalar function to $\mathbf{A}$ without changing the corresponding magnetic field. For a static homogeneous field, for example, the vector potential can be chosen in the general form

$$\mathbf{A} = \frac{1}{2}\mathbf{B} \times (\mathbf{r} - \mathbf{R}_O) \tag{74}$$

with $\mathbf{R}_O$ as an *arbitrary* parameter, the so-called gauge origin.

The freedom in the choice of $\mathbf{A}$, i.e. in fixing the *gauge*, has in principle no consequences, as $\mathbf{B}$ as observable quantity is uniquely defined. However, one should note that the freedom in choosing the gauge for a given $\mathbf{B}$ introduces an arbitrary parameter, namely the gauge-origin $\mathbf{R}_O$, into the Hamiltonian. The question is now how this arbitrary parameter $\mathbf{R}_O$ affects the computation of magnetic properties.

The basic physical laws require that values for observable quantities must be independent of the chosen gauge or gauge-origin. This statement is known as the *principle of gauge-invariance* (or in our special case as *gauge-origin independence*). It is an obvious statement, as Hamiltonians differing only in $\mathbf{R}_O$ describe the *same* physical system with the *same* physical properties. The values for the latter therefore cannot depend on $\mathbf{R}_O$ which is exactly what the principle of gauge invariance states. However, it should be noted that gauge invariance is only requested for observable properties and not for non-observable quantities such as, for example, the wavefunction. For the latter gauge invariance cannot be enforced.

It is obvious (and straightforward to demonstrate, see, for example, Ref. 73) that the exact solution to the Schrödinger equation satisfies the requirement of gauge invariance. All properties computed from the exact solution are uniquely defined and, as it should be, independent of the gauge origin $\mathbf{R}_O$. It is also obvious

Table 4. ^{1}H shielding constants (in ppm) for hydrogen fluoride calculated at the HF-SCF level using different basis sets and gauge origins.

	gauge origin		
basis set	center of mass	fluorine	hydrogen
dz+d	29.3	27.6	60.1
tz+d	28.4	27.2	50.8
qz+2d	27.7	27.0	40.4

(though it has to be discussed for each case separately) that gauge invariance is not necessarily ensured for approximate solutions of the Schrödinger equation. This is a major problem for quantum chemistry, as there we deal almost exclusively with such solutions.

Indeed, it is easily shown that none of the routinely applied quantum chemical approaches provides gauge-invariant results for magnetic properties. The origin of this deficiency is easily traced back to the finite-basis set representation used for the molecular orbitals (commonly known as LCAO approximation). To illustrate the problem, Table 4 contains results for the ^{1}H shielding constant in hydrogen fluoride computed at the HF-SCF level using different basis sets and different gauge origins. The results clearly differ for the three gauge origins. Furthermore, it is observed that the largest discrepancies appear for the smaller and more incomplete basis sets. In passing, we note that *exact* HF calculations would provide gauge-origin independent results, as the problem is the basis-set expansion and not the HF approximation.

The main problem with the gauge-dependence of the computational results is that they are no longer uniquely defined. The computed values for magnetic properties depend on a parameter (or parameters) which can be chosen in an arbitrary manner. This also means that results could even become meaningless, when, for example, computed chemical shifts for symmetry-equivalent nuclei are predicted to be different. Clearly, this is an artifact of the calculation.

However, the gauge-dependence problem is more involved, as one would expect from the discussion given so far. Naively, one would assume that unique results can be achieved by simply fixing the gauge origin, for example, to the center of mass of the considered molecule. Of course, such a choice would guarantee unique results, but on the other hand it does not resolve the fundamental problem connected with the gauge problem in the computation of magnetic properties.

A second serious problem in the calculation of magnetic properties is the slow basis set convergence of the results. Fig. 3 displays the convergence for the ^{1}H shielding in case of hydrogen fluoride. It is seen that rather large basis sets are required to obtain reliable results. Convergence to the basis set limit is much slower than in the computation of most other molecular properties. In particular, standard basis sets appear insufficient for the reliable calculation of magnetic properties.

For atoms, the nuclear position represents a natural gauge origin. With this choice, no problem in the finite-basis representation of the magnetic interaction appears, as the usually employed basis functions, i.e., spherical Gaussians centered

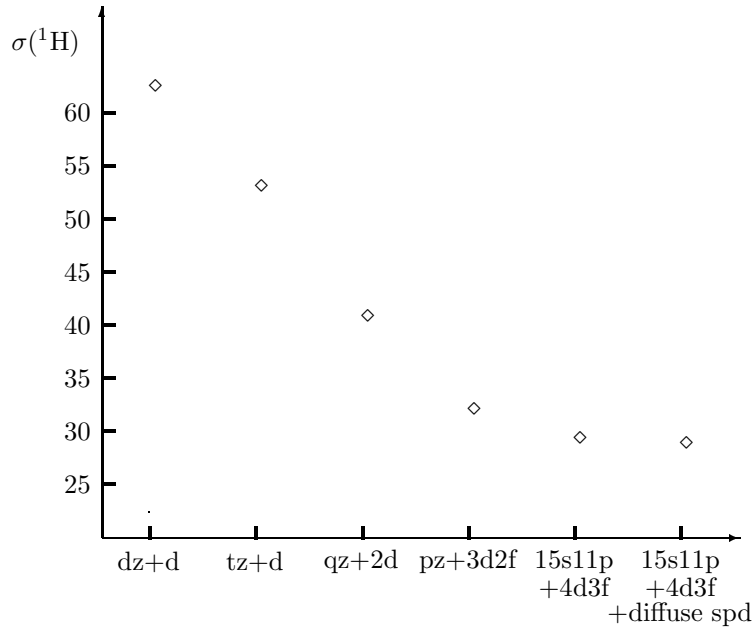


Fig. 3: Basis set convergence in the calculation of the ^{1}H shielding constant of hydrogen fluoride. All calculations have been carried out with the gauge origin at the hydrogen.

at the nuclear position, are eigenfunctions to the angular momentum operator

$$\mathbf{l} = \mathbf{r} \times \mathbf{p}. \tag{75}$$

The same is not the case as soon as the gauge origin is shifted and no longer coincides with the nuclear position. The angular momentum is then defined with respect to the displaced gauge origin

$$\mathbf{l}_O = (\mathbf{r} - \mathbf{R}_O) \times \mathbf{p} \tag{76}$$

and the following holds

$$\mathbf{l}_O|\chi_\mu\rangle = \mathbf{l}|\chi_\mu\rangle + \mathbf{R}_O \times \mathbf{p}|\chi_\mu\rangle \tag{77}$$

when $\mathbf{l}_O$ is applied to a basis function $|\chi_\mu\rangle$. The second term in Eq. (77) involves a simple differentiation of a Gaussian basis function and thus leads to basis functions with by one increased and decreased angular momentum quantum numbers. A proper basis set representation of $\mathbf{l}_O|\chi_\mu\rangle$ thus requires that additional higher angular momentum functions are included in the basis set, i.e. the corresponding p function for a s function, the corresponding d function for a p function, etc. A correct representation of this term would require a complete AO basis which is usually not provided. Shifting the gauge origin away from the nucleus thus deteriorates the

description of the magnetic interactions. The deteriorations increases with the size of the displacement.

3.5 Local gauge origin methods

From the discussion above, it is clear why the nucleus is the best gauge origin for atoms. However, a similar *natural* gauge origin does not exist for molecules. It is also obvious from the discussion above that a single gauge origin must be unsatisfactory for molecular calculations. The quality of the description for the various parts of a molecule depends on their distances to the gauge origin. As a single gauge origin can impossibly be optimal for all molecular parts, a rather unbalanced description of the magnetic interactions results.

The only viable way to overcome these problems consists in the use of more than one gauge origin for the external magnetic field. The idea behind this concept of *local* or *distributed gauge origins* is simple, though the technical realization is more involved. To introduce local gauge origins, it is first necessary to partition the molecule (or to be more specific the molecular wavefunction) into *local* fragments. For each fragment, the gauge origin can be then individually chosen in an optimal way. If the local fragments can be assigned to the various atoms in the molecule, the corresponding nuclear position would be, for example, a good choice for the gauge origin.

As within the concept of local gauge origins all molecular parts are described equally well, a balanced description is achieved. A "good" description of the magnetic interactions is ensured, as for each local fragment an optimal choice for the gauge origin is possible.

For a realization of the described concept of local gauge origins, it is necessary to introduce gauge-transformations. A shift of the gauge origin from $\mathbf{R}_O$ to $\mathbf{R}'_O$ is achieved in the theoretical description via a so-called gauge transformation. For a one-electron system, the corresponding equations are

$$\Psi \longrightarrow \Psi' = \exp(-\Lambda(\mathbf{r}))\ \Psi \tag{78}$$

$$\hat{H} \longrightarrow \hat{H}' = \exp(-\Lambda(\mathbf{r}))\ \hat{H}\ \exp(\Lambda(\mathbf{r})) \tag{79}$$

with the gauge factor defined by

$$\Lambda(\mathbf{r}) = \frac{ie}{2c\hbar}\ [(\mathbf{R}'_O - \mathbf{R}_O) \times \mathbf{B}] \cdot \mathbf{r}. \tag{80}$$

Expansion of the Hamiltonian H' using the Hausdorff formula

$$H' = H + [H, \Lambda] + \frac{1}{2}[[H, \Lambda], \Lambda] + ... \tag{81}$$

$$= H - \frac{ie\hbar}{2mc}\mathbf{B} \cdot ((\mathbf{R}_O - \mathbf{R}'_O) \times \boldsymbol{\nabla}) + ... \tag{82}$$

$$= -\frac{\hbar^2}{2m}\boldsymbol{\nabla}^2 + V(\mathbf{r}) - \frac{ie\hbar}{2mc}\mathbf{B} \cdot ((\mathbf{r} - \mathbf{R}'_O) \times \boldsymbol{\nabla}) + ... \tag{83}$$

shows that H' is indeed the Hamiltonian with the gauge origin at $\mathbf{R}'_O$ instead of $\mathbf{R}_O$.

To introduce local gauge origins, the concept of gauge transformations needs to be extended, as so far we are only concerned with the change from a given origin to another gauge origin. This extension is easily achieved by introducing more general gauge transformations of the following kind

$$\sum_A \exp(\Lambda_A(\mathbf{r}))\ \hat{P}_A \tag{84}$$

where $\hat{P}_A$ represents a suitable projector on the local fragment A and $\exp(\Lambda_A(\mathbf{r}))$ a gauge transformation to the origin chosen for A. Note that it can be shown that Eq. (84) defines indeed a valid gauge transformation and leaves for the exact solution of the Schrödinger equation all physical observables unchanged. As usual, we have to require for the projector that

$$\sum_A \hat{P}_A \ = \ 1. \tag{85}$$

One should also note that for a many-electron system the corresponding gauge transformation is given by

$$\prod_{\alpha=1}^{N} \exp(\Lambda_A(\mathbf{r}_\alpha))\hat{P}_A^\alpha \tag{86}$$

where α is the electron index and $\hat{P}_A^\alpha$ the corresponding projector.

Following Eq. (84), it is most convenient to define the local fragment at the one-electron level. With the atomic and molecular orbitals as the *one-electron* building blocks for the wavefunction, it appears natural to introduce individual gauge origins for these one-electron functions. The corresponding schemes, i.e., IGLO[74,75], LORG[76], and GIAO[77−80] are nowadays well established and routinely used for NMR chemical shift calculations[h]. In fact, it is justified to state that these schemes first enabled those calculations to be routinely carried out within chemical applications, as the trivial solution to the gauge problem, i.e., the use of very large basis sets, is necessarily restricted to small molecules.

IGLO and LORG are both based on individual gauges for molecular orbitals. However, as standard HF orbitals are usually delocalized, they are not well suited for a local gauge origin approach. It is therefore mandatory to introduce localized occupied orbitals[81] and to define individual gauges for them. This is exactly what is done in the individual gauges for localized orbital (IGLO) approach of Kutzelnigg and Schindler[74,75] and the localized orbital/local origin (LORG) scheme of Bouman and Hansen[76]. Details for both approaches can be found in the original literature.

Considering the need of localized orbitals as disadvantageous, it appears more natural to work with atomic orbitals (which are by construction localized quantities) and to assign to each of them an individual gauge origin. No further manipulations are required in this case. This choice of local gauge origins leads to what is nowadays known as the gauge-including atomic orbital (GIAO) approach[i]. As it seems

[h] It should be noted that the IGLO and GIAO approaches are also well suited for the efficient computation of magnetazibilities.

[i] The GIAO method is sometimes also called the London atomic orbital (LAO) approach.

that the GIAO approach is becoming the *standard* for the calculation of magnetic properties, we will discuss it in the following section in some detail.

3.6 The GIAO ansatz

The GIAO ansatz[77–80] consists in choosing local gauge origins for atomic orbitals $|\chi_\mu\rangle$. This means in particular that the corresponding nucleus at which the atomic orbital is centered is chosen as the "natural" gauge origin.

Within the concept of local gauge origins, the GIAO ansatz is based on the following gauge-transformation

$$\sum_\mu \exp(\Lambda_\mu(\mathbf{r}))\ \hat{P}_\mu \tag{87}$$

with the projector on $|\chi_\mu\rangle$ defined by

$$\hat{P}_\mu = \sum_\nu |\chi_\mu\rangle S^{-1}_{\mu\nu} \langle\chi_\nu| \tag{88}$$

and the gauge factor given by

$$\Lambda_\mu(\mathbf{r}) = \frac{ie}{2c\hbar}\ [(\mathbf{R}_\mu - \mathbf{R}_O) \times \mathbf{B}] \cdot \mathbf{r}. \tag{89}$$

with $\mathbf{R}_O$ as the original gauge origin in the Hamiltonian and $\mathbf{R}_\mu$ as the center of the basis function $|\chi_\mu\rangle$. It is obvious that the gauge factor given in Eq. (89) describes a shift of the gauge origin from $\mathbf{R}_O$ to $\mathbf{R}_\mu$. The projector in Eq. (88) is somewhat more involved in order to account for the non-orthogonality of the AOs.

While Eqs. (87) to (89) yield in our opinion a very elegant description of the GIAO approach, a different, though in principle equivalent description is more common.[80] Consider the equations which define the gauge transformations (Eqs. (78) and (79)). These equations can be interpreted such that there is some freedom whether the phase factors (of the gauge transformation) are assigned to the Hamiltonian or to the wavefunction. Eqs. (78) and (79) thus offer two possibilities for the interpretation of the GIAO approach. The first is to apply the local transformation and to work with a modified Hamiltonian but with the same unchanged wavefunction. This means that the latter is described in the usual way with the standard atomic orbitals, etc. The second possibility leaves the Hamiltonian unchanged, but attaches additional phase factors (those from the gauge transformation) to the wavefunction description. In particular, it appears convenient to attach these additional phase factors to the AOs and to describe the GIAO ansatz such that the calculation of magnetic properties are now carried out with the following perturbation-dependent basis functions

$$|\chi_\mu(\mathbf{B})\rangle = \exp(-\frac{ie}{2c\hbar}(\mathbf{B} \times [\mathbf{R}_\mu - \mathbf{R_O}]) \cdot \mathbf{r})|\chi_\mu(0)\rangle \tag{90}$$

instead of the usual field-independent functions $|\chi_\mu(0)\rangle$. The field-dependent basis functions of Eq. (90) have been termed in the literature as *gauge-including atomic*

orbitals[j] or *London orbitals*.

We will discuss now in some detail why GIAOs provide a satisfactory solution to the gauge-origin problem. Let us take first the *local gauge origin* view, i.e., the description of the GIAO ansatz as an approach where local gauge origins are introduced for AOs. The gauge-origin problem is *solved* here by fixing the gauge in an optimal way. This first provides unique results and second ensures fast basis set convergence, as for each AO the corresponding optimal gauge is used. However, one should avoid the term *gauge invariance* in this context, as the gauge problem is only solved by fixing the gauge in a very special way. However, the results are independent of the original gauge (characterized by the common gauge origin $\mathbf{R}_O$) and in that respect it is appropriate to talk about gauge-origin independent results in the sense that the results for magnetic properties are independent of $\mathbf{R}_O$.

Let us take the second view, i.e., that of perturbation-dependent basis functions. Here, the use of special basis functions ensures proper behavior of the wavefunction in the presence of the magnetic field. Gauge-origin independent results are obtained, as these basis functions are especially tailored to the chosen gauge (the gauge origin $\mathbf{R}_O$ appears in the definition of the GIAOs). Gauge invariance is again not ensured, as invariance is only guaranteed with respect to shifts in the gauge origin $\mathbf{R}_O$, but not with respect to more general gauge transformations. Fast basis set convergence is achieved, because the GIAOs are constructed in such a manner that they provide exact first-order solutions in the presence of the magnetic field provided the corresponding AO is the correct zeroth-order solution[82]. One can argue that the GIAOs already take care of the major effect of the magnetic field perturbation on the wavefunction and that the remaining corrections in MO coefficients, CI coefficients or CC amplitudes are rather small and easily described within the standard techniques.

The fast basis set convergence of GIAO calculations of shielding constants is demonstrated in Fig.4.

We add a discussion on some technical aspects which are best explained in the picture of field-dependent basis functions. As the basis functions now explicitly depend on the magnetic field $\mathbf{B}$, differentiation of the usual unperturbed one- and two-electron integrals necessarily involves additional terms. While a conventional approach just requires the integrals

$$\left(\frac{\partial\langle\chi_\mu|h|\chi_\nu\rangle}{\partial B_i}\right)_{conv} = -\frac{ie\hbar}{2mc}\langle\chi_\mu|(\mathbf{r}\times\boldsymbol{\nabla})_i|\chi_\nu\rangle \tag{91}$$

$$\left(\frac{\partial\langle\chi_\mu|h|\chi_\nu\rangle}{\partial m_{Ki}}\right)_{conv} = -\frac{ie\hbar}{mc}\langle\chi_\mu|\frac{[(\mathbf{r}-\mathbf{R}_K)\times\boldsymbol{\nabla}]_i}{|\mathbf{r}-\mathbf{R}_K|^3}|\chi_\nu\rangle \tag{92}$$

$$\left(\frac{\partial^2\langle\chi_\mu|h|\chi_\nu\rangle}{\partial B_i\partial m_{Kj}}\right)_{conv} = \frac{e^2}{2mc^2}\langle\chi_\mu|\frac{\mathbf{r}\cdot(\mathbf{r}-\mathbf{R}_K)\delta_{ij}-\mathbf{r}_i(\mathbf{r}-\mathbf{R}_K)_j}{|\mathbf{r}-\mathbf{R}_K|^3}|\chi_\nu\rangle \tag{93}$$

[j] It should be noted that the GIAOs were originally named in a somewhat misleading manner as gauge-invariant or gauge-independent atomic orbitals. Following a suggestion by Bouman and Hansen[76] the more appropriate name *gauge-including atomic orbitals* has become standard since the beginning of the nineties.

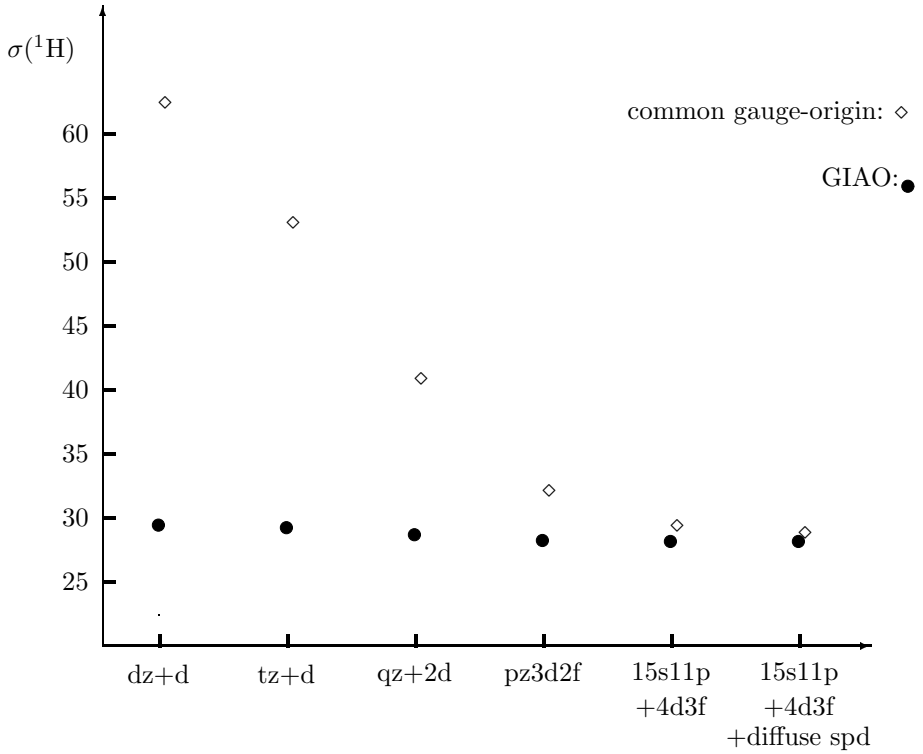


Fig. 4: Comparison of the basis set convergence in common gauge-origin and GIAO calculation of the ^{1}H shielding constant of hydrogen fluoride.

the dependence of the basis functions on B leads to following additional integrals.

$$\left(\frac{\partial\langle\chi_\mu|\chi_\nu\rangle}{\partial B_i}\right)_{GIAO} = \langle\frac{\partial\chi_\mu}{\partial B_i}|\chi_\nu\rangle + \langle\chi_\mu|\frac{\partial\chi_\nu}{\partial B_i}\rangle \tag{94}$$

$$\left(\frac{\partial\langle\chi_\mu|h|\chi_\nu\rangle}{\partial B_i}\right)_{GIAO} = \left(\frac{\partial\langle\chi_\mu|h|\chi_\nu\rangle}{\partial B_i}\right)_{conv} + \langle\frac{\partial\chi_\mu}{\partial B_i}|h|\chi_\nu\rangle + \langle\chi_\mu|h|\frac{\partial\chi_\nu}{\partial B_i}\rangle \tag{95}$$

$$\left(\frac{\partial\langle\chi_\mu|h|\chi_\nu\rangle}{\partial m_{Ki}}\right)_{GIAO} = \left(\frac{\partial\langle\chi_\mu|h|\chi_\nu\rangle}{\partial m_{Ki}}\right)_{conv} \tag{96}$$

$$\left(\frac{\partial^2\langle\chi_\mu|h|\chi_\nu\rangle}{\partial B_i\partial m_{Kj}}\right)_{GIAO} = \left(\frac{\partial^2\langle\chi_\mu|h|\chi_\nu\rangle}{\partial B_i\partial m_{Kj}}\right)_{conv} - \frac{ie\hbar}{mc}\langle\frac{\partial\chi_\mu}{\partial B_i}|\frac{[(\mathbf{r}-\mathbf{R}_K)\times\boldsymbol{\nabla}]_j}{|\mathbf{r}-\mathbf{R}_K|^3}|\chi_\nu\rangle - \frac{ie\hbar}{mc}\langle\chi_\mu|\frac{[(\mathbf{r}-\mathbf{R}_K)\times\boldsymbol{\nabla}]_j}{|\mathbf{r}-\mathbf{R}_K|^3}|\frac{\partial\chi_\nu}{\partial B_i}\rangle \tag{97}$$

$$\left(\frac{\partial\langle\chi_\mu\chi_\nu|\chi_\sigma\chi_\rho\rangle}{\partial B_i}\right)_{GIAO} = \langle\frac{\partial\chi_\mu}{\partial B_i}\chi_\nu|\chi_\sigma\chi_\rho\rangle + \langle\chi_\mu\frac{\partial\chi_\nu}{\partial B_i}|\chi_\sigma\chi_\rho\rangle + \langle\chi_\mu\chi_\nu|\frac{\partial\chi_\sigma}{\partial B_i}\chi_\rho\rangle + \langle\chi_\mu\chi_\nu|chi_\sigma\frac{\partial\chi_\rho}{\partial B_i}\rangle \tag{98}$$

For a long time, efficient calculation and handling of these integrals (and here in particular of the additional two-electron integrals) was considered a major obstacle

in the application of the GIAO approach. However, Pulay pointed out that the additional integrals are closely related to usual geometrical integral derivatives[83] and showed how modern analytic derivative techniques can be used to design an efficient GIAO code at the SCF level[80].

3.7 Availability of methods for the calculation of NMR chemical shifts

Before discussing available methods for the efficient calculation of NMR chemical shifts, let us add a few historical remarks. The suggestion of using gauge-including atomic orbitals goes back to London[77] who has used them in a study of molecular diamagnetism more than 50 years ago. At the end of the fifties, GIAOs were used by Hameka in SCF calculations of shielding constants for a few diatomic molecules[78] and in the seventies by Ditchfield[79]. However due to technical problems in calculating and handling integral derivatives, Ditchfield's work had only limited impact. The breakthrough with respect to applicability was the IGLO development by Kutzelnigg and Schindler[74,75]. By introducing local gauges for localized orbitals, the problem of additional two-electron integrals was avoided in a rather elegant way and calculations of chemical shifts for larger and chemically more interesting molecules became possible. Noteworthy are in particular the numerous applications of the IGLO approach to problems in carbocation chemistry by Schleyer and coworkers[84]. The LORG approach was suggested somewhat later within the random-phase approximation (RPA) context, but has been shown by Kutzelnigg[73] to be closely related to IGLO. The popularity of the GIAO approach in the nineties started with Pulay's seminal work[80]. As the GIAO approach can be considered the most elegant way to deal with the gauge problem, it has been adopted by most groups and implemented in many quantum chemical program packages for the calculation of magnetic properties.

Table 5 lists the currently available schemes for the calculation of NMR chemical shifts. Note that we include only those approaches which take care of the gauge-origin problem via local gauge-origin methods and ignore all other developments, as they cannot be considered well suited for most chemical applications. At the HF-SCF level, three variants are available: GIAO-SCF[79,80], IGLO[74,75], and LORG[76] and all three schemes yield results of similar quality. However, implementations at electron-correlated levels were mainly pursued using GIAOs[85–91,29]. GIAO-MP2 (also known as GIAO-MBPT(2)) provides the largest range of applicability, GIAO-CCSD(T) is a tool for highly accurate prediction of NMR chemical shifts, while the complementary GIAO-MCSCF treatment allows to tackle difficult cases with large static correlation effects. On the other hand, IGLO was generalized to MCSCF wavefunction models (MC-IGLO)[92] and the LORG scheme to second-order LORG (SOLO)[93].

