Inaugural - Dissertation

zur Erlangung der Doktorwürde der
Naturwissenschaftlich-Mathematischen Gesamtfakultät
der Ruprecht-Karls-Universität Heidelberg

Development of a modular quantum-chemistry framework for the investigation of novel basis functions

with an application to Coulomb-Sturmians

vorgelegt von

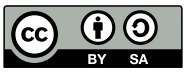
Michael F. Herbst, MA MSci
aus Grünstadt-Asselheim

im März 2018

Gutachter: Prof. Dr. Andreas Dreuw
Prof. Dr. Guido Kanschat

Tag der mündlichen Prüfung: 22. Mai 2018

Licensing and redistribution



Source code repository

This thesis was generated using LaTeX, `python` and `molsturm`. The full source code is available on `github` at `https://github.com/mfherbst/dissertation`. To generate this very document use tag `v1.1.1`.

For everyone, since knowledge is a bastion

For God, who made us curious and question

For my parents, who have always taken care

For Carine, for whom I will always be there

Abstract

State-of-the-art methods for the calculation of electronic structures of molecules predominantly use Gaussian basis functions. The algorithms employed inside existing code packages are consequently often highly optimised keeping only their numerical requirements in mind. For the investigation of novel approaches, utilising other basis functions, this is an obstacle, since requirements might differ. In contrast, this thesis develops the highly flexible program package `molsturm`, which is designed in order to facilitate rapid design, implementation and assessment of methods employing different basis function types. A key component of `molsturm` is a Hartree-Fock (HF) self-consistent field (SCF) scheme, which is suitable to be combined with any basis function type.

First the mathematical background of quantum mechanics as well as some numerical techniques are reviewed. Care is taken to emphasise the often overlooked subtleties when discretising an infinite-dimensional spectral problem in order to obtain a finite-dimensional eigenproblem. Common quantum-chemical methods such as full configuration interaction and HF are discussed providing insight into their mathematical properties. Different formulations of HF are contrasted and appropriate SCF solution schemes formulated.

Next discretisation approaches based on four different types of basis functions are compared both with respect to the computational challenges as well as their ability to describe the physical features of the wave function. Besides (1) Slater-type orbitals and (2) Gaussian-type orbitals, the discussion considers (3) finite elements, which are piecewise polynomials on a grid, as well as (4) Coulomb-Sturmians, which are the analytical solutions to a Schrödinger-like equation. A novel algorithmic approach based on matrix-vector contraction expressions is developed, which is able to adapt to the numerical requirements of all basis functions considered. It is shown that this ansatz not only allows to formulate SCF algorithms in a basis-function independent way, but furthermore improves the theoretically achievable computational scaling for finite-element-based discretisations as well as performance improvements for Coulomb-Sturmian-based discretisations. The adequacy of standard SCF algorithms with respect to a contraction-based setting is investigated and for the example of the optimal damping algorithm an approximate modification to achieve such a setting is presented.

With respect to recent trends in the development of modern computer hardware the potentials and drawbacks of contraction-based approaches are evaluated. One drawback, namely the typically more involved and harder-to-read code, is identified and a data structure named lazy matrix is introduced to overcome this. Lazy matrices are a generalisation of the usual matrix concept, suitable for encapsulating contraction expressions. Such objects still look like matrices from the user perspective, including the possibility to

perform operations like matrix sums and products. As a result programming contraction-based algorithms becomes similarly convenient as working with normal matrices. An implementation of lazy matrices in the `lazyten` linear algebra library is developed in the course of the thesis, followed by an example demonstrating the applicability in the context of the HF problem.

Building on top of the aforementioned concepts the design of `molsturm` is outlined. It is shown how a combination of lazy matrices and a contraction-based SCF scheme separates the code describing the SCF procedure from the code dealing with the basis function type. It is discussed how this allows to add a new basis function type to `molsturm` by only making code changes in a single integral interface library. On top of that, we demonstrate by the means of examples how the readily scriptable interface of `molsturm` can be employed to implement and assess novel quantum-chemical methods or to combine the features of `molsturm` with existing third-party packages.

Finally, the thesis discusses an application of `molsturm` towards the investigation of the convergence properties of Coulomb-Sturmian-based quantum-chemical calculations. Results for the convergence of the ground-state energies at HF level are reported for atoms of the second and the third period of the periodic table. Particular emphasis is put on a discussion about the required maximal angular momentum quantum numbers in order to achieve convergence of the discretisation of the angular part of the wave function. Some modifications required for a treatment at correlated level are suggested, followed by a discussion of the effect of the Coulomb-Sturmian exponent. An algorithm for obtaining an optimal exponent is devised and some optimal exponents for the atoms of the second and the third period of the periodic table at HF level are given. Furthermore, the first results of a Coulomb-Sturmian-based excited states calculation based on the algebraic-diagrammatic construction scheme for the polarisation propagator are presented.

Zusammenfassung

Für die Berechnung elektronischer Zustände in Molekülen verwenden aktuelle Methoden vor allem Gaußfunktionen. Die in den bestehenden Quantenchemiepaketen verwendeten Algorithmen sind dementsprechend oft sehr stark auf diesen Funktionstyp und dessen numerische Anforderungen angepasst. Dies ist ein Hindernis für die Verwendung anderer Basisfunktionen bei der Betrachtung dieser Methoden, da die Anforderungen durchaus unterschiedlich sein können. Im Gegensatz dazu wird in dieser Arbeit das Programmpacket `molsturm` entwickelt, welches explizit so gestaltet wurde, dass neue Basisfunktionen in der Elektronenstrukturtheorie auf einfache Weise eingebunden und getestet werden können. Ein Lösungsverfahren für das Hartree-Fock-Problem (HF), welches mit beliebigen Basisfunktionstypen verwendet werden kann, ist dazu ein wichtiger Bestandteil von `molsturm`.

Zunächst werden der mathematische Hintergrund der Quantenmechanik und einige numerische Techniken vorgestellt. Dabei wird insbesondere auf die oft unterschlagenen Feinheiten eingegangen, welche beim Diskretisieren eines unendlich-dimensionalen Spektralproblemes hin zu einem endlich-dimensionalen Eigenwertproblem entstehen. Häufig verwendete quantenchemische Methoden wie die vollständige Konfigurationswechselwirkung (full configuration interaction) oder HF werden diskutiert. Unterschiedliche Formulierungen von HF werden miteinander verglichen und Lösungsalgorithmen auf Basis des Verfahrens des selbstkonsistenten Feldes (self-consistent field, SCF) angegeben.

Im Weiteren werden Diskretisierungsansätze basierend auf vier verschiedenen Basisfunktionstypen miteinander verglichen, wobei sowohl auf Herausforderungen in Bezug auf die numerische Berechenbarkeit der entstehenden Ausdrücke eingegangen wird, als auch auf die Fähigkeit der Basen, die physikalischen Eigenschaften der Wellenfunktion zu beschreiben. Neben (1) Orbitalen vom Slatertyp und (2) Gaußorbitalen behandelt die vorgestellte Diskussion (3) finite Elemente, abschnittsweise Polynome, welche auf einem Gitter definiert sind, sowie (4) Coulomb-Sturmfunktionen (Coulomb-Sturmians), welche die analytischen Lösungen einer der Schrödingergleichung ähnlichen Differenzialgleichung sind. Ein neuartiger Algorithmus basierend auf Matrix-Vektor-Kontraktionsausdrücken wird entwickelt, welcher die numerischen Anforderungen aller betrachteten Funktionen abdeckt. Es wird gezeigt, dass mittels dieses Ansatzes nicht nur SCF-Algorithmen in einer basisfunktionsunabhängigen Weise formuliert werden können, sondern auch, dass dadurch die algorithmische Komplexität für Diskretisierungen basierend auf finiten Elementen reduziert wird und dass damit Effizienzverbesserungen für Diskretisierungen basierend auf Coulomb-Sturmfunktionen möglich sind. Die Eignung üblicherweise verwendeter SCF-Algorithmen auf eine derartige kontraktionsbasierte (contraction-based) Formulierung wird geprüft und für das Beispiel des Algorithmus der optimalen Dämp-

fung (optimal damping algorithm) wird eine zusätzliche Näherung vorgeschlagen, die diesen mit der kontraktionsbasierten Formulierung vereinbart.

Vor dem Hintergrund aktuell verfügbarer Hardware werden das Potential und die Nachteile kontraktionsbasierter Methoden diskutiert. Ein Hindernis bei der Entwicklung kontraktionsbasierter Methoden ist oft, dass die daraus entstehenden Quelltexte schwerer lesbar sind. Um dieses Problem zu umgehen wird die Datenstruktur einer "bequemen Matrix" (lazy matrix) eingeführt. Bequeme Matrizen sind eine Verallgemeinerung des üblichen Matrixkonzeptes, welche als eine Art Behältnis (container) für Kontraktionsausdrücke aufgefasst werden können. Aus Sicht eines Benutzers bequemer Matrizen, sehen diese weiterhin wie gewöhnliche Matrizen aus, beispielsweise können sie in üblicher Weise miteinander addiert oder multipliziert werden. Die Folge ist, dass man kontraktionsbasierte Algorithmen auf die gleiche Art und Weise wie bei der Verwendung von gewöhnlichen Matrizen implementieren kann. Eine Implementierung des Konzepts der bequemen Matrizen wird in Form von der Bibliothek `lazyten` vorgestellt. Ebenso wird ein Beispiel gegeben, welches die Eignung von bequemen Matrizen im Kontext der Lösung des HF-Problems demonstriert.

Basierend auf den oben erwähnten Konzepten wird die Programmstruktur von `molsturm` diskutiert. Es wird dargelegt, wie durch die Anwendung der bequemen Matrizen innerhalb eines kontraktionsbasierten SCF-Verfahrens eine Trennung des Programmcodes, welcher das SCF-Verfahren selbst beschreibt von jenem Programmcode, welcher die Diskretisierung und die Basisfunktionen betrifft, erreicht werden konnte. Des Weiteren wird gezeigt, wie ein neuer Basisfunktionstyp in `molsturm` implementiert werden kann, indem nur an einer einzigen Stelle im Programmcode Änderungen durchgeführt werden. Darüber hinaus wird mittels einiger Beispiele besprochen, wie `molsturm` über skriptbare (scriptable) Schnittstellen mit der Funktionalität bestehender Programme kombiniert werden kann, um so auf einfache Weise neue Quantenchemiemethoden zu implementieren und zu testen.

Zuletzt wird eine Anwendung von `molsturm` für die Untersuchung der Konvergenzeigenschaften von Quantenchemierechnungen basierend auf Coulomb-Sturmfunktionen vorgestellt. Erste Ergebnisse für die Konvergenz der Grundzustandsenergie auf HF-Niveau werden für die Atome der zweiten und dritten Periode des Periodensystems vorgestellt. Im Besonderen wird auf die maximale Drehimpulsquantenzahl eingegangen, welche benötigt wird, um eine Konvergenz des Winkelanteils der Wellenfunktion auf HF-Niveau zu erreichen. Mögliche Änderungen dieses Ergebnisses im Hinblick auf die Beschreibung der Atome mittels Korrelationsmethoden werden kurz angedeutet und der Effekt des Exponenten der Coulomb-Sturmfunktionen auf die Grundzustandsenergie diskutiert. Ein Algorithmus zur Bestimmung des optimalen Exponenten wird konstruiert und einige optimale Exponenten für die Beschreibung der Atome der zweiten und dritten Periode des Periodensystems auf HF-Niveau werden angegeben. Des Weiteren werden erste Ergebnisse einer Berechnung elektronisch angeregter Zustände mittels des algebraisch-diagrammatischen Konstruktionsschemas für den Polarisationspropagator basierend auf Coulomb-Sturmfunktionen vorgestellt.

Table of contents

List of Tables

List of Figures

Symbols and conventions

Δ	Laplace operator.
δ_{ij}	Kronecker delta. Equal to 0 if $i \neq j$, else 1.
$\varepsilon_{\text{conv}}$	Convergence tolerance for an iterative process. If the error is below this value, the process should be considered converged.
$\mathbb{F}$	Either the field of real numbers $\mathbb{R}$ or the field of complex numbers $\mathbb{C}$.
$\gamma_\Theta(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2)$	One-particle reduced density matrix.
$\mathcal{H}$	Arbitrary Hilbert space, typically the Hilbert space of quantum mechanics, which is $L^2(\mathbb{R}^d, \mathbb{C}^s)$ for d spatial and s spin dimensions.
$\mathcal{I}_{\text{bas}}$	Index set of the one-particle basis functions. Typically a set of multi-indices of quantum numbers.
$\text{id}_{\mathcal{H}}$	Identity operator on the Hilbert space $\mathcal{H}$.
$\mathcal{I}_{\text{occ}}$	Index set of occupied SCF orbitals.
$\mathcal{I}^\alpha_{\text{occ}}$, $\mathcal{I}^\beta_{\text{occ}}$	Index set of occupied SCF orbitals of α or β spin, respectively. Typically $\{1, \ldots, N^\alpha_{\text{elec}}\}$ and similar for $\mathcal{I}^\beta_{\text{occ}}$.
$\mathcal{I}_{\text{orb}}$	Index set of computed SCF orbitals, typically $\{1, \ldots N_{\text{orb}}\}$.
$\mathcal{I}_{\text{virt}}$	Index set of virtual, i.e. unoccupied, SCF orbitals.
k_{exp}	Coulomb-Sturmian exponent.
k_{opt}	Optimal Coulomb-Sturmian exponent k_{exp}, which yields the best description of the system given the current Coulomb-Sturmian basis and the selected quantum-chemical method.
$\mathcal{I}$	Index set.
$\mathbf{C}$	Matrix of occupied molecular orbital coefficients of size $2N_{\text{bas}} \times N_{\text{elec}}$.
$\mathbf{C}^\alpha$	Matrix of occupied spin-up molecular orbital coefficients of size $N_{\text{bas}} \times N^\alpha_{\text{elec}}$.
$\mathbf{C}_F$	Matrix of all molecular orbital coefficients. Result of the diagonalisation of $\mathbf{F}$. Size $2N_{\text{bas}} \times N_{\text{orb}}$.
$\mathbf{C}^\alpha_F$	Matrix of all spin-up molecular orbital coefficients. Result of the diagonalisation of $\mathbf{F}^\alpha$. Size $N_{\text{bas}} \times N^\alpha_{\text{orb}}$.
$\mathbf{I}_N$	Identity matrix in $\mathbb{C}^{N \times N}$.

$\mathbf{M}$	Matrix with elements denoted as M_{ij}.
N_{bas}	Cardinality of $\mathcal{I}_{\text{bas}}$, i.e. the number of one-particle basis functions.
N_{elec}	Total number of electrons.
$N_{\text{elec}}^{\alpha}, N_{\text{elec}}^{\beta}$	Number of α or β electrons, respectively. Note $N_{\text{elec}}^{\alpha} + N_{\text{elec}}^{\beta} = N_{\text{elec}}$ and $N_{\text{elec}}^{\alpha} \geq N_{\text{elec}}^{\beta}$ by convention.
N_{orb}	The number of computed SCF orbitals. Note that $N_{\text{orb}} \leq N_{\text{bas}}$.
$\|\cdot\|_2$	l_2-norm or Euclidean norm.
$\|\cdot\|_{\text{frob}}$	Frobenius norm of a matrix, which is the square root of the sum of all elements squared.
$\hat{\mathcal{H}}_{N_{\text{elec}}}$	Electronic Hamiltonian for a N_{elec}-electron system, see section 4.2.
$\hat{\mathcal{J}}$	Effective Coulomb operator.
$\hat{\mathcal{K}}$	One-electron exchange operator.
$\hat{\mathcal{T}}$	Kinetic energy operator.
$\hat{\mathcal{V}}$	Potential energy operator.
Φ	Slater determinant $\bigwedge_{i=1}^{N_{\text{elec}}} \psi_i$ or many-electron basis function.
Ψ	State of a (many-body) quantum system, typically the *exact* ground state solution to the many-electron, electronic time-independent Schrödinger equation.
ψ_i	One-particle function, typically i-th eigenfunction of the Fock operator, i.e. a Hartree-Fock orbital.
$\rho_{\Theta}(\underline{\boldsymbol{r}})$	Electron density.
RMSO_l	The root mean square occupied coefficient per angular momentum l, see definition 8.1 on page 176.
Θ	Tuple of all occupied one-particle SCF orbitals $(\psi_1, \psi_2, \ldots, \psi_{N_{\text{elec}}})$.
φ_μ	μ-th one-particle basis function of the one-particle basis $\{\varphi_\mu\}_{\mu \in \mathcal{I}_{\text{bas}}}$.
$\underline{\boldsymbol{r}}$	Vector in $\mathbb{R}^3$, which specifies the position of an electron in space.
$\underline{\boldsymbol{x}}$	Vector in $\mathbb{R}^{3N_{\text{elec}}}$, which typically specifies the positions of all electrons of the system.
$H^1(\mathbb{R}^d, \mathbb{C})$	The Sobolev space of complex-valued functions with square-integrable first derivative.
$H^2(\mathbb{R}^d, \mathbb{C})$	The Sobolev space of complex-valued functions with square-integrable second derivative.
$L^2(\mathbb{R}^d, \mathbb{C})$	The Hilbert space of square-integrable complex-valued functions.
M	Number of nuclei.
P_l^m	Associated Legendre polynomial with orders l and m.
r_{ij}	Short hand for $\left\|\underline{\boldsymbol{r}}_i - \underline{\boldsymbol{r}}_j\right\|_2$.
$Y_l^m(\theta, \varphi)$	Spherical harmonic with angular momentum quantum number l and azimuthal quantum number m.

ADC	Algebraic-diagrammatic construction
AO	Atom-centred orbital
BO	Born-Oppenheimer approximation
CBS	Complete basis set
CC	Coupled cluster
CCD	Coupled-cluster doubles
cGTO	Contracted Gaussian-type orbital
CS	Coulomb-Sturmian or Coulomb-Sturmian basis
DIIS	Direct inversion in the iterative subspace
ERI	Electron-repulsion integral
FCI	Full configuration interaction
FE	Finite element
GTO	Gaussian-type orbital
GUHF	Generalised unrestricted Hartree-Fock
iff	if and only if
LA	Linear algebra
MP	Møller-Plesset perturbation theory treatment of electron-electron correlation. Typically followed by a number to indicate the order.
ODA	Optimal damping algorithm
QM	Quantum mechanics
RHF	Restricted Hartree-Fock
RMS	Root mean square
ROHF	Restricted open-shell Hartree-Fock
STO	Slater-type orbital
TISE	Time-independent Schrödinger equation
tODA	Truncated optimal damping algorithm
UHF	Unrestricted Hartree-Fock

Chapter 1

Introduction

> *The underlying physical laws necessary for the mathematical theory of a large part of physics and the whole of chemistry are thus completely known, and the difficulty is only that the exact application of these laws leads to equations much too complicated to be soluble. It therefore becomes desirable that approximate practical methods of applying quantum mechanics should be developed, which can lead to an explanation of the main features of complex atomic systems without too much computation.*
>
> — Paul Adrien Maurice Dirac (1902–1984)

Experimental chemistry has already been performed thousands of years ago in the form of alchemy, metallurgy, pottery and dye making. Gradually, in the 17th and 18th century, chemistry transformed into a mature science, where new theories were developed based on experimental evidence rather than philosophical thought. The 19th century marked the appearance of thermodynamics, which provided a theoretical foundation to quantitatively describe the physical processes of chemical reactions. Whilst this already allowed to deductively reach predictions of chemical processes, a full understanding of the microscopic behaviour of chemical systems was not available until the appearance of quantum mechanics at the turn of the 20th century. By now the application of quantum mechanics for the modelling of chemical processes has grown into a field on its own, known as **quantum chemistry**.

Compared to experimental chemistry, quantum chemistry is thus relatively new. Still, as Dirac [1] noted already in 1929, all fundamental equations of quantum chemistry, in the form of the mathematical formulation of quantum mechanics, are known. Whilst solving these equations exactly is possible for model systems, solving them for any chemically or physically interesting system is only feasible using approximate methods. Which methods are best employed depends largely on the intended application, i.e. the complexity of the chemical system or the properties and behaviours to be described. Over the years a hierarchy of approximations has been developed for this reason, ranging from crude to numerically exact or from highly specialised to generally applicable. Nowadays the modelling of chemical processes based on quantum-chemical arguments is well-established both in industry and research.

The aforementioned aspects of accuracy and applicability are to be seen in contrast to computational demand. Generally speaking, the more specific the setting one has in mind and the more accuracy can be sacrificed, the more computationally cheap the resulting quantum-chemical methods become. In electronic structure theory, for example, one typically neglects the motion of the nuclei in order to yield a simplified equation, the electronic Schrödinger equation. With approximate solutions to this equation at hand, already many aspects of chemical reactivity or spectroscopy can be modelled, even though the nuclei are assumed to be motionless in this picture.

For most if not all practically relevant problems, the full description of a chemical system on the level of the electronic Schrödinger equation is still not possible and further approximations are required. One particular ansatz is the Hartree-Fock (HF) method, where the interaction of the electrons amongst themselves is only treated in an averaged manner. This leads to a computationally much more feasible problem, since individual electrons of a chemical system no longer couple directly, but only via a mean field generated by all electrons collectively. The many-body problem of the electronic Schrödinger equation thus becomes an effective one-electron problem. The downside of this is that some of the chemically relevant physics, namely parts of the electron-electron interaction, is lost and a second so-called Post-HF method is needed on top of the HF solution to correct for this. Various levels of Post-HF corrections are available, but in practice not all levels are reachable due to the increasing complexity of the problem. On the other hand, in order to gain insight into a particular research question, the most accurate treatment is not necessarily required. The approach of a HF calculation followed by a Post-HF method sketched here, is not the only way to model chemical systems. An alternative route is density-functional theory (DFT), where the ansatz is to directly work with the electron density instead of the wave function. Due to its good combination of accuracy and computational cost for many problems of electronic structure theory, DFT has become widely adopted.

Since the underlying function spaces of quantum mechanics are infinite dimensional all of the aforementioned methods involve infinite-dimensional spaces as well. For a numerical treatment on a computer, where only finite amounts of memory are available, this is of course an insurmountable obstacle. The remedy in practice is yet another approximation, where one restricts the problem spaces to a finite number of dimensions by only considering subspaces spanned by a finite number of single-particle basis functions. Such a restriction is called a discretisation and has the pleasant side effect that the partial differential equations dictated by quantum mechanics reduce to standard problems of linear algebra. In the case of HF, for example, the discretised problem may be solved by repetitively diagonalising the arising so-called Fock matrix.

This naturally leads to the question: Which type of basis functions should be used to span the subspace for modelling at HF, DFT or Post-HF level? Most available program packages for quantum-chemical calculations of molecules nowadays employ methodologies based on the linear combination of Gaussian-shaped atomic orbital functions. This predominance can be rationalised by the pragmatic historic developments taking place in the founding years of modern electronic structure theory. Boys [2] realised in 1950 that evaluating the integrals required for solving HF is much more feasible for Gaussian-type orbitals compared to the physically more suitable Slater-type orbitals [3]. This idea was picked up and refined later [4] and eventually set off many developments both in terms of efficient algorithms as well as methodologies centred around Gaussian basis functions, spreading their use for simulating the electronic structures of molecules.

Unfortunately Gaussian-type orbitals have major drawbacks caused by their unphysical shape: They are not able to represent all features of the electron density well. Most prominently they fail to describe the cusp of the electron density at the position of a nucleus as well as its exponential decay behaviour [5]. In most practical use cases this is acceptable, since the important quantity for understanding chemical processes is not the absolute energy of a molecule. Much rather chemistry is all about the energy differences between the involved species or electronic configurations. Since changes in the electronic structure both at the nucleus as well as the region far from the nuclei are generally much less pronounced, the errors resulting from an inadequate description of these features tend to cancel one another. For cases where they do become important one can usually compensate by employing specialised Gaussian basis sets [6, 7] up to some extent. There are strong indications, however, that the accurate computation of some physical properties like nuclear magnetic resonance (NMR) shielding tensors [8, 9] requires a correct description of both aforementioned features. Additionally, those specialised basis sets can lead to numerical instabilities in the resulting linear algebra due to their overcompleteness, making it numerically more challenging to obtain reliable results.

Multiple research groups have therefore looked into alternative types of basis functions for the modelling of molecular structures. Examples include so-called numerical basis functions [10] like wavelets [11–16] or finite elements [17–23]. Both of these are interesting because they allow for rigorous error bounds to be derived for the discretisation, essentially leading to a modelling of chemical systems with guaranteed numerical precision. Another promising approach are Sturmian-type functions like Coulomb-Sturmians [9, 24–28] or generalised Sturmians [21, 29–34], since they amount to correctly represent the physical features of the electron density and furthermore lead to more feasible integrals than Slater-type orbitals. Their use in quantum-chemistry is, however, not well-established.

There are some further aspects to consider in such a discussion about basis functions. First of all our discussion has ignored a very important feature of the electronic structure, namely the electron-electron cusp. It has been shown that a proper modelling of this feature of the wave function is important in order to reach rapid convergence for Post-HF methods [35]. A simple single-particle basis such as the Gaussian-type orbitals cannot properly account for this, since these functions have no direct notion of the electron-electron distance. Recent attempts to tackle this issue are so-called explicitly correlated methods [35]. In a nutshell these methods change the underlying basis from the usual Gaussian-type orbitals to so-called Gaussian geminals, where an explicit dependence on the electron-electron distance is directly incorporated inside the basis functions. The result is much faster convergence for Post-HF methods [35].

Furthermore we only took the viewpoint of modelling individual molecules so far, mainly because this will be the focus of the thesis. But the modelling of electronic structures is not restricted to single molecules. Periodic or extended systems are accessible for quantum-chemical simulations as well. For these a combination of DFT as the underlying quantum-chemical method with a plane-waves or a projector-augmented wave basis are well-established and highly suitable [36–39].

This overview suggests that going beyond the usual Gaussian-based discretisations and towards employing novel basis function types could potentially yield quantum-chemical methods, which might allow for a more suitable description of certain features and properties. One of the challenges for developing such new methods are the deviating

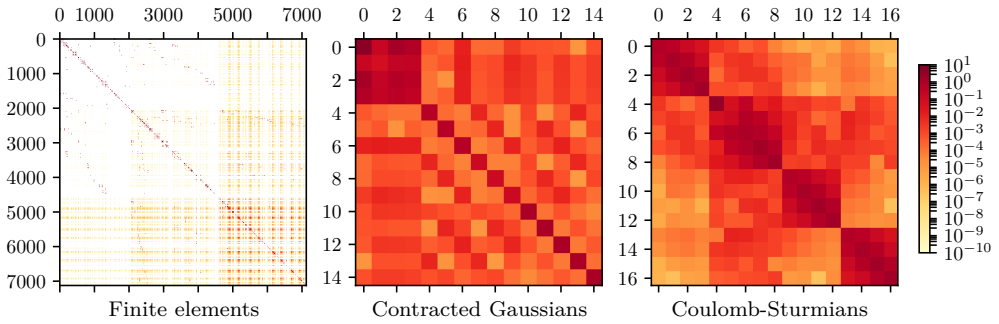


Figure 1.1: Structure of typical Fock matrices taken from a Hartree-Fock calculation for beryllium if the indicated basis function types were used for the discretisation. The absolute values of the matrices are depicted to the same scale with elements smaller than 10^{-10} coloured in white.

mathematical properties between conventional Gaussian-type orbitals as well as alternative basis functions. As an example figure 1.1 shows the structure of the Fock matrix when discretisations using finite elements, contracted Gaussians or Coulomb-Sturmians are employed. The most drastic difference can be seen for a finite-element discretisation, where the matrix is both much larger as well as sparser compared to the other two cases. In fact the discretisation which gave rise to the Fock matrix depicted in figure 1.1 is still too small to yield a sensible description of the beryllium atom density. A realistic description would need on the order of 10^5 to 10^6 basis functions [22]. For this reason it is not possible to keep the finite-element discretised Fock matrix completely in memory and diagonalise it at once, which is the standard approach when employing contracted Gaussians. Instead iterative diagonalisation methods need to be used for finite elements. Coulomb-Sturmians are some sort of a middle ground, where iterative methods are not necessarily required, but open up the possibility for more efficient algorithms, as we will discuss in more detail later in this work. The implementation of alternative basis function types therefore requires in many cases other numerical techniques compared to Gaussian-based discretisations.

As mentioned before, existing packages for quantum-chemical calculations predominantly rely on Gaussian-type orbitals and as a consequence are highly optimised towards their properties. Implicit assumptions about the structure of the discretised quantities, like the Fock matrix shown above, are thus typically scattered throughout these large codes making the implementation of alternative types of basis functions rather challenging and time consuming. Especially for the initial testing of novel methods towards their range of applicability in standard problems of electronic structure theory this is an obstacle. On the other hand starting from scratch for each new basis function type is not an ideal option either, since one is faced with the task of reprogramming all the algorithms for which already hundreds of man-years of development have been spent in existing programs.

In this thesis a different approach is presented, which is followed by the `molsturm`[1] program package [40]. Motivated by the mathematical structure of the HF problem and the self-consistent field (SCF) approach usually utilised to solve it, `molsturm` is

[1] `https://molsturm.org`

designed to be a quantum-chemical method development framework, which supports experimentation at all levels. The SCF process in `molsturm` uses a contraction-based approach, where the Fock matrix is not stored in memory, but where matrix-vector contraction expressions are sufficient. As will be discussed, this allows to separate the code dealing with the SCF scheme from the basis-function specific details, such that the SCF code is highly general, but the integral back end is still in full control over the way integral data is produced and consumed. A key result is that new types of basis functions as well as new SCF algorithms can be readily incorporated and tried within the existing framework. It is explicitly *not* the goal of `molsturm` to recode every aspect of quantum-chemical modelling, but instead to facilitate integration of the implemented SCF procedure with existing software for Post-HF methods by simple and easy-to-use interfaces. One can best think of `molsturm` as a mediator between integral libraries and Post-HF methods, where new developments on either side can be quickly connected and tested for their applicability in the quantum-chemical modelling of electronic structures.

Following along these lines chapter 2 and chapter 3 provide the background for treating the problems of quantum mechanics as well as quantum chemistry numerically. Starting from the similarities of classical and quantum physics the reader will be introduced to functional analysis and spectral theory in chapter 2, always motivated from a quantum-chemical perspective. In chapter 3 we introduce standard projection techniques for transforming from the picture of exact mathematics into the field of numerical linear algebra, which allows to solve quantum-chemical problems on a computer.

Thereafter chapter 4 deals with the mathematical and numerical structure of quantum chemistry with a strong focus on the HF problem. Chapter 5 follows with a detailed review of numerical techniques and SCF algorithms for solving HF.

Chapter 6 picks up on the lessons learned about the numerical structure of HF and discusses contraction-based methods and so-called lazy matrices in order to program algorithms in a basis-function independent way. In chapter 7 the design of the quantum-chemistry package `molsturm` is presented, emphasising its usefulness for investigating novel methods for quantum-chemical calculations and computational electronic structure theory.

In chapter 8 we employ `molsturm` to perform a systematic assessment of the convergence properties of Coulomb-Sturmians for the quantum-chemical modelling of atoms and make some suggestions regarding sensible Coulomb-Sturmian basis sets for ground state calculations of atoms at HF level of theory. Finally chapter 9 concludes the work and chapter 10 gives an outlook into further directions of research.

Chapter 2

Mathematical foundation of quantum mechanics

> *Whenever we proceed from the known into the unknown we may hope to understand, but we may have to learn at the same time a new meaning of the word "understanding".*
>
> — Werner Heisenberg (1901–1976)

At the turn of the 19th century it was discovered that classical mechanics in the formulations provided by Joseph-Louis Lagrange and William Hamilton is not able to capture all effects observed in experiments. Especially the phenomenon of the black-body radiation spectrum, but also the photoelectric effect could not be explained. Finally in 1900 Max Planck somewhat reluctantly introduced the idea of discrete, i.e. quantised, energy levels in order to explain the black-body radiation spectrum, building on earlier ideas by Ludwig Boltzmann. This started the development of a quantised theory of mechanics with major contributions by nowadays famous names such as Niels Bohr, Max Born, Louis de Broglie, Paul Dirac, Albert Einstein, Vladimir Fock, Pascual Jordan, John von Neumann, Erwin Schrödinger and others.

In this chapter we will discuss the mathematical structure of quantum mechanics in light of the problems of atomic physics and quantum chemistry. The discussion will focus on the mathematical fields of functional analysis and spectral theory as these are key in order to understand the peculiarities with computing discrete energy levels. The connection to atomic physics is made clear wherever possible. The parts of the chapter, where we build up the required mathematical language might seem rather technical, still. In the chapter we follow the excellent material by Shankar [41], Müller [42] and Helffer [43].

2.1 Correspondence of classical and quantum mechanics

According to the Hamiltonian formulation of classical mechanics a physical system with d degrees of freedom is described by a set of generalised coordinates $q_1, \ldots, q_d$ along

with their canonical momenta $p_1, \ldots, p_d$. It is assumed that any physically measurable quantity F only depends on these system parameters. In other words one may define a so-called **observable** $F(q_1, \ldots, q_d, p_1, \ldots p_d)$ as a function $\mathbb{R}^{2d} \to \mathbb{R}$, i.e. from a vector in **phase space** to the measured value. Clearly the coordinates q_i and momenta p_i are observables as well. The most important observable is the total energy function or **Hamiltonian**

$$H(q_1, \ldots, q_d, p_1, \ldots p_d) \equiv \underbrace{\frac{1}{2}\sum_{k=1}^{d} \frac{p_k^2}{m_k}}_{=T(p_1,\ldots,p_d)} + V(q_1, \ldots, q_d), \tag{2.1}$$

where m_k is the mass of the particle associated with the degree of freedom k, T is the kinetic energy observable and V the total potential energy observable. In the formalism of Hamiltonian mechanics, H governs the time evolution of the system, namely

$$\frac{\mathrm{d}p_k}{\mathrm{d}t} = -\frac{\partial H}{\partial q_k}, \qquad \frac{\mathrm{d}q_k}{\mathrm{d}t} = \frac{\partial H}{\partial p_k} \qquad \forall k \in \{1, \ldots, d\}. \tag{2.2}$$

These expressions allow to generalise the description of the time evolution to any other arbitrary observable as well. To make the connection to quantum mechanics more apparent, let us introduce for this purpose the so-called **Poisson bracket**. It is the skew-symmetric form

$$\{F, G\}_P \equiv \sum_{j=1}^{d} \left(\frac{\partial F}{\partial q_j}\frac{\partial G}{\partial p_j} - \frac{\partial F}{\partial p_j}\frac{\partial G}{\partial q_j} \right). \tag{2.3}$$

According to the Liouville equation

$$\frac{\mathrm{d}F}{\mathrm{d}t} = \frac{\partial F}{\partial t} + \{F, H\}_P . \tag{2.4}$$

it relates the Hamiltonian H to the time evolution of any arbitrary observable F. One may also show the relationships

$$\{q_k, q_l\}_P = \{p_k, p_l\}_P = 0 \qquad \{p_k, q_l\}_P = -\delta_{kl} \qquad \forall k, l \in \{1, \ldots, d\} \tag{2.5}$$

between the principle system observables.

2.1.1 Moving to quantum mechanics

We will now introduce (non-relativistic) quantum mechanics (QM) in a rather pragmatic manner, namely by stating a summary what changes in the QM formulation compared to the classical one. The full mathematical details are not yet stated at this point and not all terms defined. The reader should take this overview as a motivation for a more detailed treatment of the mathematical concepts further down this chapter.

(a) Instead of phase space vectors $(q_1, \ldots, q_d, p_1, \ldots p_d) \in \mathbb{R}^{2d}$, in QM a particular state of the system is represented by functions $\Psi : \mathbb{R}^d \to \mathbb{C}$ mapping from space coordinates to a complex number. These originate from a complex separable Hilbert space $\mathcal{H}$.

(b) A classical observable F is represented by a self-adjoint operator $\hat{\mathcal{F}}$ on the Hilbert space $\mathcal{H}$ in QM.

(c) For each classical observable one may construct an equivalent corresponding quantum-mechanical operator. For example the observable q_k corresponds[1] to [41, 42]

$$q_k \longrightarrow \hat{\mathrm{x}}_k = x_k,$$

i.e. just the multiplication with x_k, the k-th coordinate of the system. On the other hand p_k corresponds to [41, 42]

$$p_k \longrightarrow \hat{\mathrm{p}}_k = \frac{\hbar}{\imath}\frac{\partial}{\partial x_k},$$

an appropriately scaled derivative with respect to x_k.

(d) Relationships originating from classical mechanics can (usually) be transformed into their QM analogue. For this replace all occurrences of the Poisson bracket and the contained classical observables with the commutator

$$\left[\hat{\mathcal{F}}, \hat{\mathcal{G}}\right] = \hat{\mathcal{F}}\hat{\mathcal{G}} - \hat{\mathcal{G}}\hat{\mathcal{F}} \tag{2.6}$$

and corresponding operators, i.e. [41]

$$\{F, G\}_P \longrightarrow \frac{\imath}{\hbar}\left[\hat{\mathcal{F}}, \hat{\mathcal{G}}\right].$$

(e) The measured values of an observable F are the eigenvalues λ_k of $\hat{\mathcal{F}}$ *only*.

(f) Assume that we can find a complete and countable set of eigenpairs $\{(\lambda_\mu, \Psi_\mu)\}_{\mu\in\mathcal{I}}$ for the operator[2] $\hat{\mathcal{F}}$, where $\mathcal{I}$ is an appropriate index set. One may then compute the expectation value of a measurement on a normalised[3] state Φ as

$$\langle\hat{\mathcal{F}}\rangle = \left\langle \Phi \middle| \hat{\mathcal{F}}\,\Phi \right\rangle_{\mathcal{H}} = \sum_{\mu\in\mathcal{I}} \lambda_\mu \left|\langle\Psi_\mu|\Phi\rangle_{\mathcal{H}}\right|^2, \tag{2.7}$$

where $\langle\,\cdot\,|\,\cdot\,\rangle_{\mathcal{H}}$ is the inner product of the Hilbert space $\mathcal{H}$.

Now that we discussed the *ad hoc* modification of classical mechanics in order to yield a theory based on the postulates of QM, let us see how one is able to deduce useful results of QM from the analogous expressions in classical mechanics. For example from (2.5) we can immediately deduce important commutator relations between the position and momentum operators:

$$[\hat{\mathrm{x}}_k, \hat{\mathrm{x}}_l] = [\hat{\mathrm{p}}_k, \hat{\mathrm{p}}_l] = 0 \qquad\qquad [\hat{\mathrm{x}}_k, \hat{\mathrm{p}}_l] = \frac{\hbar}{\imath}\delta_{kl}. \tag{2.8}$$

Similarly from (2.4) we obtain

$$\frac{\mathrm{d}\hat{\mathcal{F}}}{\mathrm{d}t} = \frac{\partial\hat{\mathcal{F}}}{\partial t} + \frac{\imath}{\hbar}\left[\hat{\mathcal{F}}, \hat{\mathcal{H}}\right], \tag{2.9}$$

[1] In fact alternative constructions for the position and momentum operator are possible as well. In this work the so-called *position representation* is presented.

[2] We will see in section 2.3 on page 20 that this is *not* always possible. A more general treatment of spectral theory involving *spectral projectors* allows to reformulate point (f) for cases where such eigenpairs cannot be found. See [43] for details.

[3] unit normalised, i.e. $\langle\Phi|\Phi\rangle_{\mathcal{H}} = 1$

the equation of motion, which governs the time-evolution of the operator $\hat{\mathcal{F}}$ in the so-called **Heisenberg picture** of QM. Taking the statistical average over (2.9) results in the Ehrenfest theorem

$$\frac{\mathrm{d}\langle\hat{\mathcal{F}}\rangle}{\mathrm{d}t} = \frac{\partial\langle\hat{\mathcal{F}}\rangle}{\partial t} + \frac{1}{\imath\hbar}\left\langle\left[\hat{\mathcal{F}},\hat{\mathcal{H}}\right]\right\rangle. \tag{2.10}$$

This result allows to rationalise the **correspondence principle** of classical mechanics and QM, which we have developed so far. Comparing (2.10) and (2.4) and keeping in mind that the expectation value $\langle\hat{\mathcal{F}}\rangle$ as well as the classical observable F both tell us about the result of a measurement, we can deduce that — on average — classical mechanics still holds. Thus we may expect the classical expressions to carry some meaning in the QM sense as well.

2.1.2 The Schrödinger equation

Even though the Heisenberg picture developed in the previous section is descriptive for deducing the analogy between classical mechanics and QM, it is not very suitable for the kinds of problems we will be looking at in the remainder of this thesis. More suitable for our needs is the **Schrödinger picture** of QM, which differs in the way it treats time evolution. In the Heisenberg picture the state function Ψ is time-independent and the operators evolve. In the Schrödinger picture it is the other way round, i.e. Ψ may change over time and the operators are static. Both pictures are related by a unitary transformation in the Hilbert space $\mathcal{H}$ governed by the Stone-von Neumann theorem[44–46].

The equivalent expression to (2.9) in the Schrödinger picture is the **time-dependent Schrödinger equation** [41, 42]

$$\hat{\mathcal{H}}\Psi = \imath\hbar\frac{\partial}{\partial t}\Psi. \tag{2.11}$$

Similar to (2.9) the key operator governing the time-evolution of the system is $\hat{\mathcal{H}}$. By analogy to its classical counterpart $\hat{\mathcal{H}}$ is referred to as the **QM Hamiltonian** or just Hamiltonian as well. In fact many properties of a system may already be deduced by considering the eigendecomposition of its Hamiltonian $\hat{\mathcal{H}}$ alone. In light of this the ansatz

$$\hat{\mathcal{H}}\Psi_\mu = E_\mu\Psi_\mu \tag{2.12}$$

for finding the Hamiltonian's eigenpairs

$$(E_\mu, \Psi_\mu) \in \mathbb{R} \times \mathcal{H}$$

is given the name **time-independent Schrödinger equation** (TISE) as well.