Naturally, there has been also a great interest in calculating NMR chemical shifts using DFT approaches. From a pragmatic point of view, very promising results have so far been obtained[94–102], but it is necessary to add a few remarks. First, it needs to be realized that the Hohenberg-Kohn theorems do not hold in the presence of a magnetic field. They need to be extended (as described by Vignale and Rasolt[103]) to include current-dependent functionals. Most implementations,

Table 5. Available implementations for the calculations of NMR chemical shifts

quantum chemical methods		Ref.
IGLO	Kutzelnigg and Schindler (1982)	75
LORG	Bouman and Hansen (1985)	76
GIAO-SCF	Ditchfield (1974), Wolinski,Pulay, Hinton (1990)	79,80
MC-IGLO	van Wüllen and Kutzelnigg (1993)	92
GIAO-MCSCF	Ruud *et al.* (1994)	87
GIAO-MP2	Gauss (1992)	85,86
GIAO-MP3	Gauss (1994)	88
GIAO-MP4	Gauss, Stanton (1994,1996)	88,91
GIAO-CCSD	Gauss, Stanton (1995)	89,90
GIAO-CCSD(T)	Gauss, Stanton (1996)	91
GIAO-CCSDT-n	Gauss, Stanton (2000)	29
DFT-IGLO	Malkin *et al.* (1993)	94
SOS-DFPT	Malkin *et al.* (1994)	95
GIAO-DFT	Schreckenbach and Ziegler (1995),	96
	Handy *et al.* (1995),	99
	Pulay *et al.* (1996), Cheeseman *et al.* (1996)	97,98
SOLO	Bouman, Hansen (1990)	93

however, ignore that fact and treat magnetic properties by what is called uncoupled DFT. A justification for this choice might be seen in the work of Lee *et al.*[99] who showed that inclusion of the current does not necessarily improve the results[k]. Second, to rectify some problems in the uncoupled DFT approach (which certainly is incomplete from a formal point of view) Malkin *et al.* suggested an *ad hoc* modification of the energy denominator within their sum-over-states density-functional perturbation theory (SOS-DFPT) scheme[95]. Though the results give some justification for this empirical modification, it must be clearly stated that there is no rigorous theoretical justification for Malkin's approach and that a similar manipulation would not have been accepted within the more conventional approaches. Nevertheless, it can be anticipated that DFT treatments of NMR chemical shifts are of great practical importance and will become a routine tool within quantum chemistry.

We will complete our discussion by emphasizing some algorithmic developments in chemical shift calculations aiming at the treatment of larger systems. Implementation of direct methods for GIAO-SCF[105] or IGLO[106] paved the path for the routine treatment of molecular systems with up to 100 atoms. Integral-direct concepts have also been extended to the GIAO-MP2 approach[107]. Together with an efficient treatment of molecular point-group symmetry[108], thus GIAO-MP2 chemical shift calculations became possible on molecules with more than 50 atoms described by more than 600 basis functions (see also section 3.8). However, as the CPU requirements remained unchanged, it appears attractive to couple such integral-direct

[k]For a different concept for the treatment of magnetic properties within DFT, see the work by Harris *et al.*[104]

Table 6. Calculated absolute shielding constants (in ppm)

Molecule	Nucleus	σ_e	σ_0	σ(300K)	σ(300K, exp.)
H_2	^{1}H	26.667	26.312	26.298	26.288±0.002
HF	^{1}H	28.84	28.52	28.48	28.54±0.01
	^{19}F	419.6	409.6	409.2	409.6±1.0
CO	^{13}C	4.2	1.9	1.8	0.9±0.9
	^{17}O	-54.3	-60.4	-60.6	-44.8±17.2
N_2	^{15}N	-58.4	-62.5	-62.7	-61.6
F_2	^{19}F	-189.9	-220.8	-225.5	-233.02±1.0

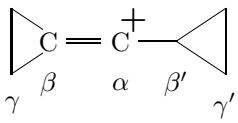
developments with other ideas such as, for example, the local-correlation treatment first suggested by Pulay[109] and recently pursued by Werner and co-workers.[110] Based on a preliminary implementation, test calculations have recently demonstrated that a local GIAO-MP2 (GIAO-LMP2) scheme should hold great promises for the treatment of larger molecules.[111]

Other active areas where methodological developments concerning chemical shift calculations are pursued are the treatment of relativistic effects (important for heavy elements)[112−116], the routine calculation of rovibrational effects (important to improve agreement between theory and experiment)[117−119] and the consideration of solvent effects[120−122]. The latter is of special importance, as the majority of the NMR spectra is measured in solution or the liquid phase.

3.8 Examples for chemical shift calculations

We will give a few examples to demonstrate the range of applications which are possible with the currently available methods for computing chemical shifts. Table 6 reports computed absolute shielding constants needed for establishing absolute NMR scales[117]. In order to provide accurate data, it is here essential to use the GIAO-CCSD(T) methods in combination with large basis sets. The error in the computed absolute shifts of any of the other methods is too large in order to provide reliable data. In addition, consideration of rovibrational and temperature effects is mandatory. To emphasize the importance of such calculations, we note that the current ^{17}O scale is based on calculations[117], as the corresponding experimental scale (based on measured spin-rotation constants) appears to be inaccurate.

For most chemical applications, computation of relative shifts is of central importance. In case of the following vinyl cation



it turned out that GIAO-CCSD(T) calculations were essential to provide an unequivocal assignment[123]. GIAO-HF-SCF and GIAO-MP2 calculations are too inaccurate and only a high-level correlation treatment provides sufficient accuracy. The computed spectra are displayed in Fig. 5:

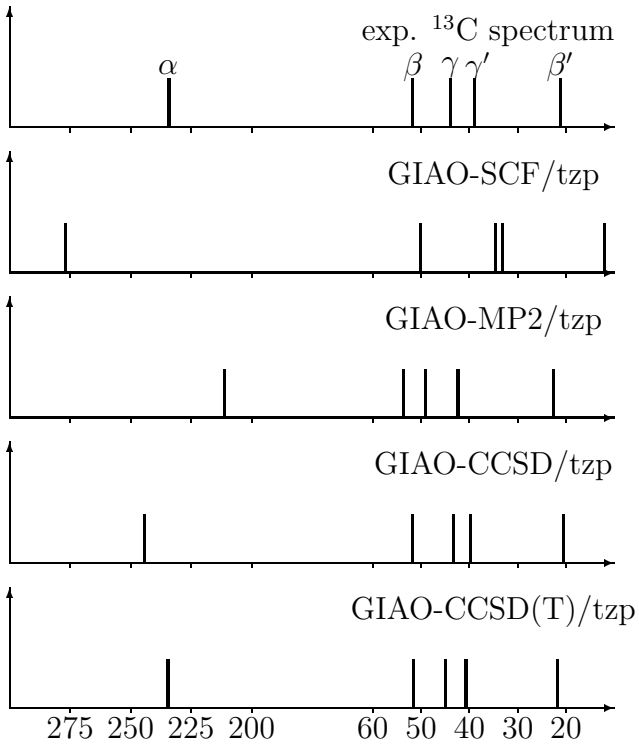


Fig. 5: Calculated and experimental ^{13}C NMR spectra for the 1-cyclopropylcyclopropylidenemethyl cation

Another example is the ^{27}Al NMR spectrum of the Al_4Cp_4 molecule (Fig. 6) measured by Schnöckel and co-workers[124].

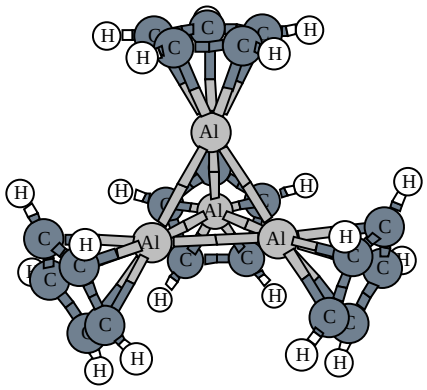
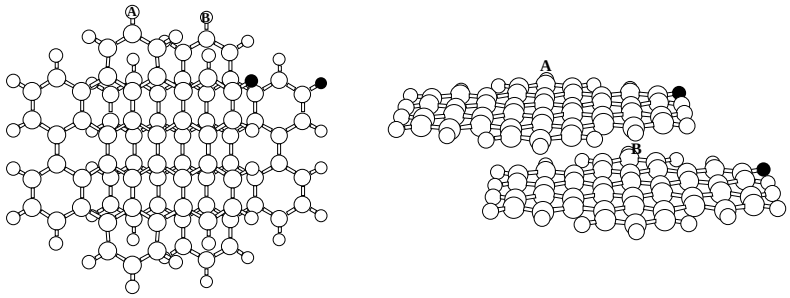


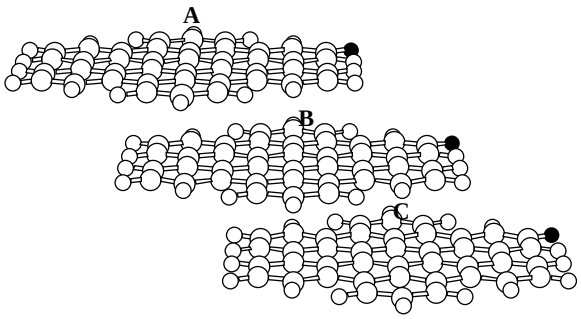
Fig. 6: Structure of Al_4Cp_4

The spectrum was unequivocally assigned to the tetrameric species on the basis of the computed ^{27}Al chemical shifts[124,108], as the experimental value of -111 ppm agrees well with the computed value of -108.5 (GIAO-MP2). Correlation effects amount in this case to about 15 ppm and, thus, are not negligible. The computational cost for the required GIAO-MP2 calculations have been rather demanding. Using a parallel version of the integral-direct program described in Ref. 108, the calculation (44 atoms, 192 electrons, 548 basis functions, molecular symmetry: D_{2d}) required about 6 days on a SGI Power Challenge (4 nodes).

A final example deals with an application of chemical shift calculations in combination with solid-state NMR measurements. The aromatic proton signal for a hexabenzocoronene (HBC) derivative splits in the solid state into three peaks (5.7, 6.9, and 8.3 ppm)[125]. GIAO-SCF calculations on dimeric and trimeric units of HBC (Figure 7) enabled an assignment of the measured ^{1}H NMR chemical shifts to a specific structural model by comparison with computed shifts for various model systems[126].



Computed ^{1}H-NMR shifts: 6.9 / 7.9 / 8.9 ppm



Computed ^{1}H-NMR shifts: 5.9 / 7.4 / 8.7 ppm

Fig. 7: Dimeric and trimeric units of HBC. The experimental chemical shifts for the aromatic protons in the investigated solid HBC derivative are 5.7, 6.9, and 8.3 ppm[125].

For the structure shown in Fig. 7, the best agreement is obtained between calculations and measured values.

3.9 Indirect spin-spin coupling constants

Besides the computation of NMR chemical shifts, theoretical prediction of indirect spin-spin coupling constants J_{KL} is equally important for a full understanding and assignment of experimental NMR spectra. The coupling constants are given as the corresponding second derivatives of the electronic energy with respect to the involved nuclear spins

$$J_{KL} = \frac{1}{3h} \sum_i \frac{d^2E}{dI_{Ki} dI_{Li}}. \tag{99}$$

It can be shown that there are four contributions to Eq. (99). The first (and for CC and CH coupling constants dominating) term is the Fermi-Contact (FC) contribution described by the following perturbed Hamiltonian

$$\left(\frac{\partial h}{\partial I_{Ki}}\right)_{FC} = -\frac{2\pi e^2 g_K g_e}{3mM_p c^2} \delta(\mathbf{r} - \mathbf{r}_K)\mathbf{s}. \tag{100}$$

The second term describes the spin-dipole (SD) interaction:

$$\left(\frac{\partial h}{\partial I_{Ki}}\right)_{SD} = \frac{e^2 g_K g_e}{4mM_p c^2} \frac{(\mathbf{r}_i - \mathbf{r}_K)^2 \mathbf{s} - 3(\mathbf{s} \cdot (\mathbf{r}_i - \mathbf{r}_K)(\mathbf{r}_i - \mathbf{r}_K)}{|\mathbf{r}_i - \mathbf{r}_K|^5}. \tag{101}$$

Third and fourth terms finally represent the so-called diamagnetic and paramagnetic spin-orbit (DSO and PSO) contributions with

$$\left(\frac{\partial^2 h}{\partial I_{Ki} \partial I_{Lj}}\right)_{DSO} = \frac{g_K g_L e^4}{8mM_p^2 c^4} \frac{(\mathbf{r} - \mathbf{r}_K) \cdot (\mathbf{r} - \mathbf{r}_L)\delta_{ij} - (\mathbf{r} - \mathbf{r}_K)_j (\mathbf{r} - \mathbf{r}_L)_i}{|\mathbf{r} - \mathbf{r}_K|^3 |\mathbf{r} - \mathbf{r}_L|^3} \tag{102}$$

and

$$\left(\frac{\partial h}{\partial I_{Ki}}\right)_{PSO} = -\frac{ie^2 \hbar g_K}{mM_p c^2} \frac{[(\mathbf{r} - \mathbf{R}_K) \times \nabla]_j}{|\mathbf{r} - \mathbf{R}_K|^3}. \tag{103}$$

In Eqs. (100) to (103), $\mathbf{s}$ denotes the operator for the electron spin, m the electron mass, M_p the proton mass, g_K the g factor of the Kth nucleus, and g_e the g factor of the electron.

Though calculation of J_{KL} is not hampered by the gauge problem, its calculation is actually even more problematic. First of all, the FC and SD terms represent so-called triplet operators (note the appearance of the electron spin $\mathbf{s}$ in the expressions given in Eqs. (100) and (101)). The calculation of these contributions to J_{KL} is thus affected by triplet instabilities of the wavefunction and accordingly the HF-SCF approach turns out to be useless in many cases[127]. Second, there are a total of 10 perturbations per nucleus which renders computation of the complete set of spin-spin coupling constants expensive. Third, the basis set convergence in the calculation of the FC term is rather slow[128–130], as the appropriate representation of the delta function operator in the FC contribution in terms of Gaussians is demanding.

As the HF approach fails in most cases, promising results have been so far only obtained at correlated levels. MCSCF calculations[131,132] as well as CCSD calculations[133,134] have been shown to provide reliable theoretical data for these properties. However, both schemes are hampered in the application to larger molecules by high computational requirements.

Therefore, DFT might offer a pragmatic alternative. A first implementation by Malkina *et al.*[135] (within their SOS-DFPT scheme) unfortunately ignores the SD term which in some cases has been proven to be important. A complete implementation within the coupled-perturbed DFT framework that considers all four contributions to J_{KL} has been recently presented by Cremer *et al.*[136] However, further work is still needed before a final conclusion concerning the applicability of DFT in the calculation of spin-spin coupling constants can be given.

4 Frequency-dependent properties

4.1 General theory

While analytic derivative theory is sufficient for the theoretical treatment of time-independent (static) properties, the underlying theory needs to be extended for the calculation of time-dependent (dynamical) properties. In particular, the fact that there is – unlike for the static case – in the time-dependent case no well-defined energy explains why the simple derivative theory discussed so far is not applicable.

Nevertheless, there is large interest in the calculation of dynamical properties. The main examples comprise frequency-dependent polarizabilities and hyperpolarizabilities which are the key quantities in the area of non-linear optics[137,138].

Starting point for the discussion of dynamical properties necessarily is the time-dependent Schrödinger equation[l]

$$H(t)|\Psi\rangle = i\frac{\partial}{\partial t}|\Psi\rangle \tag{104}$$

with the Hamiltonian H consisting of the usual time-independent molecular part H_0 and a time-dependent perturbation $V(t)$:

$$H(t) = H_0 + V(t). \tag{105}$$

For $V(t)$, one generally assumes that it can be written as a sum of periodic perturbations

$$V(t) = \sum_{k=-N}^{N} \exp(-i\omega_k t) \sum_X \epsilon_X(\omega_k) X \tag{106}$$

with ω_k as the frequencies and $\epsilon_X(\omega_k)$ and X denoting the corresponding perturbation strengths and operators. For a periodically oscillating electric field (the most common example), X is the dipole operator μ and $\epsilon_X(\omega_k)$ the corresponding electric field strength.

As $V(t)$ has to be Hermitian, the following relations must hold:

$$X^\dagger = X, \tag{107}$$

[l] Atomic units are used here and in the following.

$$\omega_{-k} = -\omega_k, \tag{108}$$

and

$$\epsilon_X^*(\omega_k) = \epsilon_X(\omega_k). \tag{109}$$

Eq. (106) for $V(t)$ can thus be rewritten as

$$V(t) = \epsilon(0)X + 2\sum_{k=1}^{N} \cos(\omega_k t) \sum_X Re(\epsilon_X(\omega_k))X + 2\sin(\omega_k t) \sum_X Im(\epsilon_X(\omega_k))X \tag{110}$$

i.e., in a more common form with the perturbations given in terms of real sine and cosine functions.

The expectation value of an operator X can now be expanded in the form[m]

$$\begin{aligned}\langle X\rangle(t) = \langle X\rangle_0 &+ \sum_{k_1} \exp(-i\omega_{k_1} t) \sum_Y \langle\langle X;Y\rangle\rangle_{\omega_{k_1}} \epsilon_Y(\omega_{k_1}) \\ &+ \frac{1}{2}\sum_{k_1 k_2} \exp(-i(\omega_{k_1}+\omega_{k_2})t) \sum_{Y,Z} \langle\langle X;Y,Z,\rangle\rangle_{\omega_{k_1},\omega_{k_2}} \epsilon_Y(\omega_{k_1})\epsilon_Z(\omega_{k_2}) + ...\end{aligned} \tag{111}$$

with the linear response function $\langle\langle X;Y\rangle\rangle_{\omega_{k_1}}$, the quadratic response function $\langle\langle X;Y,Z\rangle\rangle_{\omega_{k_1}\omega_{k_2}}$), etc. characterizing the time dependence of $\langle X\rangle$. Eq. (111) can be interpreted as an expansion of $\langle X\rangle$ with respect to the Fourier components of the perturbation $V(t)$. The response functions are denoted by $\langle\langle ...\rangle\rangle$. The operator before the semi-colon represents the operator for which the expectation value is computed, while the operators after the semi-colon denote those which are involved in the Fourier components of $V(t)$. The frequencies given as subscript are those connected with the perturbation operators $Y, Z, ...$

For the specific case of $X = \mu_i$, Eq. (111) takes the form

$$\begin{aligned}\langle\mu_i\rangle(t) = \langle\mu_i\rangle_0 &+ \sum_{k_1} \exp(-i\omega_{k_1} t) \sum_j \alpha_{ij}(-\omega_{k_1};\omega_{k_1})\epsilon_j(\omega_{k_1}) \\ &+ \frac{1}{2}\sum_{k_1,k_2} \exp(-i(\omega_{k_1}+\omega_{k_2})t) \sum_{jk} \beta_{ijk}(-\omega_{k_1}-\omega_{k_2};\omega_{k_1},\omega_{k_2})\epsilon_j(\omega_{k_1})\epsilon_k(\omega_{k_2}) \\ &+ ...\end{aligned} \tag{112}$$

with $\alpha_{ij}(-\omega;\omega)$ as the tensor elements of the frequency-dependent polarizability, $\beta_{ijk}(\omega_1;\omega_2,\omega_3)$ as the tensor elements of the frequency-dependent first hyperpolarizability, etc. Table 7 gives an overview about the various types of polarizabilities and hyperpolarizabilities as well as their relationship to physical effects (for a more detailed discussion see, for example, Ref. 138.) It is important to note in this context that the sum of the frequencies (with explicit consideration of signs!) determines the frequency of the corresponding contribution in the expectation value

[m] Note that we assume that the operator X is included in the perturbation $V(t)$.

Table 7. Definition and physical relevance of the various (frequency-dependent) polarizabilities and hyperpolarizabilities

(hyper)polarizability	physical effect
$\alpha(0;0)$	static polarizability
$\alpha(\text{-}\omega;\omega)$	frequency-dependent polarizability
$\beta(0;0,0)$	static first hyperpolarizability
$\beta(-2\omega;\omega,\omega)$	second harmonic generation (SHG)
$\beta(-\omega;\omega,0)$	dc-Pockels effect (dc-P); electro-optical Pockels Effect (EOPE)
$\beta(0;\omega,\omega)$	optical rectification (OR)
$\gamma(0;0,0,0)$	static second hyperpolarizability
$\gamma(3\omega;\omega,\omega,\omega)$	third harmonic generation (THG)
$\gamma(2\omega;\omega,\omega,0)$	dc-second harmonic generation (dc-SHG); electric field induced SHG (EFISH or ESHG)
$\gamma\ (-\omega;\omega,-\omega,\omega)$	intensity-dependent refractive index (IDRI); degenerate four wave mixing (DFWM)
$\gamma\ (-\omega_1;\omega_1,-\omega_2,\omega_2)$	ac-Kerr effect (ac-K); optical Kerr effect (OKE)
$\gamma\ (-\omega;\omega,0,0)$	dc-Kerr effect (dc-K); electro-optical Kerr effect (EOKE)
$\gamma\ (0;\omega,-\omega,0)$	dc-optical rectification (dc-OR); electric field induced optical rectification (EFIOR)

expression. For example, in case of the SHG hyperpolarizability, the resulting contribution to the dipole moment has *twice* the frequency of the originally perturbing field.

It can be shown that the response functions in Eq. (111) and thus the frequency-dependent properties of interest can be determined as derivatives of the so-called time-averaged quasi energy[139]. The latter is given as

$$Q(t) = \langle\tilde{\Psi}|(H - i\frac{\partial}{\partial t})|\tilde{\Psi}\rangle \tag{113}$$

with the phase-isolated wavefunction

$$|\Psi\rangle = \exp(-iF(t))|\tilde{\Psi}\rangle \tag{114}$$

with

$$\frac{dF(t)}{dt} = \langle\tilde{\Psi}|(H - i\frac{\partial}{\partial t})|\tilde{\Psi}\rangle \tag{115}$$

defined in such a way that it coincides in the static case with the usual time-independent wavefunction. Time averaging of $Q(t)$ is performed in such a manner

$$\{Q(t)\}_T = \lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2} Q(t)dt \tag{116}$$

that T corresponds to multiples of all periods of the considered perturbations.

It can be then demonstrated that the expectation value $\langle X\rangle_0$ corresponds to the first derivative of $\{Q(t)\}$ with respect to the corresponding field strengths[n]

$$\langle X\rangle_0 = \frac{d\{Q\}_T}{d\epsilon_X(0)}, \tag{117}$$

the linear response function to the corresponding second derivative

$$\langle\langle X;Y\rangle\rangle_{\omega_{k_1}} = \frac{d^2\{Q\}_T}{d\epsilon_X(\omega_0)d\epsilon_Y(\omega_{k_1})} \tag{118}$$

with

$$\omega_0 = -\omega_{k_1}, \tag{119}$$

the quadratic response function to the corresponding third derivative

$$\langle\langle X;Y,Z\rangle\rangle_{\omega_{k_1},\omega_{k_2}} = \frac{d^3\{Q\}_T}{d\epsilon_X(\omega_0)d\epsilon_Y(\omega_{k_1})d\epsilon_Z(\omega_{k_2})} \tag{120}$$

with

$$\omega_0 = -\omega_{k_1} - \omega_{k_2}, \tag{121}$$

etc. Time-averaging is essential in order to obtain these simple expressions for the response functions. The trick is that time averaging *extracts* the relevant terms from the quasi-energy $Q(t)$ and enforces proper matching of frequencies (as seen in Eqs. (119) and (121)).

The time dependent variation principle can be rewritten in our case as

$$\delta\{Q\}_T = 0. \tag{122}$$

To obtain explicit equations for the wavefunction parameters, it is convenient to expand them in terms of the Fourier components of the perturbations

$$\begin{aligned} c(t) = c^{(0)} &+ \sum_{k_1} \exp(-i\omega_{k_1}t)c^{(1)}(\omega_{k_1}) \\ &+ \sum_{k_1k_2} \exp(-i(\omega_{k_1}+\omega_{k_2})t)c^{(2)}(\omega_{k_1},\omega_{k_2}) + ... \end{aligned} \tag{123}$$

and then require fulfillment of the variational condition (Eq. (122)) for each order of the perturbation. This yields in first order

$$\frac{\partial\{Q\}_T}{\partial c^{(0)}} = 0, \tag{124}$$

which is identical to the usual time-independent stationarity condition for c. In higher orders, the following conditions are obtained

$$\frac{d}{d\epsilon_X(\omega_1)}\left(\frac{\partial\{Q\}_T}{\partial c^{(1)}(\omega_2)}\right) = 0 \tag{125}$$

with $\omega_1 = -\omega_2$

$$\frac{d^2}{d\epsilon_X(\omega_1)d\epsilon_Y(\omega_2)}\left(\frac{\partial\{Q\}_T}{\partial c^{(2)}(\omega_3,\omega_4)}\right) = 0 \tag{126}$$

[n] Note that all derivatives are taken at the point $\epsilon_X(\omega) = 0, \epsilon_Y(\omega) = 0, ...$

with $\omega_1 + \omega_2 = -\omega_3 - \omega_4$, etc.

For non-variational wavefunction parameters, the same technique as in the static case is used. Accordingly, the derivatives of $Q(t)$ in Eqs. (117) to (126) are replaced by the corresponding derivatives of an energy functional $\tilde{Q}(t)$ (in Ref. 139 called the Lagrangian $L(t)$) with the latter obtained by augmenting $Q(t)$ with the corresponding time-dependent equations ($g(c,t) = 0$) for the wavefunction parameters

$$\tilde{Q}(t) = Q(t) + \lambda g(c,t). \tag{127}$$

Note that in this case, it is sometimes necessary to symmetrize the corresponding expressions for the response functions in order to ensure proper symmetry relations (for a detailed discussion see, for example, section 2.C and 3D of Ref. 139). The symmetrization needs to be carried out with respect to simultaneous complex conjugation and inversion of the sign of the involved frequencies.

With Eqs. (111) to (127), the required theory (usually referred to as *response theory*) for the calculation of frequency-dependent properties is summarized. The given expressions differ from those in the static case mainly by the fact that the energy (the key quantity for static properties) is replaced by the corresponding time-averaged quasi-energy (the key quantity for dynamical properties). However, the same techniques as in the static case, i.e. $(2n+1)$ and $(2n+2)$ rules, can be used to deduce computationally efficient expressions for the calculation of dynamical properties.

4.2 Specific examples

As first example, the computation of frequency-dependent polarizabilities at the closed-shell HF-SCF level will be sketched. The corresponding quasi-energy is given in that case by

$$\begin{aligned} Q(t) = &\sum_{\mu\nu} P_{\mu\nu}(t) h_{\mu\nu} + \frac{1}{2} \sum_{\mu\nu} \sum_{\sigma\rho} P_{\mu\nu}(t) P_{\sigma\rho}(t) (\langle \mu\sigma | \nu\rho \rangle - \frac{1}{2} \langle \mu\sigma | \rho\nu \rangle) \\ &-i \sum_{i} \sum_{\mu\nu} c^*_{\nu i}(t) S_{\mu\nu} \frac{\partial c_{\mu i}}{\partial t}. \end{aligned} \tag{128}$$

In comparison to the usual HF-SCF energy expression (Eq. (19)), we note that in Eq. (128) MO coefficients (and thus also the density matrix elements) carry an explicit time-dependence and that there is one additional term, namely the one which involves the time derivative of the MO coefficients. As we need to ensure the orthonormality of the MOs, the appropriate starting point is given by the following functional

$$\tilde{Q}(t) = Q(t) - 2 \sum_{i} \varepsilon_{ij} \sum_{\mu\nu} c^*_{\mu i} S_{\mu\nu} c_{\nu j} \tag{129}$$

where we augment the quasi-energy by the orthonormality constraint multiplied with the corresponding, now time-dependent Lagrangian multipliers ε_{ij}.

Following Eq. (123), the MO coefficients $c_{\mu i}(t)$ are expanded in terms of the Fourier components of the electric field perturbations

$$c_{\mu i}(t) = c_{\mu i} + \sum_{k_1} \exp(-i\omega_{k_1} t) c^{(1)}_{\mu i}(\omega_{k_1}) + \sum_{k_1 k_2} \exp(-i(\omega_{k_1} + \omega_{k_2})t) c^{(2)}_{\mu i}(\omega_{k_1}, \omega_{k_2}) + ... \tag{130}$$

$$= c_{\mu i} + \sum_{k_1} \exp(-i\omega_{k_1} t) \sum_X \epsilon_X(\omega_{k_1}) c^X_{\mu i}(\omega_{k_1}) + \sum_{k_1 k_2} \exp(-i(\omega_{k_1} + \omega_{k_2})t) \sum_{X,Y} \epsilon_X(\omega_{k_1}) \epsilon_Y(\omega_{k_2}) c^{XY}_{\mu i}(\omega_{k_1}, \omega_{k_2}) + ... \tag{131}$$

Thus, the perturbed coefficients $c^X_{\mu i}(\omega_{k_1}), c^{XY}_{\mu i}(\omega_{k_1}, \omega_{k_2}), ...$ completely characterize the time-dependence of the HF wavefunction. Furthermore, as common in CPHF theory, these coefficients are expanded in terms of the unperturbed MO coefficients

$$c^X_{\mu i}(\omega) = \sum_p c_{\mu p} U^X_{pi}(\omega) \tag{132}$$

with the $U^X_{pi}(\omega)$ as the actual parameters to be determined in the calculation. Similar parametrizations are also used for the higher-order MO coefficients.

Differentiation of $\{\tilde{Q}(t)\}_T$ with respect to electric field components $\epsilon_X(\omega_1)$ and $\epsilon_Y(\omega_2)$ yields then for the frequency-dependent polarizability

$$\alpha_{XY}(\omega_1; \omega_2) = -2 \sum_a \sum_i U^Y_{ai}(-\omega_2) h^X_{ia} - 2 \sum_a \sum_i h^X_{ai} U^Y_{ai}(\omega_2) \tag{133}$$

with

$$\omega_1 = -\omega_2. \tag{134}$$

Corresponding expressions for the hyperpolarizabilities can be found in the literature. They are most conveniently derived using the $(2n+1)$ rule.

The required coefficients $U^X_{ai}(\omega_{k_1})$ are determined via the so-called time-dependent HF (TDHF) equations which are obtained by differentiating the quasi-energy with respect to MO coefficients $c^{(1)}_{\mu i}$ and the corresponding electric field strengths $\epsilon_X(\omega_1)$. After some rearrangements, the following equations are obtained[o]

$$\begin{aligned} \sum_{em} (2\langle am|ie\rangle - \langle am|ie\rangle - \omega) U^X_{em}(\omega) + \sum_{em} (2\langle ae|im\rangle - \langle ae|mi\rangle) U^X_{em}(-\omega) &= -h^X_{ai} \\ \sum_{em} (2\langle am|ie\rangle - \langle am|ie\rangle + \omega) U^X_{em}(-\omega) + \sum_{em} (2\langle ae|im\rangle - \langle ae|mi\rangle) U^X_{em}(\omega) &= -h^X_{ai} \end{aligned} \tag{135}$$

which can be solved in the usual iterative manner. The TDHF equations resemble very much the usual CPHF equations (compare Eq. (32)). Indeed, the CPHF equations for electric perturbations are obtained in the static limit ($\omega \to 0$), in which $U^X_{ai}(\omega) = U^X_{ai}(-\omega)$. Note that for the time-dependent case, the perturbed

[o] Note that for electric perturbations U^X_{ij} is zero.

coefficients for ω and $-\omega$, i.e. $U^X_{ai}(\omega)$ and $U^X_{ai}(-\omega)$ couple and need to determined together.

As a second example, we discuss the computation of frequency-dependent properties within CC theory. The appropriate starting point is here the following quasi-energy functional

$$\tilde{Q}(t) = \langle 0|(1+\Lambda(t))\exp(-T(t))(H - i\frac{\partial}{\partial t})\exp(T(t))|0\rangle. \tag{136}$$

obtained by augmenting the CC quasi-energy with the time-dependent CC equations multiplied by the time-dependent Lagrangian multipliers $\lambda_p(t)$. Orbital relaxation contributions are in Eq. (136) not considered. While for most static properties (geometrical derivatives, magnetic properties, i.e., in general all properties for which perturbation-dependent basis functions are used) inclusion of orbital relaxation is mandatory, the opposite is true for dynamical properties. A closer analysis reveals that a correct pole structure (see the discussion in Ref. 139) is only ensured if orbital relaxation effects are treated via the single excitations (T_1) in an indirect manner and not explicitly included. Explicit consideration of orbital relaxation on the other hand would lead to additional (artificial) second-order poles[139] and thus to an unphysical behavior of the response functions. A further advantage of the so-called *unrelaxed* approach is that the theory as well as corresponding computer implementations are considerably simplified.

For the frequency-dependent polarizability, the following expression is obtained in CC theory

$$\begin{aligned}\alpha_{XY}(-\omega;\omega) = &-\langle 0|(1+\Lambda)[\exp(-T)\frac{\partial H}{\partial Y}\exp(T), T^X(\omega)]0\rangle \\ &-\langle 0|(1+\Lambda)[\exp(-T)\frac{\partial H}{\partial X}\exp(T), T^Y(-\omega)]0\rangle \\ &-\langle 0|(1+\Lambda)[[\exp(-T)H\exp(T), T^Y(-\omega)], T^X(\omega)]|0\rangle\end{aligned} \tag{137}$$

with the perturbed cluster operator $T^X(\omega)^p$ determined as solution of the following equations

$$0 = \langle \Phi_p|\exp(-T)\frac{\partial H}{\partial X}\exp(-T)|0\rangle + \langle \Phi_p|[\exp(-T)(H-\omega)\exp(T), T^X(\omega)]|0\rangle. \tag{138}$$

Corresponding expressions for first and second hyperpolarizabilities have been given in the literature[150,151,154].

4.3 Available implementations

The time-dependent HF scheme for the calculation of frequency-dependent properties has been implemented by Sekino and Bartlett[140] in the eighties employing a general formulation that allows computation of arbitrary polarizabilities and hyperpolarizabilities. This development was later followed by other implementations

[p] In line with Eq. (123), $T^X(\omega)$ is defined via the first-order term in the Fourier expansion of $T(t)$.

(see, for example, Refs. 141 and 142); all of them nowadays allow the routine computation of frequency-dependent properties at the HF-SCF level. Concerning the treatment of electron correlation, a first major effort has been made by Rice and Handy[143] to derive and implement frequency-dependent polarizabilities at the MP2 level. As amply discussed in the literature, a correct formulation and implementation of MP2 frequency-dependent polarizabilities (within response theory) has been only achieved later[144,145]. MP2 calculations of first and second hyperpolarizabilities have recently been reported by Kobayashi *et al.*[146]

At the CC level, response theory was first formulated by Monkhorst[147] in 1977. A first implementation for the computation of frequency-dependent polarizabilities was reported in 1994 within the CCSD approximation[148]. An earlier simplified variant based on an equation-of-motion CC ansatz[149] has turned out less satisfying, as it lacked the important property of size extensivity. CCSD calculations for hyperpolarizabilities were first presented by Hättig *et al.*[150,151] in 1997 and 1998. Concerning inclusion of triple excitations, it is important to recognize that the otherwise highly popular CCSD(T) ansatz is not well suited for the calculation of unrelaxed properties. Considering this, Christiansen *et al.* devised a new hierarchy of CC models[152] consisting of CCS, CC2, CCSD, CC3, etc. for the calculation of dynamical properties. CCS only includes single excitations (CCS energies are identical to the corresponding HF-SCF energies, higher-order response-properties, however, differ), CC2 truncates the doubles equations to lowest order, CCSD involves a full treatment of single and double excitations, while CC3 includes for the first time triple excitations with the triples equations truncated to lowest order. To ensure an adequate treatment of orbital relaxation, it is mandatory in this hierarchy of CC models to consider single excitations (T_1) — contrary to usual perturbation arguments — as zeroth order. CC3 implementations for the computation of dynamical polarizabilities and hyperpolarizabilities have recently reported by Christiansen *et al.*[153,154] and for the first time allow near-quantitative predictions for these type of properties (see, for example, Ref. 155).

Finally, it should be mentioned that a lot of effort has been also devoted to compute frequency-dependent properties at the MCSCF level[156−158] and that there is — as for other properties — a great interest in DFT computations of these properties employing time-dependent DFT techniques (see, for example, Ref. 159).

4.4 Example

As an example for the computation of frequency-dependent properties, we show in Fig. 8 the dispersion curve for the refractive index n of N_2. The latter is related to the (frequency-dependent) isotropic polarizability $\overline{\alpha}$ via

$$n \ = 1 + 2\pi \ \overline{\alpha}(\omega) N. \tag{139}$$

Fig. 8 compares results obtained at SCF and various CC levels[q] with those from experimental investigations.[160] It is clearly seen how the results improve (in comparison with experiment) within the CC hierarchy and that CC3 yields by far the best agreement with experiment.

[q]All calculations have been carried out for $r(NN) = 2.068$ bohr with the aug-pVQZ basis.

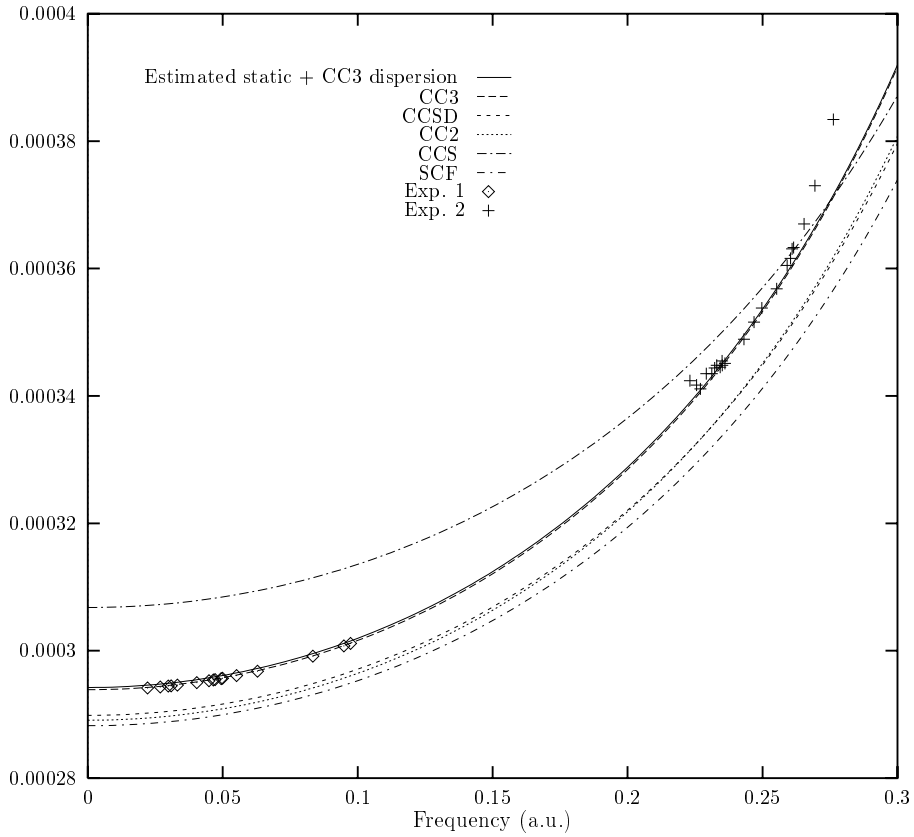


Fig. 8: Refractive index of N_2 as calculated at SCF and various CC levels in comparison with experiment.

5 Summary

The basic concepts for the quantum chemical calculation of molecular properties have been discussed. Focussing on properties specific to a given electronic state, analytic derivative techniques are the essential prerequisite for the accurate and efficient computation of the required energy derivatives. Analytic derivatives techniques have been proven especially important for the computation of magnetic properties. Problems inherent to their calculation such as gauge-origin dependence of the results and slow basis set convergence are best dealt with by using GIAOs, i.e. explicitly magnetic-field dependent basis functions.

The derivative approach to molecular properties can also be extended to dynamical properties. In the framework of response theory, these kind of properties are computed via the corresponding derivatives of the so-called time-averaged quasi energy. As the expressions for dynamical properties coincide in the zero-frequency

limit with those for the corresponding static properties, response theory represents a *unifying* concept for molecular properties that covers both the static and dynamic case but also enables computation of excitation energies (via the poles of the response functions) and transition strengths (via the residues of the response functions at the poles).

The importance of the presented concepts can be hardly overestimated, as computation of molecular properties plays a major role in almost all modern applications of quantum chemistry. In this way, it is certainly justified to consider the introduction of analytic derivative techniques (Pulay's paper on HF-SCF gradients in 1969) as an important mile stone in quantum chemistry. There is no doubt that computation of molecular properties as well as method development in this area of quantum chemistry will remain an important topic in the future.

Acknowledgments

The author thanks Dr. Christian Ochsenfeld, Alexander Auer, and Oliver Heun for helpful discussions and a careful reading of the manuscript.

References

1. See, for example, P.W. Atkins, *Molecular Quantum Mechanis*, Oxford University Press, Oxford, 1983, Chapter 13.
2. See, for example, P.W. Atkins, *Molecular Quantum Mechanis*, Oxford University Press, Oxford, 1983, Chapter 5.
3. J. Hellmann, *Einführung in die Quantenchemie*, Deuticke & Co, (1937); R.P. Feynman, *Phys. Rev.* **56**, 340 (1939).
4. See, for example, the discussion by P. Pulay in *Modern Electronic Structure Theory*, D. Yarkony (Ed.), World Scientific, Singapore, 1995, p. 1191.
5. H.B. Schlegel, in *Modern Electronic Structure Theory*, D. Yarkony (Ed.), World Scientific, Singapore, 1995, p. 459.
6. I.M. Mills, in *Modern Spectroscopy: Modern Research*, K.N. Rao, C.W. Matthews, Eds., Academic Press, New York, 1972, p. 115.
7. W.H. Flygare, *Molecular Structure and Dynamics*, Prentice-Hall, Englewood Cliffs, 1978.
8. W. Weltner, Jr., *Magnetic Atoms and Molecules*, Dover, New York, 1989.
9. W. Schneider, W. Thiel, *Chem. Phys. Lett.* **157**, 367 (1989).
10. P. Jørgensen, T. Helgaker, *J. Chem. Phys.* **89**, 1560 (1988); T. Helgaker, P. Jørgensen, *Theor. Chim. Acta* **75**, 111 (1989).
11. A. Dalgarno, A.L. Stewart, *Proc. Roy. Soc.* **A247**, 245 (1958).
12. A. Szabo, N.S. Ostlund, *Modern Quantum Chemistry*, Mc Graw-Hill, New York, 1989, chapter 3.
13. P. Pulay, *Mol. Phys.* **17**, 197 (1969).
14. See, for example, T. Helgaker, P.R. Taylor, in *Modern Electronic Structure Theory*, D. Yarkony (Ed.), World Scientific, Singapore, 1995, p. 725.
15. J. Gerrat, I.M. Mills, *J. Chem. Phys.* **49**, 1719 (1968).
16. J.A. Pople, R. Krishnan, H.B. Schlegel, J.S. Binkley, *Int. J. Quantum. Chem.*

Symp. **13**, 225 (1979).
17. For recent reviews on CC theory, see: R.J. Bartlett, J.F. Stanton, in *Reviews in Computational Chemistry*, Vol. **5**, K.B. Lipkowitz, D.B. Boyd (Eds.), VCH Publisher, New York 1994, p. 65 ff; T.J. Lee, G.E. Scuseria, in *Quantum Mechanical Electronic Structure Calculations with Chemical Accuracy*, S.R. Langhoff (Ed.), Kluwer Academic Publisher, Dordrecht, 1995, p. 47 ff; R.J. Bartlett, in *Modern Electronic Structure Theory*, D.R. Yarkony (Ed.), World Scientific, Singapore, 1995, p. 1047 ff; J. Gauss, in *Encyclopedia of Computational Chemistry*, P.v.R. Schleyer (Eds.), Wiley, New York, 1998, p. 615 ff.
18. L. Adamowicz, W.D. Laidig, R.J. Bartlett, *Int. J. Quant. Chem. Symp.* **18**, 245 (1984).
19. J.E. Rice, R.D. Amos, *Chem. Phys. Lett.* **122** 585 (1985).
20. R.J. Bartlett, in *Geometrical Derivatives of Energy Surfaces and Molecular Properties*, P. Jørgensen, J. Simons (Eds.), Reidel, Dordrecht, 1986, p. 35.
21. N.C. Handy, H.F. Schaefer, *J. Chem. Phys.* **81**, 5031 (1984).
22. A.C. Scheiner, G.E. Scuseria, J.E. Rice, T.J. Lee, H.F. Schaefer, *J. Chem. Phys.* **87**, 5361 (1987).
23. J. Gauss, J.F. Stanton, R.J. Bartlett, *J. Chem. Phys.* **95**, 2623 (1991).
24. E.A. Salter, G.W. Trucks, R.J. Bartlett, *J. Chem. Phys.* **90**, 1752 (1989).
25. G.E. Scuseria, H.F. Schaefer, *Chem. Phys. Lett.* **146**, 23 (1988).
26. G.E. Scuseria, *J. Chem. Phys.* **94**, 442 (1991).
27. T.J. Lee, A.P. Rendell, *J. Chem. Phys.* **94**, 6229 (1991).
28. J.D. Watts, J. Gauss, R.J. Bartlett, *Chem. Phys. Lett.* **200**, 1 (1992).
29. J. Gauss, J.F. Stanton, *Phys. Chem. Chem. Phys.* **2**, 2047 (2000).
30. J. Gauss, J.F. Stanton, in *Recent Advances in Coupled-Cluster Methods*, R.J. Bartlett (Ed.), World Scientific, Singapore, 1997, p. 49.
31. J. Gauss, J.F. Stanton, *Chem. Phys. Lett.* **276**, 70 (1997).
32. S. Bratoz, *Colloq. Int. CNRS* **82**, 287 (1958). See also, M. Allavena, S. Bratoz, *J. Chim. Phys.* **60**, 1199 (1963) and M. Allavena, *Theoret. Chim. Acta* **5**, 21 (1966).
33. For first applications of HF-SCF gradients see, for example, P. Pulay, *Mol. Phys.* **18**, 473 (1970); *Mol. Phys.* **21**, 329 (1971); P. Pulay, W. Meyer, *J. Mol. Spectrosc.* **41**, 59 (1971); *J. Chem. Phys.* **57**, 3337 (1972).
34. N.C. Handy, R.D. Amos, J.F. Gaw, J.E. Rice, E.D. Simandiras, T.J. Lee, R.J. Harrison, W.D. Laidig. G.B. Fitzgerald, R.J. Bartlett, in *Geometrical Derivatives of Energy Surfaces and Molecular Properties*, P. Jørgensen, J. Simons (Eds.), Reidel, Dordrecht, 1986, p. 179.
35. N.C. Handy, R.D. Amos, J.F. Gaw, J.E. Rice, E.D. Simandiras, *Chem. Phys. Lett.* **120**, 151 (1985).
36. R.J. Harrison, G.B. Fitzgerald, W.D. Laidig, R.J. Bartlett, *Chem. Phys. Lett.* **124**, 291 (1986).
37. L. Verluis, T. Ziegler, *J. Chem. Phys.* **88**, 322 (1988).
38. R. Fournier, J. Andzelm, D. Salahub, *J. Chem. Phys.* **90**, 3671 (1989).
39. B. Delley, *J. Chem. Phys.* **94**, 7245 (1991).
40. S. Kato, K. Morokuma, *Chem. Phys. Lett.* **65**, 19 (1979).
41. J.D. Goddard, N.C. Handy, H.F. Schaefer, *J. Chem. Phys.* **71**, 1525 (1979).

42. G.B. Fitzgerald, R. Harrison, W.D. Laidig, R.J. Bartlett, *J. Chem. Phys.* **82**, 4379 (1985).
43. J. Gauss, D. Cremer, *Chem. Phys. Lett.* **138**, 131 (1987).
44. J. Gauss, D. Cremer, *Chem. Phys. Lett.* **153**, 303 (1988).
45. G.W. Trucks, J.D. Watts, E.A. Salter, R.J. Bartlett, *Chem. Phys. Lett.* **153**, 490 (1988).
46. B.R. Brooks, W.D. Laidig, P. Saxe, J.D. Goddard, Y. Yamaguchi, H.F. Schaefer, *J. Chem. Phys.* **72**, 4652 (1980).
47. R. Krishnan, H.B. Schlegel, J.A. Pople, *J. Chem. Phys.* **72**, 4654 (1980).
48. J.E. Rice, T.J. Lee, N.C. Handy, *J. Chem. Phys.* **88**, 7011 (1988).
49. J. Gauss, D. Cremer, *Chem. Phys. Lett.* **150**, 280 (1988).
50. J. Gauss, D. Cremer, *Chem. Phys. Lett.* **163**, 549 (1989).
51. R. Shepard, H. Lischka, P.G. Szalay, T. Kovar, M. Ernzerhof, *J. Chem. Phys.* **96**, 351 (1991).
52. N.C. Handy, D.J. Tozer, C.W. Murray, G.J. Laming, R.D. Amos, *Isr. J. Chem.* **33**, 331 (1993).
53. B.G. Johnson, M.J. Frisch, *J. Chem. Phys.* **100**, 7429 (1994).
54. M.R. Hoffmann, D.J. Fox, J.F. Gaw, Y. Osamura, Y. Yamaguchi, R.S. Grev, G. Fitzgerald, H.F. Schaefer, P.J. Knowles, N.C. Handy, *J. Chem. Phys.* **80**, 2660 (1984).
55. D.J. Fox, Y. Osamura, M.R. Hoffmann, J.F. Gaw, G. Fitzgerald, Y. Yamaguchi, H.F. Schaefer, *Chem. Phys. Lett.* **102**, 17 (1983).
56. H. Koch, H.J.Aa. Jensen, P. Jørgensen, T. Helgaker, G.E. Scuseria, H.F. Schaefer, *J. Chem. Phys.* **92**, 4924 (1990).
57. J.F. Gaw, Y. Yamaguchi, H.F. Schaefer, *J. Chem. Phys.* **81**, 6395 (1984).
58. M.J. Frisch, M. Head-Gordon, J.A. Pople, *Chem. Phys. Lett.* **166**, 275 (1990).
59. M.J. Frisch, M. Head-Gordon, J.A. Pople *Chem. Phys. Lett.* **166**, 281 (1990).
60. F. Haase, R. Ahlrichs, *J. Comp. Chem.* **14**, 907 (1993).
61. B.G. Johnson, M.J. Frisch, *Chem. Phys. Lett.* **216**, 133 (1993).
62. S.P. Brown, I. Schnell, J.D. Brand, K. Müllen, H.W. Spiess, *J. Mol. Struct.*, in press (2000).
63. C. Ochsenfeld, *Phys. Chem. Chem. Phys.* **2**, 2153 (2000).
64. C. Ochsenfeld, *Chem. Phys. Lett.*, in press.
65. J.M. Vrtilek, C.A. Gottlieb, E.W. Gottlieb, T.C. Killian, P. Thaddeus, *Astrophys. J.* **364**, L53 (1990); J. Cernicharo, C.A. Gottlieb, M. Guélin, T.C. Killian, G. Paubert, P. Thaddeus, J.M. Vrtilek, *Astrophys. J.* **138**, L39 (1991).
66. G. Maier, H.-P. Reisenauer, W. Schwab, P. Carsky, B.A. Hess, L.J. Schaad, *J. Amer. Chem. Soc.* **109**, 5183 (1987).
67. C.A. Gottlieb, T.C. Killian, P. Thaddeus, P. Botschwina, J. Flügge, M. Oswald, *J. Chem. Phys.* **98**, 4478 (1993).
68. J. Gauss, J.F. Stanton, *J. Mol. Struct.* **485–486**, 43 (1999); J.F. Stanton, J. Gauss, *Int. Rev. Phys. Chem.* **19**, 61 (2000).
69. L.D. Landau, E.M. Lifschitz, *Lehrbuch der Theoretischen Physik, Bd. II*, Akademie-Verlag, Berlin, 1981, p 120 ff.
70. A. Abragam, *The Principles of Nuclear Magnetic Resonance*, Oxford University Press, Oxford, 1961.