2.1.3 The Hamiltonian of the hydrogen-like atom

Employing the correspondence principle it is often very convenient to construct the QM Hamiltonian of a system starting from the classical energy expression. Let us consider a hydrogen-like system, where a particle of positive charge Ze is clamped at the origin and surrounded by a single electron. In Cartesian coordinates the position of the electron can be described by the vector $\underline{\boldsymbol{r}} \equiv (x_1, x_2, x_3)^{\mathrm{T}}$ and its momentum by the vector $\underline{\boldsymbol{p}}$. The classical kinetic energy and potential energy of such a system are given by

$$T = \frac{\underline{\boldsymbol{p}} \cdot \underline{\boldsymbol{p}}}{2m_e} \qquad \text{and} \qquad V = -\frac{Ze^2}{\|\underline{\boldsymbol{r}}\|_2} = -\frac{Ze^2}{r}. \tag{2.13}$$

respectively. The appropriate QM analogues are

$$\hat{\mathcal{T}} = -\frac{\hbar^2}{2m_e}\Delta \qquad \text{and} \qquad \hat{\mathcal{V}} = -\frac{Ze^2}{r} \tag{2.14}$$

where the **Laplace operator** in 3 dimensions

$$\Delta = \sum_{i=1}^{3} \frac{\partial^2}{\partial x_i^2} \tag{2.15}$$

was introduced. The full Hamiltonian therefore reads

$$\hat{\mathcal{H}} = \hat{\mathcal{T}} + \hat{\mathcal{V}} = -\frac{\hbar^2}{2m_e}\Delta - \frac{Ze^2}{r}. \tag{2.16}$$

The eigenvalues of the hydrogen-like Hamiltonian (2.16) can be determined analytically, see section 2.3.5 on page 28 for details. The lowest eigenvalue is an energy of around $-2.18 \cdot 10^{-18}\,\mathrm{J}$, hardly a convenient number. In fact most energy values in quantum chemistry and physics are of this order of magnitude. Similarly many other relevant quantities like the charge of the involved particles or typical lengths are only very small numbers. For this reason so-called **atomic units** are typically employed. These are generated by a unitary transformation of the Hilbert space, which effectively yields

$$\hbar \equiv e \equiv m_e \equiv a_0 \equiv E_h \equiv 1.$$

See table 2.1 on the next page for the values of these quantities in terms of the usual SI units. Employing this transformation on (2.16) gives rise to

$$\hat{\mathcal{H}} = -\frac{1}{2}\Delta - \frac{Z}{r} \tag{2.17}$$

for the hydrogen-like Hamiltonian in atomic units. Its lowest energy eigenvalue is $-1/2E_h$, certainly a more pleasant number. Additionally the sketched transformation has simplified the expression of the operator from (2.16) to (2.17), which is in fact a general observation for the relevant equations of quantum physics and chemistry, providing another justification for their use. From now on we will work exclusively in these units. A more detailed discussion of atomic units, approaching the subject from a slightly different angle can be found in [47].

symbol	name	atomic unit of	value in SI units
$\hbar$	Planck constant	action	$1.055 \cdot 10^{-34}$ Js
e	elementary charge	charge	$1.602 \cdot 10^{-19}$ C
m_e	electron mass	mass	$9.109 \cdot 10^{-31}$ kg
a_0	Bohr radius	length	$5.292 \cdot 10^{-11}$ m
E_h	Hartree energy	energy	$4.360 \cdot 10^{-18}$ J

Table 2.1: Atomic units and their relationship to SI units. Values taken from [48].

2.2 Elements of functional analysis

The mathematical field of functional analysis is concerned with the study of Banach and Hilbert spaces as well as the properties of mappings between such structures. In this work we will neglect Banach spaces and focus on Hilbert spaces only due to their exceptional importance in the mathematical structure of quantum mechanics, see the previous section 2.1.1. After some general remarks, we will take a closer look at the Lebesgue space $L^2(\mathbb{R}^d, \mathbb{C})$ as well as Sobolev spaces in the context of QM.

In this section we assume familiarity with the concept of a vector space as well as some intuitive understanding of the Lebesgue integral. For a more detailed discussion developing such concepts by generalising standard Euclidean geometry, see [49].

2.2.1 Definition of Hilbert spaces

Hilbert spaces are generalising some concepts of two- or three-dimensional Euclidean space to larger vector spaces of possibly infinite dimensions. Most notably taking limits or computing lengths and angles is possible in the same way as for Euclidean geometry, thus allowing to perform vector calculus or to numerically approximate in a sound way. Same as vector spaces, Hilbert spaces are defined with respect to a field $\mathbb{F}$, see definition below. In our case $\mathbb{F}$ can be typically identified with the field of all complex numbers $\mathbb{C}$ or the real numbers $\mathbb{R}$.

The first ingredients to a Hilbert space are ways to measure angles and distances, i.e. an inner product and a norm.

Definition 2.1. An **inner product space** over a field $\mathbb{F}$ is a vector space V (over the same field) that is further equipped with an **inner product**, i.e. a map

$$\langle \cdot | \cdot \rangle_V : V \times V \to \mathbb{F}$$

that satisfies(for all vectors $x, y, z \in V$ and all $\alpha \in \mathbb{F}$)

$$\langle x|y\rangle_V^* = \langle y|x\rangle_V \qquad \textit{(conjugate symmetry)} \tag{2.18}$$

$$\langle x|\alpha y + z\rangle_V = \alpha \langle x|y\rangle_V + \langle x|y\rangle_V \qquad \textit{(linearity in the last argument)} \tag{2.19}$$

$$\langle x|x\rangle_V \geq 0 \quad \text{and} \quad \langle x|x\rangle_V = 0 \;\Rightarrow\; x = 0 \qquad \textit{(positive-definiteness)}, \tag{2.20}$$

where the asterisk "$*$" denotes complex conjugation. We typically drop the "V" subscript from the notation of the inner product if the underlying vector space is clear from context.

Remark 2.2. Some literature uses a deviating definition for the inner product, where not the second, but the first argument in (2.19) is linear, i.e. where (2.19) would be

replaced by

$$\langle \alpha y + z | x \rangle_V = \alpha \langle y | x \rangle_V + \langle z | x \rangle_V .$$

Our definition is in better agreement with the usual convention of quantum physics and quantum chemistry due to the resemblance of Dirac notation [49].

Definition 2.3. Given a vector space V over the field $\mathbb{F}$, a **norm** is a map $\|\cdot\| : V \to \mathbb{R}$ such that the following axioms hold for all vectors $x, y \in V$ and all $\alpha \in \mathbb{F}$:

$$\|\alpha x\| = |\alpha| \, \|x\| \qquad \textit{(absolute scalability)} \tag{2.21}$$

$$\|x + y\| \leq \|x\| + \|y\| \qquad \textit{(triangle inequality)} \tag{2.22}$$

$$\text{If } \|x\| = 0 \quad \Rightarrow \quad x \text{ is the zero vector} \qquad \textit{(norm separates points)} \tag{2.23}$$

If such a norm can be found for a particular vector space V, one typically refers to V as a **normed vector space** as well.

Proposition 2.4. *For every inner product space exists the so-called* ***induced norm***

$$\|x\|_V = \sqrt{\langle x | x \rangle_V} \qquad \forall x \in V. \tag{2.24}$$

One may drop the subscript on the norm if it is clear from context.

Proof. See [49]. □

The second ingredient for a Hilbert space is a property called **completeness**. Formally it is defined as such:

Definition 2.5. A vector space V is called **complete** if every **Cauchy sequence** of vectors in V has a limit in V.

Let us first recall, that a sequence $(x_n)_{n \in \mathbb{N}} \in V$ is called Cauchy if

$$\forall \varepsilon > 0 \quad \exists M \in \mathbb{N} \quad \text{such that} \quad \|x_n - x_m\|_V < \varepsilon \quad \forall n, m > M.$$

One can show that every converging sequence is Cauchy. A roughly equivalent way of phrasing definition 2.5 is therefore, that a space V is complete iff every sequence (x_n) of elements which come arbitrarily close at large enough n tend towards an element, which is from V *as well*.

Example 2.6. To make the concept of completeness more clear, let us consider a counterexample. For this let us leave the setting of vector spaces and more broadly think about sequences defined on sets of numbers[4], where the concept of completeness applies as well.

It is well known that the sequence

$$x_n = \sum_{k=0}^{n} \frac{1}{k!} \in \mathbb{Q}$$

[4]This is fine, since completeness is in fact a property on so-called metric spaces, which are related to normed vector spaces, but have much less structure.

converges to Euler's number e, i.e.

$$\lim_{n\to\infty} x_n = e \notin \mathbb{Q}.$$

In other words $\mathbb{Q}$ is not complete.

One may, however, build the **completion** of $\mathbb{Q}$ by just including all limiting points of all sequences in $\mathbb{Q}$. In fact this is one way of defining the set of real numbers $\mathbb{R}$.

Remark 2.7. A subtle point about completeness is that it depends on the norm which is used to determine whether a sequence is Cauchy or not. In other words a vector space may be complete with respect to one norm, but not with respect to another. Similarly the completion of a space with respect to two different norms may yield different spaces.

In practice the choice of the norm is only of importance for infinite-dimensional vector spaces, since for finite-dimensional real or complex vector spaces all norms are equivalent[5] anyway.

Finally we can state

Definition 2.8. A **Hilbert space** $\mathcal{H}$ is an inner product space, which is complete with respect to the induced norm.

In other words a Hilbert space is a space, where the inner product naturally defines a way to measure distances and take limits, that is perform calculus. Thinking ahead towards the integral and differential operators we will define on such Hilbert spaces, this is exactly what we will need.

Before we look into some Hilbert spaces relevant for QM, let us first clarify the concept of **denseness** and **separability**.

Definition 2.9. A subspace S of a vector space V is called **dense in** V if each vector $x \in V$ either is a member of S or one may find a Cauchy sequence in S for which x is the limit point.

In other words S is dense in V if we can — for each element of V — construct a sequence of approximations inside the smaller space S, representing the desired element up to arbitrary accuracy. Denseness is therefore one of the fundamental properties required for approximation.

Example 2.10. Returning to example 2.6 on the previous page we note, that $\mathbb{Q}$ is dense in $\mathbb{R}$. This guarantees that we may approximate any real number up to arbitrary accuracy by an appropriate sum of fractions, which is one of the assumptions behind any floating point operation performed on the computer.

Definition 2.11. A Hilbert space is **separable**[6] iff it admits a countable orthonormal basis.

Remark 2.12. If a Hilbert space is separable we can find a basis set[7] $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ of at most countably infinite cardinality, i.e. where $\mathcal{I}_{\text{bas}} \subseteq \mathbb{N}$. With this we can write for each

[5] That is they induce the same topology.

[6] In the broader context of metric spaces, a separable space has a countable, dense subset.

[7] This remark sketches the construction of a so-called *Schauder basis*, which is related, but not identical to the concept of a *Hamel basis*, which is usually employed in finite-dimensional linear algebra.

$\Psi \in \mathcal{H}$:

$$\Psi = \sum_{\mu \in \mathcal{I}_{\text{bas}}} c_\mu \varphi_\mu. \tag{2.25}$$

This in turn uniquely identifies each Ψ with a sequence $(c_\mu)_{\mu \in \mathbb{N}}$ of complex numbers. By this means each complex, separable Hilbert space is isomorphic to the space of complex-valued, square-summable sequences $l^2(\mathbb{N}, \mathbb{C})$. One can easily show that this isomorphism is even an isometry, i.e.

$$\|\Psi\|_{\mathcal{H}} = \|(c_\mu)\|_{l^2} = \sqrt{\sum_{\mu=0}^{\infty} |c_\mu|^2}.$$

By transitivity all separable Hilbert spaces are isometrically isomorphic to each other.

In our remaining discussion we will only encounter complex, separable Hilbert spaces. This implies:

- If $\Psi \in \mathcal{H}$ is a vector in a Hilbert space, we can always identify it with a (possibly infinite) column vector of complex coefficients.
- Finite-dimensional Hilbert spaces are isomorphic to $\mathbb{C}^d$, where d is the dimensionality. Their vectors are thus identified by a column of complex numbers of finite size.

Remark 2.13. A consequence of remark 2.12 is that we can numerically approximate all separable Hilbert spaces rather naturally. For example by restricting the sum in (2.25) to only a finite number of d basis functions, we can make sure that the resulting Ψ is located in only a d-dimensional subspace $\mathcal{H}^{(d)} \subset \mathcal{H}$. Moreover this subspace is dense, since in the limit of taking all basis functions, we get exactly $\mathcal{H}$. In turn since $\mathcal{H}^{(d)}$ is finite-dimensional, we can identify each approximation to Ψ with a vector in $\mathbb{C}^d$, which can be represented numerically on the computer, regardless of the structure of $\mathcal{H}$.

2.2.2 The Hilbert spaces of quantum mechanics

Now that we have the required basic concepts at hand, let us discuss the question which Hilbert space to take for quantum mechanics. In section 2.1.1 we said that the state functions $\Psi : \mathbb{R}^d \to \mathbb{C}$ are taken from a complex, separable Hilbert space. In our treatment we adhere to the **Copenhagen interpretation** or **Born interpretation** of the quantum-mechanical state Ψ, which associates the meaning of a probability density with the square of the state function $|\Psi(x_1, x_2, \dots x_d)|^2$ at each point in space $(x_1, \dots x_d)$. A more detailed analysis in light of this probabilistic meaning of Ψ suggests to take these functions from the Hilbert space of square-integrable functions $L^2(\mathbb{R}^d, \mathbb{C})$, which we will define now.

Definition 2.14. Given two suitable[8] functions $f, g : \mathbb{R}^d \to \mathbb{C}$, we can define an inner product

$$\langle f | g \rangle_{L^2} := \int_{\mathbb{R}^d} f^*(\underline{\boldsymbol{x}}) g(\underline{\boldsymbol{x}}) \, \mathrm{d}\underline{\boldsymbol{x}} \tag{2.26}$$

[8] In order for the inner product (2.26) to be well-defined, the integrand needs to be Lebesgue-measurable, i.e. f and g need to be chosen such that the integral over $f^* g$ can be performed in the Lebesgue sense.

and the corresponding induced norm function

$$|f|_{L^2} := \sqrt{\langle f|f\rangle_{L^2}} = \left(\int_{\mathbb{R}^d} |f(\underline{\boldsymbol{x}})|^2 \,\mathrm{d}\underline{\boldsymbol{x}}\right)^{1/2}, \tag{2.27}$$

where the integral — in both cases — is to be understood in the Lebesgue sense and we identified

$$\underline{\boldsymbol{x}} \equiv (x_1, \ldots, x_d)$$

for ease of notation.

Notice, that (2.27) is not yet a norm, but almost[9]. What is missing is the norm-separates-points-condition (2.23). The reason for this has to do with the subtleties of Lebesgue integration, namely that the result of a Lebesgue integral

$$\int_{\mathbb{R}^d} f \,\mathrm{d}x$$

is unchanged if we replace the integrand f by a function f', which is identical *almost everywhere*. This is meant to say, that changing f at infinitely many, well-separated places in order yield f' does not change the outcome of the integration. More mathematically one says that the Lebesgue integral is only uniquely defined up to **sets of measure zero**. Returning to (2.23), the problem is hence that there is more than one function satisfying $|f|_{L^2} = 0$, in fact infinitely many. To circumvent this problem one performs a trick, namely one puts all functions which are equivalent almost everywhere in one class and henceforth only thinks of them as one entity. In the language of mathematics, we form the quotient group under the equivalence relation of functions being identical almost everywhere. Under this procedure $|\cdot|_{L^2}$ becomes a full norm, denoted as $\|\cdot\|_{L^2}$ and we can define:

Proposition 2.15. *The set*

$$L^2(\mathbb{R}^d, \mathbb{C}) := \left\{ f : \mathbb{R}^d \to \mathbb{C} \;\middle|\; \|f\|_{L^2} < \infty \right\},$$

*where the norm $\|\cdot\|_{L^2}$ stays finite is the set of **square-integrable functions**. It forms a Hilbert space over the field $\mathbb{C}$.*

Proof. See [50]. □

Proposition 2.16. *$L^2(\mathbb{R}^d, \mathbb{C})$ is separable.*

Proof. See [51]. □

In other words $L^2(\mathbb{R}^d, \mathbb{C})$ truly satisfies all the requirements for being a suitable Hilbert space of QM as introduced in section 2.1.1 on page 8.

Remark 2.17. We did not introduce the most general theory of QM in this section, but only *non-relativistic, spin-free* QM. In a fully relativistic QM treatment each state is not a function returning a single complex value, but rather a function returning a **spinor**, which for relativistic QM and so-called spin-1/2 particles has 4 spin components per particle. In other words for an N-particle system the corresponding space would

[9] It is a so-called *semi-norm*.

be $L^2(\mathbb{R}^{3N}, \mathbb{C}^{4N})$. This work does not treat relativistic QM at all, much rather we will only deal with *non-relativistic, spin-adapted* QM, which only has 2 spin components per particle, thus states from $L^2(\mathbb{R}^{3N}, \mathbb{C}^{2N})$.

For simplifying the mathematical treatment we will nevertheless assume $\mathcal{H} = L^2(\mathbb{R}^d, \mathbb{C})$ for most of our analysis in the next few chapters and only introduce spin *ad hoc* in the form of the space $L^2(\mathbb{R}^{3N}, \mathbb{C}^{2N})$ once this is needed. We can do this without any loss of generality because of remark 2.12 on page 14, where we pointed out that all infinite-dimensional Hilbert spaces are isometrically isomorphic. This implies that all of the properties we have showed or will show based on the Hilbert space $L^2(\mathbb{R}^d, \mathbb{C})$ can be generalised to $L^2(\mathbb{R}^d, \mathbb{C}^s)$ with $s \geq 1$ with ease.

2.2.3 Sobolev spaces

Many operators of quantum mechanics including all the operators, which we will discuss in detail, involve taking derivatives of states. Whilst Lebesgue spaces are suitable for doing statistics, their mathematical structure does not make sure that derivatives of functions from $L^2(\mathbb{R}^d, \mathbb{C})$ stay in $L^2(\mathbb{R}^d, \mathbb{C})$. For example $1/r$ is square-integrable on $\mathbb{R}^3$, whilst its radial derivative $-1/r^2$ is not. One way to tackle this, is to take an appropriate subspace of $L^2(\mathbb{R}^d, \mathbb{C})$, which allows taking a certain number of derivatives. As it turns out the appropriate treatment for the numerical solution of partial differential equations, does not require the usual or **strong derivatives**, weak derivatives are sufficient. These are defined as such:

Definition 2.18. A function $f \in L^2(\mathbb{R}^d, \mathbb{C})$ has a **weak partial derivative** $g \in L^2(\mathbb{R}^d, \mathbb{C})$ with respect to x_i if

$$\forall \varphi \in C_0^\infty(\mathbb{R}^d, \mathbb{C}): \quad \langle g|\varphi\rangle_{L^2} = -\left\langle f \middle| \frac{\partial}{\partial x_i}\varphi \right\rangle_{L^2},$$

where $C_0^\infty(\mathbb{R}^d, \mathbb{C})$ is the space of all infinitely differentiable complex-valued functions with compact support. To denote the weak derivative one may write $g = \frac{\partial}{\partial x_i} f$ like in the strong case. It can further be shown that if f has a strong derivative then it also has a weak derivative, which coincides with the strong derivative. For ease of notation we also write

$$D^{\underline{\boldsymbol{\alpha}}} f = \frac{\partial^{\|\underline{\boldsymbol{\alpha}}\|_1}}{\prod_{i=1}^d \partial x_i^{\alpha_i}},$$

where $\underline{\boldsymbol{\alpha}} \in \mathbb{N}^d$ and as usual for the l_1-norm

$$\|\underline{\boldsymbol{\alpha}}\|_1 = \sum_{i=1}^d |\alpha_i|.$$

With the weak derivative at hand we can define the so-called Sobolev spaces, which allow to make certain guarantees about the number of (weak) derivatives, which can be taken. A full family of such spaces exist. We will only present two kinds here.

Definition 2.19. The **Sobolev space** defined by

$$H^k(\mathbb{R}^d, \mathbb{C}) := \left\{ f \in L^2(\mathbb{R}^d, \mathbb{C}) \;\middle|\; D^{\underline{\boldsymbol{\alpha}}} f \in L^2(\mathbb{R}^d, \mathbb{C}) \text{ for } \|\underline{\boldsymbol{\alpha}}\|_1 \leq k \right\} \tag{2.28}$$

$$\begin{array}{ccccccc} L^2(\mathbb{R}^d,\mathbb{C}) & \supset & H^1(\mathbb{R}^d,\mathbb{C}) & \supset & H^2(\mathbb{R}^d,\mathbb{C}) & \supset \cdots & \\ & & \cup & & \cup & & \\ & & H_0^1(\mathbb{R}^d,\mathbb{C}) & \supset & H_0^2(\mathbb{R}^d,\mathbb{C}) & \supset \cdots \supset & C_0^\infty(\mathbb{R}^d,\mathbb{C}) \end{array}$$

Figure 2.1: Overview of the spaces discussed in this section. Apart from $C_0^\infty(\mathbb{R}^d,\mathbb{C})$ all mentioned spaces are Hilbert spaces. In each case $A \subset B$ denotes that A is a proper, dense subspace of B.

with inner product

$$\langle f|g\rangle_{H^k} := \sum_{\|\underline{\boldsymbol{\alpha}}\|_1 \leq k} \langle D^{\underline{\boldsymbol{\alpha}}} f \mid D^{\underline{\boldsymbol{\alpha}}} g\rangle_{L^2} \tag{2.29}$$

and induced norm

$$\|f\|_{H^k} = \sum_{\|\underline{\boldsymbol{\alpha}}\|_1 \leq k} \|D^{\underline{\boldsymbol{\alpha}}} f\|_{L^2} \tag{2.30}$$

is a Hilbert space [50].

Definition 2.20. The completion of $C_0^\infty(\mathbb{R}^d,\mathbb{C})$ with respect to the norm $\|\cdot\|_{H^k}$ is the Sobolev space $H_0^k(\mathbb{R}^d,\mathbb{C})$. It is a proper subspace of $H^k(\mathbb{R}^d,\mathbb{C})$ and a Hilbert space as well [50].

Colloquially speaking if a function is a member of $H^k(\mathbb{R}^d,\mathbb{C})$ or $H_0^k(\mathbb{R}^d,\mathbb{C})$, we can assume that the k-th derivative of this function remains square-integrable. These spaces will become rather important in the next section 2.3 on page 20, where we will need them to define self-adjoint operators upon them. As a summary the relationships between the spaces we discussed in this section have been summarised in figure 2.1. Note, that by definition

$$L^2(\mathbb{R}^d,\mathbb{C}) = H^0(\mathbb{R}^d,\mathbb{C}) = H_0^0(\mathbb{R}^d,\mathbb{C}).$$

To finish our discussion of Sobolev spaces let us determine in which Sobolev space the function

$$\Psi_{1s}(\underline{\boldsymbol{r}}) = \exp\left(-\sqrt{x^2+y^2+z^2}\right) = \exp(-r) \tag{2.31}$$

is located. This function and trivial generalisations thereof will be of relevance for our future treatment, since it arises naturally as an eigenfunction of the hydrogen-like Hamiltonian (2.17) (see section 2.3.5 on page 28) and is furthermore an important building block of the Coulomb-Sturmians (see section 5.3.6 on page 115).

Example 2.21. The function Ψ_{1s} of (2.31) belongs to $H^1(\mathbb{R}^3,\mathbb{C})$.

Proof. Since the function is Riemann-integrable over $\mathbb{R}^3$, it is Lebesgue-integrable as well and as a result $\Psi_{1s} \in L^2(\mathbb{R}^3,\mathbb{C})$. Furthermore for any $\alpha \in \{x,y,z\}$:

$$\left\|\frac{\partial\Psi_{1s}}{\partial\alpha}\right\|_{L^2} = \left\|-\frac{\alpha}{r}\exp(-r)\right\|_{L^2} = \int_{\mathbb{R}^3}\frac{\alpha^2}{r^2}\exp(-2r)\,\mathrm{d}\underline{\boldsymbol{r}} \leq \int_{\mathbb{R}^3}\frac{r^2}{r^2}\exp(-2r)\,\mathrm{d}\underline{\boldsymbol{r}} \tag{2.32}$$

Due to the properties of the Lebesgue integral, we may ignore the removable discontinuity at $\underline{\boldsymbol{r}} = \underline{\boldsymbol{0}}$ and instead write

$$\left\|\frac{\partial\Psi_{1s}}{\partial\alpha}\right\|_{L^2} \leq \int_{\mathbb{R}^3}\exp(-2r)\,\mathrm{d}\underline{\boldsymbol{r}} = \|\Psi_{1s}\|_{L^2} < \infty.$$

This shows that $\exp(-r) \in H^1(\mathbb{R}^3, \mathbb{C})$, since each term of (2.30) is bound. □

For the next step, showing $\Psi_{1s} \in H^2(\mathbb{R}^3, \mathbb{C})$, we need two results relating $H^1(\mathbb{R}^3, \mathbb{C})$ and $L^2(\mathbb{R}^3, \mathbb{C})$.

Proposition 2.22 (Hardy's inequality)**.** *For all $u \in H^1(\mathbb{R}^3, \mathbb{C})$, we have*

$$\int_{\mathbb{R}^3} \|\nabla u\|_2^2 \, \mathrm{d}\underline{\boldsymbol{r}} \geq \frac{1}{4} \int_{\mathbb{R}^3} \frac{|u|^2}{r^2} \, \mathrm{d}\underline{\boldsymbol{r}}$$

Proof. For a proof of the special case $u \in C_0^\infty(\mathbb{R}^3, \mathbb{C})$ see [43, p. 30]. The more general case we claim here, follows from the denseness of $C_0^\infty(\mathbb{R}^3, \mathbb{C})$ in $H^1(\mathbb{R}^3, \mathbb{C})$ and continuity of the integrands on both sides with respect to the H^1 norm. □

Corollary 2.23. *If $u \in H^1(\mathbb{R}^3, \mathbb{C})$, then $\frac{u}{r} \in L^2(\mathbb{R}^3, \mathbb{C})$.*

Proof. One easily rewrites Hardy's inequality to

$$\|u\|_{H^1} \geq \sum_{\alpha \in \{x,y,z\}} \int_{\mathbb{R}^3} \left| \frac{\partial u}{\partial \alpha} \right| \mathrm{d}\underline{\boldsymbol{r}} \overset{\text{(triangle)}}{\geq} \int_{\mathbb{R}^3} \|\nabla u\|_2^2 \, \mathrm{d}\underline{\boldsymbol{r}} \overset{\text{(Hardy)}}{\geq} \frac{1}{4} \int_{\mathbb{R}^3} \frac{|u|^2}{r^2} \, \mathrm{d}\underline{\boldsymbol{r}} = \frac{1}{4} \left\| \frac{u}{r} \right\|_{L^2}$$

which proves the claim. □

Example 2.24. We now want to use corollary 2.23 to prove that $\Psi_{1s} \in H^2(\mathbb{R}^3, \mathbb{C})$.

Proof. Considering our result from (2.32) we find that for all $\alpha, \beta \in \{x, y, z\}$:

$$\left\| \frac{\partial^2 \Psi_{1s}}{\partial \alpha \partial \beta} \right\|_{L^2} \leq \left\| \frac{\delta_{\alpha\beta}}{r} \exp(-r) \right\|_{L^2} + \left\| \frac{\alpha\beta}{r^3} \exp(-r) \right\|_{L^2} + \left\| \frac{\alpha\beta}{r^2} \exp(-r) \right\|_{L^2}$$

Noting $|\alpha\beta| \leq r^2$ and ignoring the removable singularities in the Lebesgue integral, we arrive at

$$\begin{aligned} \left\| \frac{\partial^2 \Psi_{1s}}{\partial \alpha \partial \beta} \right\|_{L^2} &\leq \left\| \frac{1}{r} \exp(-r) \right\|_{L^2} + \left\| \frac{r^2}{r^3} \exp(-r) \right\|_{L^2} + \left\| \frac{r^2}{r^2} \exp(-r) \right\|_{L^2} \\ &= 2 \left\| \frac{1}{r} \exp(-r) \right\|_{L^2} + \|\exp(-r)\|_{L^2} \\ &< \infty, \end{aligned}$$

where in the last line we used $\exp(-r) \in H^1(\mathbb{R}^3, \mathbb{C})$ and corollary 2.23. □

Remark 2.25. Analogously to what we sketched in examples 2.21 on the facing page and 2.24, one could attempt to probe whether the one-dimensional function $f(x) = \exp(-|x|)$ is in $H^1(\mathbb{R}, \mathbb{C})$ or $H^2(\mathbb{R}, \mathbb{C})$. Whilst the former can be easily verified, one finds $f \notin H^2(\mathbb{R}, \mathbb{C})$.

This rather surprising result is a consequence of the second part of the Sobolev embedding theorem of which we only present a slightly specialised form here.

Theorem 2.26 (Sobolev embedding)**.** *Given $r, k, d \in \mathbb{N}$ with*

$$k > \frac{d}{2} > 0 \quad \textit{and} \quad k - \frac{d}{2} > r$$

one may find an embedding

$$H^k(\mathbb{R}^d) \subset C^r(\mathbb{R}^d)$$

between the Sobolev space $H^k(\mathbb{R}^d)$ and the space of the r times continuously differentiable functions, $C^r(\mathbb{R}^d)$.

This embedding theorem allows to get an idea what is to be expected about the smoothness of a function in H^k. Interestingly the smaller the dimensionality the smoother such a function has to be.

2.3 Spectral theory

In this chapter we will broaden our discussion focusing on linear operators between the state functions of a Hilbert space. We will discuss certain common classes of operators including self-adjoint and compact operators as well as their spectral properties. We will see that most operators, including the ones required for atomic physics and quantum chemistry, do not show all the nice properties we would like to rely on. For example one might not be able to find eigenfunctions for all values of the spectrum and the ones one is able to determine might not amount to span the Hilbert space completely. For this reason we will hint at techniques relevant to the Hilbert space $L^2(\mathbb{R}^d, \mathbb{C})$ and a few of the relevant operators of QM, which will allow us to recover at least part of the eigenspectrum with the numerical methods discussed in chapter 3 on page 31.

2.3.1 Bounded and self-adjoint operators

Mathematically a linear operator is defined as such:

Definition 2.27. A **linear operator** on a Hilbert space $\mathcal{H}$ is the linear map $\hat{\mathcal{A}} : D(\hat{\mathcal{A}}) \to \mathcal{H}$, where $D(\hat{\mathcal{A}}) \subset \mathcal{H}$ is a subspace called the **domain** of $\hat{\mathcal{A}}$.

Typically we employ just the term **operator** to refer to linear operators. Recall that a mapping is called **linear** if for all $u, v \in \mathcal{H}$ and all $\alpha \in \mathbb{C}$

$$\hat{\mathcal{A}}(u+v) = \hat{\mathcal{A}}u + \hat{\mathcal{A}}v \qquad \hat{\mathcal{A}}(\alpha u) = \alpha \hat{\mathcal{A}}u \tag{2.33}$$

hold. Even though not strictly necessary, we will assume for our treatment that the Hilbert space is separable and that the domain of an operator is always dense in it.

Proposition 2.28. *The inner product of $\mathcal{H}$ induces the so-called operator norm*

$$\left\|\hat{\mathcal{A}}\right\|_{\mathcal{L}(\mathcal{H})} := \sup_{\substack{u \in D(\hat{\mathcal{A}}), \\ u \neq 0}} \frac{\left\|\hat{\mathcal{A}}u\right\|_{\mathcal{H}}}{\|u\|_{\mathcal{H}}}.$$

Proof. See [51, Satz II.1.4] □

The first important classification we will discuss here is the notion of bounded and unbounded operators.

Definition 2.29. An operator $\hat{\mathcal{A}}$ on $\mathcal{H}$ is **bounded** iff

$$\left\|\hat{\mathcal{A}}\right\|_{\mathcal{L}(\mathcal{H})} < \infty,$$

i.e. if it has finite operator norm. A bounded operator is referred to as **continuous**[10] as well. Operators, which are not bounded are **unbounded operators**.

In our example of $\mathcal{H} = L^2(\mathbb{R}^d, \mathbb{C})$ an operator is hence bounded if its action on a square-integrable function yields another function, which stays square-integrable. In the introductory paragraph of section 2.2.3 on page 17 we already noted that the radial derivative of the square-integrable function $1/r$, namely $-1/r^2$, is not square-integrable. It should therefore not come as a surprise that operators containing derivatives — like the kinetic energy operator of QM— are in general not bounded.

An alternative way to think about unbounded operators is accessible via the concept of an **operator extension**.

Definition 2.30. Let $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$ be operators on $\mathcal{H}$. $\hat{\mathcal{B}}$ is an extension of $\hat{\mathcal{A}}$ if $D(\hat{\mathcal{A}}) \subset D(\hat{\mathcal{B}})$ and if $\forall u \in D(\hat{\mathcal{A}}) : \ \hat{\mathcal{A}}u = \hat{\mathcal{B}}u$.

Proposition 2.31. *Let $\hat{\mathcal{A}}$ be an operator on $\mathcal{H}$. The following statements are equivalent:*

- *$\hat{\mathcal{A}}$ is unbounded.*
- *$\hat{\mathcal{A}}$ does not possess a bounded extension.*

Proof.

$\Rightarrow$ Since $\hat{\mathcal{A}}$ is unbounded, there exists a $x \in D(\hat{\mathcal{A}})$ with

$$\nexists\, \alpha \in \mathbb{R} : \quad \left\|\hat{\mathcal{A}}x\right\|_{\mathcal{H}} = \alpha \left\|x\right\|_{\mathcal{H}}.$$

For any extension $\hat{\mathcal{B}}$ by construction $x \in D(\hat{\mathcal{B}})$ and $\hat{\mathcal{B}}x = \hat{\mathcal{A}}x$, such that for this x

$$\nexists\, \beta \in \mathbb{R} : \quad \left\|\hat{\mathcal{B}}x\right\|_{\mathcal{H}} = \beta \left\|x\right\|_{\mathcal{H}}.$$

Thus any such $\hat{\mathcal{B}}$ is unbounded.

$\Leftarrow$ For $D(\hat{\mathcal{A}}) = \mathcal{H}$ the statement is trivial and w.l.o.g we assume $D(\hat{\mathcal{A}}) \neq \mathcal{H}$. Further we proceed by proving the equivalent statement, that each bounded operator $\hat{\mathcal{A}}$ possesses at least one bounded extension.

Given $\hat{\mathcal{A}}$ with $D(\hat{\mathcal{A}}) \subsetneq \mathcal{H}$ we can choose a $v \in \mathcal{H}$ with $v \perp D(\hat{\mathcal{A}})$ and $\|v\| = 1$. An extension $\hat{\mathcal{B}}$ of $\hat{\mathcal{A}}$ can be defined pointwise

$$\hat{\mathcal{B}}u \equiv \hat{\mathcal{A}}\big(u - v\,\langle v|u\rangle\,\big)$$

with $u \in D(\hat{\mathcal{B}}) = D(\hat{\mathcal{A}}) \oplus \{v\}$. For any $u \in D(\hat{\mathcal{B}}$ by construction $u - v\,\langle v|u\rangle \in D(\hat{\mathcal{A}})$ such that we can find a $\alpha \in \mathbb{R}$ satisfying

$$\begin{aligned}\left\|\hat{\mathcal{B}}u\right\| &= \left\|\hat{\mathcal{A}}(u - v\,\langle v|u\rangle)\right\| = \alpha \left\|u - v\,\langle v|u\rangle\right\| \\ &\leq \alpha \left(\|u\| + \|v\|^2 \|u\|\right) = 2\alpha \|u\|.\end{aligned}$$

[10] In fact this is a consequence from the fact that a bounded linear operator between normed vector spaces is always continuous.

In other words $\hat{\mathcal{B}}$ is a bounded extension of $\hat{\mathcal{A}}$.

□

There also exists a middle ground, namely so-called **semi-bounded operator**s, defined as such:

Definition 2.32. An operator $\hat{\mathcal{A}}$ on $\mathcal{H}$ with domain $D(\hat{\mathcal{A}})$ is called semi-bounded from below if there exists a constant C such that for all $u \in D(\hat{\mathcal{A}})$:

$$\left\langle u \middle| \hat{\mathcal{A}} u \right\rangle = \left\langle \hat{\mathcal{A}} u \middle| u \right\rangle \quad \text{and} \quad \left\langle u \middle| \hat{\mathcal{A}} u \right\rangle \geq -C \left\langle u | u \right\rangle .$$

Starting from definition 2.29 it is easy to show that a bounded operator $\hat{\mathcal{A}}$ on a Hilbert space $\mathcal{H}$ maps Cauchy sequences to Cauchy sequences, i.e. if $(x_n) \in \mathcal{H}$ is Cauchy, so is $(\hat{\mathcal{A}} x_n)$. In this sense a somewhat stronger version of boundedness is compactness, defined as:

Definition 2.33. An operator $\hat{\mathcal{A}} : D(\hat{\mathcal{A}}) \to \mathcal{H}$ on a Hilbert space $\mathcal{H}$ is **compact** if for any sequence (x_n) that converges weakly in $D(\hat{\mathcal{A}})$, $\hat{\mathcal{A}} x_n$ converges strongly in $\mathcal{H}$.

Recall that a sequence (x_n) is called **weakly convergent** if for all $\phi \in \mathcal{H}$ the sequence (y_n) with $y_n = \langle x_n | \phi \rangle_{\mathcal{H}}$ is **strongly convergent**, i.e. Cauchy.

Compactness is of importance for us in the context of spectral theory, since compact operators have particularly nice spectral properties. As expected one may easily show, that [51]

Proposition 2.34. *A compact operator $\hat{\mathcal{A}}$ defined on a Hilbert space is bounded as well.*

Remark 2.35. Each operator $\hat{\mathcal{A}}$ on a Hilbert space $\mathcal{H}$ can be uniquely identified with a sesquilinear form $a : \mathcal{H} \times \mathcal{H} \to \mathbb{C}$, defined by

$$\mathcal{H} \times \mathcal{H} \ni (u, v) \mapsto a(u, v) := \left\langle u \middle| \hat{\mathcal{A}} v \right\rangle_{\mathcal{H}} \in \mathbb{C}. \tag{2.34}$$

This is a consequence of the Riesz representation theorem [49].

In many applications, including the numerical treatment discussed in chapter 3 on page 31, the sesquilinear form a is more intuitive to employ than the operator $\hat{\mathcal{A}}$ itself.

Using the identification of the previous remark, we may define the terms symmetric and self-adjoint.

Definition 2.36. An operator $\hat{\mathcal{A}}$ on $\mathcal{H}$ is called **symmetric** if

$$\forall (u, v) \in D(\hat{\mathcal{A}}) \times D(\hat{\mathcal{A}}) : \qquad \left\langle \hat{\mathcal{A}} u \middle| v \right\rangle = \left\langle u \middle| \hat{\mathcal{A}} v \right\rangle$$

In Physics textbooks a symmetric operator is usually called Hermitian.

Definition 2.37. Let $\hat{\mathcal{A}}$ be a linear operator on $\mathcal{H}$ with (dense) domain $D(\hat{\mathcal{A}})$ and let $D(\hat{\mathcal{A}}^\dagger)$ be the space

$$D(\hat{\mathcal{A}}^\dagger) := \left\{ v \in \mathcal{H} \,\middle|\, \exists f_v \in \mathcal{H} \text{ such that } \forall u \in D(\hat{\mathcal{A}}) : \left\langle \hat{\mathcal{A}} u \middle| v \right\rangle = \langle u | f_v \rangle \right\},$$

where for each v the f_v is unique due to the denseness of $D(\hat{\mathcal{A}})$ in $\mathcal{H}$ and the Riesz representation theorem.

Then the **adjoint** of $\hat{\mathcal{A}}$ is the linear operator $\hat{\mathcal{A}}^\dagger$ with domain $D(\hat{\mathcal{A}}^\dagger)$ defined by

$$\forall v \in D(\hat{\mathcal{A}}^\dagger) \quad \left\langle \hat{\mathcal{A}}u \middle| v \right\rangle = \left\langle u \middle| \hat{\mathcal{A}}^\dagger v \right\rangle$$

Definition 2.38. A self-adjoint operator is an operator $\hat{\mathcal{A}}$ for which $\hat{\mathcal{A}}^\dagger = \hat{\mathcal{A}}$, or equivalently an operator which is symmetric *and* where $D(\hat{\mathcal{A}}) = D(\hat{\mathcal{A}}^\dagger)$.

Remark 2.39. For a *bounded* linear operator $\hat{\mathcal{A}}$ with $D(\hat{\mathcal{A}}) = \mathcal{H}$ one can find[11] a definition for the adjoint, which is more usual in the literature of quantum physics.

Namely by means of the identification

$$\forall (u, v) \in \mathcal{H} \times \mathcal{H} \quad \left\langle \hat{\mathcal{A}}u \middle| v \right\rangle = \left\langle u \middle| \hat{\mathcal{A}}^\dagger v \right\rangle$$

one can find a unique adjoint $\hat{\mathcal{A}}^\dagger$ for each bounded operator $\hat{\mathcal{A}}$. This operator will be bounded, too[12].

Comparing with the definition of a symmetric operator we find that for bounded operators the property of symmetric and self-adjoint are equivalent.

Remark 2.40. Even though symmetric and self-adjoint are related concepts, symmetric operators are not very useful in practice. Only self-adjoint operators have the nice mathematical properties, we require for quantum mechanics, namely a real spectrum and a spectral decomposition into bound and continuous states. See the next section for details.

Most operators in QM are not self-adjoint albeit being symmetric if defined in a naïve way. In many cases this issue can be circumvented by choosing an appropriate operator extension. We will discuss this is section 2.3.3 on page 27 and 2.3.5 on page 28 considering the spectrum of the Laplace operator (2.15) and the hydrogen-like operator (2.17).

Remark 2.41. As a summary of the terms introduced in this section, we note the following implications for an operator $\hat{\mathcal{A}}$ on a Hilbert space $\mathcal{H}$.

- $\hat{\mathcal{A}}$ compact $\Rightarrow$ $\hat{\mathcal{A}}$ bounded $\Rightarrow$ $\hat{\mathcal{A}}$ semi-bounded
- $\hat{\mathcal{A}}$ self-adjoint $\Rightarrow$ $\hat{\mathcal{A}}$ symmetric
- If $\hat{\mathcal{A}}$ bounded: $\hat{\mathcal{A}}$ self-adjoint $\Leftrightarrow$ $\hat{\mathcal{A}}$ symmetric.

2.3.2 Spectra of self-adjoint operators

In this section we will clarify the notion of a spectrum for a self-adjoint operator in infinite dimensions and the connections to the probably more familiar concepts of eigenvalues and eigenvectors in finite dimensions.

[11] Since $D(\hat{\mathcal{A}})$ in our treatment is dense in $\mathcal{H}$, one can always find a unique, bounded extension of $\hat{\mathcal{A}}$ with complete domain $\mathcal{H}$ if $\hat{\mathcal{A}}$ if bounded

[12] Note, that this makes the set of bounded operators on $\mathcal{H}$ a so-called C^* algebra.

Definition 2.42. Let $\hat{\mathcal{A}}$ be a self-adjoint[13] linear operator on $\hat{\mathcal{H}}$.

- We call the open set[14]
$$\rho(\hat{\mathcal{A}}) = \left\{ \lambda \in \mathbb{C} \;\middle|\; (\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}}) \text{ is bijective on } D(\hat{\mathcal{A}}) \right\}$$
the **resolvent set** of $\hat{\mathcal{A}}$.
- The closed set $\sigma(\hat{\mathcal{A}}) = \mathbb{C} \setminus \rho(\hat{\mathcal{A}})$ is then the **spectrum** of $\hat{\mathcal{A}}$.

We can further show: [43, p. 102]

Proposition 2.43. *If $\hat{\mathcal{A}}$ is self-adjoint, then $\sigma(\hat{\mathcal{A}}) \subset \mathbb{R}$.*

Another way of phrasing definition 2.42 is that the spectrum is the set of all points where $(\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}})$ is not bijective. This implies that both points where $(\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}})$ is not injective as well as points where $(\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}})$ is not surjective are part of the spectrum.