71. L.D. Landau, E.M. Lifschitz, *Lehrbuch der Theoretischen Physik, Bd. II*, Akademie-Verlag, Berlin, 1981, p 120 ff.
72. See, for example, I. Mills, T. Cvitas, K. Homann, N. Kallay, K. Kuchitsu, *Quantities, Units and Symbols in Physical Chemistry*, Blackwell, Oxford, 1993, p. 119 ff.
73. W. Kutzelnigg, *J. Mol. Struct. (Theochem)*, **202**, 11 (1989).
74. W. Kutzelnigg, *Isr. J. Chem.* **19**, 193 (1980).
75. M. Schindler, W. Kutzelnigg, *J. Chem. Phys.* **76**, 1919 (1982).
76. A.E. Hansen, T.D. Bouman, *J. Chem. Phys.* **82**, 5035 (1985).
77. F. London, *J. Phys. Radium* **8**, 397 (1937).
78. H.F. Hameka, *Mol. Phys.* **1**, 203 (1858); H.F. Hameka, *Mol. Phys.* **2**, 64 (1959).
79. R. Ditchfield, *Mol. Phys.* **27**, 789 (1974).
80. K. Wolinski, J.F. Hinton, P. Pulay, *J. Am. Chem. Soc.* **112**, 8251 (1990).
81. See, for example, F. Boys, *Rev. Mod. Phys.* **32**, 296 (1960); J.M. Foster, S.F. Boys, *Rev. Mod. Phys.* **32**, 300 (1960); C. Edminston, K. Ruedenberg, *Rev. Mod. Phys.* **35**, 457 (1963).
82. P. Pulay, J.F. Hinton, K. Wolinski, in *Nuclear Magnetic Shieldings and Molecular Structure*, J.A. Tossell, Ed., Kluwer, Dordrecht, 1993, p. 243.
83. P. Pulay, *Adv. Chem. Phys.* **69** (1987) 241.
84. For a review (in German) , see, P. Buzek, P. von R. Schleyer, S. Sieber, *Chemie in unserer Zeit* **26**, 116 (1992).
85. J. Gauss, *Chem. Phys. Lett.* **191** , 614 (1992).
86. J. Gauss, *J. Chem. Phys.* **99**, 3629 (1993).
87. K. Ruud, T. Helgaker, R. Kobayashi, P. Jørgensen, K.L. Bak, H.J.Aa Jensen, *J. Chem. Phys.* **100**, 8178 (1994).
88. J. Gauss, *Chem. Phys. Lett.* **229**, 198 (1994).
89. J. Gauss, J.F. Stanton, *J. Chem. Phys.* **102**, 251 (1995)
90. J. Gauss, J.F. Stanton, *J. Chem. Phys.* **103**, 3561 (1995).
91. J. Gauss, J.F. Stanton, *J. Chem. Phys.* **104**, 2574 (1996).
92. C. van Wüllen, W. Kutzelnigg, *Chem. Phys. Lett.* **205**, 563 (1993).
93. T.D. Bouman, A.E. Hansen, *Chem. Phys. Lett.* **175**, 292 (1990).
94. V.G. Malkin, O.L. Malkina, D.R. Salahub, *Chem. Phys. Lett.* **204**, 80 (1993).
95. V.G. Malkin, O.L. Malkina, M.E. Casida, D.R. Salahub, *J. Am. Chem. Soc.* **116**, 5898 (1994).
96. G. Schreckenbach, T. Ziegler, *J. Phys. Chem.* **99**, 606 (1995).
97. G. Rauhut, S. Puyear, K. Wolinski, P. Pulay, *J. Phys. Chem.* **100**, 6310 (1996).
98. J.R. Cheeseman, G.W. Trucks, T.A. Keith, M.J. Frisch, *J. Chem. Phys.* **104**, 5497 (1996).
99. A.M. Lee, N.C. Handy, S.M. Colwell, *J. Chem. Phys.* **103**, 10095 (1995).
100. L. Olsson, D. Cremer, *J. Chem. Phys.* **105**, 8995 (1996).
101. L. Olsson, D. Cremer, *J. Phys. Chem.* **100**, 16881 (1996).
102. P.J. Wilson, R.D. Amos, N.C. Handy, *Chem. Phys. Lett.* **312**, 475 (1999).
103. G. Vignale, M. Rasolt, *Phys. Rev. B* **37**, 10685 (1988); G. Vignale, M. Rasolt, D.J.W. Geldart, *Adv. Quant. Chem.* **21**, 235 (1990).

104. C.J. Grayce, R.A. Harris, *Phys. Rev. A* **50**, 3089 (1994); *J. Phys. Chem.* **99**, 2724 (1995); F.R. Salsbury, R.A. Harris, *J. Chem. Phys.* **107**, 7350 (1997); *Chem. Phys. Lett.* **279**, 247 (1997).
105. M. Häser, R. Ahlrichs, H.P. Baron, P. Weis, H. Horn, *Theor. Chim. Acta* **83**, 435 (1992).
106. U. Meier, C. van Wüllen, M. Schindler *J. Comp. Chem.* **13**, 551 (1992).
107. M. Kollwitz, J. Gauss, *Chem. Phys. Lett.* **260**, 639 (1996).
108. M. Kollwitz, M. Häser, J. Gauss, *J. Chem. Phys.* **108**, 8295 (1998).
109. S. Saebø, P. Pulay, *Ann. Rev. Phys. Chem.* **44**, 213 (1993) and references therein.
110. C. Hampel, H.J. Werner, *J. Chem. Phys.* **104**, 6286 (1996); M. Schütz, G. Hetzer, H.-J. Werner, *J. Chem. Phys.* **111**, 5691 (1999); M. Schütz, G. Hetzer, H.-J. Werner, *J. Chem. Phys.*, submitted.
111. J. Gauss, H.-J. Werner, *Phys. Chem. Chem. Phys.* **2**, 2083 (2000).
112. M. Kaupp, O.L. Malkina, V.G. Malkin, P. Pyykkö, *Chem. Eur. J.* **4**, 118 (1998).
113. H. Nakatsuji, H. Takashima, M. Hada, *Chem. Phys. Lett.* **233**, 95 (1995).
114. V.G. Malkin, O.L. Malkina, D.R. Salahub, *Chem. Phys. Lett.* **261**, 335 (1996).
115. J. Vaara, K. Ruud, O. Vahtras, H. Ågren, J. Jokisaari, *J. Chem. Phys.* **109**, 1212 (1998).
116. G. Schreckenbach, T. Ziegler, *Int. J. Quantum Chem.* **60**, 753 (1996).
117. D. Sundholm, J. Gauss, A. Schäfer, *J. Chem. Phys.* **105**, 11051 (1996).
118. A. Auer, J. Gauss, J.F. Stanton, to be published.
119. K. Ruud, P.-O. Åstrand, P.R. Taylor, *J. Chem. Phys.* **112**, 2668 (2000).
120. D. Cremer, L. Olsson, F. Reichel, E. Kraka, *Isr. J. Chem.* **33**, 369 (1993).
121. K.V. Mikkelsen, P. Jørgensen, K. Ruud, T. Helgaker, *J. Chem. Phys.* **106**, 1170 (1997).
122. R. Cammi, B. Mennucci, J. Tomasi, *J. Chem. Phys.* **110**, 7627 (1999).
123. H.-U. Siehl, T. Müller, J. Gauss, P. Buzek, P.von R. Schleyer, *J. Am. Chem. Soc.* **116**, 6384 (1994); J.F. Stanton, J. Gauss, H.-U. Siehl, *Chem. Phys. Lett.* **262**, 183 (1996); J. Gauss, J.F. Stanton, *J. Mol. Struct. (Theochem)* **398–399**, 73 (1997).
124. J. Gauss, U. Schneider, R. Ahlrichs, C. Dohmeier, H. Schnöckel, *J. Am. Chem. Soc.* **115**, 2402 (1993).
125. S.P. Brown, I. Schnell, J.D. Brand, K. Müllen, H.W. Spiess, *J. Am. Chem. Soc.*, **121**, 6712 (1999).
126. C. Ochsenfeld, S.P. Brown, I. Schnell, J. Gauss, H.W. Spiess, *J. Am. Chem. Soc.*, to be submitted (2000).
127. J. Kowalewski, *Prog. NMR Spectr.* **11**, 1 (1977); J. Kowalewski, A. Laaksonen, in *Theoretical Models of Chemical Bonding, Part 3*, Z.B. Maksic (Ed.), Springer, Berlin, 1991, p. 386 ff.
128. J. Geertsen, J. Oddershede, G.E. Scuseria, *J. Chem. Phys.* **87**, 2138 (1987).
129. J. Oddershede, J. Geertsen, G.E. Scuseria, *J. Phys. Chem.* **92**, 3056 (1988); J. Geertsen, J. Oddershede, G.E. Scuseria, *Magn. Reson.* **93**, 455 (1991).
130. T. Helgaker, M. Jaszunski, K. Ruud, A. Gorska, *Theor. Chem. Acc.* **99**, 175

(1998).
131. O. Vahtras, H. Ågren, P. Jørgensen, H.J.Aa. Jensen, S.B. Padkjaedr, T. Helgaker, *J. Chem. Phys.* **96**, 6120 (1992).
132. A. Barszcewicz, M. Jaszunski, K. Trela-Kamienska, T. Helgaker, P. Jørgensen, O. Vahtras, *Theor. Chim. Acta* **87**, 19 (1993); K. Ruud, T. Helgaker, P. Jørgensen, K.L. Bak, *Chem. Phys. Lett.* **226**, 1 (1994); T. Helgaker, M. Jaszunski, K. Ruud, *Mol. Phys.* **91**, 881 (1997).
133. S.A. Perera, H. Sekino, R.J. Bartlett, *J. Chem. Phys.* **101**, 2186 (1994); S.A. Perera, M. Nooijen, R.J. Bartlett, *J. Chem. Phys.* **104**, 3290 (1996).
134. S.A. Perera, R.J. Bartlett, P.v.R. Schleyer, *J. Am. Chem. Soc.* **117**, 8476 (1995); S.A. Perera, R.J. Bartlett, *J. Am. Chem. Soc.* **118**, 7849 (1996).
135. O.L. Malkina, D.R. Salahub, V.G. Malkin, *J. Chem. Phys.* **105**, 8793 (1996).
136. V. Sychrowsky, J. Gräfenstein, D. Cremer, *J. Chem. Phys.*, in press (2000).
137. D. P. Shelton, J. E. Rice, *Chem. Rev.* **94**, 3 (1994).
138. For a review, see D.M. Bishop, P. Norman, in *Handbook of Advanced Electronic and Photonic Materials*, H.S. Nalwa (Ed.), Academic Press, San Diego, in press (2000).
139. O. Christiansen, P. Jørgensen, C. Hättig, *Int. J. Quantum Chem.* **68**, 1 (1998).
140. H. Sekino, R.J. Bartlett, *J. Chem. Phys.* **85**, 976 (1986).
141. S.P. Karna, M. Dupuis, E. Perrin, P.N. Prasad, *J. Chem. Phys.* **92**, 7418 (1990); S.P. Karna, M. Dupuis, *J. Comp. Chem.* **12**, 487 (1991).
142. J.E. Rice, R.D. Amos, S.M. Colwell, N.C. Handy, J. Sanz, *J. Chem. Phys.* **93**, 8828 (1990).
143. J.E. Rice, N.C. Handy, *J. Chem. Phys.* **94**, 4959 (1991).
144. C. Hättig, B. Heß, *Chem. Phys. Lett.* **233**, 359 (1995).
145. F. Aiga, K. Sasagane, R. Itoh, *J. Chem. Phys.* **99**, 3779 (1993).
146. T. Kobayashi, K. Sasagane, F. Aiga, K. Yamaguchi, *J. Chem. Phys.* **110**, 11720 (1999); *J. Chem. Phys.* **111**, 842 (1999).
147. H.J. Monkhorst, *Int. J. Quantum Chem. Symp.* **S11**, 421 (1977).
148. R. Kobayashi, H. Koch, P. Jørgensen, *Chem. Phys. Lett.* **219**, 30 (1994).
149. J.F. Stanton, R.J. Bartlett, *J. Chem. Phys.* **98**. 3029 (1993).
150. C. Hättig, O. Christiansen, H. Koch, P. Jørgensen, *Chem. Phys. Lett.* **269**, 428 (1997).
151. C. Hättig, O. Christiansen, P. Jørgensen, *Chem. Phys. Lett.* **282**, 139 (1998).
152. O. Christiansen, H. Koch, P, Jørgensen, *Chem. Phys. Lett.* **243**, 409 (1995).
153. O. Christiansen, J. Gauss, J.F. Stanton, *Chem. Phys. Lett.* **292**, 437 (1998).
154. J. Gauss, O. Christiansen, J.F. Stanton, *Chem. Phys. Lett.* **296**, 117 (1998).
155. H. Larsen, J. Olsen, C. Hättig, P. Jørgensen, O. Christiansen, J. Gauss, *J. Chem. Phys.* **111**, 1917 (1999).
156. D.L. Yeager, P. Jørgensen, *Chem. Phys. Lett.* **65**, 77 (1979); J. Olsen, P. Jørgensen, *J. Chem. Phys.* **82**, 3235 (1985).
157. H. Hettema, H.J. Aa. Jensen, P. Jørgensen, J. Olsen, *J. Chem. Phys.* **97**, 1174 (1992).
158. D. Jonsson, P. Norman, H. Ågren, *J. Chem. Phys.* **105**, 6401 (1995).
159. S.J.A. Gisbergen, V.P. Osinga, O.V. Gritsenko, R. van Leeuwen, J.G. Snijders, E.J. Baerends, *J. Chem. Phys.* **105**, 3142 (1996); S.J.A. Gisber-

gen, J.G. Snijders, E.J. Baerends, *J. Chem. Phys.* **109**, 10644 (1998).
160. E.R. Peck, B.N. Khanna, *J. Opt. Soc. Am.* **56**, 1059 (1966); P.G. Wilkinson, *J. Opt. Soc. Am.* **50**, 1002 (1960).

TENSORS IN ELECTRONIC STRUCTURE THEORY: BASIC CONCEPTS AND APPLICATIONS TO ELECTRON CORRELATION MODELS

MARTIN HEAD-GORDON, MICHAEL S. LEE, PAUL E. MASLEN,
TROY VAN VOORHIS, AND STEVEN R. GWALTNEY
Department of Chemistry, University of California, and
Chemical Sciences Division, Lawrence Berkeley National Laboratory,
Berkeley, CA 94720, USA
E-mail: mhg@bastille.cchem.berkeley.edu, mslee@scripps.edu
{maslen,tvan,gwaltney}@bastille.cchem.berkeley.edu

Electronic structure theories correspond to approximate Schrödinger equations, which are usually solved within finite basis expansions. Most textbook treatments of electronic structure theory present both derivations and working equations in terms of a single particle basis of orthogonal functions. All that matters, however, is the space that the functions collectively span, and observables such as the energy are naturally invariant to quite general nonunitary transformations of the underlying basis functions. Tensor methods compactly express such invariances. This chapter presents a simple introduction to tensor methods and their applications in electronic structure theory. One important message is that there is no extra algebraic effort necessary to derive electronic structure theories in terms of an underlying non-orthogonal basis when the so-called natural representation is used. In this representation there is a term-by-term correspondence with equations in an orthogonal basis. It is then straightforward to transform such equations via metric matrices into the covariant integral representation, which is sometimes more convenient for computational purposes, or as a starting point for further approximations, such as local correlation models. In the remainder of the chapter, the development of local electron correlation methods using nonorthogonal functions to span both the occupied and virtual spaces is discussed as an application of the tensor methods, and as an interesting new methodology in its own right. Additionally several other recent uses of the tensor methods from our group are briefly summarized.

1 Introduction

In this chapter, we discuss the development of electronic structure theory without requiring any orthogonalization of the underlying one-particle basis, which, in general, is naturally nonorthogonal. We employ tensor methods to permit the treatment of nonorthogonality in an efficient and general fashion. The resulting tensor equations express the working equations of any electronic structure method in a way that exhibits all of the natural invariances of such equations to underlying transformations of the basis, between either different orthogonal or nonorthogonal representations. By contrast, standard textbook treatments[1,2,3] of electronic structure theory tend to present working equations that are explicitly in an orthogonal basis, and perhaps in the end back-transform them to the original atomic orbital basis. The basic reason for employing orthogonal representations is that they appear to yield simpler equations: working equations in the atomic orbital basis are complicated by the appearance of the overlap matrix in many places. Part of the beauty of the tensor-based approach advocated here is that it allows us to obtain ex-

pressions that are general to nonorthogonal basis sets with essentially no additional complication. This has been recognized in the research literature[4,5,6,7], although it is generally not widely known amongst electronic structure theorists.

The usefulness of working equations in the nonorthogonal atomic orbital representation (or perhaps some other local nonorthogonal representation) needs little explanation these days. The atomic orbital basis consists of strongly localized functions, which permits any localization of the one and two-particle density matrices in real-space to be exploited for computational efficiency. For example, recent advances in linear scaling methods[8,9] for mean field electronic structure calculations depend critically on the use of localized nonorthogonal functions. Localized (and generally nonorthogonal) single particle functions also provide a natural representation in which the description of electron correlation via many-body theories can be cast in local terms. A spatially localized treatment of electron correlation[10,11] is one way to reduce the unphysical scaling of computational cost with molecular size that afflicts orthogonal basis formulations of many-body theories.

Beyond pragmatic considerations of computational efficiency, perhaps the primary purpose of allowing nonorthogonal functions is to obtain expressions for electronic structure theories that are as general as possible. By this we mean that an electronic structure theory normally corresponds to solving a set of equations within some given one, two or many-particle Hilbert space: this is the working model of the Schrodinger equation for a given model chemistry. The details of the individual functions that span these spaces are clearly unimportant in general: all that matters is the space they collectively define. In other words, what we are saying is that electronic structure equations are usually operator relations, which are merely represented in a given basis. Tensor methods have the important advantage of naturally expressing this invariance to nonsingular transformations of the basis functions. It is intuitive that the operators and approximate wave functions depend only on the vector space spanned by the basis functions, rather than details of the basis functions such as nonorthogonality. Their matrix representations are the tensors that we shall focus on.

The first part of this chapter is an introduction to simple concepts of tensor analysis, which does not assume any background in the area. In terms of electronic structure methods, this then lets us more or less immediately treat problems that involve functions of only one electron at a time, as an application of standard tensor methods to quantum mechanics in a finite-dimensional one-particle Hilbert space. We emphasize the fact that exact linear dependence in the basis does not pose a significant problem. The next major topic is the treatment of many-body methods as an application of tensor theory. In first quantization, generalizations of Slater's rules for matrix elements are obtained, while in second quantization, Wick's theorem holds, and lead to representations of one and two particle operators in terms of the quasiparticle reference. These two topics comprise the tutorial part of the chapter (Sections 2 and 3).

The next major part of the chapter (Section 4) describes the use of tensor concepts to treat electron correlation in a spatially localized representation. We discuss new methods that reduce the complexity of electron correlation theories by making physically motivated truncations of the wavefunction variables. In partic-

ular, the simplest wavefunction-based description of electron correlation is second order Møller-Plesset (MP2) theory. In MP2 theory, the wavefunction variables are amplitudes that describe correlated fluctuations of pairs of electrons from a mean field reference into excited states. The tensor approach allows us to represent the spaces that are occupied and unoccupied in the mean-field reference in terms of over-complete sets of atom-centered functions. In turn, this means that the double substitution amplitudes can now be represented in terms of these atom-centered functions. If no local truncation is performed, then the MP2 energy would be more expensive to evaluate in this representation than in terms of the usual "canonical" molecular orbitals (at least naively).

The overcomplete atomic representations are chosen because they are an ideal starting point for local truncations of the double substitution amplitudes. We discuss in detail the various local truncations that can be obtained by restricting the number of amplitudes retained, based on an atomic criterion. An example of such a criterion is that at least one occupied and one unoccupied index of a double substitution must belong to the same atom, otherwise that amplitude is set to zero. Such atomic truncations yield simplified "local" descriptions of the electron correlations, with inherently reduced computational complexity. Furthermore, atomic truncations have the important advantage that they yield inherently smooth potential energy surfaces, and, indeed the resulting local correlation methods satisfy all the criteria of a well-defined theoretical model chemistry. It may at first sound like these atomic truncations are very drastic, and so we additionally present a selection of numerical results to show that the resulting "local model chemistries" are quite faithful to the model chemistry obtained without any truncation (within the MP2 model).

There are, of course, innumerable other possible applications of tensor concepts to outstanding problems in electronic structure theory, either of the model development type, or of the algorithmic type. The final part of this chapter (Section 5) is a short introduction to several other areas where we have found these concepts to be helpful. This includes the simplified description of geminal wavefunctions for bond-breaking, extensions of the local correlation models discussed for MP2 theory to triple substitutions, and perturbation theory with non-Hermitian Hamiltonians. The intention is to give the reader some flavor of why nonorthogonal orbitals and tensor concepts are useful in these problems. It is important to emphasize that there are many other interesting and important applications of tensor concepts in electronic structure theory as well. Some have been pursued by other groups already, but many are yet to be investigated at all! For this reason, we believe that a general knowledge of tensor methods is useful for someone who is planning to do research in electronic structure theory, and the purpose of this chapter is to provide a starting point.

2 Basic Tensor Concepts

While we must refer to textbooks[12,13,14,15] for a full introduction to tensor analysis, we shall develop the basic concepts necessary for application to electronic structure methods in this section.

2.1 Covariant Basis Functions and the Metric

The introduction of a finite basis of one-particle functions is the so-called algebraic approximation of electronic structure theory. The one-particle basis functions are atomic orbitals, which are non-orthogonal amongst themselves. Let us term this set of given functions the covariant basis, and write them as: $\{|\phi_\mu\rangle\}$. While it is not necessary, for simplicity we shall assume that the covariant basis functions (and indeed all matrix elements involving the basis functions) are real. Other sets of one-particle covariant functions will also be used later. They will be for example a set of functions that spans only the occupied part of the one-particle space, or only the unoccupied part.

Let us introduce a first basic definition. Quantities which, upon a transformation of the basis, change in the same way as the basis functions, are termed *covariant. Covariant quantities are denoted via subscripts.* All matrix elements involving the basis functions are entirely covariant in character as they are calculated in a quantum chemistry program. In particular, the overlap matrix plays a very central role in tensor analysis and is called the *covariant metric.* It will be denoted as $g_{\mu\nu}$:

$$g_{\mu\nu} \equiv S_{\mu\nu} = \langle \phi_\mu \mid \phi_\nu \rangle \tag{1}$$

2.2 Contravariant Basis Functions and the Inverse Metric

Given a nonorthogonal set of basic functions, or, in tensor language, given a covariant basis, how will we be able to resolve a vector into components? In other words, how will the operation of projection be performed in the absence of orthogonality? The answer is that there is a matching "dual" basis that can be readily derived from the covariant functions, whose members have the property of being biorthogonal to the covariant functions.

These functions, which are called contravariant basis functions, are defined by the action of the inverse overlap matrix (or the inverse of the covariant metric), on the covariant basis functions:

$$|\phi^\mu\rangle = \sum_\nu |\phi_\nu\rangle \left(S^{-1}\right)^{\nu\mu} \tag{2}$$

The term *contravariant* is meant to imply that these functions transform on a change of basis in the opposite or inverse manner to the way in which the covariant functions transform. These transformation properties will be established shortly in the following subsection. To distinguish contravariant functions from covariant, *contravariant indexes are written as superscripts.*

A one-line proof verifies that, by construction, the contravariant functions are indeed biorthogonal to the covariant functions:

$$\langle \phi^\mu \mid \phi_\nu \rangle = \sum_\lambda \left(S^{-1}\right)^{\mu\lambda} \langle \phi_\lambda \mid \phi_\nu \rangle = \delta^\mu_\nu \tag{3}$$

The overlap matrix of the contravariant functions is in fact the inverse of the overlap matrix of the covariant functions:

$$g^{\mu\nu} \equiv \langle \phi^{\mu} \mid \phi^{\nu} \rangle = \sum_{\lambda\sigma} \left(S^{-1}\right)^{\mu\lambda} \langle \phi_{\lambda} \mid \phi_{\sigma} \rangle \left(S^{-1}\right)^{\sigma\nu} = \left(S^{-1}\right)^{\mu\nu} \tag{4}$$

The role of the metric matrices can now be clearly seen. The contravariant metric defined above has the general property that it converts a covariant index to a contravariant one. Specifically we now see that Eq. (2) can be rewritten as:

$$|\phi^{\mu}\rangle = \sum_{\nu} |\phi_{\nu}\rangle g^{\nu\mu} \tag{5}$$

Likewise the covariant metric matrix acting on a contravariant index converts that index to being of the covariant type:

$$|\phi_{\mu}\rangle = \sum_{\nu} |\phi^{\nu}\rangle g_{\nu\mu} \tag{6}$$

as is immediately proven by substituting Eq. (2) into Eq. (6). Thus the *metric matrices allow interconversion of covariant and contravariant indices.*

2.3 Invariances to Transformations and the Summation Convention

Suppose we transform from the original set of covariant basis functions, $\{|\phi_{\mu}\rangle\}$, to a modified set, $\{|\tilde{\phi}_{\mu}\rangle\}$, spanning the same space by a transformation matrix $\mathbf{T}$:

$$|\tilde{\phi}_{\mu}\rangle = \sum_{\nu} |\phi_{\nu}\rangle T^{\nu}_{\bullet\mu} \tag{7}$$

The placeholder ($\bullet$), means that the first index of the transformation is contravariant while the second index is covariant, to distinguish them since they are inequivalent. As discussed further in a subsequent subsection, we implicitly adopt the convention that the right-hand index has the character of a ket and the left-hand index has the character of a bra, in terms of Dirac's bra-ket notation.

This new covariant basis will have an overlap matrix, or covariant metric, which is related to the previous one by the following expression:

$$\tilde{g}_{\mu\nu} = \langle \tilde{\phi}_{\mu} \mid \tilde{\phi}_{\nu} \rangle = \sum_{\lambda\sigma} T^{\bullet\lambda}_{\mu} g_{\lambda\sigma} T^{\sigma}_{\bullet\nu} \tag{8}$$

For real transformations $\mathbf{T}$, the adjoint of $\mathbf{T}$ is the same as the transpose; i.e. $T^{\bullet\nu}_{\mu} = T^{\nu}_{\bullet\mu}$.

Given this new set of covariant basis functions, we can define a new contravariant basis by the prescription given in the previous subsection.

$$|\tilde{\phi}^{\mu}\rangle = \sum_{\nu} |\tilde{\phi}_{\nu}\rangle \tilde{g}^{\nu\mu} \tag{9}$$

How are these new contravariant functions related to the original set? The inverse of the new metric defined by Eq. (8) is evidently related to the original inverse metric by the following relation involving the inverse of the transformation, **T**:

$$\tilde{g}^{\mu\nu} = \sum_{\lambda\sigma} \left(T^{-1}\right)^{\mu}_{\bullet\lambda} g^{\lambda\sigma} \left(T^{-1}\right)^{\bullet\nu}_{\sigma} \tag{10}$$

Substituting Eqs. (8) and (10) into Eq. (9) we obtain the transformation property of the contravariant functions:

$$\left|\tilde{\phi}^{\mu}\right\rangle = \sum_{\nu} \left|\phi^{\nu}\right\rangle \left(T^{-1}\right)^{\bullet\mu}_{\nu} \tag{11}$$

The *contravariant functions transform inversely to the way that the covariant functions transform*. More generally, therefore, covariant and contravariant *indices* have inverse transformation properties.

This fact is important in tensor analysis. If we form a scalar quantity (or more generally reduce the number of free indices by one) by summing over one covariant index and one contravariant index:

$$c = \sum_{\mu} a^{\mu} b_{\mu} \tag{12}$$

then such a scalar will be invariant to transformations of the basis:

$$\tilde{c} = \sum_{\mu} \tilde{a}^{\mu} \tilde{b}_{\mu} = \sum_{\mu\nu\lambda} a^{\nu} \left(T^{-1}\right)^{\bullet\mu}_{\nu} T^{\bullet\lambda}_{\mu} b_{\lambda} = \sum_{\nu\lambda} a^{\nu} \delta^{\bullet\lambda}_{\nu} b_{\lambda} = c \tag{13}$$

The energy in an electronic structure theory is just such a scalar, which will be given by various sums over orbitals. Tensor notation will therefore immediately express the invariance of the energy to certain classes of transformations of the orbitals.

The *summation convention* simply states that the simple presence of a repeated index implies summation, provided *the index occurs once in covariant form and once in contravariant form*, as needed if the result is to be invariant. Specifically, we are defining:

$$a^{\mu} b_{\mu} \equiv \sum_{\mu} a^{\mu} b_{\mu} \tag{14}$$

So, at this stage we have a useful perspective. The introduction of covariant and contravariant indices that have inverse properties upon transformations of the basis permits us to sum over indices in a way that is invariant with respect to such transformations. Such a summation is termed a contraction.

More generally, tensor notation, meaning the use of the covariant and contravariant representations, will let us write express operator relations such that

their transformation properties upon a change of basis are fully implicit in the algebra. Certain rules must be followed for this to be true:

(1) Contraction must only be over index pairs where one index is covariant and one index is contravariant.
(2) Addition of one quantity to another will only be valid if they have indices that match one-to-one in covariant and contravariant character.

2.4 Flat Euclidean Spaces Versus Curved Spaces

A brief word on the geometrical character of what we have discussed to this stage is in order. We have introduced a basis, with a metric or overlap matrix that is nondiagonal. This corresponds to a set of axes that are not orthogonal to each other, but are instead skewed. We are treating the elements of the metric as constants whose value is the same regardless of where we are in the single particle space. Thus we are working in Euclidean geometry, and space is flat. For simplicity we shall retain this restriction throughout this article.

Other choices of basis vectors may define a metric whose values change depending on where we are in the space. The basis vectors corresponding to spherical polar coordinates in 3-space are a familiar example. Another example that is familiar in chemistry are the internal coordinates often used to describe molecular geometry. These basis vectors define spaces that are curved rather than flat, and it is important to properly account for this curvature. For example, to describe the shortest distance between two points, straight lines must be replaced by geodesics (the generalization of great circles). Additionally vectors can no longer be simply translated through space to change their origin, but instead must change their orientation with translation. For further consideration of this topic, we refer the reader to introductory (or not so introductory!) treatments of differential geometry[12,14,15]. An interesting and detailed recent article[16] has much relevance to minimizing self-consistent field type energy expressions.

2.5 Bras, Kets and Operators as Tensors

When we form matrix elements in electronic structure theory, we are, by definition, combining bras and kets. Therefore the tensors which we shall employ to represent operators will have associated with them a set of indices that will be divided into equal numbers of bra indices and ket indices. We adopt the useful convention that for a matrix element involving $2n$ indices, the first n indices will be associated with bras and the last n indices will be associated with kets. That is consistent with how all the matrix elements have been written in the previous parts of this section.

Let us consider the representation of some simple operators. Most basic is the operator for projection onto the space spanned by the covariant basis. The form of this projection operator follows from the biorthogonality of the covariant and contravariant functions:

$$\hat{1} = |\phi_\mu\rangle \langle\phi^\mu| \tag{15}$$

The metric matrices may be used to write this relation entirely in terms of either just the covariant basis functions or just the contravariant basis functions:

$$\hat{1} = |\phi_\mu\rangle\, g^{\mu\nu}\, \langle\phi_\nu| = |\phi^\mu\rangle\, g_{\mu\nu}\, \langle\phi^\nu| \tag{16}$$

It is simply verified that any of the three equivalent forms for the projection operator acting on a ket in the covariant space gives back that same ket. They express the "resolution of the identity" in this space. By simply inserting Eq. (15) or Eq. (16) after a bra or before a ket, one resolves it into components in either the covariant or contravariant basis in the usual way.

Expressions for other operators may also be expressed in terms of the covariant and contravariant basis functions by employing the resolution of the identity in any of the three forms given above. For example, using the simplest form, Eq. (15), to resolve a one electron operator into this nonorthogonal basis set yields:

$$\hat{F}\left(\mathbf{r_1}, \mathbf{r'_1}\right) = |\phi_\mu\rangle\, \langle\phi^\mu|\, \hat{F}\, |\phi_\nu\rangle\, \langle\phi^\nu| = |\phi_\mu\rangle\, F^\mu_{\bullet\nu}\, \langle\phi^\nu| \tag{17}$$

In the second (more compact) form of this equation, we are applying the convention that in a tensor of rank 2, the first index corresponds to the bra and the second index corresponds to the ket. A two electron operator may be treated similarly, making sure to apply the resolution of the identity to both the first and second electronic coordinates:

$$\hat{G} = |\phi_\mu\,(1)\rangle\, |\phi_\nu\,(2)\rangle\, \langle\phi^\mu\phi^\nu|\, \phi_\lambda\phi_\sigma\rangle\, \langle\phi^\sigma\,(2)|\, \left\langle\phi^\lambda\,(1)\right| \tag{18}$$

2.6 Natural Representation and the Covariant Integral Representation

Given the fact that a given ket can be represented in either the covariant or contravariant basis, as can a given bra, there are indeed many ways that a given tensor can be written. Specifically a one particle operator, such as given in Eq. (17), can be represented in 2^2=4 ways, while a two-particle operator, such as Eq. (18), can be represented in 2^4=16 different ways. Two of these ways are most useful in the context of electronic structure theory.

The first useful representation is called the *natural representation*, and adopts the convention that all ket indices are represented in the covariant basis, while all bra indices are represented in the contravariant basis. This representation is faithful in the sense that the tensor algebra representing an operator equation will be isomorphic to the operator form of the equation. Crucially, this representation also yields equations that correspond term-by-term to the form of the equations in an orthogonal basis, with the only difference lying in the covariant and contravariant character of individual indices. It is most convenient to obtain tensor equations initially in the natural representation, as they can sometimes be written down by inspection, or inferred from known equations in the orthogonal representation. The essential point is that obtaining the defining equations of an electronic structure method in terms of nonorthogonal functions in the natural representation is no harder than obtaining the equations in a conventional orthogonal representation!

We shall further develop the analogies between the natural representation and conventional orthogonal basis tools for second quantization and many-electron theory in the following sections.

The second useful representation may be called the covariant integral representation, and is the representation in which all matrix elements of one and two particle operators in the Hamiltonian are represented entirely in terms of covariant indices. This is a useful representation because in electronic structure codes, as we have already discussed, these matrix elements are always calculated with all indices covariant. If all matrix elements are defined fully covariant, then the variables that they are contracted with in the equations of a given electronic structure method must be fully contravariant. For example, in one-electron theories, the one-particle density matrix is fully contravariant in this representation. So, to summarize, as a general rule, we will *derive* in the natural representation, but *compute* in the covariant integral representation.

It is worthwhile at this point to briefly contrast what has been done with the tensor notation relative to conventional treatments of nonorthogonality in quantum chemistry. Conventionally what is done is to *assume* a particular tensor representation, such as the covariant integral representation, for the matrix elements and unknowns. This has two principal drawbacks. First, the overlap matrix must be carried explicitly, which makes the equations and derivations cumbersome relative to simply transforming the tensor character of the indices of the final equations in the natural representation with metric matrices. Second, without the explicit covariant and contravariant character of the indices, it is all too easy to violate the rules of tensor algebra.

2.7 Treatment of Exact Linear Dependence

We must extend the tensor treatment to the case where exact linear dependence exists in the single particle basis. The reason for doing so (rather than eliminating such redundancy at the outset) is that often convenient sets of occupied and virtual functions may include exact linear dependence. A simple example are *projected atomic orbitals*, defined by acting with P and Q on the parent set of atomic orbitals:

$$|\phi_i\rangle = \hat{P}\,|\phi_\mu\rangle\,\delta_i^\mu = |\phi_\mu\rangle\,P_{\bullet\nu}^{\mu}\delta_i^\nu \tag{19}$$

$$\big|\phi_a\big\rangle = \hat{Q}\big|\phi_\mu\big\rangle\delta_a^\mu = \big(\hat{I}-\hat{P}\big)\big|\phi_\mu\big\rangle\delta_a^\mu = \big|\phi_\mu\big\rangle\big(\delta_a^\mu - P_{\bullet\nu}^{\mu}\delta_a^\nu\big) \tag{20}$$

The delta functions merely preserve index conventions. Projected atomic functions are localized to the same extent as the density matrix itself[17]. Their covariant metric matrices are the covariant representations of P and Q respectively:

$$g_{ij} = \langle\phi_i \mid \phi_j\rangle = \delta_i^\mu P_{\mu\nu}\delta_j^\nu \tag{21}$$

$$g_{ab} = \langle\phi_a \mid \phi_b\rangle = \delta_a^\mu\,(g_{\mu\nu} - P_{\mu\nu})\,\delta_b^\nu \tag{22}$$

Contravariant functions may be defined by projection on the contravariant basis vectors with P and Q; their contravariant metrics are the contravariant representations of P and Q respectively.

The linear dependence associated with these representations is immediately evident. Take for example the occupied space. The *dimension* (number of linearly independent functions) of the occupied subspace is the particle number, n, but Equations (19) yields a larger set of functions (N, equal to the dimension of the atomic orbital basis itself). The metric matrix, Equation (21), is then not formally invertible, and the ability to change indices from covariant to contravariant seems to be lost. A coordinate transformation from a set of nonredundant functions to a linearly dependent representation (or vice versa) involves a rectangular matrix, which also has no formal inverse. Can we allow these redundant representations in the tensor formalism?

The answer is yes, for the following reason[18]. Trial vectors on which the inverse operates lie in the *range* of the metric (the linearly independent subspace) with no component in the *nullspace*, since a linearly dependent representation still spans the same space as a related linearly independent basis. Hence we employ a *generalized inverse* based on discarding the nullspace, as defined by the singular value decomposition (SVD). Writing out summations explicitly, the SVD for a square matrix is:

$$(S)^{-1}_{ab} = \sum_{c'} U_{ac} s_c^{-1} U_{bc} \tag{23}$$

where the terms in the sum over c are discarded if s_c (the eigenvalues of $\mathbf{S}$, with eigenvectors $\mathbf{U}$) is zero. This inverse does not satisfy $\mathbf{S}^{-1}\mathbf{S}=\mathbf{1}$, but does yield the correct solution, $\mathbf{x}=\mathbf{S}^{-1}\mathbf{b}$ to linear equations $\mathbf{Sx}=\mathbf{b}$, if $\mathbf{b}$ lies within the range of $\mathbf{S}$. The SVD is also general for rectangular matrices, as needed for transformations between redundant and nonredundant representations.

Generalized inverses of the metrics for projected atomic orbitals are the contravariant metric matrices discussed above: the density matrix and its orthogonal complement respectively:

$$g^{ij} = \left\langle \phi^i \;\middle|\; \phi^j \right\rangle = \delta^i_\mu P^{\mu\nu} \delta^j_\nu \tag{24}$$

$$g^{ab} = \left\langle \phi^a \;\middle|\; \phi^b \right\rangle = \delta^a_\mu \left(g^{\mu\nu} - P^{\mu\nu}\right) \delta^b_\nu \tag{25}$$

This is proved either by considering the overlap of the contravariant projected functions themselves or by the fact that the product of these two matrices operating on any vector lying entirely within the occupied space yields the same function unmodified.

The contraction of the product of covariant and contravariant metrics yields an idempotent matrix which represents the Kronecker delta. For example, in the occupied case, we obtain

$$g^{ik} g_{kj} = \left\langle \phi^i \;\middle|\; \phi_j \right\rangle = g^i_{\bullet j} \neq \delta^i_j \tag{26}$$

Thus redundant contravariant functions are not biorthogonal with redundant covariant functions. In the natural representation, P and Q must then be written as:

$$\hat{P} = |\phi_i\rangle \, g^{i}_{\bullet j} \, \langle \phi^j | \tag{27}$$

$$\hat{Q} = |\phi_a\rangle \, g^{a}_{\bullet b} \, \langle \phi^b | \tag{28}$$

We next consider rectangular coordinate transformations, $C^{j'}_{\bullet i}$, from a set of projected atomic orbitals, (19), (primed indices) to a linearly independent orthogonal set (unprimed indices). This transformation matrix is a set of molecular orbital coefficients. The generalized completeness relation above defines **C** and its generalized inverse **D** directly:

$$\big|\phi_i\big\rangle = \big|\phi_{j'}\big\rangle g^{j'}_{\bullet k'} \big\langle \phi^{k'} \mid \phi_i \big\rangle \equiv \big|\phi_{j'}\big\rangle C^{j'}_{\bullet i} \tag{29}$$

$$\big|\phi_{i'}\big\rangle = \big|\phi_j\big\rangle \big\langle \phi^j \mid \phi_{i'} \big\rangle \equiv \big|\phi_j\big\rangle D^{j}_{\bullet i'} = \big|\phi_j\big\rangle \big(C^{j}_{\bullet i'}\big)^* \tag{30}$$

D is both a left inverse and a right inverse. In the latter case, where **D** is undoing the transformation into the unprimed coordinates, the product of **C** and **D** is not the identity matrix, but rather the projector onto the occupied subspace in the mixed representation. Thus **D** works properly on all functions in the occupied subspace. We can also view **D** as a transformation from an orthogonal basis into the linearly dependent representation, with **C** as its generalized inverse.

This completes the demonstration that all basic operations of tensor algebra can be performed in terms of linearly dependent representations. Exact linear dependence evidently poses no practical or formal problem, because the space spanned by the linearly dependent functions remains well defined. Numerical complications can still arise from *near* linear dependence (where, for example, the eigenvalues of Equation (23) are very small, but not zero. In that case a threshold must be defined (the square root of machine precision for example) below which the eigenvalues and associated eigenvectors are discarded. Finally, while we have been discussing single-particle problems, it should also be emphasized that all of the results to be discussed in the next section on many-electron problems will also hold in the case of linear dependence, provided that the simple generalizations discussed here are applied.

3 Many-Electron Theory

We shall consider only so-called *single reference* theories of electron correlation. They begin from a single determinant of occupied orbitals, such as the mean-field Hartree-Fock solution, and treat electron correlation by substitutions of sets of occupied orbitals with virtual orbitals. As such, single reference methods are based upon an initial partitioning of the one-particle Hilbert space into two subspaces.

One consists of levels that are occupied in the reference determinant, and the other consists of unoccupied (virtual) functions:

$$\hat{I} = \hat{P} + \hat{Q} \tag{31}$$

The operators $\hat{P}$ and $\hat{Q}$ are projectors (as defined by idempotency, $\hat{P}^2 = \hat{P}$, $\hat{Q}^2 = \hat{Q}$) onto the occupied and virtual subspaces respectively. From Eq. (31) all functions which lie entirely within the occupied space are orthogonal to functions lying entirely in the virtual space (a one or two line proof). This applies to any mix of covariant and contravariant basis vectors, and is a condition of *strong orthogonality*. We shall adopt the convention that indices *i,j,k...* denote functions spanning the occupied subspace, indices *a,b,c...* denote functions spanning the virtual subspace, and indices μ,ν,λ,σ... denote functions lying in the full one-particle space. The presentation in this section generally follows our full paper on this problem[7].

3.1 Many-electron Determinants

We assume a single particle basis of dimension N, partitioned as in Eq. (31) into occupied and virtual subspaces (dimension n and N-n), each spanned by linearly independent functions (as discussed in the following section, linear dependence poses no practical problem either). The simplest many-electron wavefunction is a single determinant $|\Phi\rangle$, comprised of the n covariant functions spanning P. We call $|\Phi\rangle$ the *covariant reference ket*:

$$|\Phi\rangle = (n!)^{-1/2}\, det\left(|\phi_1\rangle\, |\phi_2\rangle \cdots |\phi_i\rangle \cdots |\phi_n\rangle\right) \tag{32}$$

What is the tensor character of $|\Phi\rangle$? This follows from how $|\Phi\rangle$ transforms under nonorthogonal transformations of the underlying single-particle basis functions spanning either P or Q. Since $|\Phi\rangle$ depends on tensor products of n single particle functions, it is an n-th rank antisymmetric tensor. The properties of antisymmetric tensors are well known as part of exterior algebra or Grassman algebra. An important fact is that if the dimension of the underlying single particle basis is S, then the dimension of the subspace of nth rank tensors which are antisymmetric is ${}^{S}\mathrm{C}_{n}$. This is 1 if S=n, as is the case for $|\Phi\rangle$ under transformations of the occupied space, or ${}^{N-n}\mathrm{C}_{0}$ for the virtual space. This permits substantial simplification, because a basis function spanning a one-dimensional subspace behaves much like a scalar under transformation of the basis.

In fact, a nonsingular linear transformation, $\mathbf{C}$, of the one-electron kets spanning the occupied space only alters the covariant reference ket by a constant, det($\mathbf{C}$). The proof is based on rewriting Eq. (32) as $|\Phi\rangle = (n!)^{-1/2}\, det\,(\Theta)$, where Θ is a matrix whose rows are electron labels, and whose columns are occupied covariant orbital labels. Under transformation with $\mathbf{C}$:

$$(n!)^{1/2}\, |\Phi\rangle' = det\,(\Theta') = det\,(\Theta C) = det\,(\Theta)\, det\,(C) = |\Phi\rangle\, det\,(C) \tag{33}$$

An antisymmetric n-th rank tensor which transforms this way is a *relative tensor* of rank 0 and weight 1 (or a pseudoscalar). Hence we can focus only on (the

small number of) single particle levels that are altered from their configuration (occupied or not) in the reference by the correlation treatment, rather than treating substituted determinants as general n-th rank tensors. The weight 1 refers to occupied basis transformations, while the tensor is of weight 0 with respect to virtual basis transformations, as is trivially proven.

Many electron expansion spaces are generated by replacing covariant occupied levels by covariant virtual levels. The set of all single substitutions of occupied levels, i, by virtual levels a, the corresponding set of all double substitutions, etc comprise well-defined subspaces of the full n-electron Hilbert space, $\mathbb{Q}_n$ (which consists of all substitutions through n-fold).

$$\mathbb{Q}_n = 0 \oplus S \oplus D \oplus T \oplus \cdots \oplus n \tag{34}$$

In general, electron correlation techniques seek an approximate solution of the Schrödinger equation that is defined only within such subspaces. For example, a trial wavefunction that depends on single and double substitutions can be completely determined by solving a Schrödinger-like equation within the $0 \oplus S \oplus D$ subspace.

We state the tensor properties of the substituted determinants that span these many-electron S, D, T subspaces, under transformations of the occupied and virtual single particle basis sets without proof. A covariant virtual level a, introduced by substitution, transforms covariantly. The hole index, i, of an occupied level that has been substituted, transforms contravariantly, consistent with creating a hole by projection with a biorthogonal contravariant bra orbital. These n-th rank antisymmetric tensors are also relative tensors of rank (m,m), for m-fold substitutions, with weight 1 for occupied basis transformations, and weight 0 with for virtual basis transformations. A single substitution of an occupied (covariant) level i by a virtual (covariant) level a is written as $\big|\Phi^i_{\bullet a}\big\rangle$, while double substitutions are $\big|\Phi^{ij}_{\bullet\bullet ab}\big\rangle$.

The space spanned by any given level of substitutions of occupied levels by virtuals is closed under separate nonsingular linear transformations of occupied and/or virtual levels amongst themselves. This follows directly from the fact that strong orthogonality between occupied and virtual subspaces is preserved under nonunitary (nonsingular) transformations of the one-particle basis set within the occupied and virtual subspaces. To give a concrete example, single substitutions are not mixed with double substitutions under transformations of this type. Note that subsets of these spaces are generally not closed under linear transformations.

The tensor character of substitution amplitudes is opposite to the many-electron basis vectors so that their contractions are properly invariant to transformations of the basis. Operators which generate invariant mixtures of substituted determinants can then be written as

$$\hat{T}_1\big|\Phi\big\rangle = t^a_{\bullet i}\big|\Phi^i_{\bullet a}\big\rangle \tag{35}$$

$$\hat{T}_2\big|\Phi\big\rangle = \tfrac{1}{4} t^{ab}_{\bullet\bullet ij}\big|\Phi^{ij}_{\bullet\bullet ab}\big\rangle \tag{36}$$

The amplitude tensors can be transformed from this natural representation with metric matrices in order to alter the character of the indices if desired.

3.2 Many-electron Bras and Slater's Rules

We define many-electron dual spaces, which preserve the biorthogonality present between contravariant and covariant single particle levels, starting from a determinant of contravariant bras, which is the n-electron *contravariant reference bra*:

$$\big\langle \Phi \big| = \big(n!\big)^{-1/2} det\big(\big\langle \phi^1 \big| \big\langle \phi^2 \big| \cdots \big\langle \phi^i \big| \cdots \big\langle \phi^n \big|\big) \tag{37}$$

There are corresponding manifolds of substituted bras, $\{\langle \Phi^a_{\bullet i}|\}$, $\big\{\big\langle \Phi^{ab}_{\bullet\bullet ij}\big|\big\}$ etc. *In the dual space, the tensor character of indices is reversed: occupied indices (holes) are covariant, virtual indices (particles) are contravariant.* For example, we replace occupied level $\big\langle \phi^i \big|$ with virtual level $\langle \phi^a|$ in the single substitution $\langle \Phi^a_{\bullet i}|$. These are all relative tensors of weight -1, because by construction they transform in a reciprocal way to the relative tensors of Sec. 3.1. They are the adjoints of the corresponding covariant many-electron kets.

Solution of Schrödinger-like equations in the many-electron subspace spanned by a given level of substituted determinants can be achieved by forming projection equations with the appropriate dual vectors. Biorthogonal representations arise in unsymmetric eigenvalue problems, such as coupled cluster excited state methods[19]. They are also sometimes employed in valence bond methods[20]. In our case, the projection equations are components of an absolute tensor which represents the many-electron equations in a subspace. It is invariant to transformations of the underlying basis, provided the projections are complete within each given substituted manifold (i.e. all double substitutions, for example).

Matrix elements of operators between many-electron bras and kets in the natural representation are slight generalizations of Slater's rules for the orthogonal case, with all new information contained within the covariant and contravariant character of the indices. The key to this close analogy is the natural representation in which covariant and contravariant character is consistently reversed between many-electron kets and bras. This is a tremendous contrast relative to the complexity of matrix elements between determinants of nonorthogonal functions in both bra and ket, which do not have the biorthogonality property.

For the identity operator (overlaps), as a result of orthonormality of single particle functions in the dual space with functions in normal (covariant) basis, only diagonal overlaps are nonzero. Thus:

$$\langle \Phi \, | \Phi \rangle = 1 \tag{38}$$

$$\big\langle \Phi^b_{\bullet j} \, \big| \, \Phi^i_{\bullet a} \big\rangle = \delta^i_j \delta^b_a \tag{39}$$

$$\big\langle \Phi^{cd}_{\bullet\bullet kl} \, \big| \, \Phi^{ij}_{\bullet\bullet ab} \big\rangle = \big(\delta^i_k \delta^j_l - \delta^i_l \delta^j_k\big)\big(\delta^c_a \delta^d_b - \delta^d_a \delta^c_b\big) = \delta^{ij}_{kl} \delta^{cd}_{ab} \tag{40}$$

In Eq. (40), we have simplified the final expression by introducing the generalized Kronecker delta (whose definition is obvious from the full equation). Nevertheless it is apparent from Eq. (40) that doubly substituted contravariant bras are not biorthogonal to doubly substituted covariant kets as we have defined them. This situtation is exactly the same as applies in the familiar orthonormal representation, and is a result of using a set of double substitutions which is redundant by roughly a factor of 4 (the permutationally related substitutions $\left|\Phi^{ij}_{\bullet\bullet ab}\right\rangle$, $\left|\Phi^{ji}_{\bullet\bullet ab}\right\rangle$, $\left|\Phi^{ij}_{\bullet\bullet ba}\right\rangle$, and $\left|\Phi^{ji}_{\bullet\bullet ba}\right\rangle$ are all equivalent). The somewhat cumbersome process of spin-adaptation can be used to define sets of orthonormal double substitutions, but we shall not pursue that issue here, preferring to use general spin orbitals. For a clear introductory discussion of how to proceed for the closed shell singlet case, see the article by Knowles, Schütz and Werner elsewhere in this volume.

For spin-orbital matrix elements involving one-electron operators, we derive results whose form is also familiar, as a direct result of the biorthogonality between many-electron basis kets and the dual basis bras:

$$\langle \Phi \,|F|\, \Phi\rangle = F^{i}_{\bullet i} = g^{ij}F_{ji} \tag{41}$$

$$\langle \Phi^{a}_{\bullet i} \,|F|\, \Phi\rangle = F^{a}_{\bullet i} = g^{ab}F_{bi} \tag{42}$$

We next consider two-electron operators. The tensor analog of the antisymmetrized ("double bar") two-electron integrals is a fourth rank tensor that in the natural representation has two covariant (ket) indices and two contravariant (bra) indices. There is also a corresponding fourth rank tensor for the parent ("single bar") two electron integrals. The definition of the antisymmetrized two electron integrals is:

$$\text{I\!I}^{pq}_{\bullet\bullet rs} = \langle \Phi^{p}\Phi^{q} \,|\Phi_{r}\Phi_{s}\rangle - \langle \Phi^{p}\Phi^{q} \,|\Phi_{s}\Phi_{r}\rangle \equiv I^{pq}_{\bullet\bullet rs} - I^{pq}_{\bullet\bullet sr} \tag{43}$$

This appears in matrix elements of the two-electron repulsion operator:

$$\langle \Phi \,|G|\, \Phi\rangle = \tfrac{1}{2}\text{I\!I}^{ij}_{\bullet\bullet ij} = \tfrac{1}{2}g^{ik}g^{jl}\text{I\!I}_{klij} \tag{44}$$

$$\langle \Phi^{a}_{\bullet i} \,|G|\, \Phi\rangle = \text{I\!I}^{aj}_{\bullet\bullet ij} = g^{ab}g^{ik}\text{I\!I}_{bkij} \tag{45}$$

$$\left\langle \Phi^{ab}_{\bullet\bullet ij}\middle|G\middle|\Phi\right\rangle = \text{I\!I}^{ab}_{\bullet\bullet ij} = g^{ac}g^{bd}\text{I\!I}_{cdij} \tag{46}$$

In addition to the natural representation, we have employed the metric tensors to re-express the results in terms of fully covariant two-electron integrals.

3.3 Second Quantization

In first quantization in the previous section, we have continued to develop a strong isomorphism between the algebra of quantum mechanics in orthogonal single particle basis sets, and nonorthogonal single particle basis sets when the natural representation is employed. It is therefore at least intuitive that other powerful tools of many-body electronic structure theory with orthonormal single particle basis sets will also carry over virtually unchanged to the nonorthogonal case provided the natural representation is employed. In particular, the tools of second quantization, and the associated diagrammatic methods for systematically treating the relatively complex algebra of many body theories is what we will want to use in general.

The purpose of this section is to describe the manner in which second quantization carries over from the case of an orthonormal single particle basis to the nonorthogonal case in the natural representation. We shall concentrate on the basic concepts, carrying them as far as necessary to show that Wick's theorem, the underpinning of diagrammatic methods, goes over unaltered. It is beyond our present scope to provide any real introduction to the use of Wick's theorem in evaluating many-body matrix elements. Rather, the key is to establish the manner in which standard textbook treatments (see for example refs. 2,3) of second quantization and diagrammatic methods can be directly employed in problems involving nonorthogonal single particle basis sets.