Recall that an eigenpair $(\lambda, v) \in \mathbb{C} \times \mathcal{H}$ satisfies

$$\hat{\mathcal{A}}v - \lambda v = 0 \quad \Leftrightarrow \quad v \in \ker(\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}}) \quad \Rightarrow \quad \ker(\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}}) \neq \{0\},$$

since $v \neq 0$. Since only non-injective operators can have a non-trivial kernel, this implies that $(\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}})$ is necessarily non-injective for (λ, v) to be an eigenpair. Unlike in finite dimensions[15] it may well happen in infinite dimensions, that $(\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}})$ is injective, but not surjective. Therefore one may find points in the spectrum, which are not eigenvalues. This is expressed more formally in the next definition.

Definition 2.44. If $\hat{\mathcal{A}}$ is self-adjoint, we can decompose $\sigma(\hat{\mathcal{A}}) = \sigma_P(\hat{\mathcal{A}}) \,\dot\cup\, \sigma_C(\hat{\mathcal{A}})$ with

- the **point spectrum**
$$\sigma_P(\hat{\mathcal{A}}) = \left\{ \lambda \in \mathbb{R} \;\middle|\; (\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}}) \text{ is non-injective} \right\} = \{\text{eigenvalues of } \hat{\mathcal{A}}\}.$$
- the **continuous spectrum**[16]
$$\sigma_C(\hat{\mathcal{A}}) = \overline{\left\{ \lambda \in \mathbb{R} \;\middle|\; (\hat{\mathcal{A}} - \lambda \,\mathrm{id}_{\mathcal{H}}) \text{ injective, but not surjective} \right\}}.$$

This definition can be understood physically by the so-called RAGE[17] theorem [52]. It draws a connection between the point spectrum $\sigma_P(\hat{\mathcal{A}})$ and the so-called **bound states** of an operator and between the continuous spectrum $\sigma_C(\hat{\mathcal{A}})$ and the so-called **scattering states**. Bound states are characterised by the property, that they have — at all times — a non-vanishing function value only in a finite region of space. Scattering states on the other hand are *not* eigenstates and they will vanish from any arbitrarily large, bounded part of space if enough time has passed.

The decomposition of the spectrum into point and continuous spectrum is not the only possibility. Especially from the point of view of numerical modelling the following, alternative approach is more helpful as we shall see later.

[13] Strictly speaking the operator only needs to be closed for this definition. An operator is closed if its graph $\{(u, \hat{\mathcal{A}}u) \,|\, u \in D(\hat{\mathcal{A}})\}$ is a closed subspace of $\mathcal{H} \times \mathcal{H}$. This is true for all self-adjoint operators.

[14] $\mathrm{id}_{\mathcal{H}}$ is the identity operator on the Hilbert space $\mathcal{H}$.

[15] In finite dimensions one can always find an operator extension for any injective operator to be surjective as well.

[16] $\overline{A}$ denotes the closure of the set A.

[17] Named after Ruelle, Amrein, Georgescu and Enss, who all worked on it.

Definition 2.45. For any self-adjoint[18] operator, we can decompose $\sigma(\hat{\mathcal{A}}) = \sigma_{\text{disc}}(\hat{\mathcal{A}}) \dot{\cup} \sigma_{\text{ess}}(\hat{\mathcal{A}})$ with

- the **discrete spectrum**[19]

$$\sigma_{\text{disc}}(\hat{\mathcal{A}}) \simeq \left\{ \lambda \in \mathbb{R} \,\middle|\, \lambda \text{ is an isolated eigenvalue of } \hat{\mathcal{A}} \text{ with finite multiplicity} \right\},$$

- the **essential spectrum**

$$\sigma_{\text{ess}}(\hat{\mathcal{A}}) = \sigma(\hat{\mathcal{A}}) \setminus \sigma_{\text{disc}}(\hat{\mathcal{A}}).$$

By construction the essential spectrum consists of

- the continuous spectrum,
- eigenvalues of infinite multiplicity,
- eigenvalues embedded inside the continuous spectrum.

It will become clear in a moment, why approximate numerical methods can only be easily used on the discrete spectrum. For this we need to discuss the special case of compact, self-adjoint operators in more detail. If an operator is compact, its spectrum has a particularly simple form.

Proposition 2.46. *If $\hat{\mathcal{A}}$ is a compact operator on the Hilbert space $\mathcal{H}$:*

- $0 \in \sigma(\hat{\mathcal{A}})$
- $\sigma(\hat{\mathcal{A}}) \setminus \{0\} = \sigma_P(\hat{\mathcal{A}}) \setminus \{0\}$
- *Only one of these cases is true:*
  - $\sigma(\hat{\mathcal{A}}) = \{0\}$
  - $\sigma(\hat{\mathcal{A}}) \setminus \{0\}$ *is finite.*
  - $\sigma(\hat{\mathcal{A}}) \setminus \{0\}$ *can be described as a sequence of points tending to* 0.

Proof. See [43, p. 56]. □

In other words the continuous spectrum of a compact operator may at most contain the value 0 — even in infinite dimensions. Furthermore there is a nice result for the eigenfunctions of a compact, self-adjoint operator:

Proposition 2.47. *Let $\mathcal{H}$ be a separable Hilbert space and $\hat{\mathcal{A}}$ a compact, self-adjoint operator on $\mathcal{H}$. The eigenfunctions of $\hat{\mathcal{A}}$, i.e. the set of all functions $\{u_k\}_{k\in\mathcal{I}} \subset \mathcal{H}$ with*

$$\hat{\mathcal{A}} u_k - \lambda_k u_k = 0$$

for a $\lambda_k \in \sigma(\hat{\mathcal{A}}) \setminus \{0\}$ are a Hilbertian basis for $\mathcal{H}$. In other words they satisfy

$$\langle u_k | u_l \rangle = \delta_{kl} \quad \forall\, k, l \in \mathcal{I} \qquad \text{and} \qquad \operatorname{span}\left(\{u_k\}_{k\in\mathcal{I}}\right) = \mathcal{H}.$$

Proof. See [43, p. 60]. □

[18] Again only closed is strictly speaking required.

[19] For a mathematically more precise description, see [43, p. 103 and p. 132].

With propositions 2.46 and 2.47 at hand, compact operators start to look a lot like the familiar case of complex square matrices. In fact one can show that [43, p. 43]

Proposition 2.48. *Any linear operator on a finite-dimensional Hilbert space is compact.*

This is essentially a consequence of the fact that in finite dimensions weak and strong convergence are equivalent.

Remark 2.49. With proposition 2.48 we can reduce the setting of self-adjoint operators $\hat{\mathcal{A}}$ on a finite-dimensional Hilbert space to the following:

- In remark 2.12 on page 14 we said that the vectors of a d-dimensional Hilbert space can be represented as column vectors from $\mathbb{C}^d$. In a similar sense $\hat{\mathcal{A}}$ can be identified by a finite matrix from $\mathbb{C}^{d\times d}$.
- The eigenfunctions of $\hat{\mathcal{A}}$ are a complete orthonormal basis for the underlying Hilbert space. $\hat{\mathcal{A}}$ has only real eigenvalues.
- Apart from zero $\hat{\mathcal{A}}$ has only a point spectrum. The essential spectrum and the continuous spectrum at most consist of 0.

Remark 2.50. As we will see in the next sections 2.3.3 on the next page and 2.3.5 on page 28 both the Laplace operator Δ as well as the Hamiltonian $\hat{\mathcal{H}}$ corresponding to hydrogen-like systems are not compact on the Hilbert space $L^2(\mathbb{R}^3, \mathbb{C})$, since both of these operators are not even bounded. Furthermore both of these operators do possess a non-trivial essential spectrum.

If a numerical approach for computing the spectra for these operators should be used, one naturally needs to restrict oneself to a finite-dimensional subspace for solving the problem. See section 3.1.2 on page 32 in the next chapter for details. Because of prop. 2.48 our *approximations* to Δ and $\hat{\mathcal{H}}$ will be compact. As we just discussed these will therefore at most have the value zero in their continuous spectrum.

Ignoring this 0 for a moment, we can state that both the point spectrum as well as the continuous spectrum of the infinite-dimensional operator will be mapped to the discrete spectrum of the approximation. For approximations to the discrete spectrum this is not a big problem. As we go to infinite accuracy in our approximation, we will recover more and more digits of the discrete eigenvalues provided that our approximation is sensible. For those eigenvalues which are part of the essential spectrum, however, things are not so simple, because they might be surrounded by discrete approximations to the continuous spectrum. In general distinguishing between true eigenvalues and so-called **spurious eigenvalues** inside the approximation to the essential spectrum is difficult. See the discussion in remark 3.8 on page 35 for some further details.

It is therefore very important to know the spectral properties of the exact operator in order to understand which part of the spectrum one may obtain. Let us discuss in the following a few examples, which are important for our treatment of QM.

2.3.3 The Laplace operator

Let us first consider the d-dimensional analogue of the Laplace operator introduced in (2.15). In Cartesian coordinates it reads

$$\Delta = \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2}. \tag{2.35}$$

Since this operator is essentially a scaled form of the kinetic energy operator $\hat{\mathcal{T}}$ (see (2.17)), we expect it to be self-adjoint and have real eigenvalues.

As it turns out, however, the naive choice of taking the domain of the operator to be the full quantum-mechanical Hilbert space $D(\Delta) = L^2(\mathbb{R}^d, \mathbb{C})$ is not helpful as this operator cannot be made self-adjoint. Only upon using the Sobolev space domain $D(\Delta) = H^2(\mathbb{R}^d, \mathbb{C})$, we get a self-adjoint operator Δ. Its spectrum is $\sigma(\Delta) = \sigma_C(\Delta) = (-\infty, 0]$ [53, example 3.2.2]. In other words it is a semi-bounded operator with no eigenvalues and no discrete spectrum at all.

2.3.4 The Laplace-Beltrami operator on the unit sphere

In contrast to the previous section, let us now consider the Laplace operator on the surface of the unit sphere

$$\mathbb{S}^2 := \left\{ \underline{\boldsymbol{r}} \in \mathbb{R}^3 \,\middle|\, x^2 + y^2 + z^2 = 1 \right\}.$$

For this it is most convenient to consider the spherical coordinate system, i.e. instead of parametrising the vector $\underline{\boldsymbol{r}}$ as a Cartesian column vector $(x, y, z)^T$, we specify it as (r, θ, φ) with

$$r = \|\underline{\boldsymbol{r}}\| = \sqrt{x^2 + y^2 + z^2} \qquad \theta = \arccos \frac{z}{r} \qquad \varphi = \arctan \frac{y}{x}.$$

The condition for the unit sphere than reduces to $r \stackrel{!}{=} 1$.

Since the sphere has no longer a Euclidean geometry but a curved manifold the operator equivalent to (2.35) takes the deviating functional form

$$\Delta_{\mathbb{S}^2} u = \frac{1}{\sin\theta} \frac{\partial}{\partial\theta} \left(\sin\theta \frac{\partial u}{\partial\theta} \right) + \frac{1}{(\sin\theta)^2} \frac{\partial^2}{\partial\varphi^2} u \tag{2.36}$$

in spherical polar coordinates. (2.36) is sometimes called the **Laplace-Beltrami operator** as well.

By taking the domain $D(\Delta_{\mathbb{S}^2}) = H^2(\mathbb{S}^2)$ the operator $\Delta_{\mathbb{S}^2}$ is self-adjoint [43, p. 120]. Furthermore one can show that the spectrum is (apart from 0) fully discrete[20]. Therefore this can be explicitly calculated by solving the ansatz $\Delta_{\mathbb{S}^2} Y = \lambda Y$ for the eigenpairs (λ, Y). This results in the **spherical harmonics**

$$Y_l^m(\theta, \varphi) = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} \, P_l^m(\cos\theta) \, e^{im\varphi} \tag{2.37}$$

[20] This follows since the inverse $\Delta_{\mathbb{S}^2}^{-1}$ is compact [43, p. 44].

where P_l^m is the associated Legendre polynomial with orders l and m. The eigenvalue corresponding to $Y_l^m(\theta,\varphi)$ is $-l(l+1)$. Due to the restriction

$$-l \leq m \leq l$$

this eigenvalue is $(2l+1)$-fold degenerate. Our spherical harmonics obviously satisfy

$$-\Delta_{\mathbb{S}^2} Y_l^m(\theta,\varphi) = l(l+1)Y_l^m(\theta,\varphi). \tag{2.38}$$

For the next section let us briefly note, that the Laplace-Beltrami operator on the unit sphere and the Laplace operator in 3 dimensions, expressed in spherical polar coordinates, are related by

$$r^2\Delta = \frac{\partial}{\partial r}\left(r^2\frac{\partial}{\partial r}\right) + \Delta_{\mathbb{S}^2}. \tag{2.39}$$

This allows to show

$$-r^2\Delta\, Y_l^m(\theta,\varphi) = -\Delta_{\mathbb{S}^2}\, Y_l^m(\theta,\varphi) = l(l+1)\, Y_l^m(\theta,\varphi). \tag{2.40}$$

An important consequence of the discreteness of the spectrum of the Laplace-Beltrami operator is that the spherical harmonics form a complete basis for $H^2(\mathbb{S}^2)$.

2.3.5 The Schrödinger operator for a hydrogen-like atom

One might wonder if a pure Laplace operator as in section 2.3.3 on the preceding page only possesses an essential spectrum, how this develops if a potential is added, like the Z/r in the case of the hydrogen-like Schrödinger operator (2.17)

$$\hat{\mathcal{H}} = -\frac{1}{2}\Delta - \frac{Z}{r}.$$

As the Hilbert space for this operator we take the QM space $L^2(\mathbb{R}^3,\mathbb{C})$ and an appropriate domain to make it self-adjoint is $H^2(\mathbb{R}^3,\mathbb{C})$ [43, p. 38]. One can show [53, 54] that $\sigma_C(\hat{\mathcal{H}}) = [0,\infty)$ and all discrete eigenvalues from $\sigma_P(\hat{\mathcal{H}})$ are less than zero. Thus $\sigma_{\text{disc}} = \sigma_P$ and $\sigma_{\text{ess}} = \sigma_C$. The point spectrum of $\hat{\mathcal{H}}$ can be conveniently determined by solving the Schrödinger equation (2.12)

$$(\hat{\mathcal{H}} - E_\mu)\Psi_\mu = 0, \tag{2.41}$$

where $\Psi_\mu \in H^2(\mathbb{R}^3,\mathbb{C})$ and $E_\mu \in (-\infty,0)$. Without jumping ahead too far let us assume that the state Ψ_μ may be uniquely identified by three quantum numbers $\mu \equiv (n,l,m)$.

Using (2.39) we may write the Hamiltonian as

$$\hat{\mathcal{H}} = -\frac{1}{2r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial}{\partial r}\right) - \frac{1}{2r^2}\Delta_{\mathbb{S}^2} - \frac{Z}{r}. \tag{2.42}$$

A careful inspection of (2.42) in contrast with (2.38) suggests a product ansatz

$$\Psi_{nlm}(\underline{\boldsymbol{r}}) = R_{nl}(r)Y_l^m(\theta,\phi).$$

With (2.40) this yields the radial equation

$$\left(-\frac{1}{2r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial}{\partial r}\right) + \frac{l(l+1)}{2r^2} - \frac{Z}{r} - E_\mu\right) R_{nl}(r) = 0 \tag{2.43}$$

which has the solutions [42]

$$R_{nl}(r) = N_{nl}\left(\frac{2Zr}{n}\right)^{l}\exp\left(-\frac{Zr}{n}\right)\ {}_1F_1\left(l+1-n\middle|2l+2\middle|\frac{2Zr}{n}\right) \tag{2.44}$$

where ${}_1F_1(a|b|\zeta)$ is a **confluent hypergeometric function**, namely [29]

$${}_1F_1\left(a|b|\zeta\right) = \sum_{k=0}^{\infty}\frac{a^{\bar{k}}}{k!\,b^{\bar{k}}}\zeta^k = 1+\frac{a}{b}\zeta+\frac{a(a+1)}{2b(b+1)}\zeta^2+\cdots \tag{2.45}$$

with $a^{\bar{k}}$ being the rising factorial of a. The normalisation constant is

$$N_{nl} = \frac{2\left(\frac{Z}{n}\right)^{3/2}}{(2l+1)!}\sqrt{\frac{(l+n)!}{n(n-l-1)!}}$$

and the corresponding energy eigenvalues are

$$E_\mu = -\frac{Z^2}{2n^2}. \tag{2.46}$$

If one follows through the derivation properly, one notices that the quantum numbers n, l and m are integer and need to satisfy the following conditions:

$$n>0 \qquad\qquad 0\leq l<n \qquad\qquad -l\leq m\leq l \tag{2.47}$$

Furthermore since all involved equations are of Sturm-Liouville form, the set of all solutions

$$\{\Psi_{nlm}\}_{n,l,m \text{ satisfy (2.47)}}$$

forms the orthonormal basis for a Hilbert space we will denote as $\mathcal{H}_{\mathrm{H}}$.

We saw in examples 2.24 on page 19 and 2.21 on page 18 that $\exp(-r)\in H^2(\mathbb{R}^3,\mathbb{C})$, which implies

$$\Psi_{1s}(r,\theta,\phi) = \Psi_{100}(r,\theta,\phi) = \sqrt{\frac{Z^3}{\pi}}\exp(-Zr)\in H^2(\mathbb{R}^3,\mathbb{C}). \tag{2.48}$$

From the functional form of R_{nl} and Y_l^m it is clear, that all eigenstates Ψ_{nlm} are infinitely differentiable everywhere except at $r=0$. See [5] and references therein for details. The polynomial in r in front of the exponential factor of the radial part R_{nl} has exponents in r in the range $[l, n-1]$ such that the eigenstate with $l=n-1=0$, i.e. Ψ_{1s}, is the least smooth. This implies $\Psi_{nlm}\in H^2(\mathbb{R}^3,\mathbb{C})$ and thus $\mathcal{H}_{\mathrm{H}}$ is a (true) subspace[21] of $H^2(\mathbb{R}^3,\mathbb{C})$.

2.4 Takeaway

Many observations of the one-particle hydrogen-like Schrödinger operator $\hat{\mathcal{H}}$ of section 2.3.5 on the preceding page generalise to the more complicated many-body atomic and molecular Hamiltonians we will introduce in chapter 4 on page 43.

[21] This is a true subspace, i.e. non-identical to $H^2(\mathbb{R}^3,\mathbb{C})$, since the scattering states are not part of it.

Most importantly all these Hamiltonians are unbounded operators, which become self-adjoint by making the domain a subspace of the Sobolev spaces $H^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C})$. Their essential spectrum is non-trival, but luckily one can show [54–57] that

$$\forall \lambda \in \sigma_{\text{disc}}(\hat{\mathcal{H}}), \mu \in \sigma_{\text{ess}}(\hat{\mathcal{H}}) \quad \lambda < \mu,$$

i.e. that the discrete spectrum always is located below the essential spectrum.

In remark 2.50 on page 26 we discussed that the essential spectrum cannot be approximated reliably by a finite-dimensional Hilbert space with only compact operators. Our best ansatz is therefore to follow a numerical approach, which aims at the description of the low end of the spectrum. This thus avoids $\sigma_{\text{ess}}(\hat{\mathcal{H}})$ and allows to obtain reliable approximations to at least a few eigenpairs corresponding to the discrete spectrum $\sigma_{\text{disc}}(\hat{\mathcal{H}})$, i.e. bound states.

Even though this is a restriction, one gets by for many cases. The rationale for this are the laws of thermodynamics, which imply that a sensible quantum-mechanical description of a system typically only requires the lowest energy state, i.e. the **ground state**, and the next few **excited state**s following most closely in energy. Assuming this is the case, care only needs to be taken to choose a sensible approximation method and a large enough approximation space. Otherwise one cannot be sure whether the obtained eigenstates are approximations to true discrete states or spurious states originating from discretising scattering states of the continuum.

This assumption does, however, break down for a couple of cases, such as plasma states, strong field physics or similar. But even without extreme energies, the description of certain processes such as resonance decayss or Rydberg-like states requires the description of high-energy bound states, which can be embedded inside the continuum, i.e. where the corresponding eigenvalues are part of the essential spectrum. This makes a numerical modelling challenging. We will mostly ignore this aspect in this work.

Chapter 3

Numerical treatment of spectral problems

> *It is a well-known experience that the only truly enjoyable and profitable way of studying mathematics is the method of "filling in details" by one's own efforts.*
>
> — Cornelius Lanczos (1893–1974)

This chapter discusses numerical techniques and algorithms, which can be used for obtaining a few of the discrete eigenvalues and corresponding eigenstates of a self-adjoint operator. For simplifying the discussion we will restrict ourselves to the cases where the eigenvalues of interest are located at the lower end of the discrete spectrum and are well-separated from the essential part of the spectrum. Notice that this is not the case for all regimes of quantum chemistry or even electronic structure theory. See the discussion in section 2.4 on page 29 for examples, where this assumption is violated.

3.1 Projection methods for eigenproblems

Let $\hat{\mathcal{A}}$ be a self-adjoint, bounded below operator on a separable Hilbert space $\mathcal{H}$ with domain $D(\hat{\mathcal{A}})$. We already saw in remark 2.35 on page 22 that $\hat{\mathcal{A}}$ uniquely defines a sesquilinear form

$$a(u, v) = \left\langle u \middle| \hat{\mathcal{A}} v \right\rangle_{\mathcal{H}}$$

for $(u, v) \in D(\hat{\mathcal{A}}) \times D(\hat{\mathcal{A}})$.

3.1.1 Form domains of operators

Even though $\hat{\mathcal{A}}$ might only be self-adjoint on the domain $D(\hat{\mathcal{A}})$, the form $a(\,\cdot\,,\,\cdot\,)$ can often be defined sensibly on a larger domain $Q(\hat{\mathcal{A}})$, called the **form domain** of $\hat{\mathcal{A}}$. Its construction will be sketched in this section. For more details see [54, p. 77] or [58, p. 276].

Since $\hat{\mathcal{A}}$ is semi-bounded from below, one can define a scalar product

$$\langle u|v\rangle_{\hat{\mathcal{A}}} \equiv \left\langle u\middle|\hat{\mathcal{A}}v + (C+1)v\right\rangle_{\mathcal{H}} = a(u,v) + (C+1)\langle u|v\rangle_{\mathcal{H}},$$

for all $u, v \in D(\hat{\mathcal{A}})$. Here C is the constant of semi-boundedness of definition 2.32 on page 22. Clearly the associated norm $\|\cdot\|_{\hat{\mathcal{A}}}$ satisfies

$$\|u\|_{\hat{\mathcal{A}}} = \left\langle u\middle|\hat{\mathcal{A}}u\right\rangle + (C+1)\|u\|_{\mathcal{H}} \overset{\text{def.2.32}}{\geq} \|u\|_{\mathcal{H}}. \tag{3.1}$$

We now take the completion of $D(\hat{\mathcal{A}})$ under the norm $\|u\|_{\hat{\mathcal{A}}}$ and call it $Q(\hat{\mathcal{A}})$. (3.1) assures that all sequences, which are Cauchy in $Q(\hat{\mathcal{A}})$ with respect to $\|\cdot\|_{\hat{\mathcal{A}}}$ are Cauchy in $\mathcal{H}$ with respect to $\|\cdot\|_{\mathcal{H}}$ as well. One can show further [54] that such sequences have the same limit in $Q(\hat{\mathcal{A}})$ irrespective of the norm used.

This allows to uniquely extend $a(\,\cdot\,,\,\cdot\,)$ to $Q(\hat{\mathcal{A}}) \times Q(\hat{\mathcal{A}})$ by setting

$$a(u,v) := \langle u|v\rangle_{\hat{\mathcal{A}}} - (C+1)\langle u|v\rangle_{\mathcal{H}}.$$

Constructed as such $Q(\hat{\mathcal{A}})$ is the largest Hilbert space on which the form $a(\,\cdot\,,\,\cdot\,)$ is defined and continuous. The form domain satisfies

$$D(\hat{\mathcal{A}}) \subseteq Q(\hat{\mathcal{A}}) \subseteq \mathcal{H},$$

where the subspaces are dense in the respective larger space.

Example 3.1. For all cases we discussed in the previous chapter, that is the Laplace operator Δ and the hydrogenic Hamiltonian $-\frac{1}{2}\Delta - \frac{Z}{r}$, the form domain is $H^1(\mathbb{R}^3)$. This can be easily verified by constructing the expression for the form $a(\,\cdot\,,\,\cdot\,)$ and applying partial integration.

3.1.2 The Ritz-Galerkin projection

The defining property of any eigenpair $(\lambda_i, v_i) \in \sigma_P(\hat{\mathcal{A}}) \times D(\hat{\mathcal{A}})$ of the operator $\hat{\mathcal{A}}$ is of course the condition

$$\hat{\mathcal{A}}v_i = \lambda_i v_i. \tag{3.2}$$

By a simple projection onto an arbitrary test function u, one can show that any such eigenpair satisfies

$$\forall u \in \mathcal{H}: \quad a(u, v_i) = \lambda_i \langle u|v_i\rangle_{\mathcal{H}}. \tag{3.3}$$

as well, the so-called **weak formulation** of the eigenproblem. In contrast to this, (3.2) is sometimes referred to as the **strong formulation**. A consequence of the Lax-Milgram theorem [43, p. 23] and the semi-boundedness of $\hat{\mathcal{A}}$ is that a solution in the weak sense implies a solution in the strong sense as well, making both formulations equivalent.

This suggests the **Ritz-Galerkin projection**, where one attempts to find an approximate solution for (3.2) by considering (3.3) in a sequence of subspaces of $Q(\hat{\mathcal{A}})$.

Definition 3.2 (Ritz-Galerkin projection)**.** Let $\hat{\mathcal{A}}$ be a self-adjoint, bounded below operator with form domain $Q(\hat{\mathcal{A}})$ and associated sesquilinear form $a(\,\cdot\,,\,\cdot\,)$. Given a sequence $(S_n)_{n\in\mathbb{N}} \subset Q(\hat{\mathcal{A}})$ of finite-dimensional subspaces satisfying

$$\forall v \in Q(\hat{\mathcal{A}}) \quad \inf_{v^{(n)} \in S_n} \left\| v - v^{(n)} \right\|_{Q(\hat{\mathcal{A}})} \xrightarrow{n\to\infty} 0, \tag{3.4}$$

one may obtain a sequence of approximate eigenspectra $\sigma^{(n)}(\hat{\mathcal{A}})$ by solving — for each n — the variational problem

$$\left\{ \begin{array}{l} \text{Search } (\lambda_i^{(n)}, v_i^{(n)}) \in \mathbb{R} \times S_n \text{ such that} \\ \qquad \forall u^{(n)} \in S_n : \quad a(u^{(n)}, v_i^{(n)}) = \lambda_i^{(n)} \left\langle u^{(n)} \middle| v_i^{(n)} \right\rangle_{\mathcal{H}} \\ \qquad\qquad \left\| v_i^{(n)} \right\|_{\mathcal{H}} = 1 \end{array} \right\}. \tag{3.5}$$

For ease of our discussion let $\hat{\mathcal{A}}^{(n)}$ denote the self-adjoint operator on a particular S_n, which is defined by the variational problem (3.5), i.e. which satisfies

$$\forall (u^{(n)}, v^{(n)}) \in S_n \times S_n \quad \left\langle u^{(n)} \middle| \hat{\mathcal{A}}^{(n)} v^{(n)} \right\rangle_{\mathcal{H}} = a(u^{(n)}, v^{(n)}).$$

Since S_n is finite-dimensional, $\hat{\mathcal{A}}^{(n)}$ is compact[1] and thus it will have a discrete spectrum $\sigma(\hat{\mathcal{A}}^{(n)})$. By definition $\sigma^{(n)}(\hat{\mathcal{A}}) = \sigma(\hat{\mathcal{A}}^{(n)})$.

Our hope is now to construct such a sequence (S_n) of subspaces, that $\sigma(\hat{\mathcal{A}}^{(n)})$ converges to $\sigma(\hat{\mathcal{A}})$. Unfortunately this is *not* possible in general, see [43] for details. What can be achieved, however, is a method to obtain sensible approximations for the lower end of the spectrum, especially all discrete eigenvalues below the essential spectrum.

Let us first state the theoretical basis in the form of the celebrated **min-max theorem** [43, p. 146]. In our discussion here, we follow the usual convention, where the eigenvalues in the discrete spectrum are indexed[2] in increasing order, i.e.

$$\lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \cdots .$$

Theorem 3.3 (Courant-Fischer min-max theorem)**.** *Let $\hat{\mathcal{A}}$ be a self-adjoint operator on $\mathcal{H}$, which is bounded below with form domain $Q(\hat{\mathcal{A}})$ and associated sesquilinear form $a(\,\cdot\,,\,\cdot\,)$. For each $0 < n \in \mathbb{N}$, we define*

$$\lambda_n(\hat{\mathcal{A}}) := \inf_{W \in \mathbb{S}_n} \sup_{u \in W \setminus \{0\}} \frac{a(u,u)}{\|u\|_{\mathcal{H}}^2} \tag{3.6}$$

where $\mathbb{S}_n$ is the set of all n-dimensional subspaces of $Q(\hat{\mathcal{A}})$. Then

- *if $\hat{\mathcal{A}}$ has at least n eigenvalues lower than $\inf \sigma_{ess}(\hat{\mathcal{A}})$ (counting multiplicities the appropriate number of times), then $\lambda_n(\hat{\mathcal{A}})$ is the n-th eigenvalue of the discrete spectrum of $\hat{\mathcal{A}}$,*
- *otherwise, $\lambda_n(\hat{\mathcal{A}}) = \inf \sigma_{ess}(\hat{\mathcal{A}})$.*

Combining this with the Ritz-Galerkin projection of definition 3.2 yields:

Corollary 3.4. *Let $\hat{\mathcal{A}}$ be a bounded below, self-adjoint operator on $\mathcal{H}$ and let (S_n) be a sequence of subspaces of the form domain, which satisfy condition* (3.4)*. If we denote with $\hat{\mathcal{A}}^{(n)}$ the approximations to $\hat{\mathcal{A}}$ according to the variational Ritz-Galerkin ansatz* (3.5)*, then*

$$\forall 0 < i \in \mathbb{N} \quad \lambda_i^{(n)} := \lambda_i(\hat{\mathcal{A}}^{(n)}) \xrightarrow{n \to \infty} \lambda_i(\hat{\mathcal{A}}),$$

where the convergence is always from above.

[1] See proposition 2.48 on page 26.

[2] This can always be done, since by definition the discrete spectrum is *always* countable.

As discussed in section 2.4 on page 29 those operators, which will be considered in this thesis, always possess a discrete spectrum located below the essential spectrum. Furthermore we will always be interested in those bound states located at the lower end of the discrete spectrum. With the aforementioned results we can sketch an approximation method for our setting.

Remark 3.5 (Approximation of the bottom of the discrete spectrum)**.** Let $\hat{\mathcal{A}}$ be a self-adjoint, bounded below operator and let us assume that we seek approximations for a few discrete eigenvalues, which are all located at the bottom of the spectrum $\sigma(\hat{\mathcal{A}})$ and well below the essential spectrum.

Let $U \subset Q(\hat{\mathcal{A}})$ be a dense subspace. We can span a sequence of subspaces $(S_{N_{\text{bas}}}) \subset Q(\hat{\mathcal{A}})$ by selecting larger and larger[3] sets $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ of $N_{\text{bas}} = |\mathcal{I}_{\text{bas}}|$ orthonormal basis functions $\varphi_\mu \in U$ as the bases. Since U is a dense subspace of the separable Hilbert space $\mathcal{H}$, it is separable as well. Therefore we know that in the limit of $N_{\text{bas}} \to \infty$, $\text{span}\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ will tend towards U. Thus $\text{span}\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ eventually allows to construct Cauchy sequences, which approximate each $v \in Q(\hat{\mathcal{A}})$ up to arbitrary accuracy. In other words the sequence $(S_{N_{\text{bas}}})$ with N_{bas} increasing satisfies condition (3.4).

Because of corollary 3.4 we can thus get arbitrarily accurate approximations to our eigenvalues of interest by solving the variational problem (3.5) in subspaces spanned by larger and larger basis sets $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}} \subset U$. This results in more and more accurate approximations of the corresponding bound eigenstates as well.

Remark 3.6 (Discrete formulation of (3.5))**.** We are again in the setting of remark 3.5. If $S_{N_{\text{bas}}} = \text{span}\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$, we can expand

$$v_i^{(n)} = \sum_{\nu\in\mathcal{I}_{\text{bas}}} C_{\nu i}^{(n)} \varphi_\nu \qquad \text{with} \quad C_{\nu i}^{(n)} \equiv \left\langle \varphi_\nu \middle| v_i^{(n)} \right\rangle$$

and thus reformulate (3.5) to become

$$\left\{ \begin{array}{l} \text{Search } \lambda_i^{(n)} \text{ and } C_{\nu i}^{(n)} \text{ such that} \\ \forall \varphi_\mu \in S_{N_{\text{bas}}} : \quad \displaystyle\sum_{\nu\in\mathcal{I}_{\text{bas}}} C_{\nu i}^{(n)}\, a(\varphi_\mu, \varphi_\nu) = \lambda_i^{(n)} \sum_{\nu\in\mathcal{I}_{\text{bas}}} C_{\nu i}^{(n)}\, \langle \varphi_\mu | \varphi_\nu \rangle \\ \qquad\qquad\qquad\qquad 1 = \displaystyle\sum_{\nu\in\mathcal{I}_{\text{bas}}} \sum_{\mu\in\mathcal{I}_{\text{bas}}} \left(C_{\mu i}^{(n)} \right)^* \langle \varphi_\mu | \varphi_\nu \rangle\, C_{\nu i}^{(n)} \end{array} \right\}. \tag{3.7}$$

Introducing the matrix $\mathbf{A}^{(n)} \in \mathbb{C}^{N_{\text{bas}} \times N_{\text{bas}}}$ and the vectors $\underline{\boldsymbol{c}}_i^{(n)} \in \mathbb{C}^{N_{\text{bas}}}$ with elements

$$A_{\mu\nu}^{(n)} = a(\varphi_\mu, \varphi_\nu) \qquad\qquad \left(c_i^{(n)}\right)_\mu = \left\langle \varphi_\mu \middle| v_i^{(n)} \right\rangle = C_{\mu i}^{(n)} \tag{3.8}$$

we can write (3.7) as the matrix eigenvalue problem[4]

$$\left\{ \begin{array}{l} \text{Search } (\lambda_i^{(n)}, \underline{\boldsymbol{c}}_i^{(n)}) \in \mathbb{R} \times \mathbb{C}^{N_{\text{bas}}} \text{ such that} \\ \qquad\qquad \mathbf{A}^{(n)} \underline{\boldsymbol{c}}_i^{(n)} = \lambda_i^{(n)} \underline{\boldsymbol{c}}_i^{(n)} \\ \qquad\qquad \left\| \underline{\boldsymbol{c}}_i^{(n)} \right\|_{\mathbb{C}^{N_{\text{bas}}}} = 1 \end{array} \right\}. \tag{3.9}$$

[3] Until the dimensionality of U is reached in case it is finite-dimensional.

[4] Recall that the functions $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ were chosen orthonormal.

In this formulation the eigenpairs $(\lambda_i^{(n)}, \underline{c}_i^{(n)})$ can be determined by standard diagonalisation schemes like the ones we will discuss in section 3.2 on the next page below.

Remark 3.7 (Requirements regarding the basis function type)**.** In light of the numerical approach sketched in remarks 3.5 and 3.6 let us summarise the requirements towards the basis functions φ_μ for solving the discretised problem (3.9).

- The basis function type should admit to construct a dense subspace of $Q(\hat{\mathcal{A}})$ if an infinitely large basis set is chosen, since this is needed in order to satisfy (3.4). Some basis function types even admit to span $Q(\hat{\mathcal{A}})$ itself in the sense of a Hilbert space basis (i.e. a Schauder basis). We shall call these **complete**.
- It should be numerically feasible to solve (3.9). In other words *both* computing $\mathbf{A}$ and determining its eigenpairs should be viable.
- The convergence in corollary 3.4 should be fast and systematic. In other words the basis type should allow to construct a suitable basis set in case certain requirements regarding accuracy, computational demands, description of properties, ... should be met. Any prior knowledge about the physical problem or the properties of $\hat{\mathcal{A}}$ can ideally be incorporated in such a basis set choice.

See chapter 5 on page 85 for some basis function types, which are used in quantum chemistry, in the light of solving the Hartree-Fock problem.

Before we discuss some basic diagonalisation algorithms in the next section, let us conclude our discussion about the discretisation of eigenvalue problems with a word of warning about the essential spectrum.

Remark 3.8. Remark 2.50 on page 26 stated that it was difficult to obtain numerical approximations to the essential spectrum. The min-max theorem 3.3 provides some theoretical justification for this. In corollary 3.4 we saw, that all eigenvalues from a Ritz-Galerkin approximation of $\hat{\mathcal{A}}$ tend to $\lambda_i(\hat{\mathcal{A}})$ as the subspace size is increased. Unfortunately this value is equal to $\inf \sigma_{\text{ess}}(\hat{\mathcal{A}})$, the infimum of the essential spectrum, as soon as we exhausted the discrete spectrum. In other words the methods we developed in this section will only help to find the bottom end of the essential spectrum, but no further information about it at all.

Another consequence of corollary 3.4 is that only a part of the eigenpairs obtained by diagonalising the matrix $\mathbf{A}^{(n)}$ of (3.9) can be trusted to carry any meaning regarding the spectrum of the exact physical operator $\hat{\mathcal{A}}$. This is because the larger eigenvalues $\lambda_i^{(n)}$ of $\mathbf{A}^{(n)}$ will only provide an *artificial* discretisation of the essential spectrum: Their values will all tend to $\inf \sigma_{\text{ess}}(\hat{\mathcal{A}})$ as the basis set is increased. Since the convergence to the bottom of the essential spectrum as well as the discrete eigenvalues is always from above, one sometimes has trouble judging whether an eigenpair of $\mathbf{A}^{(n)}$ is a true discrete eigenpair of the operator or already part of the essential spectrum. In either case the bottom end of the spectrum of $\mathbf{A}$ will carry meaning about the underlying operator $\hat{\mathcal{A}}$ if the basis set is large enough and satisfies remark 3.7.

For practical quantum-chemical applications such as the modelling of resonance processes, bound states embedded inside the continuous spectrum are required. For this reason approaches such as the so-called **stabilisation method** [59–61] have been developed, which can be used to probe bound states inside the continuum region. To the best of my knowledge a rigorous mathematical treatment, which assures that such

methods do not miss states or converge to spurious, non-physical states has not been developed yet, however.

3.2 Diagonalisation algorithms

This section discusses the key ideas of a few algorithms for obtaining approximations to the eigenpairs of a matrix $\mathbf{A}$. Whilst the regime of quantum mechanics is a complex-valued Hilbert space, in this thesis we will only consider combinations of operators and discretisation bases $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$, which have the property that

$$\forall \mu, \nu \in \mathcal{I}_{\text{bas}} : \quad a(\varphi_\mu, \varphi_\nu) \in \mathbb{R}.$$

As a result all matrices in (3.9) will be real and symmetric. In this section we will therefore only consider eigenproblems of the type

$$\mathbf{A}\underline{\boldsymbol{u}}_i = \lambda_i \underline{\boldsymbol{u}}_i,$$

where $\mathbf{A} \in \mathbb{R}^{N_{\text{bas}} \times N_{\text{bas}}}$, $\lambda_i \in \mathbb{R}$ and $\underline{\boldsymbol{u}}_i \in \mathbb{R}^{N_{\text{bas}}}$.

3.2.1 Direct methods

One approach to solve such eigenproblems are so-called **direct diagonalisation methods**. These methods directly attempt to perform a transformation

$$\mathbf{O}^{\mathrm{T}}\mathbf{A}\mathbf{O} = \mathbf{L} = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{N_{\text{bas}}}),$$

where $\mathbf{O} \in \mathbb{R}^{N_{\text{bas}} \times N_{\text{bas}}}$ is an orthogonal matrix. Typically this is performed in steps by inspecting the elements of $\mathbf{A}$ and gradually building both $\mathbf{O}$ as well as the matrix-matrix product $\mathbf{O}^{\mathrm{T}}\mathbf{A}\mathbf{O}$ using techniques such as Householder reflectors [62] or Givens rotation [62]. In either case this requires random access into the memory of $\mathbf{A}$. This is one of the reasons why direct methods are typically only suitable for either small matrices, where N_{bas} is at most on the order of 1000, or matrices with special structure, like being tridiagonal or banded. Important dense diagonalisation methods include QR factorisation [62, 63] as well as Cuppen's divide and conquer algorithm [62, 63]. They generally yield *all* eigenvalues of a matrix at once and little or no extra work is required to additionally obtain all eigenvectors as well.

3.2.2 Iterative diagonalisation methods

Unlike direct methods, which directly access the matrix elements, **iterative diagonalisation methods** only probe the matrix $\mathbf{A}$ indirectly, namely by iteratively gathering more and more information about the eigenpairs of interest. The way this is done in practice is to repetitively form the matrix-vector product

$$\underline{\boldsymbol{y}} = \mathbf{A}\underline{\boldsymbol{x}}$$

of the problem matrix $\mathbf{A}$ with suitably constructed trial vectors $\underline{\boldsymbol{x}}$. The resulting vector $\underline{\boldsymbol{y}}$ is then used to improve upon the approximation for the eigenpairs as well as to build the $\underline{\boldsymbol{x}}$ for the next step. This implies that random access into $\mathbf{A}$ is not required for such methods and thus specific storage schemes or well-crafted algorithms going beyond a typical matrix-vector product can be employed for forming $\underline{\boldsymbol{y}}$. The latter aspect is most

important for the contraction-based methods, which will be developed in chapters 5 on page 85 and 6 on page 141 of this thesis.

Iterative methods are typically not ideal for computing many or all eigenpairs of a matrix $\mathbf{A}$, which is in contrast to direct methods. They do perform, however, much better than direct methods if only few eigenpairs are desired and it is well-known *where* in the spectrum they are located. Examples for cases where iterative methods tend to work well is if one requires some of the largest eigenvalues of $\mathbf{A}$ or some of those which are closest to an estimated value σ. Some important iterative methods are sketched in the following sections.

3.2.3 The power method

The simplest iterative approach to obtain a single extremal eigenvalue from a particular matrix $\mathbf{A}$ is the power method. Starting from a random initial vector $\underline{\boldsymbol{v}}^{(0)} \in \mathbb{R}^{N_{\text{bas}}}$, the algorithm only consists of applying the matrix $\mathbf{A}$ repetitively to the current vector, i.e.

$$\begin{aligned}
\underline{\boldsymbol{v}}^{(1)} &= \mathbf{A}\underline{\boldsymbol{v}}^{(0)}, \\
\underline{\boldsymbol{v}}^{(2)} &= \mathbf{A}\underline{\boldsymbol{v}}^{(1)} = \mathbf{A}^2\underline{\boldsymbol{v}}^{(0)}, \\
\underline{\boldsymbol{v}}^{(3)} &= \mathbf{A}\underline{\boldsymbol{v}}^{(2)} = \mathbf{A}^3\underline{\boldsymbol{v}}^{(0)}, \\
&\vdots \\
\underline{\boldsymbol{v}}^{(j)} &= \mathbf{A}\underline{\boldsymbol{v}}^{(j-1)} = \mathbf{A}^j\underline{\boldsymbol{v}}^{(0)}.
\end{aligned} \tag{3.10}$$

In each step we may compute an estimate $\theta^{(j)}$ for the eigenvalue by the expression

$$\theta^{(j)} = \rho_R\left(\underline{\boldsymbol{v}}^{(j)}\right) \equiv \frac{\underline{\boldsymbol{v}}^T\mathbf{A}\underline{\boldsymbol{v}}}{\underline{\boldsymbol{v}}^T\underline{\boldsymbol{v}}}, \tag{3.11}$$

where ρ_R is the **Rayleigh quotient**, the discretised version of (3.6). In well-behaved cases this algorithm will find an approximation for the largest eigenvalue in $\theta^{(i)}$ and an approximation for the corresponding eigenvector as

$$\frac{\underline{\boldsymbol{v}}^{(i)}}{\left\|\underline{\boldsymbol{v}}^{(i)}\right\|_2}.$$

To understand this, let us write $\underline{\boldsymbol{v}}^{(0)}$ as an expansion in the exact eigenvectors $\underline{\boldsymbol{u}}_1, \underline{\boldsymbol{u}}_2, \ldots, \underline{\boldsymbol{u}}_{N_{\text{bas}}}$:

$$\underline{\boldsymbol{v}}^{(0)} = \sum_{i=1}^{N_{\text{bas}}} \alpha_i\underline{\boldsymbol{u}}_i = \alpha_{N_{\text{bas}}}\underline{\boldsymbol{u}}_{N_{\text{bas}}} + \sum_{i=1}^{N_{\text{bas}}-1} \alpha_i\underline{\boldsymbol{u}}_i \tag{3.12}$$

Without loss of generality[5] we can normalise $\underline{\boldsymbol{v}}^{(0)}$ such that $\alpha_{N_{\text{bas}}} = 1$. Keeping this in mind, the application of $\mathbf{A}$ to (3.12) results in

$$\mathbf{A}\underline{\boldsymbol{v}}^{(0)} = \lambda_{N_{\text{bas}}}\left(\underline{\boldsymbol{u}}_{N_{\text{bas}}} + \sum_{i=1}^{N_{\text{bas}}} \frac{\lambda_k}{\lambda_{N_{\text{bas}}}}\alpha_i\underline{\boldsymbol{u}}_i\right).$$

[5]The case $\alpha_{N_{\text{bas}}} = 0$ is handled by the limited precision floating point arithmetic. After a single application of $\mathbf{A}$, this is cured and we are back to the case we consider here.

After the j-th step and subsequent normalisation we hence get

$$\frac{\underline{\boldsymbol{v}}^{(j)}}{\left\|\underline{\boldsymbol{v}}^{(j)}\right\|_2} = \underline{\boldsymbol{u}}_{N_{\text{bas}}} + \mathcal{O}\left(\left(\frac{\lambda_{N_{\text{bas}}-1}}{\lambda_{N_{\text{bas}}}}\right)^j\right).$$

Provided that $|\lambda_{N_{\text{bas}}-1}| \neq |\lambda_{N_{\text{bas}}}|$, i.e. that the largest eigenvalue (by magnitude) is single, the iterate $\underline{\boldsymbol{v}}^{(j)}$ therefore converges linearly against the eigenvector corresponding to this largest eigenvalue $\lambda_{N_{\text{bas}}}$. Similarly $\theta^{(j)}$ converges against $\lambda_{N_{\text{bas}}}$ in this case.

3.2.4 Spectral transformations

With the power method at hand to obtain the largest eigenvalue, the question is now, how one could generalise this approach for getting the smallest eigenvalue or even one directly from the middle of the spectrum. This is the purpose of so-called **spectral transformations**.

Proposition 3.9. *Given a symmetric matrix $\mathbf{A} \in \mathbb{R}^{N_{bas}\times N_{bas}}$, the following holds for each eigenpair $(\lambda_i, \underline{\boldsymbol{u}}_i) \in \mathbb{R} \times \mathbb{R}^{N_{bas}}$:*

(a) If $\mathbf{A}$ is invertible, $\underline{\boldsymbol{u}}_i$ is an eigenvector of $\mathbf{A}^{-1}$ with eigenvalue $1/\lambda_i$.

(b) For every $\sigma \in \mathbb{R}$, $\underline{\boldsymbol{u}}_i$ is an eigenvector of the matrix $\mathbf{A} - \sigma\mathbf{I}_{N_{bas}}$ with eigenvalue $\lambda_i - \sigma$.

(c) If $\sigma \in \mathbb{R}$ is chosen such that $\mathbf{A} - \sigma\mathbf{I}_{N_{bas}}$ is invertible, then $\underline{\boldsymbol{u}}_i$ is an eigenvector of $(\mathbf{A} - \sigma\mathbf{I}_{N_{bas}})^{-1}$ with eigenvalue $1/(\lambda_i - \sigma)$.

Proof. All can be shown in a single line:

(a) By definition $\mathbf{I}_{N_{\text{bas}}} = \mathbf{A}^{-1}\mathbf{A}$ and thus we have

$$\frac{1}{\lambda_i}\underline{\boldsymbol{u}}_i = \frac{1}{\lambda_i}\mathbf{I}_{N_{\text{bas}}}\underline{\boldsymbol{u}}_i = \frac{1}{\lambda_i}\mathbf{A}^{-1}\mathbf{A}\underline{\boldsymbol{u}}_i = \frac{1}{\lambda_i}\mathbf{A}^{-1}\lambda_i\underline{\boldsymbol{u}}_i = \mathbf{A}^{-1}\underline{\boldsymbol{u}}_i.$$

(b) Direct calculation shows

$$(\mathbf{A} - \sigma\mathbf{I}_{N_{\text{bas}}})\,\underline{\boldsymbol{u}}_i = \mathbf{A}\underline{\boldsymbol{u}}_i - \sigma\underline{\boldsymbol{u}}_i = \lambda_i\underline{\boldsymbol{u}}_i - \sigma\underline{\boldsymbol{u}}_i = (\lambda_i - \sigma)\,\underline{\boldsymbol{u}}_i.$$

(c) Follows from (a) and (b).

□

Proposition 3.9 provides us with a toolbox for changing the spectrum of a matrix in a desired way without changing its eigenvectors. For example if we are interested in obtaining the lowest eigenvalue of a matrix $\mathbf{A}$ using the power method, we essentially only need to apply the scheme (3.10) to the inverse[6] $\mathbf{A}^{-1}$ instead of $\mathbf{A}$. Since the largest eigenvector of $\mathbf{A}^{-1}$ will be the smallest of $\mathbf{A}$, this yields the required result. Similarly, by proposition 3.9(c), we can tune the power method into a particular eigenvalue of interest by guessing an appropriate shift σ. Such spectral transformations are not restricted to the Power method, since the equivalent effect can be achieved for other iterative methods by passing them an appropriate matrix.

[6] Usually the inverse is computed iteratively as well, see discussion in section 3.2.7.

3.2.5 Krylov subspace methods

Applying the power method effectively amounts to generating a sequence of vectors

$$\underline{\boldsymbol{v}},\ \mathbf{A}\underline{\boldsymbol{v}},\ \mathbf{A}^2\underline{\boldsymbol{v}},\ \ldots, \tag{3.13}$$

starting from an initial guess $\underline{\boldsymbol{v}}$. Given that the eigenvalue of largest magnitude of $\mathbf{A}$ is not degenerate, the above sequence will approach the eigenvector corresponding to this extremal eigenvalue (see discussion in section 3.2.3). In each iteration the power method does, however, only keep one of the vectors in (3.13) and throws away all information encoded in the history of the iteration. An alternative approach which avoids doing so, is to explicitly keep all vectors in (3.13). This leads to the construction of a Krylov subspace [62]

$$\mathcal{K}_j = \left\{\underline{\boldsymbol{v}},\ \mathbf{A}\underline{\boldsymbol{v}},\ \mathbf{A}^2\underline{\boldsymbol{v}}, \ldots, \mathbf{A}^j\underline{\boldsymbol{v}}\right\}. \tag{3.14}$$

A large number of iterative methods both for solving eigenproblems as well as linear problems can be boiled down to an iterative construction of such a Krylov subspace. Once or while it is found the original problem matrix $\mathbf{A}$ is projected onto this subspace, yielding $\tilde{\mathbf{A}} \in \mathbb{R}^{j\times j}$.

A key step to exploit the notion of Krylov subspaces is the construction of an orthogonal basis for $\mathcal{K}_j$. The Arnoldi algorithm [64] was devised to achieve this in a very efficient manner. It exploits the fact that each vector (3.14) is related to its predecessor by an application of the problem matrix $\mathbf{A}$ to produce a simple recursion scheme minimising the work needed in each step. Alongside with the construction of the basis, the Arnoldi algorithm *at the same time* constructs $\tilde{\mathbf{A}}$, the projection of $\mathbf{A}$ into the Krylov subspace. Since $\tilde{\mathbf{A}}$ is both smaller than $\mathbf{A}$ and has a much simpler form[7] it can be diagonalised by a shifted QR factorisation, a direct method. This leads to the Arnoldi method for diagonalising non-symmetric real matrices, where one first uses the Arnoldi procedure to construct a sufficiently good Krylov subspace[8], followed by a dense diagonalisation of the subspace matrix to yield estimates for the eigenpairs.

A modification of the Arnoldi method for symmetric matrices $\mathbf{A}$ is the Lanczos method [65], which implicitly exploits the fact that the subspace matrix has to be tridiagonal[9] already while constructing the Krylov subspace basis.

Even though the basic idea of Arnoldi and Lanczos are comparatively easy, the implementation is still involved due to a range of subtleties. For example one can show [62] that the unmodified Lanczos procedure leads to an Arnoldi basis of poor numerical quality with potentially linearly dependent vectors roughly speaking exactly when achieving convergence for an eigenpair. Similarly both Arnoldi and Lanczos tend to have difficulties when reporting multiplicities. So if $\mathbf{A}$ has a triply degenerate eigenvalue λ_i it can happen that these algorithms only find it twice, even though the eigenspaces for λ_{i-1} and λ_{i+1}, i.e. of the next smallest and next largest eigenvalue, are completely described. For such issues a large range of remedies have been proposed over the years [62, 63], stressing the importance of Arnoldi methods in numerical linear algebra. Examples include block modifications — where not a single vector, but a collection

[7] It is a so-called upper Hessenberg matrix, i.e. only the upper triangle and a single subdiagonal in the lower triangle are non-zero.

[8] Some error estimates exist to judge this without performing the next step of actually diagonalising the upper Hessenberg matrix.

[9] Since $\mathbf{A}$ is symmetric, so is the subspace matrix and a symmetric upper Hessenberg matrix is tridiagonal.

of vectors is iterated in the Arnoldi procedure — or concepts such as implicit restart, deflation or locking.

3.2.6 The Jacobi-Davidson algorithm

Related to the Krylov subspace methods sketched above, the Jacobi-Davidson approach finds approximations to the eigenpairs of (3.9) by constructing suitable small subspaces and solving the projected problem with dense methods. The algorithm used for constructing the subspace is, however, somewhat different[10]. Let us sketch the procedure for a matrix $\mathbf{A} \in \mathbb{R}^{N_{\text{bas}} \times N_{\text{bas}}}$, where an approximation to the unknown, exact eigenpair $(\lambda_i, \underline{\boldsymbol{u}}_i)$ is desired. Following Davidson [66] we define the residual

$$\underline{\boldsymbol{r}}^{(j)} = \mathbf{A}\underline{\boldsymbol{v}}^{(j)} - \lambda_i \underline{\boldsymbol{v}}^{(j)} \tag{3.15}$$

of our current approximation $\underline{\boldsymbol{v}}^{(j)}$ to the eigenvector $\underline{\boldsymbol{u}}_i$. In order to correct, we employ the **Jacobi orthogonal component correction**, i.e. we want to add a vector $\underline{\boldsymbol{t}}^{(j)} \perp \underline{\boldsymbol{v}}^{(j)}$ to our subspace, such that

$$\mathbf{A}\left(\underline{\boldsymbol{v}}^{(j)} + \underline{\boldsymbol{t}}^{(j)}\right) = \lambda_i \left(\underline{\boldsymbol{v}}^{(j)} + \underline{\boldsymbol{t}}^{(j)}\right).$$

In other words, we attempt to find the vector missing from the subspace, such that it is able to span the exact solution, which implies that it would be able to find it the next time we solve the projected problem in the subspace.

Since λ_i is in general not known at the j-th step of the algorithm, $\underline{\boldsymbol{t}}^{(j)}$ cannot be found exactly in practice. Instead one employs the value returned by the Rayleigh quotient (3.11) instead of λ_i to make progress. Incorporating the condition $\underline{\boldsymbol{t}}^{(j)} \perp \underline{\boldsymbol{v}}^{(j)}$ leads to the **correction equation**

$$\left(\mathbf{I}_{N_{\text{bas}}} - \underline{\boldsymbol{v}}^{(j)}\underline{\boldsymbol{v}}^{(j)\,*}\right)\left(\mathbf{A}\underline{\boldsymbol{v}}^{(j)} - \theta^{(j)}\mathbf{I}_{N_{\text{bas}}}\right)\left(\mathbf{I}_{N_{\text{bas}}} - \underline{\boldsymbol{v}}^{(j)}\underline{\boldsymbol{v}}^{(j)\,*}\right)\underline{\boldsymbol{t}}^{(j)} = -\underline{\boldsymbol{r}}^{(j)}. \tag{3.16}$$

Since a vector $\underline{\boldsymbol{t}}^{(j)}$ is required in each iteration, it needs to be solved many times. Fortunately, it does, however, not need to be solved exactly. In practice, one therefore employs preconditioning techniques [62, 63, 67, 68] to speed up the performance of the iterative procedures needed to solve (3.16). An alternative is to avoid using the *exact* matrix $\mathbf{A}$ in favour of an approximation, which makes solving (3.16) easier. A combination of both is possible as well.

In the original paper Davidson [66] assumed $\mathbf{A}$ to be diagonal-dominant and thus only used the diagonal

$$\mathbf{D}_A = \operatorname{diag}\left(A_{11}A_{22}\ldots A_{N_{\text{bas}},N_{\text{bas}}}\right)$$

instead of the full $\mathbf{A}$ for the correction in (3.16). This leads to the identification

$$\underline{\boldsymbol{t}}^{(j)} = \left(\mathbf{D}_A - \theta^{(j)}\mathbf{I}_{N_{\text{bas}}}\right)^{-1}\underline{\boldsymbol{r}}^{(j)},$$

which is trivially computed elementwise as

$$\left(t^{(j)}\right)_i = \frac{\left(r^{(j)}\right)_i}{A_{ii} - \theta^{(j)}}.$$

This is the basis of many diagonalisation routines employed in quantum-chemistry packages nowadays.

[10] One should mention that similarities to the Lanczos procedure can be found, however. See [62] for details.

3.2.7 Generalised eigenvalue problems

Many eigenproblems occurring in quantum chemistry are in fact not of the form (3.9), but are so-called **generalised eigenproblems**,

$$\mathbf{A}\underline{\boldsymbol{u}}_i = \lambda_i \mathbf{S}\underline{\boldsymbol{u}}_i \tag{3.17}$$

where the right-hand side contains a real, positive-definite matrix $\mathbf{S} \in \mathbb{R}^{N_{\text{bas}} \times N_{\text{bas}}}$ as well. These typically arise because the basis set $\{\varphi_\mu\}_{\mu \in \mathcal{I}_{\text{bas}}}$ used for the discretisation is not orthogonal. For the typical basis sets employed to numerically solve the Hartree-Fock problem, one of the central aspects of this thesis, this is the usual case (see section 5.3 on page 91).

One way to deal with (3.17) is to reduce it to a normal eigenproblem by formally inverting $\mathbf{S}$ and multiplying from the right-hand side. This leads to

$$\left(\mathbf{S}^{-1}\mathbf{A}\right)\underline{\boldsymbol{u}}_i = \lambda_i \underline{\boldsymbol{u}}_i,$$

a normal eigenproblem with the problem matrix $\mathbf{S}^{-1}\mathbf{A}$. In iterative methods this amounts to replacing all occurrences of the matrix-vector product $\mathbf{A}\underline{\boldsymbol{x}}$ by the expression

$$\underline{\boldsymbol{y}} = \mathbf{S}^{-1}\mathbf{A}\underline{\boldsymbol{x}}.$$

In this expression the vector $\underline{\boldsymbol{y}}$ can be computed by solving the linear system

$$\mathbf{S}\underline{\boldsymbol{y}} = \mathbf{A}\underline{\boldsymbol{x}}$$

using an inner preconditioned iterative method. Whilst this would work, this approach is hardly ever followed in practice. The reason is that even for a real symmetric, positive-definite $\mathbf{S}$ and a real symmetric $\mathbf{A}$, the matrix $\mathbf{S}^{-1}\mathbf{A}$ might not be symmetric, which would imply that less advantageous solution algorithms need to be employed.

An alternative approach to avoid this is to try to modify the iterative procedures towards supporting the generalised eigenproblems straight away. By properly following the derivations, one finds that appropriate formulations of the algorithms in the setting of generalised eigenproblems can be achieved by replacing the explicit or implicit occurrences of the orthonormality condition

$$\underline{\boldsymbol{u}}_i^{\mathrm{T}}\underline{\boldsymbol{u}}_j = \delta_{ij}$$

by

$$\underline{\boldsymbol{u}}_i^{\mathrm{T}}\underline{\boldsymbol{u}}_j = S_{ij}.$$

In other words, only the way the orthonormalisation of the subspace vectors is performed as well as some expressions in which the identity matrix occurs, like in (3.16), need to be changed.

Yet another option is to orthogonalise the basis before performing the discretisation and thus avoid the appearance of the generalised eigenproblem all together.

Chapter 4

Solving the many-body electronic Schrödinger equation

> *The word "reality" is also a word, a word which we must learn to use correctly.*
>
> — Niels Bohr (1885–1962)

> *I am convinced that despite his slightly positivist language, Bohr believes as much as we do in the reality of phenomena of which he speaks, and then the difference between the views of Bohr and mine is more a difference of language than a difference of content.*
>
> — Vladimir Fock (1898–1974)

This chapter is concerned with the generalisation of the one-electron hydrogen-like Schrödinger Hamiltonian (2.17)

$$\hat{\mathcal{H}}_H = -\frac{1}{2}\Delta - \frac{Z}{r},$$

which we discussed in section 2.3.5 on page 28, towards the many-body problems of quantum chemistry.

Even though the spectral properties are very similar to the hydrogen-like case, solving the associated time-independent Schrödinger equation (2.12) analytically for any but the most trivial problems is impossible. Most of this chapter will therefore be devoted to discussing approximations to the exact TISE as well as numerical approaches for solving such approximations in practice.

4.1 Many-body Schrödinger equation

Let us consider a chemical system consisting of M nuclei and N_{elec} electrons. We take the nuclei to be located at mass-scaled[1] coordinates $\{\underline{\boldsymbol{R}}_A\}_{A=1,2,\ldots,M} \subset \mathbb{R}^3$ with corresponding charges $\{Z_A\}_{A=1,2,\ldots,M}$. The electron positions are denoted by the (Cartesian) coordinates $\{\underline{\boldsymbol{r}}_i\}_{i=1,2,\ldots,N_{\text{elec}}} \subset \mathbb{R}^3$. Following the correspondence to classical mechanics (see section 2.1.1 on page 8) we can construct the many-body Hamiltonian on the Hilbert space $L^2(\mathbb{R}^L, \mathbb{C})$ with dimensionality $L = 3M + 3N_{\text{elec}}$ as

$$\hat{\mathcal{H}}^{\text{MB}} = \hat{\mathcal{T}}_e + \hat{\mathcal{T}}_n + \hat{\mathcal{V}}_{nn} + \hat{\mathcal{V}}_{ne} + \hat{\mathcal{V}}_{ee}. \tag{4.1}$$

In this expression we introduced the nuclear-nuclear, electronic-electronic and nuclear-electronic Coulombic interaction potentials

$$\hat{\mathcal{V}}_{nn} = \sum_{A=1}^{M} \sum_{B=A+1}^{M} \frac{Z_A Z_B}{\|\underline{\boldsymbol{R}}_A - \underline{\boldsymbol{R}}_B\|_2}, \qquad \hat{\mathcal{V}}_{ee} = \sum_{i=1}^{N_{\text{elec}}} \sum_{j=i+1}^{N_{\text{elec}}} \frac{1}{\|\underline{\boldsymbol{r}}_i - \underline{\boldsymbol{r}}_j\|_2}, \tag{4.2}$$
$$\hat{\mathcal{V}}_{ne} = -\sum_{A=1}^{M} \sum_{i=1}^{N_{\text{elec}}} \frac{Z_A}{\|\underline{\boldsymbol{R}}_A - \underline{\boldsymbol{r}}_i\|_2},$$

respectively. Furthermore we used the electronic and nuclear kinetic energy operators

$$\hat{\mathcal{T}}_e = -\frac{1}{2} \sum_{i=1}^{N_{\text{elec}}} \Delta_{\underline{\boldsymbol{r}}_i}, \qquad \hat{\mathcal{T}}_n = -\frac{1}{2} \sum_{A=1}^{M} \Delta_{\underline{\boldsymbol{R}}_A}, \tag{4.3}$$

with the shorthand

$$\Delta_{\underline{\boldsymbol{q}}} = \sum_{\alpha=1}^{3} \frac{\partial^2}{\partial q_\alpha^2}$$

for the Laplace operator with respect to particle coordinates $\underline{\boldsymbol{q}}$. If we take the domain $D(\hat{\mathcal{H}}_{\text{MB}}) = H^2(\mathbb{R}^L, \mathbb{C})$ this operator can be made self-adjoint [69]. It is furthermore bounded below [69] with a couple of discrete states below the essential spectrum.

The operator $\hat{\mathcal{H}}^{\text{MB}}$ is the fundamental object the field of quantum chemistry investigates. Its properties allow for a full (non-relativistic) quantum-mechanical description of a chemical system. This includes important properties like stable chemical structures or reactivity, with respect to other molecules as well as external potentials. As discussed in section 2.4 on page 29 a consequence of the laws of thermodynamics is, that in many cases one already gets a reasonable idea about the chemical properties of matter if only the lowest-energy, discrete eigenstates of the relevant many-body Hamiltonian $\hat{\mathcal{H}}_{\text{MB}}$ are determined.

Let us use the vectors $\underline{\boldsymbol{x}} \in \mathbb{R}^{3N_{\text{elec}}}$ and $\underline{\boldsymbol{X}} \in \mathbb{R}^{3M}$, defined as

$$\underline{\boldsymbol{x}}^{\text{T}} \equiv \left(\underline{\boldsymbol{r}}_1^{\text{T}}, \underline{\boldsymbol{r}}_2^{\text{T}}, \ldots, \underline{\boldsymbol{r}}_{N_{\text{elec}}}^{\text{T}}\right), \qquad \underline{\boldsymbol{X}}^{\text{T}} \equiv \left(\underline{\boldsymbol{R}}_1^{\text{T}}, \underline{\boldsymbol{R}}_2^{\text{T}}, \ldots, \underline{\boldsymbol{R}}_M^{\text{T}}\right), \tag{4.4}$$

to refer to all electronic or all nuclear coordinates, respectively. Taking I to denote an appropriate multi-index of quantum numbers, our problem from the previous paragraph

[1] If $\underline{\tilde{\boldsymbol{R}}}_A$ is the Cartesian coordinate of the A-th nucleus with mass M_A, then the mass-scaled coordinates are given as $\underline{\boldsymbol{R}}_A = \sqrt{M_A}\, \underline{\tilde{\boldsymbol{R}}}_A$.

can be reformulated as finding those eigenstates $\Psi_I^{\mathrm{MB}} \in H^2(\mathbb{R}^L, \mathbb{C})$ with lowest corresponding energies $E_I^{\mathrm{MB}} \in \mathbb{R}$ by the means of solving the time-independent Schrödinger equation

$$\hat{\mathcal{H}}^{\mathrm{MB}} \Psi_I^{\mathrm{MB}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) = E_I^{\mathrm{MB}} \Psi_I^{\mathrm{MB}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}). \tag{4.5}$$

Solving this equation analytically is not possible in general. Already for the Helium atom, a 3-body problem, clever approximations are needed to get anywhere [70]. But even numerically (4.5) is intractable to solve without further approximations.

Let us illustrate this claim by an example. In water, H_2O, we have 3 nuclei and 10 electrons. The dimensionality[2] of the problem is thus $L = 3 \cdot 13 = 39$. In the numerical approach we introduce in section 3.1 on page 31 the evaluation of the inner product

$$\langle \Psi | \Phi \rangle \equiv \int_{\mathbb{R}^{3M}} \int_{\mathbb{R}^{3N_{\mathrm{elec}}}} \Psi^*(\underline{\boldsymbol{x}}, \underline{\boldsymbol{X}}) \Phi(\underline{\boldsymbol{x}}, \underline{\boldsymbol{X}}) \, \mathrm{d}\underline{\boldsymbol{x}} \, \mathrm{d}\underline{\boldsymbol{X}} \tag{4.6}$$

between two functions Ψ and Φ from the underlying Hilbert space $L^2(\mathbb{R}^L, \mathbb{C})$ appears rather prominently. Most notably the computation of the sesquilinear form $a(\,\cdot\,,\,\cdot\,)$ in order to build the discretisation matrix in (3.9) boils down to computing such integrals. The numerical evaluation of (4.6) implies a sampling of the L-dimensional space $\mathbb{R}^L$ in some way or another. Even for an extremely sophisticated discretisation method or a well-designed quadrature scheme we will probably need of the order of 10 sampling points per dimension. For a 39-dimensional problem, like our water molecule, this makes on the order of 10^{39} sampling points overall. If we want a single integration to finish within the lifetime of a human being, say 100 years, the evaluation of the integration kernel $\Psi(\underline{\boldsymbol{x}}, \underline{\boldsymbol{X}})\Phi(\underline{\boldsymbol{x}}, \underline{\boldsymbol{X}})$ may take no more than some 10^{-30} seconds, which is impossible due to the physical limitations inside a general purpose computer.

Certainly one could probably find even more clever methods in some cases, but the example illustrates the so-called **curse of dimensionality** rather well. For a general quantum-chemical investigation of matter one needs to develop approximate methodologies.

4.2 Born-Oppenheimer approximation

The masses of electrons and nuclei differ by orders of magnitude. The ratio between the mass of a proton and the electron masses is already around 1836 and this ratio increases further across the table. Already for the elements of the second period, this value is at least of the order of 10^4. This justifies an approximative treatment, where we assume the motion of the electrons and the motion of the nuclei to happen at different timescales.

For this let us consider a simplified version of (4.1) at first, namely the **electronic Hamiltonian**

$$\hat{\mathcal{H}}^{\mathrm{elec}} \equiv \hat{\mathcal{H}}^{\mathrm{MB}} - \hat{\mathcal{T}}_n = \hat{\mathcal{T}}_e + \hat{\mathcal{V}}_{ne} + \hat{\mathcal{V}}_{ee} + \hat{\mathcal{V}}_{nn}.$$

This operator is constructed from the full many-body Hamiltonian by neglecting the nuclear kinetic energy operator $\hat{\mathcal{T}}_n$ completely. Introducing the short hand notation

$$r_{AB} \equiv \|\underline{\boldsymbol{R}}_A - \underline{\boldsymbol{R}}_B\|_2 \,, \qquad r_{iA} \equiv \|\underline{\boldsymbol{r}}_i - \underline{\boldsymbol{R}}_A\|_2 \,, \qquad r_{ij} \equiv \left\|\underline{\boldsymbol{r}}_i - \underline{\boldsymbol{r}}_j\right\|_2 \,,$$

[2] Some of the 39 degrees of freedom can be factored out, namely the 3 overall translations of the molecule. This does not change the overall picture very much and we will ignore this possibility in our discussion.

we can write it as

$$\hat{\mathcal{H}}^{\text{elec}} = -\frac{1}{2}\sum_{i=1}^{N_{\text{elec}}} \Delta_{\underline{\boldsymbol{r}}_i} - \sum_{A=1}^{M}\sum_{i=1}^{N_{\text{elec}}} \frac{Z_A}{r_{iA}} + \sum_{i=1}^{N_{\text{elec}}}\sum_{j=i+1}^{N_{\text{elec}}} \frac{1}{r_{ij}} + \sum_{A=1}^{M}\sum_{B=A+1}^{M} \frac{Z_A Z_B}{r_{AB}}. \tag{4.7}$$

Even though $\hat{\mathcal{H}}^{\text{elec}}$ still depends on the nuclear coordinates $\underline{\boldsymbol{X}}$, one could interpret the elements of the vector $\underline{\boldsymbol{X}}$ not as coordinates, but much rather as parameters for the potential operators $\hat{\mathcal{V}}_{ne}$ and $\hat{\mathcal{V}}_{nn}$. Physically this means that $\hat{\mathcal{H}}^{\text{elec}}$ describes a chemical system where the nuclei are clamped at well-defined points in space. Sometimes we will write $\hat{\mathcal{H}}^{\text{elec}}(\underline{\boldsymbol{X}})$ in order to make the parametrisation of $\hat{\mathcal{H}}^{\text{elec}}$ with respect to $\underline{\boldsymbol{X}}$ visible.

Without going into details at the moment, let us assume that $\hat{\mathcal{H}}^{\text{elec}}$ becomes self-adjoint inside a suitable domain. With appropriate multi-indices I_{e} we can thus find its eigenpairs $(E^{\text{elec}}_{I_{\text{e}}}, \Psi^{\text{elec}}_{I_{\text{e}}})$ via the **electronic Schrödinger equation**

$$\hat{\mathcal{H}}^{\text{elec}}(\underline{\boldsymbol{X}})\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) = E^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}})\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}). \tag{4.8}$$

Originating from the dependence of $\hat{\mathcal{H}}^{\text{elec}}(\underline{\boldsymbol{X}})$ towards the nuclear coordinates, we can think of the resulting **electronic energies** $E^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}})$ and **electronic wave function**s $\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})$ to be dependent on $\underline{\boldsymbol{X}}$ as well. Typically one uses the term **electronic state** to refer to $\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})$.

With the electronic states at hand we are able to formulate the framework of the **Born-Oppenheimer approximation**, which consists of the following two assumptions:

- Each eigenstate of (4.1) may be written by a factorisation

$$\Psi^{\text{MB}}_{I}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \equiv \Psi^{\text{MB}}_{I_{\text{e}}I_{\text{n}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \simeq \Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})\Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}}), \tag{4.9}$$

  where the multi-indices are related by $I \equiv (I_{\text{e}}, I_{\text{n}})$. $\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})$ is a solution to the electronic Schrödinger equation (4.8) and the **nuclear wave function** $\Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{x}})$ is yet to be determined.

- The factorisation (4.9) satisfies the property[3]

$$\hat{\mathcal{T}}_n \Psi^{\text{MB}}_{I}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \simeq \hat{\mathcal{T}}_n \left(\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})\Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}})\right) \simeq \Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \left(\hat{\mathcal{T}}_n \Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}})\right). \tag{4.10}$$

By plugging ansatz (4.9) into (4.5) we can simplify

$$\begin{aligned}
0 &= \left(\hat{\mathcal{H}}^{\text{MB}} - E^{\text{MB}}_{I}\right) \Psi^{\text{MB}}_{I}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \\
&\overset{(4.9)}{\simeq} \left(\hat{\mathcal{H}}^{\text{elec}} + \hat{\mathcal{T}}_n - E^{\text{MB}}_{I}\right) \Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})\Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}}) \\
&\overset{(4.10)}{\simeq} \left(\hat{\mathcal{H}}^{\text{elec}}\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})\Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}})\right) + \Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \left(\hat{\mathcal{T}}_n \Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}})\right) - E^{\text{MB}}_{I}\Psi^{\text{MB}}_{I}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \\
&\overset{(4.8)}{=} \Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}}) \left(E^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}})\Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}}) + \hat{\mathcal{T}}_n \Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}}) - E^{\text{MB}}_{I}\Psi^{\text{nuc}}_{I_{\text{n}}}(\underline{\boldsymbol{X}})\right).
\end{aligned}$$

[3] More precisely what we assume is that the nuclear kinetic energy operator $\hat{\mathcal{T}}_n$ projected onto the basis formed by all electronic states $\Psi^{\text{elec}}_{I_{\text{e}}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})$ is diagonal with all elements equal to 1. See [71] or [72] for details.

This statement is satisfied provided that the nuclear wave function $\Psi^{\text{nuc}}_{I_\text{n}}(\underline{\boldsymbol{X}})$ follows the **nuclear Schrödinger equation**

$$\left(\hat{\mathcal{T}}_n + E^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}})\right) \Psi^{\text{nuc}}_{I_\text{n}}(\underline{\boldsymbol{X}}) = E^{\text{MB}}_I \Psi^{\text{nuc}}_{I_\text{n}}(\underline{\boldsymbol{X}}). \tag{4.11}$$

Overall the Born-Oppenheimer approximation allows to solve the many-body Schrödinger equation (4.5) in two steps. First we limit ourselves to the point of view of the electrons under the electric field induced by fixed, motionless nuclei. This leads to (4.8), which is solved for the electronic states $\Psi^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})$ along with corresponding electronic energies $E^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}})$. In the second step we consider nuclear motion by solving (4.11). In this equation the electronic energies $E^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}})$ depending on the nuclear coordinates act as the electrostatic potential in which the nuclei move. For this reason $E^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}})$ is sometimes called a **potential energy surface** as well. Note, that each electronic state characterised by quantum numbers I_e gives rise to a different potential energy surface.

Employing a more detailed treatment of the Born-Oppenheimer approximation, like in the original paper [73] or Baer [71], allows to gain more insight regarding the range of applicability of the Born-Oppenheimer approximation. Loosely speaking it is a valid approximation as long as the potential energy surfaces $E^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}})$ are well-separated from another.

From a numerical point of view this approximation allows to reduce the dimensionality of the problem somewhat. To illustrate this let us return to the water molecule, which was already discussed at the end of section 4.1 on page 44. In the exact problem we need to solve one equation, namely the many-body Schrödinger equation (4.5) of dimensionality $L = 39$. Within the Born-Oppenheimer approximation this is replaced by solving two equations, the electronic one (4.8) of dimensionality $3N_\text{elec} = 30$ and the nuclear TISE (4.11) of dimensionality $3M = 9$. In the estimate we presented in section 4.1 on page 44 for the L^2 inner products, this would roughly provide a speed-up factor of 10^9.

4.2.1 Electronic Schrödinger equation

By solving the electronic Schrödinger equation (4.8) we get access to the electronic states $\Psi^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}}, \underline{\boldsymbol{x}})$ as well as the potential energy surface $E^{\text{elec}}_{I_\text{e}}(\underline{\boldsymbol{X}})$. In many cases these quantities already provide enough insight into a chemical system in order to address many questions relevant to quantum chemistry. For this reason the nuclear Schrödinger equation (4.11) will be neglected in this work from now on and we will focus only on approximation methods for solving (4.8) instead.

For ease of notation we will usually drop the indices "e" and the superscripts "elec" from now on if we refer to electronic energies or the electronic part of the wave function. Similarly in the context of the electronic Schrödinger equation the nuclei are motionless, which makes $\underline{\boldsymbol{X}}$ a fixed quantity. Thus we drop the nuclear coordinates "$\underline{\boldsymbol{X}}$" from the function arguments, too. In this convention we would for example write the electronic Schrödinger equation (4.8) as

$$\hat{\mathcal{H}} \Psi_I(\underline{\boldsymbol{x}}) = E_I \Psi_I(\underline{\boldsymbol{x}}).$$

Another simplification we sometimes employ is to consider the simplified electronic

Hamiltonian

$$\hat{\mathcal{H}}_{N_{\text{elec}}} \equiv \hat{\mathcal{H}}^{\text{elec}} - \hat{\mathcal{V}}_{nn} = -\frac{1}{2}\sum_{i=1}^{N_{\text{elec}}} \Delta_{\underline{\boldsymbol{r}}_i} - \sum_{A=1}^{M}\sum_{i=1}^{N_{\text{elec}}} \frac{Z_A}{r_{iA}} + \sum_{i=1}^{N_{\text{elec}}}\sum_{j=i+1}^{N_{\text{elec}}} \frac{1}{r_{ij}} \tag{4.12}$$

instead of $\hat{\mathcal{H}}^{\text{elec}}$. This is possible, since the potential operator governing the Coulombic interaction amongst the nuclei

$$\hat{\mathcal{V}}_{nn} = \sum_{A=1}^{M}\sum_{B=A+1}^{M} \frac{Z_A Z_B}{r_{AB}}$$

only depends on $\underline{\boldsymbol{X}}$, which makes it a constant value for one particular chemical system. In many cases one can therefore work with $\hat{\mathcal{H}}_{N_{\text{elec}}}$ in a numerical treatment and only add the nuclear potential energy term $\hat{\mathcal{V}}_{nn}$ afterwards.

In analogy to the many-body Hamiltonian (4.1) and the hydrogen-like Hamiltonian (2.17) we choose the underlying Hilbert space of $\hat{\mathcal{H}}_{N_{\text{elec}}}$ to be $L^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C})$. Due to Kato's theorem [69] $\hat{\mathcal{H}}_{N_{\text{elec}}}$ becomes self-adjoint if we set its domain to $D(\hat{\mathcal{H}}_{N_{\text{elec}}}) = H^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C})$. Not all functions in $H^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C})$ are physical, however [41, 42]. This is due to the fact that electrons do not only show spatial degrees of freedom, but furthermore an intrinsic angular momentum degree of freedom called **spin**. More precisely electrons are so-called spin-1/2 particles. By the spin statistics theorem [41] of quantum field theory this requires the electronic wave function to be antisymmetric with respect to particle exchange. More symbolically all eigenfunctions Ψ_I of $\hat{\mathcal{H}}_{N_{\text{elec}}}$ need to satisfy the condition

$$\forall i, j \in \{1, 2, \ldots, N_{\text{elec}}\}: \qquad \Psi_I(\ldots, \underline{\boldsymbol{r}}_i, \ldots, \underline{\boldsymbol{r}}_j, \ldots) = -\Psi_I(\ldots, \underline{\boldsymbol{r}}_j, \ldots, \underline{\boldsymbol{r}}_i, \ldots). \tag{4.13}$$

It is easy to see that not all elements of $H^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C})$ satisfy this.

Given that the classical correspondence of 2.1 on page 7 did not yield any kind of spin degree of freedom for non-relativistic QM, one might wonder at this point why we need to bother with spin and the resulting antisymmetry of the wave function at all in our physical model. As it turns out many fundamental experimental results and observations made at the beginning of the 20$^{\text{th}}$ century can only be explained if proper spin statistics is taken into account. This includes the Stern-Gerlach experiment [74–76], the spectral properties of atoms [77] and Fermi-Dirac statistics [78], just to name a few. Even though spin can only be rigorously derived using more sophisticated theories like relativistic QM or quantum field theory [41], one still needs to include it *ad hoc* in non-relativistic QM as well such that above observations can be explained [41, 77, 79].

Notice, that a proper inclusion of spin in non-relativistic QM requires two modifications. First we need each wave function to include an extra spin degree of freedom [77]. Secondly, we need to make sure that (4.13) is always satisfied [78]. We will defer the first modification to remark 4.12 on page 60 in order to yield a simpler mathematical treatment for now. Unfortunately we cannot ignore (4.13) due to its tremendous impact on the mathematical structure of the emerging problems [78, 80].

There are a couple of approaches which could be followed to adhere to (4.13). Typically one abstains from modifying the Hamiltonian $\hat{\mathcal{H}}_{N_{\text{elec}}}$ and instead restricts the search space for the eigenstates Ψ_I to an appropriate subspace of $L^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C})$, which is constructed in a way to enforce the required antisymmetry with respect to the electronic

coordinates [78, 80]. Such a space is the N_{elec}-th **exterior power** of $L^2(\mathbb{R}^3, \mathbb{C})$ defined as

$$\bigwedge^{N_{\text{elec}}} L^2(\mathbb{R}^3, \mathbb{C}) \equiv \text{span}\left\{ \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_{N_{\text{elec}}} \,\middle|\, \psi_i \in L^2(\mathbb{R}^3, \mathbb{C}) \, \forall i = 1, \ldots N_{\text{elec}} \right\}. \quad (4.14)$$

The key component of this definition is the **wedge product** or **exterior product** $f \wedge g$. This product is totally antisymmetric with respect to its operands and is closely related to the tensor product $f \otimes g$. For example if $f, g \in L^2(\mathbb{R}^3, \mathbb{C})$, then $f \wedge g \in L^2(\mathbb{R}^6, \mathbb{C})$. In some sense one can think of the wedge product as a generalisation of the cross product $\underline{\boldsymbol{a}} \times \underline{\boldsymbol{b}}$ for vectors $\underline{\boldsymbol{a}}, \underline{\boldsymbol{b}} \in \mathbb{R}^3$. This is somewhat apparent from its properties. Notice for example

$$\psi_1 \wedge \psi_1 = 0, \qquad \psi_1 \wedge \psi_2 = -\psi_2 \wedge \psi_1, \qquad \psi_1 \wedge (c_1\psi_1 + c_2\psi_2) = c_2\psi_1 \wedge \psi_2 \quad (4.15)$$

for any $\psi_1, \psi_2 \in L^2(\mathbb{R}^3, \mathbb{C})$ and any $c_1, c_2 \in \mathbb{C}$. One may identify the application of a wedge product string like

$$\bigwedge_{i=1}^{N_{\text{elec}}} \psi_i \equiv \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_{N_{\text{elec}}}$$

onto the electronic coordinates $\underline{\boldsymbol{x}}$ with the evaluation of a determinant, i.e.

$$\begin{aligned} \left(\bigwedge_{i=1}^{N_{\text{elec}}} \psi_i \right)(\underline{\boldsymbol{x}}) &\equiv \left(\psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_{N_{\text{elec}}} \right)(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2, \ldots, \underline{\boldsymbol{r}}_{N_{\text{elec}}}) \\ &\equiv \frac{1}{\sqrt{N_{\text{elec}}}} \det \begin{pmatrix} \psi_1(\underline{\boldsymbol{r}}_1) & \psi_2(\underline{\boldsymbol{r}}_1) & \cdots & \psi_{N_{\text{elec}}}(\underline{\boldsymbol{r}}_1) \\ \psi_1(\underline{\boldsymbol{r}}_2) & \psi_2(\underline{\boldsymbol{r}}_2) & \cdots & \psi_{N_{\text{elec}}}(\underline{\boldsymbol{r}}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_1(\underline{\boldsymbol{r}}_{N_{\text{elec}}}) & \psi_2(\underline{\boldsymbol{r}}_{N_{\text{elec}}}) & \cdots & \psi_{N_{\text{elec}}}(\underline{\boldsymbol{r}}_{N_{\text{elec}}}) \end{pmatrix}. \end{aligned}$$

Because of this observation $\psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_{N_{\text{elec}}}$ is typically called a **Slater determinant**[4] in standard quantum-chemistry textbooks [83, 84]. The functions $\psi_i \in L^2(\mathbb{R}^3, \mathbb{C})$ are usually called **single-particle functions**, since they only depend on a single electronic coordinate. Another way of phrasing (4.14) is therefore that $\bigwedge^{N_{\text{elec}}} L^2(\mathbb{R}^3, \mathbb{C})$ is the space spanned by all Slater determinants consisting of N_{elec} single-particle functions from $L^2(\mathbb{R}^3, \mathbb{C})$. Notice, that[5]

$$\bigwedge^{N_{\text{elec}}} L^2(\mathbb{R}^3, \mathbb{C}) \subset L^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C}),$$

exterior power on the left hand side is even dense in the space on the right.