Due to our focus on the many-particle problem, it is not convenient to define creation and destruction operators with respect to the zero particle vacuum, unlike some previous work on nonorthogonal basis sets. Therefore, *we take the so-called quasiparticle (QP) vacuum as our reference*: a single determinant of n occupied functions. In the natural representation, the covariant reference ket contains the occupied set of covariant basis vectors, while the contravariant reference bra contains the set of occupied contravariant functions. *Thus for substituted many-electron kets, particles (levels outside the QP reference which become filled) are covariant in character while holes (levels within the QP reference which become vacated) are contravariant. In the dual space of substituted many-electron bras, particles are contravariant while holes are covariant.* All QP creation and destruction operators defined below obey this simple convention.

We must distinguish operations on virtual (unoccupied) functions (which can create or destroy particles in the QP vacuum) from operations on occupied functions (which will create or destroy holes in the QP vacuum). For the particle operators in the virtual space in the natural representation:

$$a_a^\dagger \equiv \text{QP particle creation operator for } |\phi_a\rangle \equiv \text{QP destruction operator for } \langle\phi^a|$$

$$a^b \equiv \text{QP particle destruction operator for } |\phi_b\rangle \equiv \text{QP creation operator for } \left\langle\phi^b\right|$$

The anticommutation relations for the particle operators follow by any of the usual textbook derivations as:

$$\left\{a^a, a_b^\dagger\right\} = \delta_b^a \tag{47}$$

$$\left\{a_a^\dagger, a_b^\dagger\right\} = \left\{a^a, a^b\right\} = 0 \tag{48}$$

As we have already seen in several other contexts, all differences relative to the usual orthogonal basis treatments are contained in the covariant and contravariant character of the indices. Of course this is only the case in the natural representation. Otherwise we would find overlaps for the anticommutator in Eq. (47). In a redundant representation, such as was discussed in the section on Linear Dependence, the idempotent matrix of overlaps of covariant and contravariant levels replaces the Kronecker delta:

$$\left\{a^a, a_b^\dagger\right\} = g^a_{\bullet b} \tag{49}$$

For functions spanning the occupied space, the identity of creation and destruction operators reverses, as we create and destroy holes in the QP vacuum:

$a^i \equiv$ QP hole creation operator for $|\phi_i\rangle \equiv$ QP hole destruction operator for $\left\langle \phi^i \right|$

$a_i^\dagger \equiv$ QP hole destruction operator for $|\phi_i\rangle \equiv$ QP hole creation operator for $\left\langle \phi^i \right|$

The anticommutation relations are:

$$\left\{a^i, a_j^\dagger\right\} = \delta^i_j \tag{50}$$

$$\left\{a_i^\dagger, a_j^\dagger\right\} = \left\{a^i, a^j\right\} = 0 \tag{51}$$

Due to strong orthogonality between levels in the occupied and virtual one-particle spaces, all anticommutators between particle and hole operators are zero, as in the orthogonal basis. In a redundant representation, $g^i_{\bullet j}$ replaces δ^i_j , analogous to how we showed above that Eq. (49) generalizes Eq. (47).

Contraction is central to evaluating matrix elements within second quantization. The contraction of operators x_1 and x_2 is defined as the difference between their product, and their QP normal ordered product. The *normal ordered* string is arranged such that all particle-hole creation operators are to the left, and all destruction operators are to the right, and is multiplied by the *parity* of the permutation (-1 for an odd number of pairwise swaps; $+1$ for an even number) necessary to take the product to normal order. Thus the contraction of a pair of operators (denoted by the overbar) is:

$$\overbrace{x_1 x_2} = x_1 x_2 - N\left[x_1 x_2\right] = \langle \Phi | \, x_1 x_2 \, | \Phi \rangle \tag{52}$$

where the N signifies that the following operator string is in normal order. Based on the anticommutation relations above, the only nonzero contractions between any pair of particle-hole creation and destruction operators are:

$$\overbrace{a^a a_b^\dagger} = \delta_b^a \tag{53}$$

$$\overbrace{a_i^\dagger a^j} = \delta_i^j \tag{54}$$

To evaluate many-electron operator matrix elements, we apply Wick's theorem (e.g. refs. 2,3 for orthogonal basis sets), which expresses a string of creation and destruction operators of length m, as a terminating sum of partial contractions of the string in normal order:

$$x_1 x_2 \cdots x_m = N\left[x_1 x_2 \cdots x_m\right] + \sum_{\substack{\text{single}\\ \text{contractions}}} N\left[x_1 \overbrace{x_2 \cdots} x_m\right] + \sum_{\substack{\text{double}\\ \text{contractions}}} N\left[x_1 \overbrace{x_2 \cdots \overbrace{\cdots} x_m}\right] + \ldots + \sum_{\substack{\text{m/2 fold}\\ \text{contractions}}} N\left[\overbrace{x_1 x_2}\ \overbrace{x_3 x_4} \cdots \overbrace{x_m}\right] \tag{55}$$

Since vacuum expectation values of normal-ordered operator strings are zero, the vacuum expectation value involves only the fully contracted terms. Again, apart from the newly explicit tensor character of the operators, these results are identical with the familiar orthogonal basis case. Relative to previous treatments of many-electron theory involving nonorthogonal single particle expansion spaces, this is the real value of the present approach.

Using either Wick's theorem directly, or from inspection of the orthogonal basis results, we can now obtain general expressions for one and two-electron operators:

$$\hat{H} = H_{\bullet i}^{i} + H_{\bullet q}^{p} N\left[a_p^\dagger a^q\right] \tag{56}$$

$$\hat{G} = \tfrac{1}{2} \mathit{\Pi}_{\bullet\bullet ij}^{ij} + \mathit{\Pi}_{\bullet\bullet qi}^{pi} N\left[a_p^\dagger a^q\right] + \tfrac{1}{4} \mathit{\Pi}_{\bullet\bullet rs}^{pq} N\left[a_p^\dagger a_q^\dagger a^s a^r\right] \tag{57}$$

In these expressions, indices *i,j,k..* are occupied levels, and *p,q,r,s* are general levels where the indices run over both occupied and virtual ranges, and the antisymmetrized two-electron integrals were defined in Eq. (43). These second quantization tools are sufficient to reduce n-electron matrix elements to mixtures of 1 and 2-electron matrix elements, as for the generalized Slater's rules given in the previous section, or the correlation methods discussed in the sections below.

4 Nonorthogonal Functions for Local Electron Correlation

The development of what may be called "fast methods" for evaluating electron correlation is a problem of both fundamental and practical importance, because of these unphysical increases in computational complexity with molecular size which afflict "exact" implementations of electron correlation methods. The purpose of

this part of the chapter is to discuss our recent progress towards developing a new family of local correlation models that are based on atomic truncations[21,22,23]. Our emphasis here is on the formulation of the theory, as an interesting and topical application of the general tensor approach to the many-body problem developed in the first part of this chapter. We restrict our discussion entirely to second order Møller-Plesset (MP2) theory[24], the simplest useful wavefunction-based electron correlation method. Local models that perform effectively at the MP2 level are expected to be transferable to other theories of electron correlation in terms of single and double substitutions, such as coupled cluster theory with single and double substitutions (CCSD). Our local MP2 models are also an interesting starting point for treating triple substitutions (as needed for example in MP4 and CCSD(T)) as we briefly describe in Section 5.1.

The MP2 method was originally proposed in 1934[24]. Revived in the mid-1970's, it remains highly popular today, because it offers systematic improvement in optimized geometries and other molecular properties relative to Hartree-Fock theory[25]. Indeed, in a recent comparative study of small closed shell molecules[26] , MP2 outperformed the much more expensive CCSD method for such properties! Relative to state-of-the-art Kohn-Sham density functional theory (DFT) methods[27] , which are the most economical methods to account for electron correlation effects, MP2 has the advantage of properly incorporating long-range dispersion forces. While its computational cost scales as the 5^{th} power of molecular size, quite efficient semidirect methods[28] permit routine applications to medium-sized molecules.

We do not attempt to comprehensively review other efforts at developing fast methods for electron correlation. However, it is important to note that, to date, the most successful approach of this type is the method pioneered by Saebø and Pulay (SP)[10,29,30]. The SP model has now been adopted and extended by several groups, both at the MP2 level[31,32,33,34] and also at the level of self-consistent treatment of single and double substitutions[35,36,37]. We shall later compare the SP method against our new methods. Two important differences are that the SP method does not yield continuous potential energy surfaces, and sometimes fails to treat symmetry equivalent atoms on an equal footing, as in benzene for example. However, to this stage the SP method has advanced much further as far as efficient implementation is concerned. An excellent discussion of the current state of the art is given in the chapter by Knowles, Schütz and Werner in this volume.

As a preliminary, we discuss how local correlation impacts theoretical and numerical modeling in quantum chemistry. Generally, it is useful to distinguish two main classes of errors in an electronic structure calculation:

(1) *Intrinsic model errors*, due to employing incomplete one-particle basis sets, and incomplete descriptions of the many particle expansion spaces associated with the one particle basis. Such errors may be large, but by virtue of understanding the chemistry of the model and systematically studying its behavior over a range of applications, one can understand the different conditions under which it may produce reliable versus unreliable results. The theoretical model should be constructed in as clean a fashion as possible, so that successes and failures of the model can be cleanly understood.
(2) *Numerical error*, due to not exactly implementing the model. There is a strong

argument for attempting to keep this source of error as small as possible. Without doing so, one cannot assess the intrinsic model errors cleanly. Furthermore, the reproducibility of calculations is impeded if there is significant numerical noise. However, in the context of fast methods, the use of looser thresholds and cutoffs leads to significant speedups. Therefore it is advisable to control precision carefully because it is strongly correlated with computational cost.

Ideally, the development of fast methods for treating electron correlation should not impact either model errors or numerical errors. Unfortunately this is not possible at present, as may be appreciated from the following rough argument. *Spatial locality* is what permits reformulations of electronic structure methods that yield the same answer as traditional methods, but faster. The one-particle density matrix decays exponentially with a rate that relates to the HOMO-LUMO gap in periodic systems. From model studies, as well as formal considerations[17] , the decay rates are fairly well understood. When length scales longer than this characteristic decay length are examined, sparsity will emerge in both the one-particle density matrix and also pair correlation amplitudes expressed in terms of localized functions. Very roughly, such a length scale[9] is about 5 to 10 atoms in a line, for a good insulator such as alkanes. Hence sparsity emerges beyond this number of atoms in 1-d, beyond this number of atoms squared in 2-d, and this number of atoms cubed in 3-d. Thus for three-dimensional systems, locality only begins to emerge for systems of between hundreds and thousands of atoms.

If we wish to accelerate calculations on systems below this size regime, we must therefore introduce additional errors into the calculation, either as numerical noise through looser tolerances, or by modifying the theoretical model, or perhaps both. Our approach to local electron correlation is based on modifying the theoretical models describing correlation with an additional well-defined local approximation. We do not attempt to accelerate the calculations by introducing more numerical error because of the difficulties of controlling the error as a function of molecule size, and the difficulty of achieving reproducible significant results.

From this perspective, local correlation becomes an integral part of specifying the electron correlation treatment. This means that the considerations necessary for a correlation treatment to qualify as a well-defined theoretical model chemistry[25] apply equally to local correlation modeling. The approximations should be:

a) *Size-consistent*: meaning that the energy of a supersystem of two noninteracting molecules should be the sum of the energy obtained from individual calculations on each molecule.
b) *Uniquely defined:* Require no input beyond nuclei, electrons, and an atomic orbital basis set. In other words, the model should be uniquely specified without customization for each molecule.
c) *Yield continuous potential energy surfaces:* The model approximations should be smooth, and not yield energies that exhibit jumps as nuclear geometries are varied.

While these criteria are simple and indeed almost trite to state, they are quite easy to violate, and indeed many procedures for performing efficient calculations of electron correlation effects do not meet the 2^{nd} and 3^{rd} criteria. For example,

in the SP local correlation method, the choice of which substitutions to retain depends on the nuclear geometry[38,39]. Therefore when consistently applied the resulting potential energy surfaces are not smooth, unless the substitutions to keep are defined at a single arbitrary geometry. Similarly the configuration selection involved in current implementations of Nakatsuji's SAC-CI method yields results that depend on geometry, and this either causes potential energy surfaces to be non-smooth or the results must depend on the reference geometry chosen[40]. In the SP method, other difficulties arise in cases where the molecular orbitals do not localize cleanly[41], leading to difficulties in the consistent treatment of molecules like benzene, and to problems in defining the substitutions to retain when treating transition structures. In the new models described later in this part of this section (Section 4.3 and onwards), the requirements of a theoretical model chemistry will be strictly satisfied, by discarding many degrees of freedom according to an atomic criterion.

4.1 Second Order Mller-Plesset Theory

First, let us begin by discussing the equations of our chosen correlation method, MP2 theory, in general terms. A key result from earlier in this chapter is that use of the natural representation leads to equations for electron correlation methods that are isomorphic to traditional orthogonal basis derivations, but substantially more general. This is what we shall illustrate in this section for MP2 theory. The usual formulation of MP2 theory involves the assumption of the canonical molecular orbitals (so that the zero order Hamiltonian is diagonal in the many-electron basis). However, more general "noncanonical" forms of MP2 theory which are invariant to unitary transformations within the occupied and virtual spaces were introduced in the context of gradient theory, and open shell problems[42]. With Pulay and Saebø's local MP2 method[10], a yet more general form was presented, in which nonorthogonal functions were employed to span the virtual function space. Our tensor formalism[7] yields the most general MP2 equations possible, as they are invariant to nonunitary transformations within the occupied and virtual spaces individually.

The basis of Møller-Plesset perturbation theory is partitioning the full Hamiltonian into a zero order (mean field) part, and a first order component which describes the fluctuations associated with electron correlation effects. This may be written in second quantized form as:

$$\hat{H} = \hat{F}^{(0)} + \lambda \hat{V}^{(1)} \tag{58}$$

$$\hat{F}^{(0)} = F^{i}_{\bullet i} + F^{a}_{\bullet b} N\left[a^{\dagger}_{a} a^{b}\right] - F^{i}_{\bullet j} N\left[a^{j} a^{\dagger}_{i}\right] \tag{59}$$

$$\hat{V}^{(1)} = -\tfrac{1}{2} \text{Ⅱ}^{ij}_{\bullet\bullet ij} + F^{i}_{\bullet a} N\left[a^{\dagger}_{i} a^{a}\right] + F^{a}_{\bullet i} N\left[a^{\dagger}_{a} a^{i}\right] + \tfrac{1}{4} \text{Ⅱ}^{pq}_{\bullet\bullet rs} N\left[a^{\dagger}_{p} a^{\dagger}_{q} a^{s} a^{r}\right] \tag{60}$$

F is the Fock operator corresponding to the reference single determinant wavefunction, and, as usual indices *i,j,k...* are occupied, and *a,b,c...* are virtual. The

anti-symmetrized two electron integrals were previously defined in Eq. (43). By placing occupied-virtual elements of the Fock operator into the fluctuation potential (Equation (60)), the single reference is an eigenket of Equation (59) even when these elements are nonzero. The matrix elements and creation and destruction operators entering Equations (59) and (60) have tensorial significance.

Performing perturbation expansions of the full wavefunction together with the Hamiltonian leads to conditions for the first order wavefunction. These first order equations for the perturbed wavefunction may be resolved by projection with singly and doubly substituted bra determinants (in the dual basis of course) to yield equations for the first order single and double substitution amplitudes (contained within the corresponding substitution operators, $\hat{T}_1^{(1)}$and $\hat{T}_2^{(1)}$):

$$\left\langle \Phi^a_{\bullet i} \left| \hat{F}^{(0)} \left| \hat{T}_1^{(1)} \Phi \right. \right\rangle + \left\langle \Phi^a_{\bullet i} \left| \hat{V}^{(1)} \right| \Phi \right\rangle = 0 \tag{61}$$

$$\left\langle \Phi^{ab}_{\bullet\bullet ij} \left| \hat{F}^{(0)} \left| \hat{T}_2^{(1)} \Phi \right. \right\rangle + \left\langle \Phi^{ab}_{\bullet\bullet ij} \left| \hat{V}^{(1)} \right| \Phi \right\rangle = 0 \tag{62}$$

The substituted bras represent the replacement of contravariant occupied functions by contravariant virtual functions.

These equations may be written out in explicit spin-orbital notation quite easily, exploiting the direct analogy between the natural representation and orthogonal basis techniques.

$$F^a_{\bullet b} t^b_{\bullet i} - F^j_{\bullet i} t^a_{\bullet j} + F^a_{\bullet i} = 0 \tag{63}$$

$$F^b_{\bullet c} t^{ac}_{\bullet\bullet ij} + F^a_{\bullet c} t^{cb}_{\bullet\bullet ij} - F^k_{\bullet j} t^{ab}_{\bullet\bullet ik} - F^k_{\bullet i} t^{ab}_{\bullet\bullet kj} + \mathbb{I}^{ab}_{\bullet\bullet ij} \tag{64}$$

Here $\boldsymbol{t}$ are the unknown first order amplitudes, that are now defined in terms of two electron integrals, and matrix elements of the Fock operator.

Finally, the second order expansion of the energy is obtained from the second order expansion of the eigenvalue equation followed by projection with the bra determinant:

$$E^{(2)} = \left\langle \Phi \left| \hat{V}^{(1)} \right| \left(\hat{T}_1^{(1)} + \hat{T}_2^{(1)} \right) \Phi \right\rangle = F^i_{\bullet a} t^a_{\bullet i} + \tfrac{1}{4} \mathbb{I}^{ij}_{\bullet\bullet ab} t^{ab}_{\bullet\bullet ij} \tag{65}$$

Equations (63), (64), and (65) embody all the invariances inherent in MP2 theory: they are invariant to general nonunitary linear transformations within the occupied and virtual subspaces respectively. These equations, despite their perturbational nature, must be solved iteratively because of the off-diagonal Fock terms in Equations (63) and (64). They can be specialized to the textbook[1] canonical case by:

(1) assuming an orthogonal basis so that the covariant-contravariant distinction vanishes),
(2) satisfying the Brillouin theorem (such that $F^a_{\bullet i} = 0$) so that the first order singles are zero, and,

(3) diagonalizing the occupied-occupied and virtual-virtual blocks of the Fock matrix (so that the doubles equations uncouple).

Via the use of metric matrices, it is straightforward to alter the tensor character of the integrals and amplitudes that enter the above equations. As mentioned earlier, the natural representation is particularly convenient for defining the nonorthogonal equations of electron correlation methods, but it is not necessarily the form that we will subsequently use. Rather, since all one and two electron matrix elements are made directly in the given covariant basis, we prefer to use the *covariant integral representation*, which treats all amplitudes as contravariant.

Let us recast the MP2 equations given above into the covariant integral representation. Acting with metric matrices to convert all indices in the matrix elements to covariant and all indices in the amplitudes to contravariant leads to the following alternative form:

$$E^{(2)} = F_{ia}t^{ai} + \tfrac{1}{4}\mathbb{I}_{ijab}t^{abij} \tag{66}$$

$$\big(F_{ab}g_{ij} - g_{ab}F_{ij}\big)t^{bj} + F_{ai} = 0 \tag{67}$$

$$\big[\big(F_{ac}g_{ik} - g_{ac}F_{ik}\big)g_{bd}g_{jl} + g_{ac}g_{ik}\big(F_{bd}g_{jl} - g_{bd}F_{jl}\big)\big]t^{ckdl} + \mathbb{I}_{aibj} \tag{68}$$

Note that in writing Equations (66) and (68) we have chosen to reorder the indexes in the integral and amplitude tensors to group occupied and virtual indexes together. This is in preparation for making local approximations based on keeping only occupied and virtual functions that share a common atom: the atomic single substitution basis that is discussed in more detail later on.

4.2 The Nonorthogonal Orbitals

As discussed in the introduction to this section, we have decided to focus on atom-centered functions, so that our new models for local correlation can be based on *atom-centered selection* of the significant pair correlation amplitudes. The use of atom-based functions is very common within single electron theories of electronic structure such as Hartree-Fock theory and density functional theory. In the context of the tensor-based electron correlation formalism, we require sets of functions spanning the occupied and virtual subspaces.

The simplest approach is to use projected atomic orbitals: projected into the occupied space and the virtual space respectively:

$$|\phi_i\rangle = \hat{P}\,|\phi_\mu\rangle\,\delta_i^\mu = |\phi_\mu\rangle\,P_{\bullet\nu}^{\mu}\delta_i^\nu \tag{69}$$

$$\big|\phi_a\big\rangle = \hat{Q}\big|\phi_\mu\big\rangle\delta_a^\mu = \big(\hat{I} - \hat{P}\big)\big|\phi_\mu\big\rangle\delta_a^\mu = \big|\phi_\mu\big\rangle\big(\delta_a^\mu - P_{\bullet\nu}^{\mu}\delta_a^\nu\big) \tag{70}$$

These functions are assigned to atoms by the atom label of the parent atomic orbital, thus partitioning the redundant functions amongst the atoms according to

the number of AO's on each atom in a manner that is well-defined for any atomic arrangement. The resulting functions are exponentially localized for insulators, as they have the decay properties of the one-particle density matrix **P** itself. The use of functions of this type is satisfactory for the virtual space and was pioneered successfully in the earlier local correlation models of Pulay and Saebø. However projected atomic orbitals are highly redundant (and hence extremely inefficient) for the occupied space. Increasing the number of basis functions per atom would increase the number of functions spanning the occupied space, even though the dimension (rank) of the occupied space itself is unaltered!

The simplest atom-centered basis that is capable of spanning the occupied space is a *minimal basis* of core and valence atomic orbitals on each atom. Such a basis is necessarily redundant because it also contains sufficient flexibility to describe the empty valence antibonding orbitals necessary to correctly account for nondynamical electron correlation effects such as bond-breaking. Atom-optimized minimal basis sets (e.g. of the STO-nG type) are of course notoriously poor in practical calculations. Yet it is still quite possible to define a localized minimal basis that spans the occupied space at the end of a large basis set calculation. We term such functions *extracted polarized atomic orbitals* (EPAO's). It is even possible to explicitly perform an SCF calculation in terms of a molecule-optimized minimal basis of *polarized atomic orbitals* (PAO's). These two approaches each have strengths and weaknesses, and we discuss them in turn below.

First we summarize the situation. The number of functions spanning the occupied subspace will be the minimal basis set dimension, M, which is greater than the number of occupied orbitals, O, by a factor of up to about 2. The virtual space is spanned by the set of projected atomic orbitals whose number is the atomic orbital basis set size N, which is fractionally greater than the number of virtuals V=N-O. The number of double substitutions in such a redundant representation will be typically 3 to 5 times larger than the usual total. This will be more than compensated by reducing the number of retained substitutions by a factor of the number of atoms, A, in the local triatomics in molecules model, or a factor of A^2 in the diatomics in molecules model.

4.2.1 Extracted Polarized Atomic Orbitals

The *extracted polarized atomic orbitals* (EPAO's)[43] of a molecule are a minimal basis set that is derived subsequent to an extended basis electronic structure calculation. Thus the EPAO's are themselves linear combinations of AO's: $|\alpha\rangle = \sum_\mu B_{\alpha\mu} |\mu\rangle$. Based on numbers of core and valence atomic orbitals, the number of EPAO's for any given atom is 1 for H and He, 5 for first row elements, 9 for second row elements, etc. The set of EPAO's on each atom will be invariant to rotations amongst individual EPAO's on that atom. For simplicity, we will constrain the full set of EPAO's to form an orthonormal set, $\langle \alpha \mid \beta \rangle = \delta_{\alpha\beta}$.

EPAO's will be determined analogously to localized occupied orbitals in that an EPAO delocalization functional, L, is minimized with respect to variations of the atomic subspaces. Furthermore, these variations are subject to the constraint that the full space of EPAO's completely contains the one-particle density projector,

$\hat{\rho} = \sum_i |i\rangle\langle i|$, which can be abbreviated by the expression $\hat{\rho} \subset \hat{\tau}$, where the full space of EPAO's, $\hat{\tau}$, is the union of all the atomic subspaces in a system, $\hat{\tau} = \sum_A \hat{\tau}_A$.

Many definitions of the delocalization functional are possible, and a number have been explored. The choice we shall employ here is to minimize the second moment of the EPAO's, with the imposed condition that their centers coincide with the atomic centers:

$$L[\hat{\tau}_A, \hat{\tau}_B, ...] = \underset{\hat{\rho} \subset \hat{\tau}}{Min} \left\{ \sum_A k_A \sum_{\alpha,\beta \in A} \langle \alpha | (\hat{r} - z_A)^2 | \beta \rangle D_{\alpha\beta} \right\} \tag{71}$$

$D_{\alpha\beta} = \langle \alpha | \hat{\rho} | \beta \rangle$ is the one-particle density matrix evaluated in the EPAO basis. This occupancy weighting reduces the sensitivity of the results to the presence of empty or nearly empty orbitals in the minimum basis. The virtual part of the basis is still fully determined because the occupancy weighting is not a full projection into the occupied space. Equation (14) contains an atomic scaling factor, k_A, which we have included to make the definition of the localization criterion as general as possible.

A reasonable choice of the scaling factor k_A, will be to roughly normalize the radii of valence EPAO's of elements from different rows of the periodic table. We use a very simple scaling factor that achieves this objective, which is the inverse of the principal quantum number, n, of the atom. Thus, for elements H and He, the factor is 1; for elements Li through Ne, the factor is 1/2; for elements Na through Ar, the factor is 1/3, and so on. This choice very roughly mirrors trends in the sizes of the noble gas atoms, and has the merit of involving no adjustable parameters. Finally, the localized atom-centered occupied orbitals are obtained by projection of the EPAO's into the occupied space as $|i_A\rangle = \hat{P}|\alpha_A\rangle$. The projected functions remain atom-centered in general, but are linearly dependent as their number M is greater than the dimension of the occupied space, O.

4.2.2 Polarized Atomic Orbitals

Instead of extracting the minimal basis after the conclusion of an SCF calculation in an extended basis, can one directly perform the SCF calculation in terms of a molecule-optimized minimal basis set of polarized atomic orbitals? We have shown[44] that such an approach is feasible as long as a constraint is imposed on the form of the PAO's: namely that the PAO's are linear combinations only of the atomic orbitals on the corresponding atom. So, given a (large) secondary basis $\{|\phi_\mu\rangle\}$, the (small) basis of PAO's $\{|\phi_\alpha\rangle\}$ are defined via an atom-blocked (let PAO index α belong to atom A) rectangular transformation, $\mathbf{B}$:

$$|\phi_\alpha\rangle = \sum_{\mu \in A} |\phi_\mu\rangle B_{\mu\alpha} \tag{72}$$

The transformation coefficients are determined as part of the variational Hartree-Fock calculation, in which the density matrix or molecular orbitals are defined in terms of PAO's.