If we want to encode condition (4.13) into our problem an easy solution is to combine this with Kato's theorem and employ the domain

$$D(\hat{\mathcal{H}}_{N_{\text{elec}}}) = H^2(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C}) \cap \bigwedge^{N_{\text{elec}}} L^2(\mathbb{R}^3, \mathbb{C}) \quad (4.16)$$

[4] After John Slater, who introduced it. [81, 82].

[5] This observation is the reason why the single-particle functions need to be square-integrable, i.e. from $L^2(\mathbb{R}^3, \mathbb{C})$.

for the electronic Hamiltonian $\hat{\mathcal{H}}_{N_\mathrm{elec}}$. This makes both the operator self-adjoint and the electronic wave function comply with the spin statistics theorem. Applying similar arguments to section 3.1.1 on page 31, we can deduce the analogous form domain of $\hat{\mathcal{H}}_{N_\mathrm{elec}}$ as

$$Q(\hat{\mathcal{H}}_{N_\mathrm{elec}}) = H^1(\mathbb{R}^{3N_\mathrm{elec}}, \mathbb{C}) \cap \bigwedge^{N_\mathrm{elec}} L^2(\mathbb{R}^3, \mathbb{C}).$$

For establishing the spectral properties of $\hat{\mathcal{H}}_{N_\mathrm{elec}}$, there is first the important HVZ theorem [54–57] after Hunziker, van Winter and Zhislin.

Theorem 4.1 (HVZ)**.** *Let $\hat{\mathcal{H}}_{N_{elec}}$ be the self-adjoint operator of* (4.12) *on the Hilbert space $L^2(\mathbb{R}^{3N_{elec}}, \mathbb{C})$ with the domain given in* (4.16)*. Then $\hat{\mathcal{H}}_{N_{elec}}$ is bounded from below and its essential spectrum*[6] *is*

$$\sigma_{ess}\left(\hat{\mathcal{H}}_{N_{elec}}\right) = [\Sigma_{N_{elec}}, +\infty)$$

with

$$\Sigma_{N_{elec}} = \begin{cases} 0 & \text{if } N_{elec} = 1 \\ \inf \sigma\left(\hat{\mathcal{H}}_{N_{elec}-1}\right) < 0 & \text{if } N_{elec} \geq 2 \end{cases}.$$

This theorem establishes a link between the lower bound of the essential spectrum σ_ess of the electronic Hamiltonian of a N_elec-electron system and the lower bound of the complete spectrum σ of a corresponding $N_\mathrm{elec} - 1$ electron system with the same nuclear arrangement.

For characterising the discrete spectrum of $\hat{\mathcal{H}}_{N_\mathrm{elec}}$ we employ the important results by Zhislin [85] and Yafaev [86], summarised by the following proposition.

Proposition 4.2. *Let $\hat{\mathcal{H}}_{N_{elec}}$ be the N_{elec}-electron electronic Hamiltonian operator of* (4.12) *with domain as stated in* (4.16)*. Let further $Z_{tot} \equiv \sum_{A=1}^{M} Z_A$ denote the total nuclear charge.*

- *If $N_{elec} \leq Z_{tot}$, i.e. we consider a* neutral *or* positively charged *system, then $\hat{\mathcal{H}}_{N_{elec}}$ has an infinite number of discrete eigenvalues below the essential spectrum.*
- *If $N_{elec} \geq 1 + Z_{tot}$ (*negatively charged *system), then $\hat{\mathcal{H}}_{N_{elec}}$ has* at most *a finite number of discrete states below the essential spectrum.*

Proof. See [85, 86]. □

Before we discuss the physical interpretation of these results, let us first introduce some terminology. If we are either concerned with a neutral or positively charged N_elec-electron system or a negatively charged system with at least a single discrete eigenvalue, we can define a **ground-state energy**

$$E_0^{N_\mathrm{elec}} = \min \sigma\left(\hat{\mathcal{H}}_{N_\mathrm{elec}}\right) = \Sigma_{N_\mathrm{elec}+1}. \tag{4.17}$$

The energies of the discrete spectrum are ordered as usual

$$E_0^{N_\mathrm{elec}} \leq E_1^{N_\mathrm{elec}} \leq E_2^{N_\mathrm{elec}} \leq \cdots$$

[6] For a definition see 2.45 on page 25.

and associated with these eigenvalues are the corresponding (bound) eigenstates

$$\Psi_0, \Psi_1, \Psi_2, \ldots.$$

All states Ψ_i which have an energy eigenvalue $E_i^{N_\text{elec}} = E_0^{N_\text{elec}}$ are commonly referred to as the **ground state**. If $E_0^{N_\text{elec}}$ is not degenerate by construction Ψ_0 is the ground state. All other states Ψ_i with $E_i^{N_\text{elec}} \neq E_0^{N_\text{elec}}$ are called **excited states**.

For negatively charged systems we similarly use the term "ground state" to refer to the state or states corresponding to the lowest eigenvalue of $\sigma_P\left(\hat{\mathcal{H}}_{N_\text{elec}}\right)$ and "excited states" for the other eigenfunctions of $\hat{\mathcal{H}}_{N_\text{elec}}$. Note, however, that for negatively charged systems the case $\inf\sigma\left(\hat{\mathcal{H}}_{N_\text{elec}}\right) \leq E_0^{N_\text{elec}}$ is possible (see proposition 4.2), i.e. that the lowest-energy bound state is already embedded inside the continuum.

Remark 4.3 (Physical interpretation of the spectrum)**.** In this remark we will summarise the results, which can be deduced from the HVZ theorem 4.1 and from proposition 4.2 on the facing page.

Let us first consider a neutral or positively charged system with N_elec electrons. It has a ground states as well as an infinite number of discrete and bound excited states until the ground-state energy $E_0^{N_\text{elec}-1}$ of the corresponding $(N_\text{elec}-1)$-electron with the same nuclear arrangement is hit. Note, that we can be sure that $E_0^{N_\text{elec}-1}$ exists, because the $(N_\text{elec}-1)$-electron system is positively charged. This behaviour is easy to understand physically. As soon as the energy $E_0^{N_\text{elec}-1}$ is reached our N_elec-electron system can always separate into a stable system with $N_\text{elec}-1$ bound electrons and an unbound N_elec-th electron taking the excess energy into the continuum. From this we can easily understand the energy difference $E_0^{N_\text{elec}-1} - E_0^{N_\text{elec}}$ as the ionisation energy. Note, that embedded inside the emerging continuum at energies beyond $E_0^{N_\text{elec}-1}$ may still be bound states of the $(N_\text{elec}-1)$-electron system or the N_elec-system. In other words in general we have

$$\sigma_C\left(\hat{\mathcal{H}}_{N_\text{elec}}\right) \subsetneq \sigma_\text{ess}\left(\hat{\mathcal{H}}_{N_\text{elec}}\right).$$

If the N_elec-system is of single negative charge and possesses no bound states below the essential spectrum, this implies

$$\inf\sigma\left(\hat{\mathcal{H}}_{N_\text{elec}}\right) = \inf\sigma\left(\hat{\mathcal{H}}_{N_\text{elec}-1}\right) = E_0^{N_\text{elec}-1},$$

since the $N_\text{elec}-1$-electron system is neutral, thus possesses a ground state. In other words all bound states of this system are embedded inside the continuum. The system thus may separate into a bound $(N_\text{elec}-1)$-electron system and an unbound electron at all energies: This negative ion is not stable. Conversely for stable negative ions we would expect at least a single bound state to exist.

Unlike neutral or positively charged systems, negatively charged systems in each case only possess a finite number of bound states below the essential spectrum.

To summarise this remark, let us note the following interesting observations, which are now backed by rigorous mathematical treatment:

- The essential spectrum marks the energies at which a chemical system is unstable, because it can separate into (one or more) unbound electron plus a stable system with a reduced number of bound electrons.

- Forming a positive ion always costs energy.
- All systems with more than one electron will produce unbound electrons, i.e. ionise, at large-enough energies (in vacuum).
- Not all negative ions possess a stable ground state (in vacuum).
- All positive ions possess a stable ground state (in vacuum).

Remark 4.4 (Consequences for a numerical treatment)**.** From proposition 4.2 we can immediately deduce, that there are no problems with neutral or positively charged systems under a variational numerical treatment as discussed in section 3.1 on page 31. Both the ground state as well as the first few excited states are located below the essential spectrum and thus accessible for the treatment described in remark 3.5 on page 34.

For negative ions we might get into trouble. If the ion is stable, then at least its ground state can be approximated numerically via remark 3.5. If it is not, we might not even be able to get its ground state. The problem is, that in a variational numerical treatment we cannot easily distinguish between approximations to bound states and approximations to continuum states if they are located in the same energy range. So if the lowest-energy bound state is embedded inside the continuum, both are inside the essential spectrum. A variational treatment will converge to the bottom end of the essential spectrum (see theorem 3.3), which might not be the lowest-energy bound state in this case.

4.3 Full configuration interaction

In this section we want to develop a numerical treatment for solving the electronic Schrödinger equation (4.8) under the Ritz-Galerkin projection ansatz of section 3.1.2 on page 32. In the previous section we analysed the mathematical implications of the spin statistics theorem for electrons as fermionic systems, which lead us to choose the form domain

$$Q(\hat{\mathcal{H}}_{N_{\text{elec}}}) = H^1(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C}) \cap \bigwedge^{N_{\text{elec}}} L^2(\mathbb{R}^3, \mathbb{C}).$$

for the electronic Schrödinger operator $\hat{\mathcal{H}}_{N_{\text{elec}}}$.

For simplifying our treatment we will not try to discretise this domain in the Ritz-Galerkin ansatz of definition 3.2, much rather we will develop methods to sample only the subspace

$$\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}}) \equiv \bigwedge^{N_{\text{elec}}} H^1(\mathbb{R}^3, \mathbb{C}) \subset Q(\hat{\mathcal{H}}_{N_{\text{elec}}})$$

due to its simpler structure. Since this subspace is dense we will not suffer from any loss of numerical accuracy in the approximate treatment later on. By definition of the exterior power

$$\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}}) = \operatorname{span}\left\{ \bigwedge_{i=1}^{N_{\text{elec}}} \psi_i \,\middle|\, \psi_i \in H^1(\mathbb{R}^3, \mathbb{C})\, \forall i = 1, \ldots, N_{\text{elec}} \right\}. \tag{4.18}$$

Since $H^1(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C})$ is separable, we can find a countable basis set

$$\mathbb{B}_1 \equiv \{\psi_i\}_{i\in\mathbb{N}} \qquad \text{with } \langle \psi_i | \psi_j \rangle_1 = \delta_{ij} \quad \text{and} \quad \operatorname{span} \mathbb{B}_1 = H^1(\mathbb{R}^{3N_{\text{elec}}}, \mathbb{C}), \tag{4.19}$$

where we used the abbreviated notation $\langle \cdot \,|\, \cdot \rangle_1 \equiv \langle \cdot \,|\, \cdot \rangle_{L^2(\mathbb{R}^3,\mathbb{C})}$ for the one-particle inner product. Taking the properties of the wedge product (4.15) into account allows to deduce the equivalent construction

$$\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}}) = \text{span}\left\{ \bigwedge_{i=1}^{N_{\text{elec}}} \psi_i, \,\middle|\, \psi_i \in \mathbb{B}_1 \,\forall i = 1, \ldots, N_{\text{elec}} \right\}, \tag{4.20}$$

which builds $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ as the span over all Slater determinants built by selecting N_{elec} functions from $\mathbb{B}_1$. Since nothing stops us from selecting the same basis function twice from $\mathbb{B}_1$ in this construction, many of the constructed determinants $\bigwedge_{i=1}^{N_{\text{elec}}} \psi_i$ are zero. In other words these determinants amount to span $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$, but they are not a basis for this space. In the following we want to fix this and construct an orthonormal basis of suitable Slater determinants. This requires an appropriate inner product.

Definition 4.5. Let $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ be defined as in (4.18). We define an inner product on $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ by requiring for any two arbitrary Slater determinants

$$\Psi = \bigwedge_{i=1}^{N_{\text{elec}}} \psi_i \qquad \text{and} \qquad \Xi = \bigwedge_{i=1}^{N_{\text{elec}}} \xi_i$$

with $\psi_i, \xi_i \in H^1(\mathbb{R}^3, \mathbb{C})$ for all $i \in 1, \ldots, N_{\text{elec}}$:

$$\langle \Psi | \Xi \rangle_{N_{\text{elec}}} \equiv \det \mathbf{G} \qquad \text{where}^7 G_{ij} = \langle \psi_i | \xi_j \rangle_{L^2(\mathbb{R}^3,\mathbb{C})} \,\forall i, j \in 1, \ldots, N_{\text{elec}}, \tag{4.21}$$

The inner product for other elements from $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ is then constructed in accordance with the axioms shown in definition 2.1 on page 12.

With this inner product at hand we can construct an orthonormal basis for $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$.

Remark 4.6 (Orthonormal basis for $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$)**.** Let $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ be defined as in (4.18) and let $\mathbb{B}_1$ be an arbitrary basis for $H^1(\mathbb{R}^3, \mathbb{C})$. We take one arbitrary, non-trivial Slater determinant $0 \neq \Phi_0 \in \tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$, such that

$$\Phi_0 = \tilde{\psi}_1 \wedge \tilde{\psi}_2 \wedge \cdots \tilde{\psi}_i \cdots \wedge \tilde{\psi}_{N_{\text{elec}}}$$

for appropriate $\tilde{\psi}_i \in \mathbb{B}_1$. This determinant can always be found due to the alternative construction for $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ sketched in (4.20). Let us call Φ_0 the **reference determinant** in the following.

The functions of the (countable) basis set $\mathbb{B}_1 = \{\psi_i\}_{i \in \mathbb{N}}$ can be indexed in such a way that the first N_{elec} functions coincide with $(\tilde{\psi}_1, \tilde{\psi}_2, \ldots, \tilde{\psi}_{N_{\text{elec}}})$. In other words

$$\Phi_0 = \psi_1 \wedge \psi_2 \wedge \cdots \psi_i \cdots \wedge \psi_{N_{\text{elec}}}$$

as well. We further define the index sets[8]

$$\mathcal{I}_{\text{occ}} = \{1, \ldots, N_{\text{elec}}\} \qquad \text{and} \qquad \mathcal{I}_{\text{virt}} = \{i \in \mathbb{N} \,|\, i > N_{\text{elec}}\}.$$

[7] In this $\mathbf{G}$ is the so-called Gramian matrix.

[8] The subscript "occ" stands for *occupied* and "virt" for *virtual*. These terms will become clear when we discuss the Hartree-Fock ansatz in the next section.

With reference to Φ_0 we can construct for each $i \in \mathcal{I}_{\text{occ}}$ and each $a \in \mathcal{I}_{\text{virt}}$ a so-called singly **excited determinant**

$$\Phi_i^a = \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_a \wedge \cdots \wedge \psi_{N_{\text{elec}}}$$

by replacing the i-th function of the Slater determinant wedge string by the a-th function of $\mathbb{B}_1$ without changing the order. Analogously one may define doubly or higher excited determinants

$$\Phi_{ij}^{ab} = \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_a \wedge \cdots \wedge \psi_b \wedge \cdots \wedge \psi_{N_{\text{elec}}}$$
$$\Phi_{ijk}^{abc} = \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_a \wedge \cdots \wedge \psi_b \wedge \cdots \wedge \psi_c \wedge \cdots \wedge \psi_{N_{\text{elec}}}$$

where[9] $i, j, k \in \mathcal{I}_{\text{occ}}$ and $a, b, c \in \mathcal{I}_{\text{virt}}$ In this case one has to additionally require that $i < j < k < \cdots$ and $a < b < c < \cdots$, because otherwise no new determinants are generated (if $i = j$ or $i = k$ or $\dots$) or a zero determinant is generated (if $a = b$ or similar). Constructed in this way all determinants in the set

$$\mathbb{B}_{N_{\text{elec}}} \equiv \left\{\Phi_0, \Phi_i^a, \Phi_{ij}^{ab}, \Phi_{ijk}^{abc}, \cdots\right\}$$

are unique. Still it is not hard to see that $\operatorname{span} \mathbb{B}_{N_{\text{elec}}} = \tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$, since we only took away those determinants adding redundant information in the construction (4.20).

With the inner product defined in (4.21) we notice for all $r, s \in \mathbb{N}$

$$\langle \Phi_0 | \Phi_r^s \rangle_{N_{\text{elec}}} = \langle \psi_r | \psi_s \rangle_1 = \delta_{rs},$$

since by (4.19) all functions in $\mathbb{B}_1$ are orthonormal to each other. In other words $\mathbb{B}_{N_{\text{elec}}}$ is an orthonormal basis for $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$.

The set $\mathbb{B}_{N_{\text{elec}}}$ is sometimes called the **N_{elec}-particle basis** or **many-particle basis** corresponding to $\mathbb{B}_1$ and the reference determinant Φ_0. Albeit the precise entries in $\mathbb{B}_{N_{\text{elec}}}$ might differ from case to case the end result $\operatorname{span} \mathbb{B}_{N_{\text{elec}}} = \tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ is always true regardless of the choice of Φ_0 or $\mathbb{B}_1$.

Remark 4.7. Given a many-particle basis $\mathbb{B}_{N_{\text{elec}}}$ consisting of normalised Slater determinants, any function $\Psi \in \tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ can be expanded as such

$$\Psi = \sum_{\mu} c_\mu \Phi_\mu \qquad \text{where } \forall \mu \in \mathbb{N}: \quad \Phi_\mu \in \mathbb{B}_{N_{\text{elec}}}, c_\mu \in \mathbb{C}. \tag{4.22}$$

If one is interested in emphasising the particular basis of one-particle functions $\mathbb{B}_1$ and the particular reference determinants Φ_0 this can be written equivalently as

$$\Psi = c_0 \Phi_0 + \sum_{ia} c_i^a \Phi_i^a + \sum_{\substack{i<j \\ a<b}} c_{ij}^{ab} \Phi_{ij}^{ab} + \sum_{\substack{i<j<k \\ a<b<c}} c_{ijk}^{abc} \Phi_{ijk}^{abc} + \cdots, \tag{4.23}$$

where $i, j, k, \ldots \in \mathcal{I}_{\text{occ}}$ and $a, b, c, \ldots \in \mathcal{I}_{\text{virt}}$.

This expansion is commonly referred to as the **CI expansion**, where CI stands for configuration interaction, a term which will become more clear after the next remark.

Remark 4.8 (Full CI)**.** The discrete formulation of the Ritz-Galerkin scheme of remark 3.6 on page 34 can now be applied rather easily to the electronic Schrödinger equation. This leads to a procedure called **full CI** or full configuration interaction (FCI).

[9] This is the typical indexing convention in quantum chemistry. Indices $i, j, k, l, m, \ldots$ stand for occupied indices and $a, b, c, d, e, \ldots$ for virtual indices.

For $n = 1, 2, \ldots$:

- Take a finite-sized basis set of orthonormal one-particle functions

$$\mathbb{B}_1^{(n)} \equiv \{\psi_i\}_{i \in \mathcal{I}_{\text{bas}}} \subset U,$$

where $U \subset H^1(\mathbb{R}^3, \mathbb{C})$ is dense.

- Choose — at random or using prior knowledge — an arbitrary reference determinant

$$\Phi_0 = \psi_1 \wedge \psi_2 \wedge \ldots \wedge \psi_{N_{\text{elec}}}$$

where

$$(\psi_1, \psi_2, \ldots, \psi_{N_{\text{elec}}}) \in \left(\mathbb{B}_1^{(n)}\right)^{N_{\text{elec}}}$$

and construct the finite N_{elec}-electron basis

$$\mathbb{B}_{N_{\text{elec}}}^{(n)} \equiv \{\Phi_0, \Phi_i^a, \Phi_{ij}^{ab}, \ldots\} \tag{4.24}$$

using substitutions of the functions from $\mathbb{B}_1^{(n)}$ according to the procedure described in remark 4.6 on page 53. As usual we take

$$i, j, k, l, \ldots \in \mathcal{I}_{\text{occ}} \qquad \text{with} \qquad i < j < k < l < \cdots, \tag{4.25}$$

$$a, b, c, d, \ldots \in \mathcal{I}_{\text{virt}} \qquad \text{with} \qquad a < b < c < d < \cdots \tag{4.26}$$

where in the finite case

$$\mathcal{I}_{\text{occ}} = \{1, \ldots, N_{\text{elec}}\} \qquad \text{and} \qquad \mathcal{I}_{\text{virt}} = \{N_{\text{elec}} + 1, \ldots, N_{\text{bas}}\}.$$

- Construct the full CI matrix $\mathbf{A}_{\text{FCI}} \in \mathbb{C}^{N_{\text{FCI}} \times N_{\text{FCI}}}$ consisting of elements

$$(A_{\text{FCI}})_{IJ} = a(\Phi_I, \Phi_J) = \left\langle \Phi_I \middle| \hat{\mathcal{T}}_e + \hat{\mathcal{V}}_{ne} + \hat{\mathcal{V}}_{ee} \middle| \Phi_J \right\rangle_{N_{\text{elec}}} \tag{4.27}$$

for all combinations $\Phi_I, \Phi_J \in \mathbb{B}_{N_{\text{elec}}}^{(n)}$. There are

$$N_{\text{FCI}} = \binom{N_{\text{bas}}}{N_{\text{elec}}} \leq N_{\text{bas}}^{N_{\text{elec}}}$$

such Slater determinants.

- Diagonalise $\mathbf{A}_{\text{FCI}}$ to find a few energy eigenvalues $E_i^{(n)} \in \mathbb{R}$ and corresponding **CI vectors** $\underline{\boldsymbol{c}}_i^{(n)} \in \mathbb{C}^{N_{\text{FCI}}}$.

- Repeat with a larger basis $\mathbb{B}_1^{(n+1)}$ until convergence of eigenstates up to desired accuracy has been achieved. In many cases one already selects a suitable basis set $\mathbb{B}_1^{(n)}$ and only performs the calculation top-to-bottom once.

Notice that the subspace sequence $\operatorname{span} \mathbb{B}_{N_{\text{elec}}}^{(n)} \subset \tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ satisfies the required condition (3.4) since U is dense in $H^1(\mathbb{R}^3, \mathbb{C})$, which makes $\operatorname{span} \mathbb{B}_{N_{\text{elec}}}^{(n)}$ dense in $\tilde{Q}(\hat{\mathcal{H}}_{N_{\text{elec}}})$ and thus transitively dense in $Q(\hat{\mathcal{H}}_{N_{\text{elec}}})$. If $\hat{\mathcal{H}}_{N_{\text{elec}}}$ thus has a discrete ground and some discrete excited states below the essential spectrum, we can approximate it by this procedure up to arbitrary accuracy. This is satisfied for all neutral or positively charged systems and some negatively charged systems. Recall remark 4.4 on page 52 for details.

Equation (4.27) helps to understand where the term full configuration interaction for the sketched method comes from. In some sense a basis function in the single-particle basis $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ describes the behaviour of a single electron. In turn a Slater determinant can be interpreted physically as one sensible configuration of N_{elec} electrons amongst the available single-particle functions. The full CI matrix (4.27) now couples these configurations via the electronic Hamiltonian $\hat{\mathcal{H}}_{N_{\text{elec}}}$, which describes the interaction of the electrons in the chemical system with another. By diagonalising the matrix $\mathbf{A}_{\text{FCI}}$ we thus determine the electronic eigenstates taking the full range of interactions between all configurations into account, explaining the name full CI.

Even though FCI allows to compute the solution of the electronic Schrödinger equation up to arbitrary accuracy, it is only employed for the simplest problems or for benchmark purposes. The main reason for this is its enormous computational cost. Already for small molecules like water with only $N_{\text{elec}} = 10$ electrons and a rather small def2-SV(P)[87] basis set $\mathbb{B}_1^{(n)}$ with $N_{\text{bas}} = 18$ basis functions this makes $N_{\text{FCI}} = 43758$ and thus around $N_{\text{FCI}}^2 = 2 \cdot 10^9$ entries in $\mathbf{A}_{\text{FCI}}$ in an extremely naïve implementation, where known zero entries are stored as well. Of course this can be improved by exploiting some symmetries or the rather sparse structure of $\mathbf{A}_{\text{FCI}}$, which we will discuss in the next remark. Nevertheless the computational cost scales exponentially and allows only treatment of small systems[10]

Remark 4.9 (Structure of the FCI matrix)**.** Recall the expression

$$\hat{\mathcal{H}}_{N_{\text{elec}}} = \hat{\mathcal{T}}_e + \hat{\mathcal{V}}_{en} + \hat{\mathcal{V}}_{ee} = \sum_{i=1}^{N_{\text{elec}}} \left(-\frac{1}{2}\Delta_{\underline{\boldsymbol{r}}_i} + \sum_{A=1}^{M} \frac{Z_A}{r_{iA}} \right) + \sum_{i=1}^{N_{\text{elec}}} \sum_{j=i+1}^{N_{\text{elec}}} \frac{1}{r_{ij}}$$

for the electronic Hamiltonian. The goal of this remark will be to write the many electron integrals $\left\langle \Phi_1 \middle| \hat{\mathcal{H}}_{N_{\text{elec}}} \Phi_2 \right\rangle$ between two Slater determinants $\Phi_1, \Phi_2 \in \mathbb{B}_{N_{\text{elec}}}$ in terms of integrals over the one-electron functions ψ_i these determinants are composed of.

For this we will make use of the so-called Slater-Condon rules [83]. For applying these rules we need to differentiate between so-called **one-electron operator**s and **two-electron operator**s. One-particle operators like $\hat{\mathcal{T}}_e$ and $\hat{\mathcal{V}}_{en}$ can be written as a sum of operators like $\Delta_{\underline{\boldsymbol{r}}_i}$ or r_{iA}^{-1}, which act only on the coordinate $\underline{\boldsymbol{r}}_i$ of a single electron at once. Two-particle operators like $\hat{\mathcal{V}}_{ee}$, however, are built as a sum of terms r_{ij}^{-1} making reference to the coordinates of two electrons.

For our discussion here, let us take Φ_0 to be an arbitrary reference determinant constructed from the single-particle basis $\mathbb{B}_1$. We construct excited determinants Φ_i^a, Φ_{ij}^{ab}, ... under the index conventions (4.25) and (4.26).

For the one-electron operator $\hat{\mathcal{T}}_e + \hat{\mathcal{V}}_{en}$ the Slater-Condon rules yield

$$\begin{aligned}
\left\langle \Phi_0 \middle| \left(\hat{\mathcal{T}}_e + \hat{\mathcal{V}}_{en} \right) \Phi_0 \right\rangle_{N_{\text{elec}}} &= \sum_{i\in\mathcal{I}_{\text{occ}}} \left\langle \psi_i \middle| \hat{\mathcal{H}}_{\text{core}}\, \psi_i \right\rangle_1 \\
\left\langle \Phi_0 \middle| \left(\hat{\mathcal{T}}_e + \hat{\mathcal{V}}_{en} \right) \Phi_i^a \right\rangle_{N_{\text{elec}}} &= \left\langle \psi_i \middle| \hat{\mathcal{H}}_{\text{core}}\, \psi_a \right\rangle_1 \\
\left\langle \Phi_0 \middle| \left(\hat{\mathcal{T}}_e + \hat{\mathcal{V}}_{en} \right) \Phi_{ij}^{ab} \right\rangle_{N_{\text{elec}}} &= 0.
\end{aligned} \tag{4.28}$$

[10] Our water case is definitely still feasible with modern FCI techniques. A benchmark calculation from 1999 for example treated a system with $N_{\text{FCI}} \simeq 9.7 \cdot 10^9$ [88].

In this result we made use of the **core Hamiltonian** operator

$$\hat{\mathcal{H}}_{\text{core}} = \hat{\mathcal{T}} + \hat{\mathcal{V}}_0 = -\frac{1}{2}\Delta - \sum_{A=1}^{M} \frac{Z_A}{\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_A\|_2}, \tag{4.29}$$

which is just the sum of the kinetic operator $\hat{\mathcal{T}}$ and the nuclear attraction operator $\hat{\mathcal{V}}_0$ contribution from a single electron. Since the choice of the reference determinant Φ_0 was arbitrary, we can state more generally that the element $\left\langle \Phi_1 \middle| \hat{\mathcal{A}}_1 \Phi_2 \right\rangle_{N_{\text{elec}}}$ of a one-particle operator $\hat{\mathcal{A}}_1$ is only non-zero for determinants Φ_1, Φ_2, which differ in at most one single-particle function.

On the other hand for two-electron operators like $\hat{\mathcal{V}}_{ee}$ the Slater-Condon rules yield

$$\begin{aligned}
\left\langle \Phi_0 \middle| \hat{\mathcal{V}}_{ee}\, \Phi_0 \right\rangle_{N_{\text{elec}}} &= \frac{1}{2} \sum_{i \in \mathcal{I}_{\text{occ}}} \sum_{j \in \mathcal{I}_{\text{occ}}} (\psi_i \psi_i | \psi_j \psi_j) - (\psi_j \psi_i | \psi_i \psi_j)\,, \\
\left\langle \Phi_0 \middle| \hat{\mathcal{V}}_{ee}\, \Phi_i^a \right\rangle_{N_{\text{elec}}} &= \sum_{j \in \mathcal{I}_{\text{occ}}} (\psi_i \psi_a | \psi_j \psi_j) - (\psi_j \psi_a | \psi_i \psi_j)\,, \\
\left\langle \Phi_0 \middle| \hat{\mathcal{V}}_{ee}\, \Phi_{ij}^{ab} \right\rangle_{N_{\text{elec}}} &= (\psi_i \psi_j | \psi_a \psi_b) - (\psi_a \psi_j | \psi_i \psi_b)\,, \\
\left\langle \Phi_0 \middle| \hat{\mathcal{V}}_{ee}\, \Phi_{ijk}^{abc} \right\rangle_{N_{\text{elec}}} &= 0.
\end{aligned} \tag{4.30}$$

where the **electron-repulsion integrals** (ERIs) in Mulliken's indexing convention are given by the expression

$$(\psi_i \psi_j | \psi_k \psi_l) = \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} \frac{\psi_i^*(\underline{\boldsymbol{r}}_1)\psi_j(\underline{\boldsymbol{r}}_1)\, \psi_k^*(\underline{\boldsymbol{r}}_2)\psi_l(\underline{\boldsymbol{r}}_2)}{\|\underline{\boldsymbol{r}}_1 - \underline{\boldsymbol{r}}_2\|_2}\, \mathrm{d}\underline{\boldsymbol{r}}_1\, \mathrm{d}\underline{\boldsymbol{r}}_2. \tag{4.31}$$

Again this result generalises in the sense that for a two particle operator $\hat{\mathcal{A}}_2$ and any determinants Φ_1 and Φ_2 the elmement $\left\langle \Phi_1 \middle| \hat{\mathcal{A}}_2 \Phi_2 \right\rangle_{N_{\text{elec}}}$ is only non-zero if the determinants differ in at most two single-particle functions.

Both these observations combined allow to deduce that the full CI matrix $\mathbf{A}_{\text{FCI}}$ must be rather sparse. Originating from the two-electron Coulomb term $\hat{\mathcal{V}}_{ee}$ all entries $a(\Phi_1, \Phi_2)$ where the determinants differ by more than two functions vanish. If we pick an arbitrary reference determinant Φ_0 and order the N_{elec}-electron basis as in equation (4.24) a banded structure as in fig. 4.1 on the following page results. Of course the dimensionality is still large, but a combination of the iterative methods sketched in section 3.2 on page 36 and a contraction-based ansatz like the one sketched in section 6.1 on page 142 allow to obtain a few eigenvalues of $\mathbf{A}_{\text{FCI}}$ exploiting the sparsity structure.

The electron-repulsion integral tensor introduced in (4.31) is a very important quantity in computational chemistry. We will employ it at various occasions throughout the thesis. In the standard literature a number of deviating conventions are used for denoting this tensor. The following remark provides a summary.

Remark 4.10 (Formulation of the repulsion integrals)**.** In equation (4.31) we already met the electron-repulsion integral $(\psi_i \psi_j | \psi_k \psi_l)$ in **Mulliken notation**. Alternative names for this indexing convention are **shell pair notation** or **chemists' notation**. If the one-particle basis and its indexing convention is clear from context one sometimes writes this integral abbreviated as $(ij|kl)$ as well.

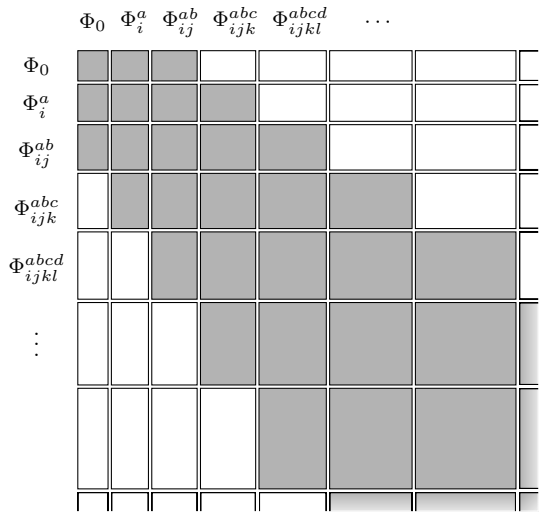


Figure 4.1: Sketch of the upper left part of the FCI matrix $\mathbf{A}_{\mathrm{FCI}}$. The identified blocks denote the interaction of equivalent classes of excited determinants under the electronic Hamiltonian $\hat{\mathcal{H}}_{N_{\mathrm{elec}}}$. The size of the blocks increases left to right and top to bottom and is not depicted to scale. Blocks with white background are identically zero and blocks with grey background may contain non-zero elements. The grey blocks show further sparsity, which is not fully depicted here. See [89] for details.

An alternative convention is physicists' notation

$$\langle ij|kl\rangle \equiv \langle \psi_i\psi_j|\psi_k\psi_l\rangle = \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} \frac{\psi_i^*(\underline{\boldsymbol{r}}_1)\psi_j^*(\underline{\boldsymbol{r}}_2)\,\psi_k(\underline{\boldsymbol{r}}_1)\psi_l(\underline{\boldsymbol{r}}_2)}{\|\underline{\boldsymbol{r}}_1 - \underline{\boldsymbol{r}}_2\|_2}\,\mathrm{d}\underline{\boldsymbol{r}}_1\,\mathrm{d}\underline{\boldsymbol{r}}_2.$$

Both conventions are related by

$$\langle ij|kl\rangle = (ik|jl)\,. \tag{4.32}$$

It is a rather common feature that the ERI integrals appear in pairs like in (4.30), where the indices are only slightly permuted. For this reason one typically defines an **antisymmetrised electron-repulsion tensor** with elements

$$\langle ij||kl\rangle \equiv \langle \psi_i\psi_j||\psi_k\psi_l\rangle \equiv \langle \psi_i\psi_j|\psi_k\psi_l\rangle - \langle \psi_j\psi_i|\psi_k\psi_l\rangle = (\psi_i\psi_k|\psi_j\psi_l) - (\psi_j\psi_k|\psi_i\psi_l)$$

as well. With this quantity the element $a(\Phi,\Phi)$, where the quadratic form is applied to an arbitrary normalised determinant $\Phi = \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_{N_{\mathrm{elec}}}$ can be written as

$$a(\Phi,\Phi) = \left\langle \Phi \middle| \hat{\mathcal{H}}_{N_{\mathrm{elec}}}\,\Phi \right\rangle_{N_{\mathrm{elec}}} = \sum_{i\in\mathcal{I}_{\mathrm{occ}}} \left\langle \psi_i \middle| \hat{\mathcal{H}}_{\mathrm{core}}\,\psi_i \right\rangle_1 + \frac{1}{2}\sum_{i\in\mathcal{I}_{\mathrm{occ}}}\sum_{j\in\mathcal{I}_{\mathrm{occ}}} \langle ij||ij\rangle\,. \tag{4.33}$$

Originating from the integral expression (4.31) both the ERI tensor as well as the antisymmetrised ERI tensor show a lot of symmetry with respect to index permutations. An overview of these symmetry properties provides appendix A on page 209.

4.4 Single-determinant ansatz

In the previous section we noted that even an approximate solution to the electronic Schrödinger equation (4.8) via the full CI ansatz is hardly feasible. Even if comparatively small one-electron basis sets $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}} \subset H^1(\mathbb{R}^3,\mathbb{C})$ are used, the dimensionality of the matrix $\mathbf{A}_{\text{FCI}}$ becomes simply too large. In this section we discuss the opposite end of the scale and only consider one-dimensional subspaces of the form domain

$$Q(\hat{\mathcal{H}}_{N_{\text{elec}}}) = H^1(\mathbb{R}^{3N_{\text{elec}}},\mathbb{C}) \cap \bigwedge^{N_{\text{elec}}} L^2(\mathbb{R}^3,\mathbb{C})$$

for solving the electronic problem. Formally by the Courant-Fischer theorem (3.3) the ground state electronic energy E_0 can be obtained by a variational minimisation over all subspaces of dimension 1. In other words

$$E_0 = \inf_{\Psi\in\mathcal{W}_{N_{\text{elec}}}} \left\langle \Psi \middle| \hat{\mathcal{H}}_{N_{\text{elec}}}\Psi \right\rangle_{N_{\text{elec}}}, \tag{4.34}$$

where

$$\mathcal{W}_{N_{\text{elec}}} = \left\{ \Psi \in Q(\hat{\mathcal{H}}_{N_{\text{elec}}}) \;\middle|\; \|\Psi\|_{L^2(\mathbb{R}^{3N_{\text{elec}}},\mathbb{C})} = 1 \right\}. \tag{4.35}$$

denotes the subspace of all normalised functions from $Q(\hat{\mathcal{H}}_{N_{\text{elec}}})$. If we restrict the search to only run over the space

$$\mathcal{R}^1_{N_{\text{elec}}} = \left\{ \bigwedge_{i=1}^{N_{\text{elec}}} \psi_i \;\middle|\; \psi_i \in H^1(\mathbb{R}^3,\mathbb{C}), \langle\psi_i|\psi_j\rangle_1 = \delta_{ij}, \quad \forall\, 1\leq i,j\leq N_{\text{elec}} \right\} \tag{4.36}$$

of all normalised Slater determinants, which is a proper subspace of $\mathcal{W}_{N_{\text{elec}}}$, we no longer yield the exact energy. According to corollary 3.4 on page 33 we merely obtain an upper bound

$$E_0 \leq E_0^{\text{HF}} = \inf_{\Phi\in\mathcal{R}^1_{N_{\text{elec}}}} \left\langle \Phi \middle| \hat{\mathcal{H}}_{N_{\text{elec}}}\Phi \right\rangle_{N_{\text{elec}}}. \tag{4.37}$$

The implied procedure, where an approximation to the electronic ground state is computed by minimising the sesquilinear form of $\hat{\mathcal{H}}_{N_{\text{elec}}}$ over the space spanned by all normalised Slater determinants, is the celebrated **Hartree-Fock** (HF) approximation [80]. The resulting minimal energy E_0^{HF} is the HF ground-state energy and the corresponding minimising determinant Φ_0 the HF ground state.

The HF approach is named both after Douglas Hartree and Vladimir Fock. Historically Hartree [90] proposed an ansatz for the electronic problem based on symmetric products of one-electron functions, which is algorithmically very similar to the procedure followed nowadays (see remark 5.1 on page 86). Slater [82] and Fock [80] then both noted the issue arising from the use of symmetric products in the context of modelling spin-$1/2$ particles like electrons (see section 4.2.1 on page 47) and subsequently Fock reformulated the procedure using Slater determinants.

Notice, that Φ_0 is — by construction — the best possible single Slater determinant to approximate the electronic ground state. Mathematically speaking the set of Slater determinants $\mathcal{R}^1_{N_{\text{elec}}}$ is exactly the set of all elements from $\mathcal{W}_{N_{\text{elec}}}$, which are of rank 1. For this reason one sometimes refers to the Hartree-Fock ground state Φ_0 as a **rank-1 approximation** to the exact electronic ground state.

Remark 4.11 (Molecular orbital formulation of HF)**.** Let

$$\Theta \equiv (\psi_1, \psi_2, \ldots, \psi_{N_\mathrm{elec}}) \in \left(H^1(\mathbb{R}^3, \mathbb{C})\right)^{N_\mathrm{elec}}$$

denote an arbitrary tuple of single-particle functions. It uniquely identifies a Slater determinant $\Phi_\Theta = \bigwedge_{i=1}^{N_\mathrm{elec}} \psi_i$. Inserting this ansatz into the energy expression (4.33) for a single determinant yields the **Hartree-Fock energy functional**

$$\begin{aligned}\mathcal{E}^\mathrm{HF}(\Theta) = \frac{1}{2}\sum_{i=1}^{N_\mathrm{elec}} \int_{\mathbb{R}^3} \|\nabla\psi_i\|_2^2 \,\mathrm{d}\underline{\boldsymbol{r}} - \int_{\mathbb{R}^3} \sum_{A=1}^{M} \frac{Z_A\,\rho_\Theta(\underline{\boldsymbol{r}})}{\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_A\|_2}\,\mathrm{d}\underline{\boldsymbol{r}} \\ + \frac{1}{2}\int_{\mathbb{R}^3}\int_{\mathbb{R}^3} \frac{\rho_\Theta(\underline{\boldsymbol{r}}_1)\rho_\Theta(\underline{\boldsymbol{r}}_2)}{\|\underline{\boldsymbol{r}}_1 - \underline{\boldsymbol{r}}_2\|_2}\,\mathrm{d}\underline{\boldsymbol{r}}_1\,\mathrm{d}\underline{\boldsymbol{r}}_2 - \frac{1}{2}\int_{\mathbb{R}^3}\int_{\mathbb{R}^3} \frac{|\gamma_\Theta(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2)|^2}{\|\underline{\boldsymbol{r}}_1 - \underline{\boldsymbol{r}}_2\|_2}\,\mathrm{d}\underline{\boldsymbol{r}}_1\,\mathrm{d}\underline{\boldsymbol{r}}_2,\end{aligned} \tag{4.38}$$

where

$$\rho_\Theta(\underline{\boldsymbol{r}}) = \sum_{i=1}^{N_\mathrm{elec}} |\psi_i(\underline{\boldsymbol{r}})|^2 \qquad \text{and} \qquad \gamma_\Theta(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2) = \sum_{i=1}^{N_\mathrm{elec}} \psi_i^*(\underline{\boldsymbol{r}}_1)\psi_i(\underline{\boldsymbol{r}}_2) \tag{4.39}$$

are the **electron density** and the **one-particle reduced density matrix**, respectively. The HF ansatz (4.36) thus becomes

$$E_0 \leq E_0^\mathrm{HF} = \inf\left\{\mathcal{E}^\mathrm{HF}(\Theta) \,\middle|\, \Theta \in \left(H^1(\mathbb{R}^3, \mathbb{C})\right)^{N_\mathrm{elec}}, \forall i,j\ \langle\psi_i|\psi_j\rangle_1 = \delta_{ij}\right\}. \tag{4.40}$$

The minimiser Θ^0, i.e. the tuple for which the minimum energy $E_0^\mathrm{HF} = \mathcal{E}^\mathrm{HF}(\Theta^0)$ is exactly obtained, defines the HF ground state $\Phi_0 \equiv \Phi_{\Theta^0}$.