The PAO's appear very promising as a method for performing Hartree-Fock (HF) (or other self-consistent field methods such as density functional theory) calculations on large molecules. The computational requirements are reduced significantly because a PAO calculation does not require full matrices to be evaluated in the large atomic orbital basis. Additionally a PAO calculation by definition provides a minimal basis set of atom-centered functions that span the occupied space (after projection with the density operator).

It is important to emphasize that as a result of the atom-blocked constraint, the PAO-HF energy is necessarily slightly higher than the HF energy directly computed in the secondary atomic orbital basis. We have reported two studies of the magnitudes of these differences both for absolute and relative energies[44,45]. In terms of absolute energies, the error introduced by the PAO model is generally smaller than normal basis set incompleteness effects. Relative energies are quite well reproduced, such that the PAO model dramatically outperforms any conventional minimal basis set. Errors in absolute and relative energies can be reduced by employing larger secondary basis sets in the PAO calculation. Some preliminary studies of the convergence of results with size and composition of the secondary basis have been completed[45].

The appeal of PAO's for local MP2 calculations is that the atom-blocked form of the transformation to PAO's can potentially accelerate the first two steps of the atomic orbital to molecular orbital two-electron integral transformation. These steps are often rate-determining, particularly for large basis sets. We do not report any electron correlation calculations using PAO's in this chapter, but this subject is under active study in our laboratory[46].

4.3 Atoms in Molecules Models of Local Correlation

We are now at the stage where local models of electron correlation can be formulated and tested. The developments described in the previous sections constitute a formally exact re-expression of standard theories of electron correlation, using MP2 theory as the working example. In the traditional formulation, the nonredundant canonical molecular orbitals are employed to express many-electron theories. The canonical MO's are in general fully delocalized throughout the system. In our re-expression of electron correlation theories, redundant sets of atom-centered functions are used to express the working equations. This takes advantage of the fact that correlation theories are fundamentally independent of the choice of the functions used to span a given occupied and virtual space. As described above, the occupied space is spanned by a minimal basis set of distorted (polarized) atomic orbitals that are projected into the occupied space. This is the smallest set of atom-centered functions capable of performing this role. The virtual space is spanned by the full atomic orbital basis projected into the virtual space. As a result of using atom-centered functions, we can truncate the *overcomplete* set of double substitutions based on an atomic criterion, so as to satisfy the requirements of a theoretical model chemistry, as discussed at the start of Section 4.

At a conceptual level, we can distinguish four possible levels of truncation, based on increasingly severe atomic truncations:

a) *Tetra-atomics in molecules.* If no truncation (apart from numerical cutoffs) is employed, substitutions will generally occur from occupied atomic-like orbitals centered on two atoms to virtual atomic-like orbitals centered on two other atoms. As discussed above, based on sparsity considerations, we expect cross-overs relative to conventional MP2 will occur only for quite large systems. Again, this is the motivation for considering stronger, ansatz-based truncations.

b) *Triatomics in molecules* (TRIM) is the level of modeling where only double substitutions with one occupied and one virtual orbital on a common atom are retained. This immediately reduces the number of retained double substitutions to no more than A^3 in the number of atoms, A. Substitutions that are doubly ionic (and often identified with basis set superposition error) are eliminated, while singly ionic and covalent substitutions are retained.

c) *Diatomics in molecules* (DIM) is the case where one occupied-virtual substitution is restricted to be on a common atom, while the other occupied-virtual substitution is restricted to be on another atom. This eliminates all ionic substitutions, and so may alternatively be called the covalent ansatz[21,22]. The correct long-range dispersion force can still be recovered at this level of truncation.

d) *Atoms in molecules* is the case where the only retained double substitutions are those for which all occupieds and virtuals are on a common atom. Long-range dispersion is no longer recovered, and therefore we shall not consider this model further in the present context. However, it is of interest for treating nondynamical correlation associated with bond-breaking, as we discuss briefly in Section 5.2.

For practical purposes, there are then three levels of local correlation model treatment that may be distinguished in the context of MP2 theory: exact, triatomics, and diatomics. In the following sections, we shall develop the explicit expressions necessary to evaluate the energy associated with these models of local electron correlation.

4.4 Atomic Single Substitution Basis

Any set of orbital substitutions describing correlated fluctuations of electrons from a mean field single determinant starting point can be represented in terms of direct products of single substitutions. For example the set of double substitutions is the direct product of the space of single substitutions with itself, the set of triple substitutions is a 3-way direct product of the set of single substitutions, and so forth.

We have expressed the space of occupied functions in terms of a minimal basis of atom-centered functions, and the space of virtual (unoccupied) functions is spanned by a redundant set of projected atomic orbitals. This means that the set of single substitutions may now be divided into two categories:

(1) *Atomic Single Substitutions* (ASIS): These are single replacements where both the occupied function and the virtual function belong to the same atom.

(2) *Interatomic Single Substitutions*: Single replacements where the occupied function and the virtual function belong to different atoms.

The basis of our local truncations will be to retain only the ASIS basis for some (or all) of the single substitutions that enter into electron correlation theories. This approximation will express the locality inherent in electron correlation: namely excitations that involve transporting electrons between spatially different regions of a molecule are energetically insignificant. Retention of only the ASIS substitutions is actually not quite as drastic an approximation as it first appears. Since the functions spanning both the occupied and the virtual spaces are both nonorthogonal and linearly dependent, the ASIS space and the space spanned by the interatomic single substitutions are not disjoint. In fact the ASIS space includes a significant fraction of the excitations on neighboring (bonded) atoms.

The dimension of the ASIS set grows linearly with the number of atoms, with the coefficient being the average number of occupied-virtual product functions on an atom. By contrast, of course, the full set of single substutions grows quadratically with the number of atoms. Therefore we are going to reduce the number of amplitudes by one power of system size for each pair of occupied-virtual indices for which we make an ASIS approximation. Specifically, we can now symbolically write the form of both the DIM and TRIM models.

The full MP2 model involves the direct product of all single substitutions with themselves:

$$\{aibj\}_{full} = \{a\} \otimes \{i\} \otimes \{b\} \otimes \{j\} \tag{73}$$

The DIM model applies the atomic single substitution approximation to the two sets of single substitutions whose direct product defines the double substitutions. Denoting an ASIS substitution as (ai) where the use of the parenthesis reminds us that both occupied and virtual functions are centered on common atoms, we may write the space of DIM substitutions as:

$$\{aibj\}_{DIM} = \{(ai)\} \otimes \{(bj)\}\,. \tag{74}$$

Clearly electrons on pairs of atoms are correlated in this way, and, as discussed below, this is sufficient to correctly describe long-range dispersion interactions.

The TRIM model corresponds to applying the atomic single substitution approximation only to one of the two single substitutions whose direct product comprises the double substitutions of MP2 theory. To make such a model consistent the correlation space should be the union of the two possible ways of doing this. The TRIM set of double substitutions may then be written as follows:

$$\{aibj\}_{TRIM} = \{(ai)\} \otimes \{b\} \otimes \{j\} \oplus \{a\} \otimes \{i\} \otimes \{(bj)\}\,. \tag{75}$$

The TRIM model expands the DIM substitutions to explicitly include single electron transfers between atoms that can be separated. Full double substitutions, TRIM doubles and DIM doubles form a well-defined hierarchy of models for local correlation.

Roughly how do we expect the DIM and TRIM models to compare to the existing Saebø-Pulay (SP) model for local correlation? Or, alternatively, what does that model look like in terms of an expression like Equation (74) or (75)? If it were implemented using atomic truncation criteria, the set of retained double substitutions in the SP method could be expressed as follows:

$$\{aibj\}_{SP} = \{(ai)\} \otimes \{(bj)\} \oplus \{i\} \otimes \{(abj)\} \oplus \{(iab)\} \otimes \{j\}\,. \tag{76}$$

In Equation (76), quantities in parentheses are centered on a common atom. Thus the SP model can be expected to lie intermediate between our DIM and TRIM models. The first term in Equation (76) is the direct product of ASIS terms that are responsible for long-range correlation, and are included in the DIM model. The 2^{nd} and 3^{rd} terms are single electron transfers between pairs of atoms, in which charge is consolidated from one electron on the atom of orbitals i and j to two electrons on the atom of either orbital i or orbital j. These single electron transfers are a *subset* of those included in the TRIM model. TRIM additionally includes substitutions in which charge is separated from one atom to two atoms, as well as singly ionic transfers that couple three atoms together rather than two.

4.5 Orthogonalizing and Canonicalizing the ASIS Basis

Recall for a moment how the molecular orbitals are treated in conventional electronic structure theory. Normally they are first orthogonalized, which eliminates all overlap matrices from the equations, and then they are canonicalized, meaning that they are transformed to a basis which makes the Fock operator diagonal. As a result, the MP2 equations uncouple (for example, refer back to the steps listed after Equations (65), and the MP2 correlation energy can be obtained without iterations.

It is possible (and indeed, probably necessary) to do exactly the same things with the ASIS basis. We can orthogonalize, such that the overlap matrix (or metric) becomes diagonal. The ASIS overlap matrix is merely the direct product of the occupied and virtual overlaps. Denoting indexes in the ASIS basis as *K,L...* (they are equivalent to atomic (ai), (bj)... pairs), we can write:

$$G_{KL} \equiv G_{(ai)(bj)} = g_{ab} g_{ij} \tag{77}$$

This overlap matrix can be explicitly diagonalized (and thus inverted by singular value decomposition) quite readily for systems of up to 150 heavy atoms or so (the computational effort scales with the cube of molecule size). This yields the orthogonalizing transformation $\mathbf{G}^{-1/2}$. The orthogonalized ASIS basis will be called the OASIS basis.

Why the need for orthogonalizing the ASIS basis via singular value decomposition? The reason that explicit orthogonalization is probably necessary is that the condition number of the ASIS metric, Equation (77), is tremendously large, as a result of making the atomic truncation, and using nonorthogonal and linearly dependent functions. This means that while the matrix described by Equation (77) becomes sparse as the system size grows large, iterative methods to invert the metric will perform tremendously poorly. If we had not made the ASIS approximation

this would not necessarily be the case, since then we could invert separately the occupied and virtual metrics entering Equation (77).

The relevant energy matrix in the ASIS basis is evident in Equations (68), and is essentially a matrix of energy differences, which we shall denote as **D**:

$$D_{KL} \equiv D_{(ai)(bj)} = \left(F_{ac}g_{ik} - g_{ac}F_{ik}\right) \tag{78}$$

Canonicalizing this matrix requires two steps. First, **D** must be transformed into the OASIS basis (which we shall denote with overbars):

$$D_{\bar{K}\bar{L}} = \sum_{K,L} G^{-1/2}_{K\bar{K}} D_{KL} G^{-1/2}_{L\bar{L}} \tag{79}$$

The second step is to diagonalize **D** in this orthogonalized basis:

$$D_{\bar{K}\bar{L}} = \sum_{\bar{P}} U_{\bar{K}\bar{P}} E_{\bar{P}} U_{\bar{L}\bar{P}} \tag{80}$$

The product of these two transformations is the matrix **T** that transforms from the ASIS basis to the canonicalized OASIS (or COASIS) basis:

$$T_{K\bar{P}} = \sum_{K,\bar{K}} G^{-1/2}_{K\bar{K}} U_{\bar{K}\bar{P}} \tag{81}$$

4.6 Energy Expression

The MP2 spin-orbital expression for the energy that we shall use is:

$$E_{MP2} = \tfrac{1}{2} I_{aibj} t^{aibj} \tag{82}$$

where $I_{aibj} \equiv (ai\,|bj\,) = \langle ab\,|ij\,\rangle$ are unsymmetrized two-electron integrals. In Mulliken notation, such integrals correspond simply to the Coulomb interaction between the charge distribution corresponding to the product of virtual a and occupied i, and the charge distribution due to the product of virtual b and occupied j. The first order amplitudes are denoted as t^{aibj}. Relative to our expression in Equations (65) and (66), we are:

(a) Using unsymmetrized rather than symmetrized integrals. This leads to the factor of 1/2 instead of 1/4, and of course makes no difference in the absence of any local truncation. In the presence of local truncation, as discussed further below, Equation (82) is a preferable starting point.
(b) Omitting the singles substitution contribution to the energy. This is because, if present, we shall treat it exactly, and we wish to focus attention on the critical doubles contribution.

While Equation (82) is equivalent to the previous expressions in the absence of local truncations, it is important to emphasize that the result of making local truncations will be different depending on which starting point is used. This should

not be disturbing upon reflection: it is simply a consequence of the fact that local modeling breaks many of the invariances that are present in the original nonlocal equations. Equation (82) is a preferable starting point because it will yield 100% of the long-range correlation energy in the context of the DIM model. By contrast, using Equation (66) will yield only 50% of the long-range correlation energy under DIM truncation!

4.7 Diatomics in Molecules (DIM)

With the definitions made already it is now straightforward to write explicit expressions for the diatomics in molecules (DIM) local correlation model for the MP2 energy. Instead of retaining all significant doubles amplitudes, we shall instead retain only the direct product of ASIS functions, as specified by Equation (74). Therefore Equation (68) for the doubles amplitudes may be rewritten as follows in the ASIS basis, while Equation (82) for the MP2 energy is given immediately below:

$$\left[D_{KM}G_{LN} + G_{KM}D_{LN}\right]t^{MN} + \mathbb{I}_{KL} \tag{83}$$

$$E_{DIM-MP2} = \tfrac{1}{2}I_{KL}t^{KL} \tag{84}$$

We have introduced the ASIS metric matrix $\mathbf{G}$, and energy difference matrix $\mathbf{D}$ defined previously in Equations (77) and (78) respectively.

As was implicit in the previous section describing canonicalizing the ASIS basis, iterative solution of Equations (83) is a very challenging problem due to the very large condition number of the ASIS overlap matrix, $\mathbf{G}$. Hence, at present the only viable method for obtaining the DIM energy is to do it non-iteratively, by direct evaluation in the COASIS basis. The unsymmetrized and antisymmetrized integrals must be transformed to the COASIS basis, using the transformation defined by Equation (81), and the energy may then be obtained explicitly as:

$$E_{DIM-MP2} = -\frac{1}{2}\sum_{\bar{P}\bar{Q}}\frac{I_{\bar{P}\bar{Q}}\mathbb{I}_{\bar{P}\bar{Q}}}{E_{\bar{P}}E_{\bar{Q}}} \tag{85}$$

Equation (85) for the DIM-MP2 energy bears considerable similarity to the normal expression for the canonical MP2 energy, but as a result of performing manipulations in the atom-centered ASIS basis, we find:

(1) There are a quadratic number of energy contributions instead of the usual quartic number. We shall use the notation that capital letters denote extensive quantities such the AO basis size N, or the minimal basis size, M, while lower case letters denote intensive quantities, such as the number of minimal basis functions per atom, m, or the number of AO's per atom, n. The number of substitutions has been reduced from the usual O^2V^2 to m^2N^2 this way, which is a saving proportional to the number of atoms squared.

(2) The rate-determining computation steps will be cubic rather than 5^{th} order in the size of the molecule. Such steps will correspond to the COASIS canonicalization, and transformations of the unsymmetrized and antisymmetrized two-electron integrals into the COASIS basis, as required for Equation (85). These steps involve computational effort proportional to m^3N^3.

4.8 Triatomics in Molecules (TRIM)

The TRIM substitutions were described symbolically in Equation (75): this local correlation model applies the ASIS approximation to only one of the two pairs of single substitution indices that comprise the double substitutions. Unfortunately, the energy associated with the TRIM model cannot easily be obtained without iterations. The TRIM ansatz is not a direct product, and therefore the coefficient matrix coupling the retained substitutions in for example Equation (68) does not decompose into direct products of smaller matrices. However since the model still involves application of the ASIS approximation, we expect the condition number of the coefficient matrix to be extremely large (even after discarding any null space of zero eigenvalues). Therefore we are not optimistic about being able to efficiently converge such equations by iterative methods. The focus of our efforts has been to develop a viable noniterative alternative. The result of these investigations is described below.

4.8.1 Half-TRIM Energy

If we adopt a triatomics in molecules model that included only half of the substitutions that are incorporated in Equation (75), the resulting model has direct product structure:

$$\{aibj\}_{half-TRIM} = \{(ai)\} \otimes \{b\} \otimes \{j\} \tag{86}$$

As a result, it can be solved noniteratively by the same COASIS recanonicalization we developed earlier, and then applied above to develop an explicit expression for the DIM-MP2 energy.

The half-TRIM energy and amplitude equations may be written out in general form as restrictions of Equations (66) and (68), using the ASIS overlap and energy difference matrices **G** and **D**. The resulting equations may also be compared and contrasted with the corresponding DIM equations, (83) and (84):

$$\big[D_{KL}g_{bc}g_{jk} + G_{KL}\big(F_{bc}g_{jk} - g_{bc}F_{jk}\big)\big]t^{Lck} + I\!I_{Kbj} \tag{87}$$

$$E_{half-TRIM-MP2} = \tfrac{1}{2}I_{Kbj}t^{Kbj} \tag{88}$$

The noniterative solution to the half-TRIM MP2 problem employs the COASIS basis for the atomic single substitution indexes, thus bringing **G** and **D** to diagonal form. The remaining occupied and virtual index can be represented in the conventional canonical orthogonal basis (denoted also by overbars), so that the energy may be directly evaluated from the following expression:

$$E_{half-TRIM-MP2} = -\frac{1}{2}\sum_{\bar{P}\bar{b}\bar{j}} \frac{I_{\bar{P}\bar{b}\bar{j}} I\!I_{\bar{P}\bar{b}\bar{j}}}{E_{\bar{P}} + \epsilon_{\bar{b}} - \epsilon_{\bar{j}}} \tag{89}$$

The computational requirements of Equation (89) are roughly as follows:

(1) The number of substitutions is reduced from growing as the 4^{th} power of molecule size (O^2V^2) to the 3^{rd} power of molecule size: specifically $mNOV$. This saves a factor proportional to the number of atoms.
(2) The significant computational steps scale either with the 3^{rd} and 4^{th} powers of the size of the molecule. Obtaining the transformation to the COASIS basis is m^3N^3, as for the DIM model, but the transformation of the two-electron integrals now involves a step proportional to the 4^{th} power of molecule size, in addition to steps scaling with the 3^{rd} power of molecule size. A factor at least proportional to the number of atoms can be saved relative to conventional transformations.

It is also worth noting that a compromise for the evaluation of the half-TRIM energy is possible, in which the ASIS indices are canonicalized (because the problem of large condition numbers has its origin in the ASIS approximation), but the remaining occupied and virtual indices are treated in a local basis. This may be the method of choice for the largest calculations using this method, because sparsity can then be exploited in the treatment of the nonlocal substitutions. Asymptotically the computational effort in this formulation will scale as the cube of molecule size, corresponding to the COASIS steps.

4.8.2 TRIM Energy Additivity Expression

The half-TRIM model will not be very close to the full TRIM model. We can predict that it will give an energy that is approximately half way between the DIM and TRIM models, because it includes only half of the additional substitutions in TRIM, while nevertheless using an unmodified energy expression. The energy expression, Equation (89), includes a factor of 1/2 based on double-counting all contributions, as is done in the DIM model. Yet the half-TRIM model includes only one copy of the new substitutions, and therefore their contribution to the energy will be undercounted by a factor of two. Recovering only about 50% of the TRIM-DIM correction is an unsatisfactory approximation, and we must do better.

The most obvious solution is to consider approximating the TRIM model by modifying the energy expression to correct such undercounting. We cannot simply double the half-TRIM energy, because then we over-count the DIM substitutions by a factor of two. However this can be corrected by subtracting the DIM energy itself, under the assumption that the omitted substitutions do not affect the calculated values. This result is the TRIM model we shall implement in practice. The associated energy is simply:

$$E_{TRIM-MP2} = 2E_{half-TRIM-MP2} - E_{DIM-MP2} \tag{90}$$

While Equation (90) is probably the simplest noniterative way to approximate the TRIM model, there are also other possibilities. However, based on numerical tests, we have not yet found any alternative that is superior to Equation (90).

4.9 Numerical Tests of the DIM And TRIM Models

Our primary purpose in this part of the chapter has been to discuss the development of the theory underlying the atoms in molecules models of local electron correlation. It is nonetheless still interesting to present a small selection of results that establishes the comparative performance of the local DIM and TRIM models relative to untruncated ("full") MP2 theory. The local MP2 methods have been implemented in a development version of the Q-Chem program suite[47] , which was used for all calculations. The results reported here are taken from our recent report on the TRIM model[23]. All calculations employ EPAO's to span the occupied space.

Details of the efficient implementation of the theory described above are reported in a recent thesis[48], and a paper in preparation. Here we simply summarize the capabilities of our program. The computational advantage associated with these local MP2 methods varies depending upon the size of molecule and the basis set. As a rough general estimate, TRIM-MP2 calculations are feasible on molecule sizes about twice as large as those for which conventional MP2 calculations are feasible on a given computer, and this is their primary advantage. Our implementation is well suited for large basis set calculations, because the memory requirement for the integral transformation does not exceed OON, and is thresholded so that it asymptotically grows linearly with molecule size. Additional memory of approximately $32N^2$ is required to complete the local MP2 energy evaluation. The disk space requirement is only about $8OVN$, but is not thresholded. The integrals are evaluated four times. DIM-MP2 calculations are faster than TRIM-MP2 and do not require disk storage, but have similar memory requirements.

4.9.1 Asymptotic Correlation Energy Recovery in Polyenes

The simplest measurement of accuracy for any local MP2 method is the percentage of untruncated MP2 correlation energy recovered. Any local method is exact for an atom, and many are exact for diatomic molecules. Therefore the quantity of interest is the limiting percentage of correlation energy recovered as molecule size becomes large. In Figure 1, the percentage of the full MP2 correlation energy recovered by the DIM and TRIM methods is plotted for each additional increment to a polyene chain respectively. The 6-31G* basis was used for these calculations.

First, let us examine the TRIM results, which are quite remarkable. It is clear that virtually no correlation energy is lost as a result of discarding what is asymptotically almost all of the double substitutions (the fraction kept is proportional to the inverse of the number of atoms). So little correlation energy is lost that the asymptotic correlation energy recovery is more than 5 times better than is commonly reported with the Saebø-Pulay model, which is perhaps typically 98%. This very high fidelity in recovering absolute correlation energy bodes well for the ability of the TRIM model to reliably reproduce relative energies also, as we investigate in the following two subsections. We also note that while the TRIM correlation energy

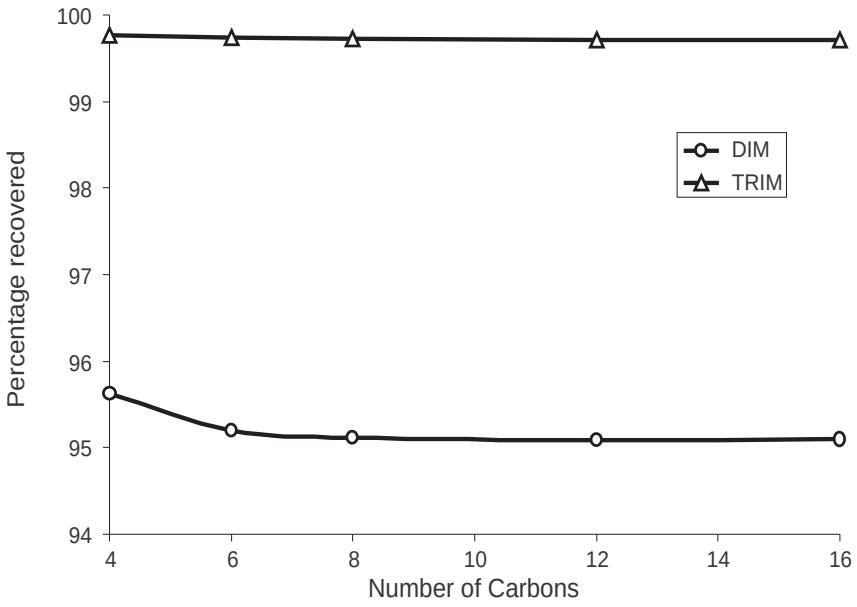


Figure 1. Percentage of the full (untruncated) MP2 correlation energy recovered by the diatomics in molecules (DIM) and triatomics in molecules (TRIM) local correlation methods for all-trans polyene chains of increasing length. Since local correlation models are usually exact for atoms and exhibit gradually decreasing fractional correlation energy recovery as the size of the molecule increases, we report the fractional correlation energy recovery for just the new increment of chain length. The 6-31G* basis was used for these calculations, and all orbitals were correlated.

recovery is always below 100% in the results reported here, it does not appear to obey a Hylleraas type variational principle with respect to the full MP2 correlation energy.

The second point of interest is the performance of the DIM method, which keeps a fraction of the double substitutions that is proportional to the inverse of the number of atoms squared. While no explicitly ionic substitutions are retained, the fraction of correlation energy recovered drops only by about 5% or so. This is about as good as could reasonably be expected, and suggests that the use of linearly dependent nonorthogonal functions must be accounting for a good fraction of local (nearest neighbor) ionic substitutions. Of course, one must also remember the large magnitude of the correlation energy (roughly 1 eV per spatially proximate pair of electrons) versus the small values of relative energies that are often of interest in chemistry. In this light it is nevertheless unclear if the DIM correlation energy recovery is sufficient to reliably treat relative energies.

4.9.2 Rotational Barriers in Conjugated Molecules

The second set of test calculations are MP2 corrections to the rotational barriers of vinyl and phenyl- substituted molecules[49]. This is a rather stringent test of the accuracy of a local MP2 method given that the barrier is a small difference in energy relative to the absolute energy. Thus it provides a reasonable assessment of the stability of the correlation energy recovery in a local method with respect to changes in the potential energy surface. Furthermore, the molecules are conjugated and the origin of the barriers is (nonlocal) resonance stabilization, which is also particularly challenging for local correlation methods. Thus success in this class of test problems bodes extremely well for success in the reproduction of relative energies in general.

Table 1 contains a summary of the individual results and the RMS and maximum errors (relative to untruncated MP2) as a result of (a) neglecting correlation, (b) employing DIM, and, (c) employing TRIM. The results indicate that TRIM deviates on average by only 0.03 kcal/mol, when compared to the full MP2 results. The DIM results show deviations nearly ten times as large (0.2 kcal/mol). The DIM deviations are nevertheless substantially smaller than simply neglecting electron correlation effects. Therefore the main conclusions are that the TRIM results reproduce untruncated MP2 to a precision which is more than satisfactory for routine chemical applications, and even DIM is certainly acceptable for this quite challenging set of problems.