Before we discuss the mathematical properties of the HF ansatz, let us first pick up on our discussion of spin in section 4.2.1 on page 47 and generalise the formalism.

Remark 4.12 (Spin-adapted formulation of HF)**.** The mathematical treatment up to this point only includes one property resulting from the spin-1/2 nature of electrons, namely the antisymmetry of the wave function. The missing property is the explicit inclusion of the spin degree of freedom. For a single spin-1/2 particle the spin degree of freedom spans the two-dimensional Hilbert space $\mathbb{C}^2$, which can be probed by the spin operator

$$\hat{\underline{\mathrm{s}}} \equiv (\hat{\mathrm{s}}_x, \hat{\mathrm{s}}_y, \hat{\mathrm{s}}_z) = \frac{1}{2}(\sigma_x, \sigma_y, \sigma_z).$$

In this expression we used the **Pauli matrices** defined as

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_y = \begin{pmatrix} 0 & -\imath \\ \imath & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{4.41}$$

The operator $\hat{\mathrm{s}}_z$ has two eigenstates

$$\uparrow \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad\qquad \downarrow \equiv \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

which are called **spin-up** and **spin-down**, respectively.

One way to incorporate spin into our present treatment is the **spinor** formalism, where a one-particle function is now written as a function of two **spin components**

$$\psi(\underline{\boldsymbol{r}}) \equiv \begin{pmatrix} \psi^\alpha(\underline{\boldsymbol{r}}) \\ \psi^\beta(\underline{\boldsymbol{r}}) \end{pmatrix}, \tag{4.42}$$

mapping each real coordinate $\underline{\boldsymbol{r}}$ to a complex spinor from $\mathbb{C}^2$. All results from functional analysis and spectral theory which we derived for the spin-free case can be adapted to the spinor formalism, simply moving from the function space $L^2(\mathbb{R}^3, \mathbb{C})$ (and its subspaces) to $L^2(\mathbb{R}^3, \mathbb{C}^2)$ (and equivalent subspaces). For example for two spin-adapted one-particle functions

$$\psi \equiv \begin{pmatrix} \psi^\alpha \\ \psi^\beta \end{pmatrix} \in L^2(\mathbb{R}^3, \mathbb{C}^2) \qquad \text{and} \qquad \varphi \equiv \begin{pmatrix} \varphi^\alpha \\ \varphi^\beta \end{pmatrix} \in L^2(\mathbb{R}^3, \mathbb{C}^2)$$

the one-particle inner product $\langle \cdot \,|\, \cdot \rangle$ becomes the $L^2(\mathbb{R}^3, \mathbb{C}^2)$ inner product

$$\langle \psi | \varphi \rangle_1 \equiv \int_{\mathbb{R}^3} \langle \psi(\underline{\boldsymbol{r}}) | \varphi(\underline{\boldsymbol{r}}) \rangle_2 \, \mathrm{d}\underline{\boldsymbol{r}} = \int_{\mathbb{R}^3} \left(\psi^\alpha(\underline{\boldsymbol{r}})\right)^* \varphi^\alpha(\underline{\boldsymbol{r}}) + \left(\psi^\beta(\underline{\boldsymbol{r}})\right)^* \varphi^\beta(\underline{\boldsymbol{r}}) \, \mathrm{d}\underline{\boldsymbol{r}}$$

in analogy to the spin-free case. In a similar fashion one may construct the exterior power $\bigwedge_{i=1}^{N_\mathrm{elec}} L^2(\mathbb{R}^3, \mathbb{C}^2)$ and Slater determinants $\bigwedge_{i=1}^{N_\mathrm{elec}} \psi_i$ from functions $\psi_i \in H^1(\mathbb{R}^3, \mathbb{C}^2)$ as well. Notice that the tensor product nature of the exterior power implies

$$\bigwedge_{i=1}^{N_\mathrm{elec}} L^2(\mathbb{R}^3, \mathbb{C}^2) \subset L^2(\mathbb{R}^{3N_\mathrm{elec}}, \mathbb{C}^{2N_\mathrm{elec}})$$

and

$$\bigwedge_{i=1}^{N_\mathrm{elec}} \psi_i \in \bigwedge_{i=1}^{N_\mathrm{elec}} H^1(\mathbb{R}^3, \mathbb{C}^2) \subset H^1(\mathbb{R}^{3N_\mathrm{elec}}, \mathbb{C}^{2N_\mathrm{elec}}). \tag{4.43}$$

In this sense the derived expressions from sections 4.1 and 4.3 can be brought forward to the spin-adapted case with only minor modifications. For example, the expression of the HF energy functional $\mathcal{E}^\mathrm{HF}(\Theta)$ can be used exactly as stated in (4.38) for a tuple

$$\Theta \equiv (\psi_1, \psi_2, \ldots, \psi_{N_\mathrm{elec}}) \in \left(H^1(\mathbb{R}^3, \mathbb{C}^2)\right)^{N_\mathrm{elec}}$$

of spin-adapted one-particle functions as well. We only need to understand the gradient

$$\nabla \psi_1 \equiv \begin{pmatrix} \nabla \psi_1^\alpha \\ \nabla \psi_1^\beta \end{pmatrix} \in \mathbb{C}^6$$

as a vector from $\mathbb{C}^6$ and define the density as

$$\rho_\Theta(\underline{\boldsymbol{r}}) = \sum_{i=1}^{N_\mathrm{elec}} \|\psi_i(\underline{\boldsymbol{r}})\|_2^2 = \sum_{i=1}^{N_\mathrm{elec}} |\psi_i^\alpha(\underline{\boldsymbol{r}})|^2 + \left|\psi_i^\beta(\underline{\boldsymbol{r}})\right|^2 \tag{4.44}$$

and the one-particle density matrix as

$$\gamma_\Theta(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2) = \sum_{i=1}^{N_\mathrm{elec}} \langle \psi_i(\underline{\boldsymbol{r}}_1) | \psi_i(\underline{\boldsymbol{r}}_2) \rangle_2 = \sum_{i=1}^{N_\mathrm{elec}} (\psi_i^\alpha)^*(\underline{\boldsymbol{r}}_1) \; \psi_i^\alpha(\underline{\boldsymbol{r}}_2) + \left(\psi_i^\beta\right)^*(\underline{\boldsymbol{r}}_1) \; \psi_i^\beta(\underline{\boldsymbol{r}}_2). \tag{4.45}$$

The extra spin component introduces another complication into the HF procedure. Analogously to the one-particle spin operator $\hat{\underline{\mathrm{s}}}$ and component operators $\hat{\mathrm{s}}_x, \hat{\mathrm{s}}_y, \hat{\mathrm{s}}_z$ for

a single particle, one can define the total spin operator $\hat{\underline{S}}$ as well as Cartesian spin components $\hat{\mathcal{S}}_x, \hat{\mathcal{S}}_y, \hat{\mathcal{S}}_z$ for an N_{elec}-electron system. Originating from $\hat{\underline{S}}$ one may define

$$\hat{\mathcal{S}}^2 = \hat{\underline{S}} \cdot \hat{\underline{S}} = \left(\hat{\mathcal{S}}_x^2, \hat{\mathcal{S}}_y^2, \hat{\mathcal{S}}_z^2\right).$$

One can show [41] that

$$\left[\hat{\mathcal{S}}^2, \hat{\mathcal{S}}_z\right] = 0,$$

i.e. that the total spin squared operator commutes with $\hat{\mathcal{S}}_z$. Since the Hamiltonian $\hat{\mathcal{H}}_{N_{\text{elec}}}$ makes no explicit reference to any particular spin component, we necessarily have

$$\left[\hat{\mathcal{S}}_z, \hat{\mathcal{H}}_{N_{\text{elec}}}\right] = \left[\hat{\mathcal{S}}^2, \hat{\mathcal{H}}_{N_{\text{elec}}}\right] = 0 \tag{4.46}$$

as well. This implies[11] that one is able to find simultaneous eigenfunctions of $\hat{\mathcal{H}}_{N_{\text{elec}}}$, $\hat{\mathcal{S}}_z$ and $\hat{\mathcal{S}}^2$. For many applications of electronic structure theory, e.g. the interpretation of certain spectroscopic results or for understanding the aforementioned Stern-Gerlach experiment, the determination of simultaneous eigenstates of these three operators at once is crucial. (4.46) naturally implies that it is possible to obtain the ground state or ground states of $\hat{\mathcal{H}}_{N_{\text{elec}}}$ in a way that they are eigenfunctions of the spin operators. There is, however, no guarantee that the HF ansatz (4.40) gives rise to a HF ground state Φ_{Θ^0}, which is an eigenfunction of $\hat{\mathcal{S}}_z$ or $\hat{\mathcal{S}}^2$ [91, 92]. In fact it is rather easy to construct Slater determinants, which are neither an eigenstate of $\hat{\mathcal{S}}_z$ nor of $\hat{\mathcal{S}}^2$.

There are two common approaches to deal with this issue [92]. One is to minimise according to (4.40) and then use appropriate projections in order to yield the required eigenstates with respect to $\hat{\mathcal{S}}_z$ and $\hat{\mathcal{S}}^2$. The other ansatz is to impose conditions on the ansatz space (4.43), such that the resulting HF ground states are eigenfunctions of $\hat{\mathcal{S}}_z$ and $\hat{\mathcal{S}}^2$. This is what we will discuss when we move to a discretised treatment of the HF ansatz in 4.15 on page 64.

Even though the Hartree-Fock ansatz was already proposed by Fock [80] in 1930, its fundamental mathematical properties were only rigorously characterised and proved by Lieb [93] and Lions [94, 95] in the 70s and 80s for the general spin-adapted case. These are summarised in the following.

Remark 4.13 (Invariance under orbital rotations)**.** Let

$$\Theta = (\psi_1, \psi_2, \ldots, \psi_{N_{\text{elec}}}) \in \left(H^1(\mathbb{R}^3, \mathbb{C}^2)\right)^{N_{\text{elec}}}$$

be a tuple, which satisfies the orthonormality condition

$$\forall i, j \in 1, \ldots, N_{\text{elec}} : \quad \langle \psi_i | \psi_j \rangle_1 = \delta_{ij}. \tag{4.47}$$

One can easily show [95], that for any unitary matrix $\mathbf{U} \in \mathbb{C}^{N_{\text{elec}} \times N_{\text{elec}}}$ it holds:

- $\Theta' = \Theta\mathbf{U}$ satisfies (4.47) as well.
- $\mathcal{E}^{\text{HF}}(\Theta\mathbf{U}) = \mathcal{E}(\mathbf{U})$

In other words all properties of HF can only be stated up to a unitary rotation amongst the constituents of the ground state Slater determinant Φ_0.

[11] Since $\hat{\mathcal{A}}\hat{\mathcal{B}}f = \hat{\mathcal{B}}\hat{\mathcal{A}}f \Leftrightarrow \left[\hat{\mathcal{A}}, \hat{\mathcal{B}}\right] = 0$.

Theorem 4.14 (Mathematical properties of HF)**.** *Assume $N_{elec} \leq \sum_{A=1}^{M} Z_A$, i.e. a neutral or positively charged chemical system.*

(a) A minimiser

$$\Theta^0 = \left(\psi_1^0, \psi_2^0, \ldots, \psi_{N_{elec}}^0\right) \in \left(H^1(\mathbb{R}^3, \mathbb{C}^2)\right)^{N_{elec}} \tag{4.48}$$

to $\mathcal{E}^{HF}$ exists [93], i.e. the HF model (4.40) has a ground state.

(b) Let us define the ***Fock operator***

$$\hat{\mathcal{F}}_{\Theta^0} = \hat{\mathcal{T}} + \hat{\mathcal{V}}_0 + \hat{\mathcal{J}}_{\Theta^0} + \hat{\mathcal{K}}_{\Theta^0} \tag{4.49}$$

consisting of the kinetic energy operator $\hat{\mathcal{T}}$ and the nuclear attraction operator $\hat{\mathcal{V}}_0$ as defined in (4.29) as well as the effective ***Coulomb operator***

$$\hat{\mathcal{J}}_{\Theta^0} = \int_{\mathbb{R}^3} \frac{\rho_{\Theta^0}(\underline{\boldsymbol{r}}_2)}{\|\cdot - \underline{\boldsymbol{r}}_2\|_2} \, \mathrm{d}\underline{\boldsymbol{r}}_2 \tag{4.50}$$

and the ***exchange operator****, implicitly defined by*

$$\left(\hat{\mathcal{K}}_{\Theta^0}\chi\right)(\underline{\boldsymbol{r}}) = -\int_{\mathbb{R}^3} \frac{\gamma_{\Theta^0}(\underline{\boldsymbol{r}}, \underline{\boldsymbol{r}}_2)}{\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{r}}_2\|_2} \chi(\underline{\boldsymbol{r}}_2) \, \mathrm{d}\underline{\boldsymbol{r}}_2. \tag{4.51}$$

To find a stationary point of $\mathcal{E}^{HF}$ with respect to Θ one needs to solve the Euler-Lagrange equations corresponding to the minimisation problem (4.40). By their means one finds that Θ^0 as defined in (4.48) can only be a stationary point of $\mathcal{E}^{HF}$ if there exists a Hermitian matrix $\lambda \in \mathbb{C}^{N_{elec} \times N_{elec}}$, such that for all $i, j \in \{1, \ldots, N_{elec}\}$

$$\hat{\mathcal{F}}_{\Theta^0}\psi_i^0 = \sum_i \lambda_{ij}\psi_j^0 \qquad \text{and} \qquad \left\langle \psi_i^0 \middle| \psi_j^0 \right\rangle = \delta_{ij} \tag{4.52}$$

hold. This is a necessary condition for Θ^0 to be a minimiser.

Once we found the ground state, the application of the Fock operator will thus only rotate us around the space spanned by the minimising functions from Θ^0.

(c) Due to the elliptic regularity theorem [93]

$$\psi_i^0 \in H^2(\mathbb{R}^3, \mathbb{C}^2) \cap C^\infty(\mathbb{R}^3 \backslash \{\underline{\boldsymbol{R}}_A\}_{A=1,\ldots,M}, \mathbb{C}^2),$$

which implies that a solution to (4.52) will always be smooth everywhere but the nuclei and globally in $H^2(\mathbb{R}^3, \mathbb{C}^2)$.

(d) The Fock operator $\hat{\mathcal{F}}_{\Theta^0}$ as defined in (4.49) is a self-adjoint operator on $L^2(\mathbb{R}^3, \mathbb{C}^2)$ with domain $D(\hat{\mathcal{F}}_{\Theta^0}) = H^2(\mathbb{R}^3, \mathbb{C}^2)$ and form domain $Q(\hat{\mathcal{F}}_{\Theta^0}) = H^1(\mathbb{R}^3, \mathbb{C}^2)$ [94]. It is bounded below and $\sigma_{ess} = [0, +\infty)$.

(e) Up to replacing Θ^0 by $\Theta^0\mathbf{U}$ for some unitary matrix $\mathbf{U} \in \mathbb{C}^{N_{elec} \times N_{elec}}$, the canonical ***Hartree-Fock equations*** *hold*

$$\hat{\mathcal{F}}_{\Theta^0}\psi_i^0 = \varepsilon_i\psi_i^0 \qquad \text{and} \qquad \left\langle \psi_i^0 \middle| \psi_j^0 \right\rangle = \delta_{ij} \tag{4.53}$$

with

$$\varepsilon_1 \leq \varepsilon_2 \leq \cdots \leq \varepsilon_{N_{elec}} < 0.$$

*(f) The **Aufbau principle** is satisfied: The $\{\varepsilon_1, \ldots, \varepsilon_{N_{elec}}\}$ are the lowest N_{elec} eigenvalues of $\hat{\mathcal{F}}_{\Theta^0}$.*

*(g) Let $\varepsilon_{N_{elec}+1}$ be the $(N_{elec}+1)$-th eigenvalue of $\hat{\mathcal{F}}_{\Theta^0}$ if the Fock operator has $(N_{elec}+1)$ negative eigenvalue (counting multiplicities) otherwise set $\varepsilon_{N_{elec}+1} = 0$. The **no unfilled-shell** property*

$$\varepsilon_{N_{elec}} < \varepsilon_{N_{elec}+1} \leq 0$$

is satisfied [96].

The proofs for these results in the general setting are rather involved and can be found in the cited works. Theorem 4.14 provides the mathematical justification for the HF procedure as it is performed in almost every quantum-chemistry program these days for neutral or positively charged system. I am unaware of a similar mathematical statement making a guarantee that the HF procedure gives a sensible ground state for negatively charged systems. In fact one can even show that no solution to the HF problem (4.40) exists for negative ions with $N_{\mathrm{elec}} > 2Z_{\mathrm{tot}} + M$ [97]. This holds for example for H^{2-}. To the best of my knowledge there is furthermore no uniqueness proof for the solution (4.48) in the infinite-dimensional setting up to today, not even for the resulting ground state density.

4.4.1 Discretised Hartree-Fock

A central result of theorem 4.14 is that the HF ansatz (4.40) can be seen as a variational problem towards finding the best molecular orbitals for a single Slater determinant, which at the optimal point reduces to the spectral problem of the Fock operator $\hat{\mathcal{F}}$. This section deals with the discretisation of both representations of HF.

Remark 4.15 (Discretised HF variational problem)**.** Assume a one-particle basis $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\mathrm{bas}}}$ consisting of $N_{\mathrm{bas}} = |\mathcal{I}_{\mathrm{bas}}|$ basis functions taken from a dense subspace of $H^1(\mathbb{R}^3, \mathbb{C})$. The space

$$\mathcal{S}_1 = \mathrm{span}\left\{ \begin{pmatrix} \varphi_\mu \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \varphi_\mu \end{pmatrix} \middle| \, \mu \in \mathcal{I}_{\mathrm{bas}} \right\}$$

spanned by spin-adapted linear combinations from $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\mathrm{bas}}}$ is a subspace of $H^1(\mathbb{R}^3, \mathbb{C}^2)$. Consequently

$$\left\{ \bigwedge_{i=1}^{N_{\mathrm{elec}}} \psi_i \middle| \, \psi_i \in \mathcal{S}_1, \langle\psi_i|\psi_j\rangle_1 = \delta_{ij} \right\} \subset \mathcal{R}^1_{N_{\mathrm{elec}}},$$

which implies

$$E_0 \leq E_0^{\mathrm{HF}} \leq \tilde{E}_0^{\mathrm{HF}} = \inf\left\{ \mathcal{E}^{\mathrm{HF}}(\Theta) \,\middle|\, \Theta = (\psi_1, \ldots, \psi_{N_{\mathrm{elec}}}) \in (\mathcal{S}_1)^{N_{\mathrm{elec}}}, \langle\psi_i|\psi_j\rangle_1 = \delta_{ij} \right\}. \tag{4.54}$$

Let us denote with

$$\boldsymbol{\varphi} = \begin{pmatrix} \varphi_1 & \varphi_2 & \cdots & \varphi_{N_{\mathrm{bas}}} \end{pmatrix}$$

the row vector of all basis functions written one after another. Then each element ψ_i of the tuple

$$\Theta = (\psi_1, \ldots, \psi_{N_{\mathrm{elec}}}) \in (\mathcal{S}_1)^{N_{\mathrm{elec}}} \tag{4.55}$$

can be expanded as

$$\psi_i = \begin{pmatrix} \psi_i^\alpha \\ \psi_i^\beta \end{pmatrix} = \sum_{\mu \in \mathcal{I}_{\text{bas}}} \varphi_\mu \begin{pmatrix} \tilde{C}_{\mu i}^\alpha \\ \tilde{C}_{\mu i}^\beta \end{pmatrix} = \underbrace{\begin{pmatrix} \boldsymbol{\varphi} & 0 \\ 0 & \boldsymbol{\varphi} \end{pmatrix}}_{\substack{\text{2 rows,} \\ 2N_{\text{bas}} \text{ columns}}} \begin{pmatrix} \tilde{C}_{1i}^\alpha \\ \vdots \\ \tilde{C}_{N_{\text{bas}},i}^\alpha \\ \tilde{C}_{1i}^\beta \\ \vdots \\ \tilde{C}_{N_{\text{bas}},i}^\beta \end{pmatrix}. \tag{4.56}$$

This allows to write

$$\Theta = \begin{pmatrix} \boldsymbol{\varphi} & 0 \\ 0 & \boldsymbol{\varphi} \end{pmatrix} \mathbf{C} \qquad \text{where} \qquad \mathbf{C} = \begin{pmatrix} \tilde{\mathbf{C}}^\alpha \\ \tilde{\mathbf{C}}^\beta \end{pmatrix} \in \mathbb{C}^{2N_{\text{bas}} \times N_{\text{elec}}} \tag{4.57}$$

is the **coefficient matrix**, which is built by pasting the row matrices at the right hand side of (4.56) one after another.

It is not hard to imagine that one could insert (4.57) into the expression for the HF energy functional $\mathcal{E}^{\text{HF}}$ in order to obtain an expression of the HF energy in terms of $\mathbf{C}$. This expression could be minimised with respect to the coefficients $\mathbf{C}$ in order to obtain an approximate HF ground-state energy and a corresponding approximate HF ground state. Even though this could be done such a **generalised unrestricted Hartree-Fock** (GUHF) procedure is hardly ever performed in practice [92]. The reason is that it suffers exactly from the issues raised at the end of remark 4.12 on page 60, namely that the resulting HF ground state is neither an eigenfunction of $\hat{\mathcal{S}}^2$ nor $\hat{\mathcal{S}}_z$.

Instead one typically selects a target eigenvalue M_S of the projected spin operator $\hat{\mathcal{S}}_z$ *before* performing the HF procedure. From this value M_S as well as the number of electrons N_{elec} one determines two parameters N_{elec}^α and N_{elec}^β, the number of spin-up and the number of spin-down electrons, such that

$$N_{\text{elec}}^\alpha + N_{\text{elec}}^\beta = N_{\text{elec}}, \qquad M_S = \frac{1}{2}\left(N_{\text{elec}}^\alpha - N_{\text{elec}}^\beta\right).$$

One can show [83, 84], that any Slater determinant Φ_Θ made from a tuple like (4.55) is an eigenfunction of $\hat{\mathcal{S}}_z$ with eigenvalue M_S if it consists of N_{elec}^α single-particle functions with zero β component and N_{elec}^β single-particle functions with zero α component. Invoking remark 4.13 on page 62 we can always reorder the single-particle functions such that the N_{elec}^α functions with zero β component are first and the other functions with zero α-component follow thereafter or

$$\psi_i^\beta = 0 \quad \forall i \in \{1, \ldots, N_{\text{elec}}^\alpha\} \qquad \text{and} \qquad \psi_i^\alpha = 0 \quad \forall i \in \{N_{\text{elec}}^\alpha + 1, \ldots, N_{\text{elec}}\}.$$

Applying these conditions to the generalised unrestricted Hartree-Fock ansatz (4.57) leads to the **unrestricted Hartree-Fock** (UHF) method[12] [98]. In UHF the coefficient matrix $\mathbf{C}$ of (4.57) becomes block-diagonal

$$\mathbf{C} = \begin{pmatrix} \mathbf{C}^\alpha & 0 \\ 0 & \mathbf{C}^\beta \end{pmatrix} \in \mathbb{C}^{2N_{\text{bas}} \times N_{\text{elec}}}, \tag{4.58}$$

[12] Beware that even though the UHF ansatz is termed *unrestricted* it implies a restriction of the search space due to spin symmetry. This naming is an unfortunate historic consequence.

where $\mathbf{C}^\alpha \in \mathbb{C}^{N_{\text{bas}} \times N^\alpha_{\text{elec}}}$ and $\mathbf{C}^\beta \in \mathbb{C}^{N_{\text{bas}} \times N^\beta_{\text{elec}}}$ are the spin-up and spin-down occupied coefficient matrices, respectively. Inserting (4.57) and (4.58) into (4.38) we obtain the HF energy functional in terms of of the coefficient matrix $\mathbf{C}$

$$\mathcal{E}^{\text{HF}}_C(\mathbf{C}) = \operatorname{tr}\left(\mathbf{C}^\dagger \left(\mathbf{T} + \mathbf{V}_0\right) \mathbf{C}\right) + \frac{1}{2} \operatorname{tr}\left(\mathbf{C}^\dagger \left(\mathbf{J}\big[\mathbf{C}\mathbf{C}^\dagger\big] + \mathbf{K}\big[\mathbf{C}\mathbf{C}^\dagger\big]\right) \mathbf{C}\right), \tag{4.59}$$

where all involved matrices are α-β block-diagonal, just like the coefficient matrix (4.58). Furthermore

- the **kinetic energy matrix T** has identical α and β blocks with elements

$$T^\alpha_{\mu\nu} = T^\beta_{\mu\nu} = \frac{1}{2} \int_{\mathbb{R}^3} (\nabla \varphi_\mu)^* \cdot \nabla \varphi_\nu \, \mathrm{d}\underline{\boldsymbol{r}}. \tag{4.60}$$

- the **nuclear attraction matrix $\mathbf{V}_0$** has identical α and β blocks of elements

$$(V_0^\alpha)_{\mu\nu} = \left(V_0^\beta\right)_{\mu\nu} = -\int_{\mathbb{R}^3} \sum_{A=1}^{M} Z_A \frac{\varphi_\mu(\underline{\boldsymbol{r}})^* \varphi_\nu(\underline{\boldsymbol{r}})}{\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_A\|_2} \, \mathrm{d}\underline{\boldsymbol{r}}. \tag{4.61}$$

- the **Coulomb matrix $\mathbf{J}\big[\mathbf{C}\mathbf{C}^\dagger\big]$** depends explicitly on the coefficient matrix $\mathbf{C}$ as expressed by the term in the square brackets. It has an identical α and β block with elements

$$J^\alpha_{\mu\nu}\big[\mathbf{C}\mathbf{C}^\dagger\big] = J^\beta_{\mu\nu}\big[\mathbf{C}\mathbf{C}^\dagger\big] = \sum_{\lambda,\kappa \in \mathcal{I}_{\text{bas}}} \sum_{\sigma \in \{\alpha\beta\}} \sum_{i=1}^{N^\sigma_{\text{elec}}} C^\sigma_{\lambda i} \left(C^\sigma_{\kappa i}\right)^* (\varphi_\mu \varphi_\nu | \varphi_\kappa \varphi_\lambda). \tag{4.62}$$

  Here as usual $(\cdot\,\cdot|\cdot\,\cdot)$ denotes the electron-repulsion integrals as defined in (4.31).

- the **exchange matrix $\mathbf{K}\big[\mathbf{C}\mathbf{C}^\dagger\big]$** has *deviating* α and β blocks, both depending on the coefficients. For $\sigma \in \{\alpha\beta\}$ their elements are

$$K^\sigma_{\mu\nu}\big[\mathbf{C}\mathbf{C}^\dagger\big] = -\sum_{\lambda,\kappa \in \mathcal{I}_{\text{bas}}} \sum_{i=1}^{N^\sigma_{\text{elec}}} C^\sigma_{\lambda i} \left(C^\sigma_{\kappa i}\right)^* (\varphi_\kappa \varphi_\nu | \varphi_\mu \varphi_\lambda). \tag{4.63}$$

Let us further define a block-diagonal **overlap matrix**

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}^\alpha & 0 \\ 0 & \mathbf{S}^\beta \end{pmatrix}$$

with elements

$$S^\alpha_{\mu\nu} = S^\beta_{\mu\nu} = \int_{\mathbb{R}^3} \varphi^*_\mu(\underline{\boldsymbol{r}}) \varphi_\nu(\underline{\boldsymbol{r}}) \, \mathrm{d}\underline{\boldsymbol{r}}. \tag{4.64}$$

Altogether definitions (4.59) to (4.64) allow to rewrite (4.54) as an optimisation problem with respect to the coefficients $\mathbf{C}$

$$\tilde{E}^{\text{HF}}_0 = \inf\left\{\mathcal{E}^{\text{HF}}_C(\mathbf{C}) \,\middle|\, \mathbf{C} \in \mathcal{C}\right\} \tag{4.65}$$

where

$$\mathcal{C} = \left\{ \begin{pmatrix} \mathbf{C}^\alpha & 0 \\ 0 & \mathbf{C}^\beta \end{pmatrix} \,\middle|\, \mathbf{C}^\alpha \in \mathbb{C}^{N_{\text{bas}} \times N^\alpha_{\text{elec}}}, \, \mathbf{C}^\beta \in \mathbb{C}^{N_{\text{bas}} \times N^\beta_{\text{elec}}}, \, \mathbf{C}^\dagger \mathbf{S} \mathbf{C} = \mathbf{I}_{N_{\text{elec}}} \right\}. \tag{4.66}$$

One can show [97] in the discrete case, that the minimiser $\mathbf{C}_0$ is even unique up to unitary rotations, i.e. up to multiplications with unitary matrices $\mathbf{U}^\alpha \in \mathbb{C}^{N^\alpha_{\text{elec}} \times N^\alpha_{\text{elec}}}$, $\mathbf{U}^\beta \in \mathbb{C}^{N^\beta_{\text{elec}} \times N^\beta_{\text{elec}}}$ with

$$\begin{pmatrix} \mathbf{C}^\alpha & 0 \\ 0 & \mathbf{C}^\beta \end{pmatrix} \to \begin{pmatrix} \mathbf{C}^\alpha \mathbf{U}^\alpha & 0 \\ 0 & \mathbf{C}^\beta \mathbf{U}^\beta \end{pmatrix}$$

In many cases an alternative formulation of (4.65) in terms of the **density matrix**

$$\mathbf{D} = \mathbf{C}\mathbf{C}^\dagger \in \mathbb{C}^{2N_{\text{bas}} \times 2N_{\text{bas}}} \quad \text{where} \quad \mathbf{D}^\alpha = \mathbf{C}^\alpha \left(\mathbf{C}^\alpha\right)^\dagger, \quad \mathbf{D}^\beta = \mathbf{C}^\beta \left(\mathbf{C}^\beta\right)^\dagger, \tag{4.67}$$

is desirable. The coefficient matrices from $\mathcal{C}$ span

$$\begin{aligned} \mathcal{P} = \Bigg\{ \begin{pmatrix} \mathbf{D}^\alpha & 0 \\ 0 & \mathbf{D}^\beta \end{pmatrix} \Bigg| \, \forall \sigma \in \{\alpha, \beta\} \; \mathbf{D}^\sigma \in \mathbb{C}^{N_{\text{bas}} \times N_{\text{bas}}}, \\ \operatorname{tr}\left(\mathbf{S}^\sigma \mathbf{D}^\sigma\right) = N^\sigma_{\text{elec}}, \, \mathbf{D}^\sigma \mathbf{S}^\sigma \mathbf{D}^\sigma = \mathbf{D}^\sigma \Bigg\}. \end{aligned} \tag{4.68}$$

With these definitions we can recast (4.65) as

$$\tilde{E}_0^{\text{HF}} = \inf \left\{ \mathcal{E}_D^{\text{HF}}(\mathbf{D}) \,\middle|\, \mathbf{D} \in \mathcal{P} \right\}, \tag{4.69}$$

where the energy functional in terms of the density matrix is

$$\mathcal{E}_D^{\text{HF}}(\mathbf{D}) = \operatorname{tr}\big(\left(\mathbf{T} + \mathbf{V}_0\right) \mathbf{D} \big) + \frac{1}{2} \operatorname{tr}\big(\left(\mathbf{J}[\mathbf{D}] + \mathbf{K}[\mathbf{D}]\right) \mathbf{D} \big). \tag{4.70}$$

The respective expressions for $\mathbf{J}[\mathbf{D}]$ and $\mathbf{K}[\mathbf{D}]$ can be obtained from (4.62) and (4.63) by replacing

$$\mathbf{C}\mathbf{C}^\dagger \to \mathbf{D} \qquad \text{and} \qquad \sum_{i=1}^{N^\sigma_{\text{elec}}} C^\sigma_{\lambda i} \left(C^\sigma_{\kappa i}\right)^* \to D^\sigma_{\lambda\kappa}. \tag{4.71}$$

Notice, that the first trace term of $\mathcal{E}_D^{\text{HF}}(\mathbf{D})$ is linear in the density matrix, whereas the second trace term is quadratic in the density matrix. Again the minimiser $\mathbf{D}_0$ of (4.69) is unique [97] if the Aufbau principle ordering of orbital energies is chosen when building $\mathbf{D}$ from $\mathbf{C}$.

Remark 4.16. All matrices arising from a discretisation of the HF ansatz (4.40) in the sense of UHF give rise to block-diagonal matrices, with the α-block describing the spin-up component and the β-block describing the spin-down component. Apart from the exchange matrix $\mathbf{K}$, the density matrix $\mathbf{D}$ as well as the coefficient matrix $\mathbf{C}$, all matrices arising in remark 4.15 have identical entries in both blocks. Inside the HF energy functional it is the exchange term $\operatorname{tr}\left(\mathbf{C}^\dagger \mathbf{K} \mathbf{C}\right)$ where the α and β block lead to non-symmetrical energy contributions. In a minimisation it is thus this term, which distinguishes spin-up and spin-down electrons and makes them become subject to deviating physics. In other words this term gives rise to the non-classical effects inside the Hartree-Fock approximation.

The UHF procedure automatically assures that the minimiser $\mathbf{C}_0$ gives rise to a Slater determinant, which is an eigenfunctions of $\hat{\mathcal{S}}_z$. It is not assured, however, that

it is an eigenfunction of $\hat{\mathcal{S}}^2$. In fact one can show [83], that the value obtained for the total spin squared for the discretised HF ground state Φ_0 is

$$S^2 = \left\langle \Phi_0 \middle| \hat{\mathcal{S}}^2 \Phi_0 \right\rangle = S^2_{\text{exact}} + N^\beta_{\text{elec}} - \left\| (\mathbf{C}^\alpha)^\dagger \, \mathbf{S}^\beta \mathbf{C}^\beta \right\|^2_{\text{frob}} \tag{4.72}$$

where $\|\cdot\|_{\text{frob}}$ is the Frobenius norm defined as

$$\|\mathbf{M}\|_{\text{frob}} \equiv \sqrt{\sum_{i=1}^{N_{\text{bas}}} \sum_{j=1}^{N_{\text{bas}}} |M_{ij}|^2} \tag{4.73}$$

and

$$S^2_{\text{exact}} = \left(\frac{N^\alpha_{\text{elec}} - N^\beta_{\text{elec}}}{2} \right) \left(\frac{N^\alpha_{\text{elec}} - N^\beta_{\text{elec}}}{2} + 1 \right).$$

The observed deviation typically becomes larger if the basis gets larger.

The mathematical structure of the minimisation problems (4.65) and (4.69) are comparatively complex. One reason for this is that the spaces $\mathcal{C}$ and $\mathcal{P}$, spanned by the coefficient or the density matrix parameters sets, are not vector spaces. Much rather they are **manifolds**, i.e. geometrical objects which locally look like vector spaces, but globally show less structure. More precisely $\mathcal{C}$ is a subset of a Stiefel manifold and $\mathcal{P}$ is a subset of a Grassmann manifold. This aspect becomes apparent when designing rigorous algorithms for solving the HF problem since the topological properties of the HF parameter spaces imply that intuitive approaches to the problem may not always work.

Remark 4.17 (Discretised HF equations)**.** Theorem 4.14 on page 63 allows to recast the HF ansatz (4.40) into an equivalent spectral problem (4.53) for the Fock operator $\hat{\mathcal{F}}$ at the minimal point. It guarantees further that $\hat{\mathcal{F}}$ shows the spectral requirements for applying the Ritz-Galerkin ansatz of remark 3.6 on page 34, namely that it is self-adjoint and shows a discrete spectrum below the essential spectrum. Choosing the same basis $\{\varphi_\mu\}_{\mu \in \mathcal{I}_{\text{bas}}}$ as in remark 4.15 and projecting problem (4.53) onto this basis yields the discretised HF equations

$$\forall i, j \in \{1, \ldots, N_{\text{elec}}\} \quad \begin{cases} \mathbf{F}\big[\mathbf{C}\mathbf{C}^\dagger\big] \, \underline{\boldsymbol{c}}_i = \varepsilon_i \mathbf{S} \underline{\boldsymbol{c}}_i \\ \underline{\boldsymbol{c}}_i^\dagger \underline{\boldsymbol{c}}_j = \delta_{ij} \end{cases}, \tag{4.74}$$

where the elements of the **Fock matrix** $\mathbf{F}\big[\mathbf{C}\mathbf{C}^\dagger\big]$ are computed by applying the sesquilinear form

$$\begin{aligned} a_\Theta(\phi, \chi) &= \int_{\mathbb{R}^3} \left\langle \phi(\underline{\boldsymbol{r}}) \middle| \left(\hat{\mathcal{F}}_\Theta \chi \right)(\underline{\boldsymbol{r}}) \right\rangle_2 \mathrm{d}\underline{\boldsymbol{r}} \\ &= \int_{\mathbb{R}^3} \left(\phi^\alpha(\underline{\boldsymbol{r}}) \right)^* \left(\hat{\mathcal{F}}_\Theta \chi \right)^\alpha(\underline{\boldsymbol{r}}) + \left(\phi^\beta(\underline{\boldsymbol{r}}) \right)^* \left(\hat{\mathcal{F}}_\Theta \chi \right)^\beta(\underline{\boldsymbol{r}}) \, \mathrm{d}\underline{\boldsymbol{r}} \end{aligned} \tag{4.75}$$

to all pairs of basis spinors (ϕ, χ) with

$$\phi, \chi \in \left\{ \begin{pmatrix} \varphi \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \varphi \end{pmatrix} \middle| \, \varphi \in \{\varphi_\mu\}_{\mu \in \mathcal{I}_{\text{bas}}} \right\}.$$

This matrix is Hermitian and block-diagonal[13]

$$\mathbf{F} = \begin{pmatrix} \mathbf{F}^\alpha & 0 \\ 0 & \mathbf{F}^\beta \end{pmatrix}$$

and can be written as

$$\mathbf{F}\big[\mathbf{C}\mathbf{C}^\dagger\big] = \mathbf{T} + \mathbf{V}_0 + \mathbf{J}\big[\mathbf{C}\mathbf{C}^\dagger\big] + \mathbf{K}\big[\mathbf{C}\mathbf{C}^\dagger\big] \in \mathbb{C}^{2N_{\text{bas}} \times 2N_{\text{bas}}}, \tag{4.76}$$

where the matrix terms are defined as (4.60) to (4.63).

The generalised eigenvalue problem (4.74) can be solved for up $N_{\text{orb}} \leq N_{\text{bas}}$ eigenpairs $(\varepsilon_i, \underline{\boldsymbol{c}}_i)$ using one of the algorithms described in section 3.2 on page 36 incorporating the modifications discussed in section 3.2.7. Let us assume the usual ordering

$$\varepsilon_1 \leq \varepsilon_2 \leq \cdots \leq \varepsilon_{N_{\text{elec}}}.$$

By the Aufbau principle of theorem 4.14 the coefficient matrix is

$$\mathbf{C} = \begin{pmatrix} \underline{\boldsymbol{c}}_1 & \underline{\boldsymbol{c}}_2 & \cdots & \underline{\boldsymbol{c}}_{N_{\text{elec}}} \end{pmatrix},$$

i.e. the first N_{elec} eigenvectors pasted together. In analogy we define a full coefficient matrix

$$\mathbf{C}_F = \begin{pmatrix} \underline{\boldsymbol{c}}_1 & \underline{\boldsymbol{c}}_2 & \cdots & \underline{\boldsymbol{c}}_{N_{\text{elec}}} & \cdots & \underline{\boldsymbol{c}}_{N_{\text{orb}}} \end{pmatrix}, \tag{4.77}$$

which contains all N_{orb} eigenvectors we solved (4.74) for.

Applying the spin restrictions of unrestricted HF exactly as in remark 4.15, we know that that we expect N_{elec}^α eigenvectors with only α components and N_{elec}^β eigenvectors with only β components. Let us take N_{orb} to be even and such that $N_{\text{elec}}^\alpha, N_{\text{elec}}^\beta > N_{\text{orb}}/2$. Since both $\mathbf{F}$ and $\mathbf{S}$ are block-diagonal we can solve (4.74) block-wise, i.e. we solve

$$\mathbf{F}^\sigma\big[\mathbf{C}\mathbf{C}^\dagger\big]\, \underline{\boldsymbol{c}}_i^\sigma = \varepsilon_i \mathbf{S}^\alpha \underline{\boldsymbol{c}}_i^\sigma$$

for $\sigma = \alpha$ and $\sigma = \beta$ separately for $N_{\text{orb}}/2$ eigenpairs each. In analogy to (4.77) we proceed to define

$$\mathbf{C}_F^\sigma = \begin{pmatrix} \underline{\boldsymbol{c}}_1^\sigma & \underline{\boldsymbol{c}}_2^\sigma & \cdots & \underline{\boldsymbol{c}}_{N_{\text{orb}}}^\sigma \end{pmatrix}$$

and consequently get a block-diagonal full coefficients matrix

$$\mathbf{C}_F = \begin{pmatrix} \mathbf{C}_F^\alpha & 0 \\ 0 & \mathbf{C}_F^\beta \end{pmatrix} \in \mathbb{C}^{2N_{\text{bas}} \times N_{\text{orb}}}.$$

To adhere with the restriction to N_{elec}^α spin-up and N_{elec}^β spin-down orbitals the **occupied coefficient matrix** $\mathbf{C}^\alpha$ is obtained as the first N_{elec}^α columns of $\mathbf{C}_F^\alpha$ and likewise $\mathbf{C}^\beta$ as the first N_{elec}^β columns of $\mathbf{C}_F^\beta$.

The eigenfunctions $\underline{\boldsymbol{c}}_i$ of $\mathbf{F}$ are called **HF orbitals** or **SCF orbitals**[14]. Those orbitals, which are part of $\mathbf{C}$ according to the Aufbau principle, are called **occupied orbitals** as they are in some sense occupied by electrons. Conversely all other orbitals

[13] This is true even in the case of generalised unrestricted Hartree-Fock.
[14] The term SCF will defined in the next remark.

are called unoccupied or **virtual orbitals**. For later convenience let us define the index sets

$$\begin{aligned}
&\mathcal{I}_{\mathrm{orb}} \equiv \{1, \dots, N_{\mathrm{orb}}\} \\
&\mathcal{I}^{\alpha}_{\mathrm{occ}} \equiv \{1, \dots, N^{\alpha}_{\mathrm{elec}}\} && \mathcal{I}^{\alpha}_{\mathrm{virt}} \equiv \{N^{\alpha}_{\mathrm{elec}} + 1, \dots, N_{\mathrm{orb}}/2\} \\
&\mathcal{I}^{\beta}_{\mathrm{occ}} \equiv \{N_{\mathrm{orb}}/2 + 1, \dots, N_{\mathrm{orb}}\} && \mathcal{I}^{\beta}_{\mathrm{virt}} \equiv \{N^{\alpha}_{\mathrm{elec}} + N_{\mathrm{orb}}/2 + 1, \dots, N_{\mathrm{orb}}\} \\
&\mathcal{I}_{\mathrm{occ}} \equiv \mathcal{I}^{\alpha}_{\mathrm{occ}} \cup \mathcal{I}^{\beta}_{\mathrm{occ}} && \mathcal{I}_{\mathrm{virt}} \equiv \mathcal{I}_{\mathrm{orb}} \backslash \mathcal{I}_{\mathrm{occ}},
\end{aligned}$$

whose meaning should be self-explanatory.