It is also important to emphasize that both the TRIM and DIM results for relative energies are significantly better than the reproduction of total MP2 correlation energies might imply. In other words there is significant cancellation of errors in the evaluation of relative energies. The extent to which this is so can be partly answered from Table 1. Suppose the absolute MP2 correction to the energy of a vinyl-substituted molecule is 0.8 hartree or 500 kcal/mol (or about double, or 1000 kcal/mol for a substituted benzene). The TRIM method, for instance, might produce an error anywhere from 0.1% to 0.2% leading to a range of 0.5 kcal/mol. (or 1 kcal/mol for the substituted benzenes). Nonetheless, the TRIM rotational barriers are well within 0.1 kcal/mol. Thus the percentage of MP2 correlation energy recovered is quite stable with respect to conformational changes, due in large part to the associated smoothness of the EPAO description of the occupied space. This argument is even more important in accounting for the fairly reasonable results obtained with the DIM model.

4.9.3 Water Dimer Interaction Energy

The water dimer exhibits a significant correlation contribution to the hydrogen-bonding energy of binding. Furthermore due to the donor-acceptor nature of hydrogen bonding, it is possible that the correlation effects are not highly localized. The water dimer is thus a good system to assess the performance of the DIM and TRIM models for treating hydrogen-bonding interactions. MP2 energy corrections to the association energies of water clusters were assessed with the aug-cc-pVXZ series of basis sets, where X=D,T,Q. The geometry of the water dimer was obtained from a CCSD(T)/aug-cc-pVTZ calculation[50] . The results, for the correlation con-

Table 1. MP2/6-311G** corrections to Hartree-Fock relative energy differences in twelve conjugated molecules (kcal/mol)[a,b]. The conjugated molecules are substituted ethylenes of the form Vi-X, where Vi is the vinyl group, and substituted benzenes of the form Ph-X, where Ph is the phenyl group.

Molecule	**Conformation**	**TRIM**	**DIM**	**null**	**Full MP2**
Vi-C_2H_3	rotation	-0.249	0.030	0.000	-0.288
Vi-C_2H_3	gauche	-0.508	-0.182	0.000	-0.530
Vi-CFO	rotation	-0.913	-0.926	0.000	-0.930
Vi-CHO	rotation	-0.064	-0.028	0.000	-0.085
Vi-NH_2	rotation	0.770	0.605	0.000	0.806
Vi-NO_2	rotation	-1.795	-1.717	0.000	-1.834
Vi-OH	rotation	1.012	1.063	0.000	0.999
Vi-OH	anti	0.058	-0.084	0.000	0.059
Ph-OH	rotation	0.824	1.033	0.000	0.810
Ph-NO_2	rotation	-2.296	-2.524	0.000	-2.316
Ph-C_2H_3	rotation	-0.423	0.156	0.000	-0.506
Ph-CFO	rotation	-1.423	-1.694	0.000	-1.423
Ph-CHO	rotation	-0.459	-0.405	0.000	-0.493
Ph-NH_2	rotation	0.579	0.616	0.000	0.572
Ph-NH_2	inversion	0.735	0.548	0.000	0.741
RMS Error		0.031	0.252	1.022	
Max Error		0.083	0.662	2.316	

a All calculations were performed at HF/6-31G* optimized geometries, with all electrons correlated.

b The results exclude the Hartree-Fock contribution, values for which have been reported in ref. 49.

Table 2. Comparison of several local MP2 models for the calculation of the MP2 correction to the water dimer[a] association energy[b] (kcal/mol). These models are the triatomics in molecules (TRIM) model, as developed here, the Saebø-Pulay model (SP), and the diatomics in molecules (DIM) model.

Basis Set	**Full**	**TRIM**	**% Full**	**SP**[c]	**% Full**	**DIM**	**% Full**
aug-cc-pVDZ	-1.406	-1.316	94%	-0.52	37%	-0.191	14%
aug-cc-pVTZ	-1.541	-1.478	96%	-0.96	62%	-0.503	33%
aug-cc-pVQZ	-1.482	-1.432	97%	-1.17	79%	-0.804	54%

a All calculations were performed at the CCSD(T)/aug-cc-pVTZ optimized geometry[50].
b Only valence orbitals are correlated. The current best estimate[50,53] of the complete basis set limit frozen core MP2 value at the Halkier geometry[50] is 1.37 kcal/mol.
c Frozen core Saebø-Pulay local MP2 results from the calculations of Schütz *et al.*[54], also at the Halkier geometry.

tribution only, are reported in table 2. No counterpoise corrections were performed, because we wish to assess the convergence of the calculations towards the estimated complete basis set limit of 1.37 kcal/mol[50,51,52,53].

From table 2, the TRIM model closely tracks the untruncated MP2 values, with the difference gradually decreasing to 0.06 kcal/mol by the aug-cc-pVQZ basis. Table 2 also contains calculations using the SP model from ref. 54 and DIM calculations. The SP model yields results that are intermediate between the DIM and TRIM models, as might be anticipated from the discussion in Sec. 4.4.

It is noteworthy that the fraction of the correlation contribution to the binding energy that is recovered by the SP and DIM models increases strongly as the basis set size increases. This reflects an important point about these (and indeed all) local truncations. Both DIM and SP are exact in the limit in which each atom's set of basis functions approach global completeness. Therefore the quality of results obtained by these truncations improves markedly as the basis set size increases. The much milder truncation involved in the TRIM model results in less pronounced basis set dependence of the truncation effects. The SP model, and in particular the DIM model, recover a disturbingly low fraction of the interaction energy for the smaller basis sets, and therefore should probably not be used for hydrogen-bonding problems except with the larger basis sets.

5 An Overview of Other Applications

Space and time considerations prohibit us from giving an equally detailed discussion of our other recent applications of tensor ideas to electronic structure theory of many-electron systems. Instead, the following short discussion provides an intro-

ductory overview of several related research projects that we have been pursuing. Full details are available in the primary literature. The purpose of these short discussions is to give more examples of the usefulness of tensor methods in electronic structure theory.

5.1 Local Electron Correlation Models for Triple Substitutions

The local correlation methods discussed above for double substitutions, in the context of MP2 theory, can be extended to yield local correlation methods for triple substitutions[55,56]. Triple substitutions are essential for predicting chemical reaction energies to 1 kcal/mol or better, and are employed in the widely used G2 and G3 thermochemical methods, where the rate-determining step is usually the triples contribution to the fourth order Møller-Plesset energy: MP4(T). The cost of methods including triple substitutions, such as the widely used MP4(T) and CCSD(T) methods, scales as the seventh power of molecule size, and this limits their application to roughly ten first row atoms.

The basis of the local triples method is a generalization of the TRIM method discussed above for MP2 theory. The same nonorthogonal sets of functions are used to span the occupied space (the molecule-adapted minimal basis, projected into the occupied space), and the virtual space (the full atomic orbital basis, projected into the virtual space). Three electrons are simultaneously promoted in a triple substitution, and in the local model we have developed, two of those promotions are restricted to a single atom, while the third is permitted to be non-local. This reduces the number of triple substitutions from rising with the 6^{th} power of molecule size to rising with the 4^{th} power. The recanonicalization described in the section on the TRIM model for MP2 theory is even more valuable in the context of triple substitutions, because it means that the local triples amplitudes do not need to be stored. Instead they can be made in the recanonicalized representation in batches, their contribution to the triples energy can be immediately obtained, and they can be discarded prior to commencing the next batch.

Benchmark thermochemical calculations on 105 molecules in the G2/97 database indicate that local truncation recovers at least 95% of the untruncated triples energy. The local error introduced into the G2 binding energies is typically 0.1 - 0.2 kcal/mol, with a maximum error of 0.26 kcal/mol. This error is small enough for most applications. Finally, while the efficient computational implementation of this local triples method is a challenging algorithmic problem, the results are worthwhile[56]. The local triples algorithm reduces the computational cost from growing with the seventh power of molecule size to the fifth power of molecule size. In practice, this enables triples calculations on molecules two to three times larger than previously feasible. It crosses over with the conventional triples algorithm around 25 occupied orbitals, or roughly 150 basis functions. The local algorithm requires all the doubles amplitudes to be stored on disk, and also requires disk storage for a cubic number of local integrals. A relatively small (quadratic) amount of memory is required.

The discussion of atoms in molecules models of local correlation in Section 4.3 listed several different atomic truncations of the full set of double substitutions. The gentlest truncation is triatomics in molecules, where one occupied to unoccupied orbital substitution must occur on a single atom, so that the overall double substitution cannot involve more than 3 atoms. In the more strongly truncated diatomics in molecules model, two orbital substitutions are each required to be on single atoms, so that the double substitution cannot involve more than a pair of atoms. The most drastic truncation, which we termed atoms in molecules, requires all 4 orbitals involved in the double substitution to be on a single atom. Such an approximation would not correctly describe the dispersion energy, and therefore we did not consider it as a useful approximation in the context of MP2 theory.

Are there other contexts in which the very drastic atoms in molecules approximation to electron correlation (or something very similar to it) could be useful? The answer is yes, and in this section we shall describe one such context: namely the so-called perfect pairing approaches to electron correlation[57,58]. In perfect pairing, the (unnormalized) wavefunction is taken to be an antisymmetrized (A) product of doubly occupied "core" functions, and geminal (2-electron) functions, g_i that describe electron correlations within each valence bonded pair (and each lone pair) of electrons:

$$\big|\Psi\big\rangle = A\big[\varphi_1\bar{\varphi}_1\varphi_2\bar{\varphi}_2\cdots g_1 g_2\cdots\big] \tag{91}$$

$$g_i = A\big[\varphi_{C+i}\bar{\varphi}_{C+i} + t_i\varphi^*_{C+i}\bar{\varphi}^*_{C+i}\big] \tag{92}$$

Here C is the number of doubly occupied (C)ore orbitals. Clearly a wavefunction of this form is well suited to breaking single bonds, where at dissociation, the bonding and antibonding orbitals, φ_i and φ^*_i, approach degeneracy.

It has been known for some time[59], although it is still not widely appreciated, that the PP wavefunction can be exactly rewritten in the form of a simplified coupled cluster doubles wavefunction; no approximation is involved. Specifically:

$$\big|\Psi\big\rangle = exp\big(T_2^{PP}\big)\big|\Phi_0\big\rangle \tag{93}$$

$$\big|\Phi_0\big\rangle = A\big[\varphi_1\bar{\varphi}_1\varphi_2\bar{\varphi}_2\cdots\varphi_O\bar{\varphi}_O\big] \tag{94}$$

$$\hat{T}_2^{PP}\big|\Phi_0\big\rangle = \sum_i t_i\big|\Phi_i\big\rangle \tag{95}$$

The double substitution $|\Phi_i\rangle$ is the determinant $|\Phi_0\rangle$ with the occupied pair $\varphi_{C+i}\bar{\varphi}_{C+i}$ replaced by the unoccupied (correlating) pair $\varphi^*_{C+i}\bar{\varphi}^*_{C+i}$. Viewed in this light, the PP wavefunction is evidently a form of local correlation! It is a form in which each occupied pair is treated separately (the pair is the atomic analog here).

Conventionally the bonding and antibonding functions that comprise the wavefunction are taken to be strongly orthogonal to each other. With this constraint the energy can be solved for either variationally, or nonvariationally via the machinery of coupled cluster theory. The resulting method is qualitatively quite successful in describing potential energy curves. Single, double, and even triple bond-breaking curves are qualitatively correct using perfect pairing. This applies whether the energy is obtained via the variational method, or via the coupled cluster approach, which is not formally variational.

At the same time, PP does have some notable deficiencies, which are directly connected to the limited description of electron correlation that it contains. For example, applying PP to the benzene molecule leads to symmetry breaking: PP predicts that the most stable geometry has alternating carbon-carbon bond lengths rather than equal ones. This arises from the fact that there are two equivalent sets of localized occupied orbitals that can describe the π electrons, which can be loosely mapped to the two Kekule structures. When one of these sets is chosen, and a corresponding set of correlating orbitals are defined, this favors localization of the electronic structure. The extent of symmetry breaking is relatively small in energy terms[60]: about 3 kcal/mol is the energy difference between D_{6h} and D_{3h} structures. Nevertheless this is clearly an undesirable result.

Given our use of nonorthogonal functions to develop local correlation models for double substitutions, we thought it natural to ask whether nonorthogonal functions might be useful for alleviating the problem discussed above for the PP model. In particular, we have explored the possibility of using the following redefinition of the orbitals in Equation (92). Instead of permitting one orthogonal occupied and one correlating orbital for each valence pair of *electrons*, we prefer to use one occupied and one correlating orbital for each valence atomic *orbital*[60]. Since the number of valence atomic orbitals is larger than the number of electrons, this is clearly a redundant, or linearly dependent representation. The occupied valence orbitals will be nonorthogonal to each other, and the unoccupied correlating orbitals will also be nonorthogonal to each other. The two sets will remain strongly orthogonal, however, as is necessary to make the occupied space well defined.

The use of redundant representations for the *one-particle* (orbital) space permits additional flexibility in the *two-particle* space, because there are now more amplitudes, and these additional amplitudes are not entirely redundant. In the case of benzene in the π space, there will now be one occupied orbital for each C atom (corresponding essentially to a p_z function projected into the occupied space), and one correlating orbital. Clearly, each symmetry-equivalent atom is treated equivalently as far as correlation is concerned, and so it is not surprising that the resulting nonorthogonal PP model[60] (solved via coupled cluster methods), eliminates the artifactual symmetry breaking seen in the traditional PP model applied to benzene.

5.3 Second Order Corrections to Singles and Doubles Coupled Cluster Energies

To approach chemical accuracy in reaction energies and related properties, it is necessary to account for electron correlation effects that involve 3 electrons simul-

taneously, as represented by triple substitutions relative to the mean field single determinant reference, which arise in MP4. The best standard methods for including triple substitutions are the CCSD(T) type methods[61]. The accuracy of these methods is well-documented for many cases[62] and in general is a very significant improvement relative to the starting point. While the (T) corrections have been extraordinarily successful, there is nonetheless still room for improvement. They contain judiciously chosen terms from 4^{th} and 5^{th} order Møller-Plesset perturbation theory, as well as higher order terms that result from the fact that the converged cluster amplitudes are employed to evaluate the 4^{th} and 5^{th} order terms. The correction therefore depends upon the bare reference orbitals and orbital energies, and in this way its effectiveness still depends on the quality of the reference determinant. Since we are correcting a coupled cluster solution rather than a single determinant, this is an aspect of the (T) corrections that can be improved.

Such an improvement has recently been reported[63]. The new correction is a true second order correction to a coupled cluster starting point and is therefore denoted as (2). It was first developed for the coupled cluster doubles method using optimized orbitals[64] (denoted as OD) and is also straightforward to implement for either CCSD or QCISD, given the general theory already presented[65]. The basis of the (2) method is to partition not the regular Hamiltonian into perturbed and unperturbed parts, but rather to partition a similarity-transformed Hamiltonian, defined as $\hat{\bar{H}} = e^{-\hat{T}} \hat{H} e^{\hat{T}}$. In the truncated space (call it the p-space) within which the cluster problem is solved (e.g. singles and doubles for CCSD), the coupled cluster wavefunction is a true eigenvalue of $\hat{\bar{H}}$. Therefore we take the zero order Hamiltonian, $\hat{\bar{H}}^{(0)}$, to be the full $\hat{\bar{H}}$ in the p-space, while in the space of excluded substitutions (the q-space) we take only the one-body part of $\hat{\bar{H}}$ (which can be made diagonal). The fluctuation potential describing electron correlations in the q-space is $\hat{\bar{H}} - \hat{\bar{H}}^{(0)}$, and the (2) correction then follows from second order perturbation theory.

Tensor methods are naturally useful in the development of the detailed expressions for the (2) correction[65]. This is because the similarity transformed Hamiltonian, $\hat{\bar{H}}$, is non-Hermitian. As a result, when the one-body part of $\hat{\bar{H}}$ is made diagonal, the single particle eigenfunctions that result are different on the left and right sides. In other words, the single particle bras and kets are different in the basis in which we work. Furthermore, in this semicanonical basis, the bras and kets are biorthogonal to each other, but each set contains functions that are nonorthogonal amongst themselves. This basis defines the natural representation that is appropriate for this problem, and in ref. 65, the working equations of the (2) methods are derived and presented in this representation, as well as more general biorthogonal natural representations (in which the single particle $\hat{\bar{H}}$ is not diagonal).

The new partitioning of terms between the perturbed and unperturbed Hamiltonians inherent in the (2) correction leads to a correction that show both similarities and differences relative to the existing (T) corrections. There are two types of higher correlations that enter at second order: not only triple substitutions, but also quadruple substitutions. The quadruples are treated with a factorization ansatz, that is exact in 5^{th} order Møller-Plesset theory[66] , to reduce their computational

cost from N^9 to N^6. For large basis sets this can still be larger than the cost of the triples terms, which scale as the 7^{th} power of molecule size, with a factor twice as large as the usual (T) corrections. These corrections are feasible for molecules containing between four and ten first row atoms, depending on computer resources, and the size of the basis set chosen. There is early evidence that the (2) corrections are superior to the (T) corrections for highly correlated systems[63]. This shows up in improved potential curves, particularly at long range. For such problems, with restricted orbitals, the (T) corrections tend to diverge, but the new (2) corrections are stable provided the coupled cluster reference itself is stable. This advantage may also extend to improved energetic and structural properties at equilibrium in problematical cases. It will be some time before sufficient testing on the new (2) corrections has been done to permit a general assessment of the performance of these methods, but they are clearly very promising.

6 Conclusions

In this chapter we have tried to accomplish several objectives. First, we have provided a general introduction to tensor methods in the context of electronic structure theory, where they are particularly well-suited to dealing with nonorthogonal single particle basis sets. Second, we have discussed local electron correlation models for second order Møller-Plesset theory, that are based entirely on nonorthogonal single particle basis sets both as an application of the tensor algebra and as an illustration of the potential usefulness of nonorthogonal functions. Third, we have provided a brief description of several other applications in electronic structure theory where we have found the tensor methods useful, to further illustrate their value. Fourth, we would like to emphasize that there are many other potential applications awaiting us in the future. Perhaps this is the best reason to at least be generally aware of how to apply tensor methods in electronic structure theory.

Acknowledgments

This work was supported by grants from the National Science Foundation, the Air Force Office of Scientific Research, and the Department of Energy. MHG gratefully acknowledges a Packard Fellowship (1995-2000).

References

1. For an introductory treatment of electron correlation methods, see for example, A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry* (McGraw-Hill, New-York, 1989).
2. R. McWeeny, *Methods of Molecular Quantum Mechanics* (Academic Press, London, 1989).
3. F. E. Harris, H. J. Monkhorst, and D. L. Freeman, *Algebraic and Diagrammatic Methods in Many-Fermion Theory* (Oxford University Press, Oxford, 1992).
4. E. B. Stechel, T. A. Schmalz, and J. C. Light, J. Chem. Phys. **70**, 5640 (1979).
5. O. Sinanoglu, Theor. Chim. Acta **65**, 223 (1984).

6. E. Artacho and L. Milans del Bosch, Phys. Rev. A **43**, 5770 (1991).
7. M. Head-Gordon, P. E. Maslen and C. A. White, J. Chem. Phys. **108**, 616 (1998).
8. For examples of formation of the effective Hamiltonian with linear scaling computational effort, see, C. A. White, B. G. Johnson, P. M. W. Gill and M. Head-Gordon, Chem. Phys. Lett. **253**, 268 (1996), and references therein.
9. For a discussion of linear scaling methods that replace diagonalizing the Hamiltonian, see, R. Baer and M. Head-Gordon, Phys. Rev. Lett. **79**, 3962 (1997), and references therein.
10. P. Pulay and S. Saeb, Ann. Rev. Phys. Chem. **44**, 213 (1993).
11. P. Fulde, *Electron Correlations in Molecules and Solids*, 3rd edition (Springer-Verlag, Berlin, 1995).
12. J. L. Synge and A. Schild, *Tensor Calculus* (Dover, New York, 1978).
13. A. D. Myskis, *Advanced Mathematics for Engineers* (Mir, Moscow, 1979) Chapter 5.
14. D. Lovelock and H. Rund, *Tensors, Differential Forms and Variational Principles* (Dover, New York, 1988).
15. T. Frankel, *The Geometry of Physics: An Introduction* (Cambridge, Cambridge, 1997).
16. A. Edelman, T. A. Arias, and S. T. Smith, SIAM J. Matrix Anal. Appl. **20**, 303 (1998).
17. W. Kohn, Phys. Rev. **115**, 809 (1959).
18. S. L. Campbell and C. D. Meyer, Junior. *Generalized Inverses of Linear Transformations*, (Dover Publications, New York, 1991).
19. See for example, J. F. Stanton and R. J. Bartlett, J. Chem. Phys.**98**, 7029 (1993), and references therein.
20. See for example, J. J. W. McDouall, Theor. Chim. Acta **83**, 339 (1992); **85**, 395 (1993), and references therein.
21. P. E. Maslen and M. Head-Gordon, Chem. Phys. Lett. **283**, 102 (1998).
22. P. E. Maslen and M. Head-Gordon, J. Chem. Phys. **109**, 7093 (1998).
23. M. S. Lee, P. E. Maslen and M. Head-Gordon, J. Chem. Phys. **112**, 3592 (2000).
24. C. Møller and M. S. Plesset, Phys. Rev. **46**, 618 (1934).
25. W. J. Hehre, L. Radom, P. v. R. Schleyer, and J. A. Pople, *Ab Initio Molecular Orbital Theory* (Wiley, New York 1986).
26. T. Helgaker, J. Gauss, P. Jorgensen, and J. Olsen, J. Chem. Phys. **106**, 6430 (1997).
27. W. Kohn, A. D. Becke and R. G. Parr, J. Phys. Chem. **100**, 12974 (1996).
28. For the most recent developments in semidirect methods, together with a review of earlier work, see, M. Head-Gordon, Mol. Phys. **96**, 673 (1999).
29. P. Pulay, Chem. Phys. Lett. **100**, 151 (1983).
30. S. Saebo, P. Pulay, J. Chem. Phys. **86**, 914 (1987).
31. R. B. Murphy, M. B. Beachy, R. A. Friesner, and M. N. Ringnalda, J. Chem. Phys. **103**, 1481 (1995).
32. G. Hetzer, P. Pulay, and H. J. Werner, Chem. Phys. Lett. **290**, 143 (1998).
33. G. Rauhut, P. Pulay, and H. J. Werner, J. Comput. Chem. **11**, 1241 (1998).

34. M. Schütz, G. Hetzer, and H. J. Werner, J. Chem. Phys. **111**, 5691 (1999).
35. C. Hampel and H. J. Werner, J. Chem. Phys. **104**, 6286 (1996).
36. G. Reynolds, T. J. Martinez, and E. A. Carter, J. Chem. Phys. **105**, 6455 (1996).
37. G. Reynolds and E. A. Carter, Chem. Phys. Lett. **265**, 660 (1997).
38. J. W. Boughton and P. Pulay, J. Comput. Chem. **14**, 736 (1998).
39. A. El Azhary, G. Rauhut, P. Pulay and H.-J. Werner, J. Chem. Phys. **108**, 5185 (1998).
40. T. Nakajima and H. Nakatsuji, Chem. Phys. **242**, 177 (1999).
41. R. A. Friesner, R. B. Murphy, M. D. Beachy, M. N. Ringnalda, W. T. Pollard, B. D. Dunietz, Y. X. Cao, J. Phys. Chem. **A103**, 1913 (1999).
42. For one example, see, W. J. Lauderdale, J. F. Stanton, J. Gauss, J. D. Watts, and R. J. Bartlett, Chem. Phys. Lett. **187**, 21 (1991).
43. M. S. Lee and M. Head-Gordon, Int. J. Quantum Chem. **76**, 169 (2000).
44. M. S. Lee and M. Head-Gordon, J. Chem. Phys. **107**, 9085 (1997).
45. M. S. Lee and M. Head-Gordon, Comp. Chem. **24**, 295 (2000).
46. M. S. Lee and M. Head-Gordon (work in progress).
47. C. A. White, J. Kong, D. R. Maurice, T. R. Adams, J. Baker, M. Challacombe, E. Schwegler, J. P. Dombroski, C. Ochsenfeld, M. Oumi, T. R. Furlani, J. Florian, R. D. Adamson, N. Nair, A. M. Lee, N. Ishikawa, R. L. Graham, A. Warshel, B. G. Johnson, P. M. W. Gill and M. Head-Gordon, Q-Chem, Version 1.2, Q-Chem, Inc., Pittsburgh, PA (1998).
48. M. S. Lee, Ph. D. Thesis, University of California, Berkeley, 2000.
49. M. Head-Gordon and J. A. Pople, J. Phys. Chem. **97**, 1147 (1993).
50. A. Halkier, H. Koch, P. Jorgensen, O. Christiansen, I. M. Beck Nielsen and T. Helgaker, Theor. Chem. Acc. **97**, 150 (1997).
51. S. S. Xantheas, J. Chem. Phys. **104**, 8821 (1996).
52. M. Schütz, S. Brdarski, P. O. Widmark, R. Lindh, and G. Karstrom, J. Chem. Phys. **107**, 4597 (1997).
53. W. Klopper, and H. P. Lüthi, Mol. Phys. **96**, 559 (1999).
54. M. Schütz, G. Rauhut, and H.-J. Werner, J. Phys. Chem. **A102**, 5997 (1998).
55. P. E. Maslen, M. S. Lee and M. Head-Gordon, Chem. Phys. Lett. **319**, 205 (2000).
56. P. E. Maslen, M. S. Lee and M. Head-Gordon (in preparation).
57. A. C. Hurley, J. Lennard-Jones and J. A. Pople, Proc. Roy. Soc. London, Ser. A **220**, 446 (1953).
58. W. A. Goddard III, and L. B. Harding, Ann. Rev. Phys. Chem. **29**, 363 (1978).
59. J. Cullen, Chem. Phys. Lett. **202**, 217 (1996).
60. T. Van Voorhis and M. Head-Gordon, J. Chem. Phys. **112**, 5633 (2000).
61. K. Raghavachari, G. W. Trucks, J. A. Pople and M. Head-Gordon, Chem. Phys. Lett. **157**, 479 (1989).
62. See for example, T. J. Lee and G. E. Scuseria, in Quantum Mechanical Calculations with Chemical Accuracy, edited by S. R. Langhoff (Kluwer, Dordrecht, 1995) p 47.
63. S. R. Gwaltney and M. Head-Gordon, Chem. Phys. Lett. **323**, 21 (2000).

64. C. D. Sherrill, A. I. Krylov, E. F. C. Byrd, and M. Head-Gordon, J. Chem. Phys. **109**, 4171 (1998).
65. S. R. Gwaltney, C. D. Sherrill, M. Head-Gordon, and A. I. Krylov, J. Chem. Phys. **113**, 3548 (2000).
66. S. A. Kucharski and R. J. Bartlett, J. Chem. Phys. **108**, 5243 (1998); 108, 9221 (1998).