Remark 4.18 (Properties of the discretised HF ansatz)**.** By theorem 4.14 on page 63 the minimiser $\mathbf{C}_0 \in \mathcal{C}$ of (4.66) satisfies exactly the discretised HF equations (4.74) such that

$$\mathbf{F}\Big[\mathbf{C}_0\mathbf{C}_0^\dagger\Big]\,\mathbf{C}_0 = \mathbf{S}\mathbf{C}_0\mathbf{E}$$

where

$$\mathbf{E} = \mathrm{diag}(\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{N_{\mathrm{elec}}}) \in \mathbb{R}^{N_{\mathrm{elec}} \times N_{\mathrm{elec}}}.$$

This condition can be equivalently expressed as [99]

$$\mathbf{F}\Big[\mathbf{C}_0\mathbf{C}_0^\dagger\Big]\,\mathbf{C}_0\mathbf{C}_0^\dagger\,\mathbf{S} - \mathbf{S}\,\mathbf{C}_0\mathbf{C}_0^\dagger\,\mathbf{F}\Big[\mathbf{C}_0\mathbf{C}_0^\dagger\Big] = \mathbf{0} \tag{4.78}$$

and is always satisfied if $\mathbf{C}_0$ is a minimiser. The reverse statement is not true, however, since all stationary points of the energy functional $\mathcal{E}_C$ optimised on the manifold $\mathcal{C}$ satisfy (4.78).

Nevertheless, nothing stops us to pick any other $\mathbf{C}^{(n)} \in \mathcal{C}$ and build the Fock matrix $\mathbf{F}\Big[\mathbf{C}^{(n)}\big(\mathbf{C}^{(n)}\big)^\dagger\Big]$ according to (4.76). We can solve for its eigenpairs, i.e. find a matrix $\mathbf{C}_F^{(n+1)}$ of eigenvectors and corresponding eigenvalues

$$\mathbf{E}^{(n+1)} = \mathrm{diag}\left(\varepsilon_1^{(n+1)}, \varepsilon_2^{(n+1)}, \dots, \varepsilon_{N_{\mathrm{orb}}}^{(n+1)}\right) \in \mathbb{R}^{N_{\mathrm{orb}} \times N_{\mathrm{orb}}}.$$

such that

$$\mathbf{F}\left[\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\right]\mathbf{C}_F^{(n+1)} = \mathbf{S}\mathbf{C}_F^{(n+1)}\mathbf{E}^{(n+1)}. \tag{4.79}$$

In such a case, we can in general not expect the expression

$$\mathbf{e}^{(n)} = \mathbf{F}\left[\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\right]\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\,\mathbf{S} - \mathbf{S}\,\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\mathbf{F}\left[\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\right] \tag{4.80}$$

to be zero. Typically, however, $\mathbf{C}^{(n+1)}$ is much closer to the minimum $\mathbf{C}_0$ than $\mathbf{C}^{(n)}$ was — see section 5.4.1 on page 128 for details. This suggests an iterative approach to find the minimum $\mathbf{C}$ starting from a guess $\mathbf{C}^{(0)}$. Such an approach is called **self-consistent field** (SCF) procedure and we will discuss it in more detail in the next chapter. In the context of such an iterative approach to solve (4.74) expression (4.80) is called the **Pulay error** after Peter Pulay [99].

There are two more aspects of theorem 4.14 worth pointing out. From the unfilled shell property (g) $\varepsilon_{N_{\mathrm{elec}}} < \varepsilon_{N_{\mathrm{elec}}+1}$ [96], we can deduce that there is always a gap between the highest occupied orbital (HOMO) and the lowest unoccupied orbital (LUMO) in a

converged HF result. Together with (e), we see that only eigenfunctions ψ_i with negative orbital energy eigenvalue ε_i may be part of $\mathbf{C}$ in the converged case [95]. Secondly the smoothness property (c) makes sure that all solutions of the HF equations are numerically easy to model. Especially for numerical basis functions like finite elements, where the rate of convergence depends on the smoothness of the function, this is important such that the problem can be modelled employing a reasonable number of basis functions. Furthermore care needs to be taken to place the grid points in a way that the nuclei sit *on* a grid point and not in between.

In this section we discussed roughly three ways to view the HF problem in the discrete setting of a finite basis set $\{\varphi_i\}_{i\in\mathcal{I}_{\text{bas}}}$. Firstly, as a minimisation on a Stiefel manifold (4.65), where the energy is optimised with respect to the occupied orbital coefficients $\mathbf{C}$. The second option is to minimise on a Grassmann manifold (4.69) and optimise the energy with respect to the density matrix $\mathbf{D}$. The last option is to solve the non-linear HF equations (4.74) in a self-consistent field approach, which would construct a sequence $\mathbf{C}^{(n)}$ of orbital coefficients or of density matrices $\mathbf{D}^{(n)}$ until (4.74) is satisfied within a certain error. No guarantee is made that this converges to the minimum, but in many cases it does. For all these approaches we will discuss practical algorithms in 5.4 on page 127.

4.4.2 Restricted Hartree-Fock

By the means of equation (4.72) we already stated that the unrestricted HF ansatz does not always yield an HF ground state Ψ_0, which is an eigenfunction of $\hat{\mathcal{S}}^2$. Following our discussion in remark 4.12 on page 60 we could fix this either by projection onto the space of eigenstates of $\hat{\mathcal{S}}^2$ or by imposing extra conditions on the HF ansatz.

In the case of closed-shell chemical systems the latter is in fact rather simple. For closed-shell atoms $N^\alpha_{\text{elec}} = N^\beta_{\text{elec}}$, which implies that (4.72) gives the correct value S^2_{exact} if we enforce $\mathbf{C}^\alpha = \mathbf{C}^\beta$. The condition $\mathbf{C}^\alpha = \mathbf{C}^\beta$ not only yields an eigenfunction of $\hat{\mathcal{S}}^2$, but furthermore implies that $\mathbf{F}^\alpha = \mathbf{F}^\beta$ as well. In other words in this **restricted Hartree-Fock** (RHF) [100] ansatz one only needs to solve one block of the Fock matrix in (4.74) and may use the result for both spin-up and spin-down functions.

Enforcing the correct S^2 value (4.72) for open-shell electronic systems with[15] $N^\alpha_{\text{elec}} > N^\beta_{\text{elec}}$ is considerably more involved. A first approach was published by Roothaan [101, 102]. In this celebrated work he distinguishes doubly occupied, singly occupied and virtual orbitals. He then replaces the block-diagonal Fock matrix from (4.74) by a specially crafted Fock matrix consisting of nine blocks, each block modelling the interaction between pairs of two of the aforementioned orbital subspaces. A block is build as a linear combination of certain parts of $\mathbf{F}^\alpha$ and $\mathbf{F}^\beta$, made in such a way to ensure, that the resulting SCF minimum is an eigenfunction of S^2 with exactly the desired eigenvalue. In other words the way this linear combinations are done depends on the spin state to be computed. Multiple ways to perform the linear combinations is possible [103] and depending on which method is chosen, results can deviate. This restricted open-shell Hartree-Fock (ROHF) approach will not be considered much further in this thesis. Instead we will treat all open-shell systems with the UHF procedure, since UHF is simpler to implement, computationally cheaper and overall more widespread. Nevertheless some issues, which appeared in our convergence analysis of our Coulomb-Sturmian-based

[15] By convention there are always more spin-up than spin-down electrons

Hartree-Fock ansatz (see chapter 8) turned out to originate from our UHF treatment and could be potentially avoided in an ROHF formalism. We will discuss this further in section 8.2.1 on page 179.

4.4.3 Real-valued Hartree-Fock

Our discussion of Hartree-Fock up to this point leads to numerical problems taking place in complex arithmetic, since both the coefficient matrix $\mathbf{C}$ as well as the Fock matrix $\mathbf{F}$ are so far taken to have complex entries. Whilst doing this is possible, it is typically nevertheless avoided by reducing the HF problem to a problem of equivalent structure, but situated in real Hilbert spaces. The major motivation for this is that computations amongst complex numbers are slower since effectively more floating point operations need to be performed in order to treat the real and the imaginary part as required.

Let $(\varepsilon_i, \psi_i) \in \mathbb{R} \times H^2(\mathbb{R}^3, \mathbb{C}^2)$ be an eigenpair of the HF problem, i.e.

$$0 = \left(\hat{\mathcal{F}}_{\Theta^0} - \varepsilon_i\right) \psi_i \tag{4.81}$$

Choosing appropriate functions $\psi_i^R, \psi_i^I \in H^2(\mathbb{R}^3, \mathbb{R}^2)$ we can write $\psi_i = \psi_i^R + \imath\psi_i^I$ such that by linearity of the Fock operator

$$0 = \left(\hat{\mathcal{F}}_{\Theta^0} - \varepsilon_i\right) \psi_i^R + \imath \left(\hat{\mathcal{F}}_{\Theta^0} - \varepsilon_i\right) \psi_i^I. \tag{4.82}$$

For (4.82) to be satisfied, we need the real and the complex part of the right hand side to be equal to the zero function.

Since all terms of the Fock operator only contain real factors or real differential operators, it is clear that the Fock operator maps real-valued functions to real-valued functions. In other words

$$\chi \in H^2(\mathbb{R}^3, \mathbb{R}^2) \quad \Rightarrow \quad \left(\hat{\mathcal{F}}\chi\right) \in L^2(\mathbb{R}^3, \mathbb{R}^2).$$

This implies that (4.82) is true iff simultaneously

$$0 = \left(\hat{\mathcal{F}}_{\Theta^0} - \varepsilon_i\right) \psi_i^R \tag{4.83}$$

$$0 = \left(\hat{\mathcal{F}}_{\Theta^0} - \varepsilon_i\right) \psi_i^I. \tag{4.84}$$

If ψ_i is not already real-valued and thus $\psi_i^I = 0$, its real part ψ_i^R and its imaginary part ψ_i^I must both be solutions to the HF equations as well. Furthermore ψ_i^R and ψ_i^I are associated to the same eigenvalue ε_i as ψ.

As a result one can obtain the solutions for the complex-valued problem (4.81) by only looking for eigenpairs $(\varepsilon_i, \psi_i) \in \mathbb{R} \times H^2(\mathbb{R}^3, \mathbb{R}^2)$. This completely avoids the need for complex arithmetic as $\mathbf{C}_F$, $\mathbf{D}$, $\mathbf{F}$ and all other matrices we defined previously in this section will only consist of real elements in this case. Apart from this simplification no extra care needs to be taken, since the real eigenfunctions corresponding to an eigenvalue ε_i still span the full complex eigenspace one would obtain from solving the original complex-valued problem — provided that complex coefficients are used. This makes sure that we neither miss anything nor get spurious results by using a real-valued ansatz, thus still get exactly the physical eigenstates we are after.

The overall approach sketched here is more general than just the Hartree-Fock problem and can be applied to many other spectral problems of quantum physics in order to yield equivalent real-valued versions for a numerical treatment. One should stress, however, that the operator under consideration not only needs to be Hermitian, but it further needs to map functions from a real Hilbert space to functions from a real Hilbert space. Let us illustrate this by the example of the single-particle spin operator

$$\hat{\mathrm{s}}_y = \begin{pmatrix} 0 & -\imath \\ \imath & 0 \end{pmatrix}.$$

It is easy to see that on the Hilbert space $\mathbb{C}^2$, $\hat{\mathrm{s}}_y$ is a Hermitian matrix, hence a self-adjoint operator. Its eigenvalues are -1 and 1 with corresponding normalised eigenvectors

$$\frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ \imath \end{pmatrix} \quad \text{and} \quad \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ -\imath \end{pmatrix}.$$

Even though this operator is Hermitian, trying to find the aforementioned eigenpairs by solving the spectral problem

$$\hat{\mathrm{s}}_y \underline{\boldsymbol{v}} = \lambda \underline{\boldsymbol{v}}$$

in real arithmetic only, i.e. with $\underline{\boldsymbol{v}} \in \mathbb{R}^2$, will not yield a single eigenpair.

4.5 Capturing electronic correlation

In the previous section we talked at length about the HF approximation for solving the non-relativistic electronic Schrödinger equation. Since the search space for the variational minimisation is much more restricted compared to the FCI ansatz, we necessarily make a larger approximation error in the HF case. Nevertheless it should be noted that HF yields a rather good rank-1 approximation to the full Schrödinger problem, where up to 99% of the FCI energy [104] in a respective basis is obtained at a fraction of the cost. Unfortunately chemistry is about energy differences and not about absolute energies. For example to a good approximation, chemical reactivity can be determined by looking at the energy barrier between reactants and products, i.e. the difference in energy between the reactants and the maximal energy, which is obtained along the reaction path transforming them to products. As the difference matters getting 99% of the absolute energy typically still leads to much larger errors than 1% for the reaction barrier. One might therefore wonder what part of the exact physical picture HF is missing and how one could improve on that.

4.5.1 What does Hartree-Fock miss?

Even though the Fock operator (4.49) describes a many-electron system, it is a one-particle operator, since it only acts on single-electron functions. The many-particle aspect is only treated via the Coulomb term (4.50) and the exchange term (4.51), where the interaction with other electrons is included in the form of integrals over the electron density ρ_{Θ^0} or the density matrix γ_{Θ^0}. Overall an electron thus does not see the exact position of all its neighbours via the Fock matrix, but only some kind of an average electron field. In this sense the HF ansatz is sometimes called a **mean-field approximation**. In the light of this the SCF can be thought of as an adjustment procedure, where the electronic arrangement in the form of the occupied SCF orbitals $\{\psi_i\}_{i\in\mathcal{I}_{\mathrm{occ}}}$ is

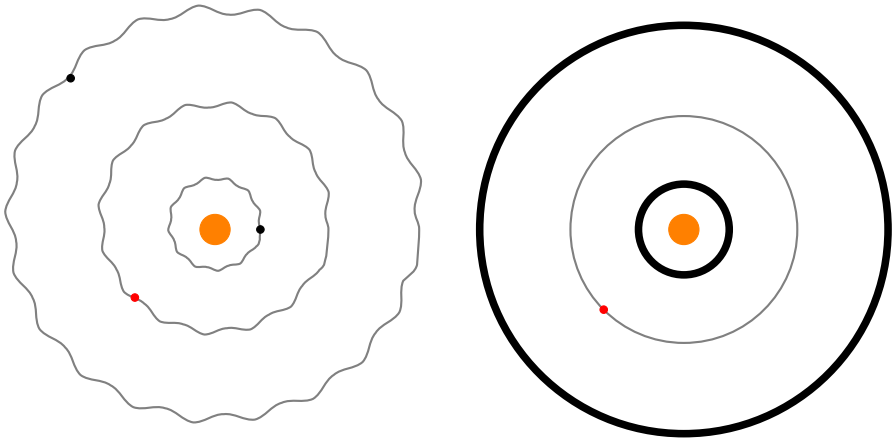


Figure 4.2: Real planetary system and mean-field model in the spirit of the HF approximation shown side-by-side. The mean-field picture on the right-hand side is shown from the point of view of the red planet, such that its neighbours are smeared out as thick black circles over their respective orbits. Adapted from [104].

adjusted until their generated mean field is no longer changing this arrangement, i.e. is self-consistent.

To visualise this issue better, let us consider in analogy a planetary system[16], where multiple planets are revolving around a central sun. In the real system, which is depicted on the left-hand side of figure 4.2, the individual planets feel each other at all times at their exact positions. As a result their orbit around the sun is not a perfect circle but shows pronounced wiggles due to the interaction between the planets. In other words the motion of the planets around the sun is highly correlated. In contrast to this the right-hand side depicts the scenario drawn for the red planet in a HF-like mean-field model. Its neighbours are no longer visible at their exact positions and the red planet thus only amounts to interact with some sort of smeared out particle density, where their position is averaged over their complete orbits. This interaction is almost as strong at all points and thus the mean-field orbit of the red planet is much more smooth.

In the electronic system the situation is similar in sense that the behaviour of individual electrons is indeed very much correlated. Due to its mean-field nature the HF ansatz largely misses the description of this so-called **electron correlation**[17]. In fact one typically refers to the difference

$$E_0^{\text{corr}} = E_0^{\text{FCI}} - E_0^{\text{HF}} \tag{4.85}$$

between the HF and FCI energies in a particular basis set as the **correlation energy**. As mentioned before E_0^{corr} is typically rather small compared to E_0^{HF}. Nevertheless the

[16] The idea is taken from [104].

[17] Conventionally one calls the HF treatment of a chemical system the *uncorrelated* treatment of the electronic structure. This is not perfectly sound in my opinion, as for example the Pauli principle is fulfilled in HF. This implies for example that two electrons of the same spin cannot occupy the same orbital, which implies in turn that the motion of electrons of the same spin is at least to this extend correlated.

effects of the electron-electron interaction are very important for a proper description of the electronic structure of a chemical system and can therefore not be neglected [83, 84, 104].

In practice one sometimes divides correlation effects into two subclasses. The first kind, the so-called **dynamic correlation**, is the aforementioned failure of the HF approximation to treat the communal, correlated motion of electrons properly. The second kind, **static correlation**, occurs if the number of Slater determinants, which is available for the description of a degenerate or near-degenerate state is not sufficient. For the HF approximation, where only one determinant for the description of the ground state is available, this defect becomes apparent in situations with a low-lying excited state, for example. A classic example would be a molecule close to bond breaking. In such a case the ground state resulting from a full CI treatment has relevant contributions from more than one determinant. As a result even the best restricted HF ground state determinant misses a substantial part of the full CI ground state and thus provides a wrong description of the physics. In the remainder of this discussion about electron correlation we will ignore static correlation and assume that a single determinant HF ground state is already a pretty decent description of the electronic structure. Detailed discussions of so-called multi-reference or multi-configurational methods tackling static correlation can be found for example in [84, 92, 104, 105].

4.5.2 Truncated configuration interaction

In section 4.3 on page 52 about the FCI method we already mentioned that the exact ground state Ψ_0 to the electronic Schrödinger equation can be expressed as a CI expansion (4.23)[18]

$$\Psi = c_0\Phi_0 + \sum_{ia} c_i^a \Phi_i^a + \sum_{\substack{i<j\\a<b}} c_{ij}^{ab}\Phi_{ij}^{ab} + \sum_{\substack{i<j<k\\a<b<c}} c_{ijk}^{abc}\Phi_{ijk}^{abc} + \cdots ,$$

starting from an arbitrary reference determinant Φ_0. A very natural choice for this is to take the reference determinant to be the HF ground state, i.e. the best single determinant for describing the electronic ground state. In this way the remaining contributions of the excited determinants Φ_i^a, Φ_{ij}^{ab}, Φ_{ijk}^{abc}, ... can be expected to be small, which makes it numerically more feasible to diagonalise the FCI matrix $\mathbf{A}_{\mathrm{FCI}}$. Furthermore this justifies truncating the CI expansion (4.23) prematurely to yield some sort of an intermediate approximation between HF and FCI. For example CISD, configuration interaction singles-doubles [106], truncates the above expansion in a way that only singles and doubles excitations are taken into account. This would lead to the ansatz wave function

$$\Psi_0^{\mathrm{CISD}} = \Phi_0 + \sum_{ia} c_i^a \Phi_i^a + \sum_{\substack{i<j\\a<b}} c_{ij}^{ab}\Phi_{ij}^{ab}$$

where one assumes the individual determinants are normalised in a way that $c_0 = 1$.

Even though truncated CI methods are conceptionally very simple, they are not used much any more for capturing dynamic correlation[19] The main reason for this is the

[18] In this section about correlation methods we will adhere to the usual index conventions where occupied indices are denoted with letters $i, j, k, l \in \mathcal{I}_{\mathrm{occ}}$ and virtual indices with letters $a, b, c, d \in \mathcal{I}_{\mathrm{virt}}$, see remark 4.8 on page 54 for details. To avoid clutter we will usually not indicate the index set in sums explicitly, e.g. write $\sum_a$ instead of $\sum_{a\in\mathcal{I}_{\mathrm{virt}}}$.

[19] In contrast to this statement the related multi-reference CI ansatz [107] *is* a state-of-the-art method for dealing with *statically* correlated systems.

so-called **size-consistency problem**. Unlike FCI the CISD energy of two molecular fragments is in general not additive, even if these fragments do not interact. Put more mathematically one can show [84] the following: If E_A is the energy corresponding to the CISD ground state Ψ_0^{CISD} for a molecule A and E_B is the analogous energy for another molecule B, then the CISD ground-state energy E_{AB} for a system consisting of both A *and* B separated by an infinite distance is *not* $E_A + E_B$. One refers to this unphysical behaviour as **size-inconsistent**. Including higher excitations does not fix this problem, such that all canonical truncated CI methods are size-inconsistent. For the modelling of chemical reactions or even large molecules, size-inconsistency is a major problem. Nowadays better, size-consistent alternatives like the coupled-cluster ansatz (see below) exist and are usually preferred.

4.5.3 Second order Møller-Plesset perturbation theory

Starting from the reasonable assumption that the HF ground state determinant Φ_0 is a very good approximation to the exact electronic ground state it is a sensible ansatz to employ Rayleigh-Schrödinger perturbation theory [83, 84] and correct perturbatively for the missing correlation contribution to the energy as well as the wave function. The typical perturbation theory ansatz is to partition the electronic Schrödinger Hamlitonian (4.12) into

$$\hat{\mathcal{H}}_{N_{\mathrm{elec}}} = \hat{\mathcal{H}}^0 + \hat{\mathcal{H}}^1,$$

i.e. a zeroth order Hamiltonian $\hat{\mathcal{H}}^0$, which is easy to compute, and the perturbation $\hat{\mathcal{H}}^1$, which is the part missed in $\hat{\mathcal{H}}^0$, assumed to be small.

One way this partitioning can be achieved is Møller-Plesset perturbation theory [108], where the unperturbed operator is taken to be the direct sum of N_{elec} Fock operators at the orbital configuration corresponding to the HF ground state Φ_0,

$$\hat{\mathcal{H}}^0 = \bigoplus_{i=1}^{N_{\mathrm{elec}}} \hat{\mathcal{F}}_{\Theta^0},$$

and the perturbation is

$$\hat{\mathcal{H}}^1 = \hat{\mathcal{V}}_{ee} - \bigoplus_{i=1}^{N_{\mathrm{elec}}} \left(\hat{\mathcal{J}}_{\Theta^0} + \hat{\mathcal{K}}_{\Theta^0} \right),$$

i.e. whatever the HF operator misses. In the discretised setting of a finite-dimensional one-particle basis $\{\varphi_\mu\}_{\mu \in N_{\mathrm{bas}}}$ one may easily derive the zeroth to second order energy contributions [83]

$$\begin{aligned} E_0^0 &= \sum_i \varepsilon_i, \\ E_0^1 &= -\frac{1}{2} \sum_{ij} \langle ij||ij \rangle, \\ E_0^2 &= \frac{1}{4} \sum_{ijab} \frac{|\langle ij||ab \rangle|^2}{\varepsilon_i + \varepsilon_j - \varepsilon_a - \varepsilon_b} \end{aligned}$$

to the ground-state energy. In other words up to zeroth order we obtain the sum of the orbital energies. The first order correction accounts for the double counting of

the electron-electron interactions and recovers the HF energy expression. The first real improvement to HF results at **second order Møller-Plesset perturbation theory** (MP2). For reasons, which will become clear in the next section one often introduces the so-called T_2 **amplitude**

$$t_{ij}^{ab} \equiv \frac{\langle ij||ab\rangle}{\varepsilon_i + \varepsilon_j - \varepsilon_a - \varepsilon_b} \tag{4.86}$$

and writes the MP2 energy as

$$E_0^{\mathrm{MP2}} = E_0^{\mathrm{HF}} + \frac{1}{4}\sum_{ijab} \langle ij||ab\rangle^* \, t_{ij}^{ab}. \tag{4.87}$$

The MP methods do have some issues as well. Most notably the perturbation expansion of energies does in general not converge [84], making it hard to properly justify these methods from a mathematical basis. In practice MP2 is still vividly employed, mainly because it gives a decent guess towards the exact energy of the electronic ground state at manageable computational cost[20]. The other MP methods on the other hands are nowadays used only rarely.

One should mention that due to its perturbative nature, there is no guarantee that $E_0^{\mathrm{MP2}} \geq E_0$, the exact ground-state energy of the electronic Schrödinger equation (4.8). In the community of quantum chemistry one often refers to this fact as MP2 being non-variational. This saying is, however, a bit inaccurate, since the method MP2 is indeed variational in the sense of the Courant-Fischer theorem (3.6), namely that larger basis sets will always lead a lower-energy MP2 ground state, which is furthermore closer to the exact MP2 ground-state wave function. This is not that much apparent in the outlined derivation, but can be seen from an alternative route employing the Hylleraas functional [84]. In contrast it is not variational in the sense that a larger basis yields an MP2 energy which approaches E_0 from above.

4.5.4 Coupled-cluster theory

The main idea of coupled-cluster theory is to employ a more elaborate ansatz for the ground-state wave function with the overall aim to reach a size-consistent method. In this work coupled-cluster only plays a minor role. This section will therefore be limited to the absolutely necessary steps to get the rough idea. For a more thorough introduction the reader is directed to the excellent review by Crawford and Schaefer [110] as well as numerous other works [84, 111] dealing with the topic.

In coupled-cluster theory one starts from the so-called exponential ansatz

$$\Psi^{\mathrm{CC}} = \exp(\hat{\mathrm{T}})\Phi_0 \tag{4.88}$$

to generate the coupled-cluster wave function Ψ^{CC} from a HF ground state reference determinant Φ_0. In this equation

$$\hat{\mathrm{T}} = \hat{\mathrm{T}}_1 + \hat{\mathrm{T}}_2 + \cdots + \hat{\mathrm{T}}_{N_{\mathrm{elec}}} \tag{4.89}$$

is the **excitation operator** consisting of all singles excitations

$$\hat{\mathrm{T}}_1 = \sum_{ia} t_i^a \hat{\tau}_i^a$$

[20] Using sensible approximations linear-scaling MP2 is possible [109].

with $\hat{\tau}_i^a$ defined, such that $\Phi_i^a = \hat{\tau}_i^a \Phi_0$, all doubles excitations

$$\hat{\mathrm{T}}_2 = \sum_{\substack{i<j \\ a<b}} t_{ij}^{ab} \hat{\tau}_{ij}^{ab} \qquad \text{with} \qquad \Phi_{ij}^{ab} = \hat{\tau}_{ij}^{ab} \Phi_0,$$

all triples

$$\hat{\mathrm{T}}_3 = \sum_{\substack{i<j<k \\ a<b<c}} t_{ijk}^{abc} \hat{\tau}_{ijk}^{abc} \qquad \text{with} \qquad \Phi_{ijk}^{abc} = \hat{\tau}_{ijk}^{abc} \Phi_0,$$

and so forth. In these sums the coefficients t_i^a, t_{ij}^{ab}, t_{ijk}^{abc} and so forth are called **cluster amplitudes**. In a similar notation to (4.22) the excitation operator is often directly written as a sum of the operators $\hat{\tau}_i^a$, $\hat{\tau}_{ij}^{ab}$, $\hat{\tau}_{ijk}^{abc}$ …, namely as

$$\hat{\mathrm{T}} = \sum_{\mu} t_\mu \hat{\tau}_\mu, \tag{4.90}$$

where μ is an appropriately chosen multi-index and the sum is implicitly taken to have sensible limits.

If we allow all possible excitations in (4.89), i.e. do not truncate the sum, the space spanned by all possible coupled-cluster wave functions Ψ^{CC} is exactly equivalent to the space of all Slater determinants, namely the form domain[21] $\tilde{Q}(\hat{\mathcal{H}}_{N_{\mathrm{elec}}})$. Without truncation CC is thus equivalent to FCI, moreover the exponential ansatz in this case just provides an alternative to the standard parametrisation of $\tilde{Q}(\hat{\mathcal{H}}_{N_{\mathrm{elec}}})$ in terms of the CI expansion (see remark 4.6 on page 53).

In the corresponding discretised setting, Φ_0 is the solution to the discretised HF problem (section 4.4.1 on page 64). In a similar fashion to full CI one would expect a good ansatz for obtaining a CC approximation to the ground state of the electronic Schrödinger equation to use the Ritz-Galerkin ansatz of remark 3.6 on page 34. In other words, one would attempt to solve the variational minimisation problem

$$E_0 \leq E_0^{\mathrm{CC}} = \inf_{\{t_\mu\}_\mu} \frac{\left\langle \exp(\hat{\mathrm{T}})\Phi_0 \middle| \hat{\mathcal{H}}_{N_{\mathrm{elec}}} \exp(\hat{\mathrm{T}})\Phi_0 \right\rangle_{N_{\mathrm{elec}}}}{\left\langle \exp(\hat{\mathrm{T}})\Phi_0 \middle| \exp(\hat{\mathrm{T}})\Phi_0 \right\rangle_{N_{\mathrm{elec}}}}, \tag{4.91}$$

where there resulting minimising amplitudes give the CC ground state wave function corresponding to the minimal ground-state energy E_0^{CC}. Without truncation of (4.89) this is again equivalent to discretised full CI. Even with truncation to, for example, $\hat{\mathrm{T}} = \hat{\mathrm{T}}_1 + \hat{\mathrm{T}}_2$, equation (4.91) is intractable to solve. The reason for this is the number of parameters in the problem. Even with truncation the exponential ansatz $\exp(\hat{\mathrm{T}})\Phi_0$ generates *every* Slater determinant, such that (4.91) yields a high-dimensional, non-linear problem, where products of the individual amplitudes $\{t_\mu\}_\mu$ often occur in the resulting system of equations.

[21] Recall the definition in (4.18).

For this reason one usually employs a different, so-called **projection** approach, which shall only be sketched here[22]. If one plugs the exponential ansatz directly into the electronic Schrödinger equation (4.8) for the ground state one obtains

$$\hat{\mathcal{H}}_{N_\mathrm{elec}} \exp(\hat{\mathrm{T}})\Phi_0 = E_0^{\mathrm{CC}} \exp(\hat{\mathrm{T}})\Phi_0, \tag{4.92}$$

where E_0^{CC} is the coupled-cluster ground-state energy. By a simple rearrangement this can be written as

$$E_0^{\mathrm{CC}} = \left\langle \Phi_0 \middle| \hat{\mathcal{H}}_T \middle| \Phi_0 \right\rangle_{N_\mathrm{elec}} \tag{4.93}$$

where we introduced the similarity-transformed Hamiltonian

$$\hat{\mathcal{H}}_T = \exp(-\hat{\mathrm{T}})\hat{\mathcal{H}}_{N_\mathrm{elec}} \exp(\hat{\mathrm{T}}).$$

For making use of equation (4.93) at all, the unknown amplitudes $\{t_\mu\}_\mu$ still need to be found. This is done by projecting (4.92) onto determinants $\exp(-\hat{\mathrm{T}})\Phi_\mu = \exp(-\hat{\mathrm{T}})\hat{\tau}_\mu\Phi_0$, which yields equations

$$\left\langle \Phi_\mu \middle| \hat{\mathcal{H}}_T \middle| \Phi_0 \right\rangle_{N_\mathrm{elec}} = 0 \tag{4.94}$$

one for each μ. In truncated CC methods, where only some of the terms of (4.89) are kept, we can use (4.90) to generate exactly one equation for each amplitude μ. In other words, the μ in (4.94) is just taken to run over the same index range as in the expansion (4.90) for the truncated excitation operator $\hat{\mathrm{T}}$.

Numerically solving for the CC amplitudes in (4.94) amounts to a root-finding problem, where the parameters are the set of all amplitudes $\{t_\mu\}_\mu$. This is typically approached by minimising the residuals

$$r_\mu = \left\langle \Phi_\mu \middle| \hat{\mathcal{H}}_T \middle| \Phi_0 \right\rangle_{N_\mathrm{elec}} \tag{4.95}$$

iteratively until numerically $r_\mu = 0$ for all μ. Even though this problem is easier compared to the variational CC ansatz, the working equations resulting from the expressions (4.95) are typically all but simple. For example, let us consider one of the simplest coupled-cluster approaches, where

$$\hat{\mathrm{T}} = \hat{\mathrm{T}}_2 = \sum_{\substack{i<j \\ a<b}} t_{ij}^{ab} \hat{\tau}_{ij}^{ab},$$

called coupled-cluster doubles (CCD). A proper derivation [110–113] starting from (4.94)

[22] Notice, that some mathematical rigour is dropped here. The expression $\hat{\mathcal{H}}_{N_\mathrm{elec}} \exp(\hat{\mathrm{T}})\Phi_0$ is only defined properly if $\Phi_0 \in H^2(\mathbb{C}^{3N_\mathrm{elec}}, \mathbb{C}^{2N_\mathrm{elec}})$. This is, however, not true in general. Even in the discretised case one may choose a one-particle basis $\{\varphi_\mu\}_{\mu\in\mathcal{I}_\mathrm{bas}}$, where some functions are not members of $H^2(\mathbb{C}^{3N_\mathrm{elec}}, \mathbb{C})$. As a result $\Phi_0 \notin H^2(\mathbb{C}^{3N_\mathrm{elec}}, \mathbb{C}^{2N_\mathrm{elec}})$.

yields the equations

$$\begin{aligned}
r_{ij}^{ab} &= \langle ab||ij\rangle \\
&+ \sum_e f_{ae}\, t_{ij}^{eb} - \sum_e f_{be}\, t_{ij}^{ea} - \sum_m f_{mi}\, t_{mj}^{ab} + \sum_m f_{mj}\, t_{mi}^{ab} \\
&+ \frac{1}{2}\sum_{mn} \langle mn||ij\rangle\; t_{mn}^{ab} + \frac{1}{2}\sum_{ef} \langle ab||ef\rangle\; t_{ij}^{ef} \\
&+ \sum_{me} \langle mb||ej\rangle\; t_{im}^{ae} - \sum_{me} \langle mb||ei\rangle\; t_{jm}^{ae} \\
&- \sum_{me} \langle ma||ej\rangle\; t_{im}^{be} + \sum_{me} \langle ma||ei\rangle\; t_{jm}^{be} \\
&- \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{mn}^{af}\, t_{ij}^{eb} + \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{mn}^{bf}\, t_{ij}^{ea}. \\
&- \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{in}^{ef}\, t_{mj}^{ab} + \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{jn}^{ef}\, t_{mi}^{ab} \\
&+ \frac{1}{4}\sum_{mnef} \langle mn||ef\rangle\; t_{mn}^{ab}\, t_{ij}^{ef} + \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{im}^{ae}\, t_{jn}^{bf} \\
&- \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{jm}^{ae}\, t_{in}^{bf} - \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{im}^{be}\, t_{jn}^{af} \\
&+ \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\; t_{jm}^{be}\, t_{in}^{af}
\end{aligned} \tag{4.96}$$

for the CCD residual r_{ij}^{ab}. They involve multiple contractions over the antisymmetrised ERI tensor from remark 4.10 on page 57, the amplitudes t_{ij}^{eb} and elements of the Fock matrix $\mathbf{f}$ in the SCF orbital basis. This latter matrix is defined as

$$\mathbf{f} = \mathbf{C}_F^\dagger \mathbf{F} \mathbf{C}_F \in \mathbb{C}^{N_\mathrm{orb} \times N_\mathrm{orb}}.$$

If the canonical HF ansatz of (4.53) is used, $\mathbf{f}$ will be diagonal and equivalent to $\mathrm{diag}(\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_{N_\mathrm{orb}})$. The corresponding CCD energy expression

$$E_\mathrm{CCD} = \frac{1}{4}\sum_{ijab} \langle ij||ab\rangle\, t_{ij}^{ab}. \tag{4.97}$$

can be obtained by simplifying (4.93). Since the rank-4 tensor t_{ij}^{ab} occurs in the expression for the $\hat{\mathrm{T}}_2$ excitation operator, this tensor is usually called the T_2-amplitudes tensor as well. Comparing the structure of (4.87) and (4.97), the name of the expression (4.86) in MP2 finally becomes apparent.

For higher-order methods like CCSD, where $\hat{\mathrm{T}} = \hat{\mathrm{T}}_1 + \hat{\mathrm{T}}_2$, or CCSDT, where $\hat{\mathrm{T}}_3$ is considered on top, the expressions for the working equations (4.95) are even more involved. In turn these methods become rather expensive as well, e.g. CCSD scales as $\mathcal{O}(N_\mathrm{bas}^6)$ and CCSDT as $\mathcal{O}(N_\mathrm{bas}^8)$. Nevertheless, CC methods are very popular and widely adopted in quantum chemistry. Firstly because they converge systematically towards the FCI energy as higher and higher excitations are considered in (4.89) and secondly because *all* CC methods are size-consistent — unlike the truncated CI methods we mentioned

above. One particular approach named CCSD(T), where the triples excitations are perturbatively added on top of CCSD, has been named the *gold standard* of chemistry as it generally yields highly accurate results with an expensive, but an acceptable scaling of $\mathcal{O}(N_{\text{bas}}^7)$, where the most costly $\mathcal{O}(N_{\text{bas}}^7)$ step is not iterative. Recent improvements [114] within the framework of pair-natural orbital approaches, has brought down the apparent scaling of CCSD(T) to linear, allowing to compute the energies of complete proteins on the level of CCSD(T).

4.5.5 Excited states methods

In most of our discussion up to this point we have only focused on obtaining an approximation to the ground state of the electronic Schrödinger equation. In some applications of electronic structure theory, however, electronic excitations play a role. Examples include the interaction of UV photons or photons of visible light with the electronic structure in a dye or a solar cell or more generally any photo-activated chemical reaction. Whenever this is the case the modelling of multiple electronic states on an equal footing is required.

For FCI or truncated CI methods, this can be achieved without additional modification by solving the respective full or truncated CI matrix for more than one eigenpair. All but the lowest-energy eigenpair describe excited states. These are not the only excited states methods in existence. In fact to each of the other methods we have discussed so far one is able to appoint at least one analogue [115]. For example for Hartree-Fock, there is configuration-interaction singles (CIS) or time-dependent HF (TDHF) and for coupled-cluster there are the equation-of-motion and linear-response coupled-cluster theories [116, 117]. Last but not least, the algebraic-diagrammatic construction scheme (ADC) for the polarisation propagator at various orders [118, 119] can be seen as a CI-like scheme on top of a Møller-Plesset ground state. Its excited states are generally in good agreement with the MP description of the ground state.

4.6 Density-functional theory

In this section we want to briefly look at a different approach towards modelling the electronic structure. Instead of solving for the wave function Ψ_0 associated to the ground state of the electronic Hamiltonian $\hat{\mathcal{H}}_{N_{\text{elec}}}$, the idea behind **density-functional theory** is to solve for the state's electronic density ρ_0 instead.

The rationale for this is twofold. Firstly the density contains all information about the chemical system. The integral $\int_{\mathbb{R}^3} \rho(\underline{\boldsymbol{r}})\, \mathrm{d}\underline{\boldsymbol{r}}$ evaluates to the number of electrons N_{elec} and via Kato's cusp condition [69] one may obtain the nuclear charges Z_A via the derivatives of the electron density at the cusp points. Secondly the Hohnberg-Kohn theorems [120] as well as the Levy constrained search ansatz [121] provide a unique identification between a particular ground state electron density and the potential, which generates this density. Even from a mathematical point of view solving for the ground state density $\rho_0(\underline{\boldsymbol{r}})$ is thus sufficient to characterise all properties of the ground state of a system.

The Levy constrained search ansatz [121] provides a conceptionally rather intuitive route to obtain the ground state density, namely by a constrained minimisation of the energy with respect to all possible densities. The issue with this procedure is that a closed-form expression for the energy functional $\mathcal{E}(\rho)$, which returns the energy of a given density, is not known for any relevant chemical system. In other words Levy constrained

search in the form presented so far cannot be applied to chemical systems.

Further progress can be made with the Kohn-Sham ansatz [122], however. The idea is to consider a fictitious system of N_{elec} non-interacting electrons, which still has the property that it reproduces the exact ground state density of the full, interacting system. Ignoring spin in our discussion, in this model system the exact wave function is a single determinant

$$\Psi = \Phi_\Theta = \bigwedge_{i=1}^{N_{\text{elec}}} \psi_i \qquad \text{where} \qquad \Theta \equiv (\psi_1, \psi_2, \ldots, \psi_{N_{\text{elec}}}) \in \left(H^1(\mathbb{R}^3, \mathbb{C})\right)^{N_{\text{elec}}}$$

is a tuple of N_{elec} single-particle functions. Ignoring spin the resulting ground state density is

$$\rho_\Theta(\underline{\boldsymbol{r}}) = \sum_{i=1}^{N_{\text{elec}}} |\psi_i(\underline{\boldsymbol{r}})|^2 ,$$

which allows to write the Kohn-Sham energy functional as

$$\begin{aligned} \mathcal{E}^{\text{KS}}(\Theta) = \frac{1}{2} \sum_{i=1}^{N_{\text{elec}}} \int_{\mathbb{R}^3} \|\nabla \psi_i\|_2^2 \, \mathrm{d}\underline{\boldsymbol{r}} + \int_{\mathbb{R}^3} \sum_{A=1}^{M} \frac{Z_A \, \rho_\Theta(\underline{\boldsymbol{r}})}{\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_A\|_2} \, \mathrm{d}\underline{\boldsymbol{r}} \\ + \frac{1}{2} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} \frac{\rho_\Theta(\underline{\boldsymbol{r}}_1)\rho_\Theta(\underline{\boldsymbol{r}}_2)}{\|\underline{\boldsymbol{r}}_1 - \underline{\boldsymbol{r}}_2\|_2} \, \mathrm{d}\underline{\boldsymbol{r}}_1 \, \mathrm{d}\underline{\boldsymbol{r}}_2 + E_{xc}(\rho_\Theta). \end{aligned} \tag{4.98}$$

In this expression E_{xc} is the **exchange-correlation functional**, which depends only on the density function ρ. This term is supposed to describe the non-local many-body effects not yet contained in the other terms, which is threefold, (1) the part of the kinetic energy missed by the non-interacting electrons, (2) the exchange interaction as well as (3) correlation effects. The crux with Kohn-Sham DFT is that its exact functional form is unknown, such that one has to live with approximations. Which exchange-correlation functional is sensible for a particular problem depends very much on the context of the chemical system, the property one is interested in and is still subject of debate in quantum-chemical literature. Notice, however, that if the exact exchange-correlation functional was to be found, (4.98) would yield the exact ground-state energy.

Following the original Levy constrained search, we want to find the density corresponding to the minimal energy, which in the Kohn-Sham picture implies the minimisation of $\mathcal{E}^{\text{KS}}(\Theta)$ with respect to the orbitals, thus the problem

$$E_0 \leq E_0^{\text{KS}} = \inf \left\{ \mathcal{E}^{\text{KS}}(\Theta) \,\middle|\, \Theta \in \left(H^1(\mathbb{R}^3, \mathbb{C})\right)^{N_{\text{elec}}}, \, \forall i, j \; \langle \psi_i | \psi_j \rangle_1 = \delta_{ij}. \right\}. \tag{4.99}$$

Both the energy functional (4.98) as well as the Kohn-Sham minimisation problem (4.99) are closely related to the HF problem (4.40). In fact the only difference is the substitution of the exchange energy term by the exchange-correlation functional. As such it should not be very surprising that the methods employed to solve (4.99) are very similar to HF as well. The conditions to obtain the stationary points of (4.99), the Euler-Lagrange equations, can be reformulated as

$$\hat{\mathcal{F}}^{\text{KS}}_{\Theta^0} \psi_i^0 = \varepsilon_i \psi_i^0 \qquad \text{and} \qquad \langle \psi_i^0 | \psi_j^0 \rangle = \delta_{ij} \tag{4.100}$$

where Θ^0 is the minimiser of (4.99) and

$$\hat{\mathcal{F}}^{\text{KS}}_{\Theta^0} = \hat{\mathcal{T}} + \hat{\mathcal{V}}_0 + \hat{\mathcal{J}}_{\Theta^0} + V_{xc} \tag{4.101}$$

is the Kohn-Sham operator. Its difference to the Fock operator (4.49) is again simply the replacement of the exchange operator $\hat{\mathrm{K}}_{\Theta^0}$ by the **exchange-correlation potential** $V_{xc}(\underline{\boldsymbol{r}})$, which is the derivative of the exchange-correlation energy $E_{xc}(\rho)$ with respect to the density function ρ. Equation (4.100) as well as the minimisation problem (4.99) can now be discretised similar to the procedure outlined in section 4.4.1 on page 64 for Hartree-Fock, which leads to an iterative self-consistent field procedure, which is very similar to the Hartree-Fock SCF outlined in remark 4.18 on page 70. Algorithmically both for Kohn-Sham DFT as well as HF the same type of problem needs to be solved, such that all of the numerical procedures discussed in the next chapters for HF could be applied to Kohn-Sham DFT with only very few changes.

Even though the mathematical problem of the Kohn-Sham DFT ansatz is related to HF, one should mention that DFT in combination with modern exchange-correlation functionals [123–128] is much more exact than HF for common applications of quantum-chemical calculations. Since the cost is comparable to HF, it has thus become by far the most widely used method of electronic structure theory.

Chapter 5

Numerical approaches for solving the Hartree-Fock problem

> *I believe there is no philosophical high-road in science, with epistemological signposts. No, we are in a jungle and find our way by trial and error, building our road behind us as we proceed.*
>
> — Max Born (1882–1970)

This chapter is devoted to an in-depth discussion of numerical approaches for solving the HF problem both when it comes to the basis function type used for the discretisation and the algorithms for solving the discretised problem. We will discuss how different basis function types lead to numerical problems of vastly different structure and how therefore not every algorithmic ansatz works for every type of basis function.

In section 4.4.1 we noted that there are roughly three ways to view the discretised HF problem. One way would be to think of it as a minimisation of the energy with respect to the orbital coefficients, another as a minimisation with respect to the density matrix and yet a third as a non-linear eigenproblem, which needs to be solved self-consistently. Our discussion here will generally take the third viewpoint and only switch to the others when this aids our argument. Furthermore we will implicitly assume a real-valued UHF ansatz in this chapter. The adaption of the presented results to RHF or ROHF is usually straightforward[1].

5.1 Overview of the self-consistent field procedure

In remark 4.18 of the previous chapter we suggested a simple procedure for iteratively solving the HF equations. The idea was to start from an initial guess $\mathbf{C}^{(0)}$ from the Stiefel

[1] To go from UHF to RHF one just needs to consider both blocks of the relevant Fock, coefficient, and density matrices to be equivalent. Going from UHF to ROHF only amounts to replacing the UHF Fock matrix by the appropriately constructed ROHF Fock matrix before performing the diagonalisation for getting the new coefficients.

manifold $\mathcal{C}$ as defined in (4.66) and repetitively construct occupied coefficient matrices $\mathbf{C}^{(1)}, \mathbf{C}^{(2)}, \ldots, \mathbf{C}^{(n-1)} \in \mathcal{C}$ by solving the discretised HF equations (4.79) and considering the Aufbau principle. Since the minimiser of the discretised HF problem (4.65) is unique[2], there is no need to diagonalise exactly $\mathbf{F}\Big[\mathbf{C}^{(n)}\big(\mathbf{C}^{(n)}\big)^\dagger\Big]$ in each iteration. Instead we could well diagonalise an arbitrary matrix $\tilde{\mathbf{F}}^{(n)}$ for obtaining the new coefficients $\mathbf{C}^{(n)}$. It is important, however, to ensure that the final coefficients, say $\mathbf{C}_0$, satisfy the necessary conditions for being a minimiser of $\mathcal{E}_C$, namely that $\mathbf{C}_0 \in \mathcal{C}$ and that the Pulay error (4.80) vanishes. At least its norm should stay within a finite value. Notice, that for the computation of the Pulay error in each case the unmodified matrix $\mathbf{F}\Big[\mathbf{C}^{(n)}\big(\mathbf{C}^{(n)}\big)^\dagger\Big]$ needs to be employed in order for the resulting value to be meaningful. As discussed in remark 4.18 on page 70 even if both these conditions are satisfied, this is no guarantee, however, that $\mathbf{C}^{(n)}$ is a minimiser for (4.65), since both are only *necessary* but no sufficient conditions. Ignoring this fact for a moment, this leads to the following general approach.

Remark 5.1 (SCF procedure)**.** Pick a **convergence threshold** $\varepsilon_{\text{conv}} \in \mathbb{R}$, a **basis set** $\{\varphi_\mu\}_{\mu \in \mathcal{I}_{\text{bas}}} \subset H^1(\mathbb{R}^3, \mathbb{R})$ and an **initial guess** $\mathbf{C}^{(0)} \in \mathcal{C}$ of occupied coefficients. From this build an initial Fock matrix $\tilde{\mathbf{F}}^{(0)} = \mathbf{F}\Big[\mathbf{C}^{(0)}\left(\mathbf{C}^{(0)}\right)^\dagger\Big]$.

For $n = 1, 2, 3, \ldots$

- Diagonalise
$$\tilde{\mathbf{F}}^{(n-1)}\mathbf{C}_F^{(n)} = \mathbf{S}\mathbf{C}_F^{(n)}\mathbf{E}^{(n)}$$
under the condition
$$\left(\mathbf{C}_F^{(n)}\right)^\dagger \mathbf{S}\mathbf{C}_F^{(n)} = \mathbf{I}_{N_{\text{orb}}}$$
where
$$\mathbf{E}^{(n)} = \operatorname{diag}\left(\varepsilon_1^{(n)}, \varepsilon_2^{(n)}, \ldots, \varepsilon_{N_{\text{orb}}}^{(n)}\right)$$
is the diagonal matrix of orbital eigenvalues.
- Construct the occupied matrix $\mathbf{C}^{(n)}$ from the full matrix $\mathbf{C}_F^{(n)}$ by the Aufbau principle.
- Build the Fock matrix $\mathbf{F}\Big[\mathbf{C}^{(n)}\big(\mathbf{C}^{(n)}\big)^\dagger\Big]$.
- Compute $\mathbf{e}^{(n)}$ according to (4.80)
$$\mathbf{e}^{(n)} = \mathbf{F}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right]\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\mathbf{S} - \mathbf{S}\,\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\mathbf{F}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right]$$
Check the necessary condition: If $\left\|\mathbf{e}^{(n)}\right\|_{\text{frob}} \leq \varepsilon_{\text{conv}}$ the procedure is considered converged[3] with final coefficients $\mathbf{C}_0 \equiv \mathbf{C}^{(n)}$.
- Build a Fock matrix $\tilde{\mathbf{F}}^{(n)}$ somehow using $\mathbf{C}^{(n)}$ and all insight into the problem gathered so far.

The final HF energy is given by $\mathcal{E}_C(\mathbf{C}_0)$ according to (4.59) and the final SCF orbitals Θ^0 by (4.57).

[2] This is only true in the discrete setting.

[3] In finite dimensions all norms are equivalent, so the choice of the Frobenius norm is arbitrary here.

This scheme still leaves a couple of important questions unanswered, which we will address in the following sections, namely:

- What is a suitable method for choosing the initial guess $\mathbf{C}^{(0)}$?
- What type of basis function is suitable?
- What algorithms are sensible for building the next Fock matrix $\tilde{\mathbf{F}}^{(n)}$?

Furthermore remark 5.1 considers the HF problem to be parametrised in terms of the occupied coefficients $\mathbf{C}^{(n)}$ and solves it by producing a sequence of coefficients $\mathbf{C}^{(1)}, \mathbf{C}^{(2)}, \ldots, \mathbf{C}^{(n)} \in \mathcal{C}$ until convergence. By the arguments discussed in section 4.4.1 one can alternatively parametrise the HF problem in terms of density matrices $\mathbf{D}^{(n)}$. In this light some SCF algorithms are better understood if one thinks about them as schemes producing a sequence of density matrices $\mathbf{D}^{(0)}, \mathbf{D}^{(1)}, \ldots, \mathbf{D}^{(n)} \in \mathcal{P}$ instead. As an example see the optimal damping algorithm in section 5.4.3 on page 129. To distinguish both approaches, the first kinds of algorithms iterating $\mathbf{C}^{(n)}$ will be called **coefficient-based SCF** schemes whilst the second kind of algorithms iterating $\mathbf{D}^{(n)}$ we will call **density-matrix-based SCF** algorithms.

The identification

$$\mathbf{D}^{(n)} = \mathbf{C}^{(n)} \left(\mathbf{C}^{(n)}\right)^{\dagger},$$

which we already presented in (4.71) in section 4.4.1, allows to build the density matrix from the coefficients by a matrix-matrix product and in the reverse direction we can find matching coefficients for each density matrix by a factorisation, e.g. a diagonalisation or a singular-value decomposition[4]. This allows — at least theoretically — to convert every density-matrix-based algorithms into a coefficient-based scheme like remark 5.1 and vice versa. In practice, the factorisation from density matrices to coefficients could become rather costly and might not be always applicable.

5.2 Guess methods

A good guess for an iterative procedure like the SCF is characterised by two things. Firstly, it should already be close to the expected solution. Otherwise one might as well start from a random initial set of coefficients. Secondly, it should be cheap to obtain, at least considerably cheaper than the SCF itself. Otherwise again a totally random guess does just as well.

Notice that in general random guesses have not much application from a practical point of view, but for investigating the stability of an SCF procedure they are really helpful. For example, one could check whether a combination of guess method and SCF algorithm yields a true local minimum or just a stationary point of the HF problem (4.65) by trying a few random guesses and checking the resulting energies.

The next sections present a non-exhaustive list of commonly used guess methods for starting SCF procedures.

[4] Thanks to Eric Cancès for pointing this out to me.

5.2.1 Core Hamiltonian guess

Only the Coulomb and exchange matrix terms of the Fock matrix expression (4.76)

$$\mathbf{F}[\mathbf{C}\mathbf{C}^\dagger] = \mathbf{T} + \mathbf{V}_0 + \mathbf{J}[\mathbf{C}\mathbf{C}^\dagger] + \mathbf{K}[\mathbf{C}\mathbf{C}^\dagger]$$

depend on the coefficients $\mathbf{C}$. Furthermore the entries of the kinetic matrix $\mathbf{T}$ and the nuclear attraction matrix $\mathbf{V}_0$ are typically larger than the entries of the Coulomb and exchange matrices. A reasonable approximation, which avoids the SCF procedure as a whole is therefore to find an initial guess $\mathbf{C}^{(0)}$ by diagonalising the core Hamiltonian $\mathbf{T} + \mathbf{V}_0$ and keeping the N_{elec} lowest eigenvalue solutions.

Since electrons in this model do not repel each other the resulting approximate orbitals are typically too contracted and thus not extremely physical. In my calculations with Coulomb-Sturmian-type basis functions (see section 5.3.6) for example I found core Hamiltonian guesses to often converge to stationary points in the SCF process, which are *not* minima of the HF problem.

Such issues become worse with larger basis sets or larger molecules. An ad-hoc way to fix this is to scale the nuclear attraction matrix by a factor $0 < \alpha \leq 1$ in order to mimic the shielding of the nuclear charge somewhat. Nevertheless this guess method is typically only used if other options are not available. An advantage is, however, that it can always be done.

5.2.2 Guesses by projection

A common procedure in many discretisation approaches is to first obtain a quick and crude solution using only a small basis set and to refine the result later in a larger basis. Ideally as much of the information gained in the crude result is used for starting the large calculation.

In the context of the SCF procedure one would, for example, like to use the final coefficients from a calculation in the small basis $\{\chi_{\tilde{\nu}}\}_{\tilde{\nu}=1,\ldots,N_g}$ with only N_g functions to obtain a guess for the more refined calculation in the basis $\{\varphi_\mu\}_{\mu=1,\ldots,N_b}$ with $N_b \geq N_g$. Conceptionally this requires to operate on the (full) coefficient matrix $\tilde{\mathbf{c}}_0 \in \mathbb{R}^{N_g \times N_{\text{orb}}}$ with the transformation matrix $\mathbf{U} \in \mathbb{C}^{N_b \times N_g}$ consisting of elements

$$U_{\mu\tilde{\nu}} = \langle \varphi_\mu | \chi_{\tilde{\nu}} \rangle_1$$

to project the result onto the larger basis. Since $U_{\mu\tilde{\nu}}$ is not necessarily unitary, the simple matrix-matrix product $\mathbf{U}\tilde{\mathbf{c}}_0$ will in general not be orthonormal with respect to the overlap matrix of the new basis $\mathbf{S} \in \mathbb{R}^{N_b \times N_b}$ and is thus not directly usable. Instead, a more involved treatment is required, which directly leads to a properly orthonormalised guess. For example one may compute the guess coefficients as [129]

$$\mathbf{C}^{(0)} = \mathbf{S}^{-1}\mathbf{U}\tilde{\mathbf{c}}_0\mathbf{N}^{-1/2} \tag{5.1}$$

where the matrix

$$\mathbf{N} = \tilde{\mathbf{c}}_0^\dagger \mathbf{U}^\dagger \mathbf{S}^{-1} \mathbf{U} \tilde{\mathbf{c}}_0 \tag{5.2}$$

takes care of proper normalisation. The computation (and diagonalisation) of $\mathbf{N}$ can be avoided if other techniques for orthogonalising the N_{orb} column vectors $\mathbf{S}^{-1}\mathbf{U}\tilde{\mathbf{c}}_0$ are used, like a Gram-Schmidt procedure or a singular-value decomposition.

A modification of this procedure would be to alternatively build

$$\mathbf{f} = \mathbf{U}\tilde{\mathbf{c}}_0\widetilde{\mathbf{E}}\tilde{\mathbf{c}}_0^\dagger\mathbf{U}^\dagger, \tag{5.3}$$

where (assuming $N_g \geq N_{\text{orb}}$)

$$\widetilde{\mathbf{E}} = \text{diag}\left(\tilde{\varepsilon}_1, \tilde{\varepsilon}_2, \ldots \tilde{\varepsilon}_{N_{\text{orb}}}\right)$$

are the orbital energies obtained from the SCF in the old basis. Diagonalisation of this matrix with respect to $\mathbf{S}$ yields a set of initial guess coefficients $\mathbf{C}^{(0)}$ as the eigenvectors and the modified orbital energies as the eigenvalues. For example the quantum-chemistry program ORCA uses the latter approach by default [130].

5.2.3 Extended Hückel guess

The extended Hückel (EH) procedure for obtaining estimates of molecular orbitals was developed in the 1960s by Hoffmann [131] based on the extended Hückel Hamiltonian matrix defined in the earlier work by Wolfsberg and Helmholz [132]. Sometimes this procedure is called **Generalised Wolfsberg-Helmholz procedure** for this reason as well.

The idea is here to start from a minimal set of orbitals $\{\phi_i\}_{i=1,\ldots,N_{\text{trial}}}$, originally exponential-type orbitals, and build the model Hamiltonian

$$H_{ij}^{\text{EH}} = \frac{1}{2}KS_{ij}^{\text{EH}}\left(H_{ii}^{\text{EH}} + H_{jj}^{\text{EH}}\right),$$

from the EH overlap matrix $\mathbf{S}^{\text{EH}}$ with elements

$$S_{ij}^{\text{EH}} = \int_{\mathbb{R}^3} \phi_i(\underline{\boldsymbol{r}})\phi_j(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}},$$

an empirical parameter K typically set to 1.75 and the diagonal elements H_{ii}^{EH}, which should be a rough estimate for the trial orbital energies ϕ_i. For this a range of methodologies are employed in practice, including the diagonal elements of the core Hamiltonian matrix of the trial basis, experimental atomic ionisation energies [133] or even the results from a cheap SCF procedure [130]. The obtained matrix H_{ij}^{EH} is diagonalised with respect to the EH overlap matrix $\mathbf{S}^{\text{HF}}$ yielding trial coefficients $\mathbf{C}^{\text{HF}}$. Following the procedures of the previous section 5.2.2 one may project these onto the basis set of the problem of interest and use them as an initial guess $\mathbf{C}^{(0)}$ for the SCF procedure.

Despite its age the EH method is still subject to active research. For example Lee et al. [134] have constructed a scheme combining the extended Hückel method and Slater's rules [3] by which decent guesses for finite-element-based density-functional theory calculations may be obtained.

Typically the EH method only works reasonably well for small basis sets and small molecular systems. This drawback is overcome if an approach related to the superposition of atomic densities is used for obtaining the diagonal elements of $\mathbf{H}^{\text{EH}}$. In the quantum-chemistry package ORCA [130] for example one can use both the atomic orbitals as well as the orbital energies from pre-calculated atomic STO-3G [4] calculations to drive the EH guess: The trial basis set $\{\phi_i\}_{i=1,\ldots,N_{\text{trial}}}$ in their approach is just the combination of all atomic STO-3G orbitals and the diagonal elements H_{ii}^{EH} the corresponding STO-3G orbital energies.

5.2.4 Superposition of atomic densities

The idea of the superposition of atomic densities (SAD) [135] is that molecules are to a very large extend just a collection of atoms, such that the molecular electron density can be obtained approximately just by adding up the densities of all constituting atoms. If atom-centred basis functions are used this process is almost trivial. Let us illustrate the procedure by a chemical system made up of M atoms labelled $1, 2, \ldots, M$. We first perform atomic ROHF calculations on each atom using the same basis set we want to employ for the molecular calculation, but only the basis functions of the atom in question. This yields converged atomic SCF density matrices $\mathbf{D}_1, \mathbf{D}_2, \ldots \mathbf{D}_M$. In the SAD guess method as described in [135] the trial density matrix $\tilde{\mathbf{D}}$ is the sum of all density matrix $\mathbf{D}_1^\alpha$, $\mathbf{D}_1^\beta$, $\mathbf{D}_2^\alpha$ after they have been projected from the atomic basis onto the basis used for the molecular calculation. If we compose the basis of the molecular system in the usual manner, i.e. by pasting together all basis functions a basis set defines for each atom, in the order atom by atom, $\tilde{\mathbf{D}}$ would be block-diagonal

$$\tilde{\mathbf{D}} = \begin{pmatrix} \mathbf{D}_1^\alpha + \mathbf{D}_1^\beta & 0 & \cdots & 0 \\ 0 & \mathbf{D}_2^\alpha + \mathbf{D}_2^\beta & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{D}_M^\alpha + \mathbf{D}_M^\beta \end{pmatrix}.$$

Replicating $\tilde{\mathbf{D}}$ twice on the α and the β block we can construct the trial density matrix

$$\mathbf{D}^\mathrm{t} = \begin{pmatrix} \tilde{\mathbf{D}} & 0 \\ 0 & \tilde{\mathbf{D}} \end{pmatrix}$$

and with it a trial Fock matrix $\mathbf{F}^\mathrm{t} = \mathbf{F}[\mathbf{D}^\mathrm{t}]$. A diagonalisation

$$\mathbf{F}^\mathrm{t}\mathbf{C}^{(0)} = \mathbf{C}^{(0)}\mathbf{E}^\mathrm{t}$$

finally yields the initial coefficients $\mathbf{C}^{(0)}$ along with some trial energies along the diagonal matrix $\mathbf{E}^\mathrm{t}$.

A few remarks about the SAD guess method:

- This whole procedure costs roughly as much as a single SCF step plus the time needed for the atomic calculation, which is typically negligible.
- Furthermore many quantum-chemistry programs store precomputed atomic densities $\mathbf{D}_1, \mathbf{D}_2, \ldots$ for their supported basis functions and all relevant elements of the periodic table, such that the cost of the SAD guess procedure is typically even lower in practice.
- The quality of the guess is in general rather good [135].
- Since $\mathbf{D}^\mathrm{t} \notin \mathcal{P}$, the orbital energies $\mathbf{E}^\mathrm{t}$ obtained from the diagonalisation of $\mathbf{F}^\mathrm{t}$ are not variational with respect to the overall HF problem. At least one further SCF step is therefore required.
- For molecular UHF and ROHF calculations, one might want to perturb the α and β parts of $\mathbf{D}^\mathrm{t}$ slightly in order to enforce breaking the spin symmetry in the α and β blocks.

5.3 Basis function types

This section tries to address the question, which classes of functions can be used in order to build a basis set $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ for solving the HF problem. For this we will first discuss some desirable properties for a basis set, both motivated from the aim to represent the physics of the electronic Schrödinger equation as good as possible as well as requirements from the numerical side. In the light of this, we will discuss four types of basis functions in depth, namely the Slater-type orbitals (STOs), the most commonly employed contracted Gaussian-type orbitals (cGTOs), a finite-element-based discretisation method as well as so-called Coulomb-Sturmian-type orbitals.

Even though we mostly concentrate on the HF problem in this section, quite a few of the observations made here apply to DFT or methods going beyond Hartree-Fock as well. In this sense the outlined discussion can be seen as an example case for the use of the mentioned basis function types in electronic structure theory as a whole.

5.3.1 Desirable properties

The central aspect of the Ritz-Galerkin procedure for approximately solving a spectral problem is the evaluation of the $a(\,\cdot\,,\,\cdot\,)$ corresponding to the operator for all pairs of basis functions, compare with remark 3.7 on page 35 for details. For this procedure to be mathematically meaningful at all, this requires the basis functions $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ to be taken from a dense subspace of the form domain of the operator. For the real-valued HF problem this is the Sobolev space $H^1(\mathbb{R}^3,\mathbb{R})$, thus a hard requirement for all types of basis functions used for Hartree-Fock and quantum chemistry in general is that they originate from $H^1(\mathbb{R}^3,\mathbb{R})$. Furthermore, in some or another sense we will need compute the elements of the Fock matrix $\mathbf{F}\big[\mathbf{C}\mathbf{C}^\dagger\big]$ (4.76), which in turn boils down to computing the integrals of the constituent matrix expressions (4.60) to (4.63), as well as the overlap matrix (4.64). The challenging step for this is typically the evaluation of the electron-repulsion tensor (4.31)

$$(\varphi_\mu\varphi_\nu|\varphi_\kappa\varphi_\lambda) = \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} \frac{\varphi_\mu^*(\underline{\boldsymbol{r}}_1)\varphi_\nu(\underline{\boldsymbol{r}}_1)\,\varphi_\kappa^*(\underline{\boldsymbol{r}}_2)\varphi_\lambda(\underline{\boldsymbol{r}}_2)}{\|\underline{\boldsymbol{r}}_1-\underline{\boldsymbol{r}}_2\|_2}\,\mathrm{d}\underline{\boldsymbol{r}}_1\,\mathrm{d}\underline{\boldsymbol{r}}_2.$$

as it involves a double integral over space incorporating a singularity at the origin as well as the product over four basis functions. Additionally, the discretised HF equations (4.74) need to be solvable numerically as well. We will see in the next sections that the main reason why contracted Gaussian-type orbitals have become so popular in quantum chemistry is that both evaluating the ERI tensor as well as solving the resulting eigenproblem is rather easy compared to the other cases.

Apart from the mathematical and numerical feasibility we would like to get meaningful results with as little effort as possible, i.e. a good description of a chemical system should already be achievable with small basis sets. Usually this goes hand in hand with a basis function which by itself represents the physics of the chemical system very well already, such that as much prior knowledge and chemical intuition as possible could be incorporated already into the basis. Ideally this would not bias the solution procedure, such that unexpected or unintuitive results can still be found.

Last but not least we would like to be able to know how wrong our HF results are compared to the exact HF ground state, possible even with a pointer how to increase the basis, such that results can be systematically improved. The aspired scenario would

be a rigorous and tight *a priori* or even better *a posteriori* error estimate for the chosen basis function type in the context of HF.

Of course this just sketches an ideal scenario. In reality one needs a good compromise, typically even a different compromise for different applications. Especially the *a priori* and *a posteriori* error estimates are not easy to derive rigorously for HF and I am not aware of any work achieving this for the basis function types I will discuss here in detail.

5.3.2 Local energy

Before we start discussing individual basis types, let us briefly pause and think about ways to quantitatively judge a particular basis function type. A natural choice is to consider a model system, where the analytical solution can be found, and compare it with the Ritz-Galerkin HF result produced by a particular basis on the same system. In this chapter, we will compare against the hydrogen atom. Without a doubt this does not probe all aspects of the physical interactions happening inside the electronic structure. Most importantly it does miss an evaluation how a basis set deals with electron correlation. All results therefore need to be taken with care: In more complex systems the situation will be deviating.

For comparing our numerical answers in the form of the HF ground-state Slater determinant Φ_0 to the exact electronic Schrödinger equation solution Ψ_0, we will use absolute errors and relative errors in the ground-state wave function as well as the ground-state energy. Additionally, we will consider a quantity called **local energy**, which is defined below.

Definition 5.2. Let Φ_0 be an approximation to the ground state of the operator $\hat{\mathcal{H}}_{N_{\text{elec}}}$. The local energy is defined by the quotient

$$E_L(\underline{\boldsymbol{x}}) \equiv \frac{\hat{\mathcal{H}}_{N_{\text{elec}}}\Phi_0(\underline{\boldsymbol{x}})}{\Phi_0(\underline{\boldsymbol{x}})}, \tag{5.4}$$

which is constant for an exact eigenstate of $\hat{\mathcal{H}}_{N_{\text{elec}}}$ and approximately constant for good approximations. Since the potential energy operator terms are only multiplicative, this expression can be alternatively written as

$$E_L(\underline{\boldsymbol{x}}) = -\frac{1}{2}\sum_{i=1}^{N_{\text{elec}}}\frac{\Delta_{\underline{\boldsymbol{r}}_i}\Phi_0(\underline{\boldsymbol{x}})}{\Phi_0(\underline{\boldsymbol{x}})} - \sum_{i=1}^{N_{\text{elec}}}\sum_{A=1}^{M}\frac{Z_a}{\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_A\|_2} + \sum_{i=1}^{N_{\text{elec}}}\sum_{j=1+1}^{N_{\text{elec}}}\frac{1}{r_{ij}}.$$

The concept of local energy originates from the quantum Monte Carlo community [136, 137], where its sampling by a Monte Carlo procedure plays a central role for obtaining the correlation energy. It is related to the relative residual

$$\frac{1}{\Phi_0(\underline{\boldsymbol{x}})}\left(\hat{\mathcal{H}}_{N_{\text{elec}}} - E_0\right)\Phi_0(\underline{\boldsymbol{x}}) = \frac{\hat{\mathcal{H}}_{N_{\text{elec}}}\Phi_0(\underline{\boldsymbol{x}}) - E_0\Phi_0(\underline{\boldsymbol{x}})}{\Phi_0(\underline{\boldsymbol{x}})} = E_L(\underline{\boldsymbol{x}}) - E_0,$$

where E_0 is the *exact* ground-state energy of $\hat{\mathcal{H}}_{N_{\text{elec}}}$. This implies first of all that $E_L(\underline{\boldsymbol{x}}) = E_0$ is necessary for $\Phi_0(\underline{\boldsymbol{x}})$ being the exact ground state. Furthermore the fluctuations of $E_L(\underline{\boldsymbol{x}})$ around the exact constant value E_0 provide a measure how far $\Phi_0(\underline{\boldsymbol{x}})$ is off from being an exact eigenstate of $\hat{\mathcal{H}}_{N_{\text{elec}}}$ at a particular point $\underline{\boldsymbol{x}}$. In this

sense $E_L(\underline{\boldsymbol{x}})$ can thus be seen as a *local* measure for the accuracy of $\Phi_0(\underline{\boldsymbol{x}})$ [9]. Inside regions where $E_L(\underline{\boldsymbol{x}})$ is close to being constant, the basis $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$ provides a sensible description of an eigenstate of $\hat{\mathcal{H}}_{N_{\text{elec}}}$. $E_L(\underline{\boldsymbol{x}})$ is without a doubt conceptionally related to the relative error in the ground-state wave function $1-\Phi_0(\underline{\boldsymbol{r}})/\Psi_0(\underline{\boldsymbol{r}})$. Compared to the latter quantity, $E_L(\underline{\boldsymbol{x}})$ has the additional advantage that one is able to notice which eigenstate $\Phi_0(\underline{\boldsymbol{r}})$ approximates in each region of space. For example, if it fluctuates around E_0 in some areas and around E_1 in others, we can see that $\Phi_0(\underline{\boldsymbol{r}})$ sometimes represents the first excited state better than the ground state. Additionally, $E_L(\underline{\boldsymbol{x}})$ can be applied even for cases where the exact solution is not known and thus the relative error cannot be found.

5.3.3 Slater-type orbitals

In section 2.3.5 on page 28 we discussed the analytical solution of the simplest chemical systems, namely the hydrogen-like atoms or ions with only a single nucleus and a single electron. Their solutions were functions

$$\Psi_{nlm}(\underline{\boldsymbol{r}}) = N_{nl}\tilde{P}_{nl}\left(\frac{2Zr}{n}\right)Y_l^m(\theta,\phi)\exp\left(-\frac{Zr}{n}\right)$$

where $\tilde{P}_{nl}$ is polynomial of degree $n-1$ in $\frac{2Zr}{n}$, see (2.44) for details. Characteristic for the functional form of these solutions is both the exponential decay as $r\to\infty$ as well as the discontinuity at the origin, i.e. the position of the nucleus. These two fundamental observations can be generalised to the setting of the full electronic Hamiltonian $\hat{\mathcal{H}}_{N_{\text{elec}}}$ as summarised in the following remark.

Remark 5.3. Let $\Psi_i(\underline{\boldsymbol{x}})$ be an exact eigenstate of the electronic Hamiltonian $\hat{\mathcal{H}}_{N_{\text{elec}}}$ with eigenenergy $E_i^{N_{\text{elec}}}$. It holds:

- Kato's electron-nucleus cusp condition [5]:

$$\left.\frac{\partial\langle\Psi(\underline{\boldsymbol{x}})\rangle}{\partial r_i}\right|_{\underline{\boldsymbol{r}}_i=\underline{\boldsymbol{R}}_A} = -Z_A\,\langle\Psi(\underline{\boldsymbol{x}})\rangle|_{\underline{\boldsymbol{r}}_i=\underline{\boldsymbol{R}}_A}$$

  where $\langle\Psi(\underline{\boldsymbol{x}})\rangle|_{\underline{\boldsymbol{r}}_i=\underline{\boldsymbol{R}}_A}$ denotes the average value on a hypersphere with $\underline{\boldsymbol{r}}_i=\underline{\boldsymbol{R}}_A$ fixed. Notice, that this expression can be reformulated to yield the more well-known result

$$\left.\frac{\partial\rho(\underline{\boldsymbol{r}})}{\partial\underline{\boldsymbol{r}}}\right|_{\underline{\boldsymbol{r}}=\underline{\boldsymbol{R}}_A} = -2Z_A\rho(\underline{\boldsymbol{R}}_A).$$

- Taking the limit $r_1\to\infty$, while keeping all other electronic and nuclear coordinates finite, the first electron is essentially decoupled from the motion of the other particles and only sees them as a point charge of value

$$Z^{\text{net}} = \sum_{A=1}^{M} Z_A - (N_{\text{elec}}-1). \tag{5.5}$$

  The other particles of the system, i.e. excluding electron 1, effectively forms a $N_{\text{elec}}-1$-electron system. This allows to approximately write

$$\Psi_i(\underline{\boldsymbol{x}}) \simeq \tilde{\Psi}_j^{\text{rest}}(\underline{\boldsymbol{x}})\tilde{\Psi}_i^1(\underline{\boldsymbol{r}}_1),$$

  where $\tilde{\Psi}_j^{\text{rest}}$ has only parametric dependence[5] on $\underline{\boldsymbol{r}}_1$. Approximately it is an eigen-

[5] Notice that this ansatz is somewhat related to the Born-Oppenheimer approximation.

function to $\hat{\mathcal{H}}_{N_{\text{elec}}-1}$. With this ansatz the electronic Schrödinger equation at large r_1 reduces for $\tilde{\Psi}_i^1(\underline{\boldsymbol{r}}_1)$ to

$$\left(-\frac{1}{2}\Delta_{\underline{\boldsymbol{r}}_1} - \frac{Z^{\text{net}}}{r_1} - E_i^{\text{net}}\right)\tilde{\Psi}_i^1(\underline{\boldsymbol{r}}_1) \simeq 0, \tag{5.6}$$

where Z^{net} is the net charge as in (5.5) and E_i^{net} is the net energy eigenvalue roughly equal to $E_i^{N_{\text{elec}}} - E_j^{N_{\text{elec}}-1} < 0$. For a neutral N_{elec}-system $Z_{\text{net}} = 1$, such that the solution of (5.6) is

$$\tilde{\Psi}_i^1(\underline{\boldsymbol{r}}_1) \simeq \exp\left(-\sqrt{-2E_i^{\text{net}}}\, r_1\right), \tag{5.7}$$

i.e. an energy-dependent exponential decay.

Both these results motivate the use of exponential-type atomic orbitals involving a factor $\exp\left(-\zeta \left\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_A\right\|\right)$ as basis functions for molecular calculations, since such a basis will give rise to solutions, which satisfy both conditions if the factor ζ is chosen correctly.

The first attempt to do this predates the rigorous results by Kato [5, 69] by over two decades. In 1930 Slater [3] obtained approximate solutions to the electronic Schrödinger equation for many atoms of the periodic table. He employed basis sets $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}} \subset H^1(\mathbb{R}^3, \mathbb{R})$ made out of exponentially decaying functions. Motivated from the solution to the Schrödinger equation of the Hydrogen atom, his ansatz was to write each basis function as

$$\varphi_\mu^{\text{STO}}(\underline{\boldsymbol{r}}) = R_\mu^{\text{STO}}(r_\mu)\, Y_{l_\mu}^{m_\mu}(\theta_\mu, \phi_\mu) \tag{5.8}$$

i.e. as a product of radial part R_μ^{STO} and real-valued[6] spherical harmonic $Y_{l_\mu}^{m_\mu}$, where

$$(r_\mu, \theta_\mu, \phi_\mu) \equiv \underline{\boldsymbol{r}}_\mu = \underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_\mu$$

is the distance vector to the nucleus located at $\underline{\boldsymbol{R}}_\mu$. For the radial part he used a polynomial times exponential form

$$R_\mu(r) = N_\mu r^{n_\mu - 1} \exp(-\zeta_\mu r), \tag{5.9}$$

where ζ_μ is a constant depending on the nuclear charge as well as the orbital in question and N_μ is the normalisation factor

$$N_\mu = (2\zeta_\mu)^{n_\mu} \sqrt{\frac{2\zeta_\mu}{(2n_\mu)!}}.$$

He was able to construct rules of thumb for obtaining the exponents ζ_μ for many elements by introducing a concept now known as **shielding** [3]. A more detailed discussion about shielding constants can be found in section 8.4.2 on page 191. In his honour exponential-type atomic orbitals of the form (5.8) with radial part (5.9) are called Slater-type orbitals (STOs).

As mentioned above STOs are physically rather sound and as such in many cases only few of them are required to achieve good results as errors are generally small and

[6] Keeping in mind that all $2l + 1$ spherical harmonics Y_l^m with the same value for l correspond to the same eigenspace one can always find an alternative representation to the functional form given in (2.37), where all spherical harmonics are real-valued functions. See [138] for details.

convergence fast [8, 9, 139, 140]. Their big drawback, however, is that evaluating the electron-repulsion tensor $(\varphi_\mu\varphi_\nu|\varphi_\kappa\varphi_\lambda)$ is challenging, such that STO-based methods are not amongst the most commonly used quantum-chemistry methods nowadays. Nevertheless, their promising properties and fast convergence has motivated many people to work on optimising STO expansions and on designing efficient evaluation schemes for the ERI tensor [9, 32, 140–142]. As a result, a number of packages like STOP [143], SMILES [144] and ADF [145] have become available, which employ basis sets composed of STOs.

5.3.4 Contracted Gaussian-type orbitals

In 1950 Boys [2] suggested to replace the exponential factor $\exp(-\zeta r)$ in the radial part (5.9) by a Gaussian factor $\exp(-\alpha r^2)$, resulting in the so-called Gaussian-type orbitals (GTOs). Such GTO basis functions still follow the ansatz radial part times real-valued spherical harmonic

$$\varphi_\mu^{\mathrm{GTO}}(\underline{\boldsymbol{r}}) = R_\mu^{\mathrm{GTO}}(r_\mu)\, Y_{l_\mu}^{m_\mu}(\theta_\mu, \phi_\mu) \tag{5.10}$$

but their radial part is now given as

$$R_\mu^{\mathrm{GTO}}(r) = N_\mu r^{l_\mu} \exp(-\alpha_\mu r^2) \tag{5.11}$$

with Gaussian exponent α_μ and normalisation constant

$$N_\mu = \sqrt{\frac{2^{l+2}}{(2l+1)!!}} \sqrt[4]{\frac{(2\alpha)^{2l+3}}{\pi}},$$

where

$$(2l+1)!! = (2l+1)(2l-1)(2l-3)\cdots 1$$

This replacement allows to perform the evaluation of the integrals involved in building the Fock matrix $\mathbf{F}$ much more efficiently. Because of the **Gaussian product theorem** [2, 83, 146], the product of two Gaussians may be expressed *exactly* as

$$R_\mu^{\mathrm{GTO}}\big(\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_\mu\|\big)\; R_\nu^{\mathrm{GTO}}(\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_\nu\|) = R_\kappa^{\mathrm{GTO}}(\|\underline{\boldsymbol{r}} - \underline{\boldsymbol{R}}_\kappa\|)$$

where l_κ, α_κ and $\underline{\boldsymbol{R}}_\kappa$ are chosen appropriately. With this result the evaluation of all ERI integrals (4.31) can be done analytically [2]. An example would be those involving four basis functions with $l_\mu = 0$. All other ERI integrals, potentially involving higher angular momentum l_μ, can be computed from the initial ones employing a set of recursion formulas [147]. Similar strategies can be found for the one-electron integrals in order to build $\mathbf{T}$ and $\mathbf{V}_0$. Overall the construction of $\mathbf{F}$ therefore becomes much more feasible for larger basis sets of Gaussians compared to large sets of STOs.

Unfortunately, certain physical aspects like the exponential decay or the cusp are no longer directly built into the basis set if such GTO basis functions are used. Since $\varphi_\mu^{\mathrm{GTO}}(\underline{\boldsymbol{r}}) \in C^\infty(\mathbb{R}^3, \mathbb{R})$, which is a dense subset of $H^1(\mathbb{R}^3, \mathbb{R})$ this is not *per se* a problem: The denseness ensures that we can still represent every function from $H^1(\mathbb{R}^3, \mathbb{R})$ up to arbitrary accuracy if we use enough GTOs. In other words, the Ritz-Galerkin ansatz still allows us to solve problems like HF or FCI up to arbitrary accuracy, but since the physics is not completely represented, more basis function might be required to model for it.

As a remedy Hehre et al. [4] introduced so-called **contracted Gaussian-type orbitals** (cGTOs), where the radial part of a basis function φ_μ is expressed as a fixed linear combination of N_{contr} **primitive Gaussians**[7]

$$R_\mu^{\text{cGTO}}(r) = r^{l_\mu} \sum_i^{N_{\text{contr}}} c_{\mu,i} \exp(-\alpha_{\mu,i} r^2).$$

The idea is to get the best out of both worlds: The easily solvable integrals in terms of primitive GTOs and an accurate description of the wave function by using predetermined sets of **contraction coefficient**s $c_{\mu,i}$ and exponents $\alpha_{\mu,i}$, known to give a good basis set $\{\varphi_\mu^{\text{cGTO}}\}_{\mu\in\mathcal{I}_{\text{bas}}}$. By the means of this trick one is able to effectively split the parameter space of the variational problem (4.65) into two parts. One — the contraction coefficients — is fitted once and for all in order to fit a large range of problems and another — the coefficient matrix (4.58) — is the search space over which one minimises during the actual calculation.

Out of the pragmatic desire to perform molecular calculations on systems larger than what was feasible with STO basis sets at that time, Hehre et al. [4] initially focused on contracting primitive Gaussians in a way that they most closely resembled a particular STO function. This resulted in the famous STO-nG family of basis sets. Later it was realised that more accurate basis sets could be constructed by trying to minimise the energy, which is resulted from an actual HF or an MP2 calculation. Other strategies included a rigorous construction of the basis set in order to obtain convergence in the amount of recovered correlation energy, or to be consistent in certain computed properties. These deviating approaches have led to a number of different basis set families over the years, most of which share common concepts, however. Our discussion here should remain rather brief. Interested readers are referred to the excellent reviews by Hill [7] and Jensen [6].

All basis sets, which are considered state-of-the-art nowadays, are so-called **split-valence basis sets**, which is meant to indicate that multiple contracted Gaussians are available for describing the valence shell of an atom. How many are used is typically referred to by the ζ-level, e.g. a double-ζ basis set contains two *contracted* Gaussians for each valence orbital, a triple-ζ basis set three and so on. For this characters like `D`, `T`, `Q`, `5`, ... — for double, triple, quadruple, quintuple level — may be found in the name of the basis set. Notice that each contracted Gaussian inside such basis sets is typically in turn made up from multiple primitives. For a particular basis set family the error generally decreases going to higher zeta levels. For some families like Dunning's correlation-consistent basis sets [148] empirical formulas for estimating the error at a particular zeta level exist [149]. These results have been used for many years to estimate properties at the so-called **complete basis set** (CBS) limit, i.e. the theoretical value obtained if an infinitely large basis set of cGTOs were employed for the calculation. A recent work by Bachmayr et al. [150] provides some mathematical support for such formulas by rigorously deriving error estimates in the relevant $H^1(\mathbb{R}^3, \mathbb{R})$-norm. One should note, however, that these results strictly speaking only apply to a basis of uncontracted even-tempered GTOs. The authors point out, however, that a generalisation towards cGTOs should be possible.

A large range of cGTO basis sets are available nowadays, which offer a spectrum of

[7] Here we follow the usual convention to include the normalisation constant inside the contraction coefficients.

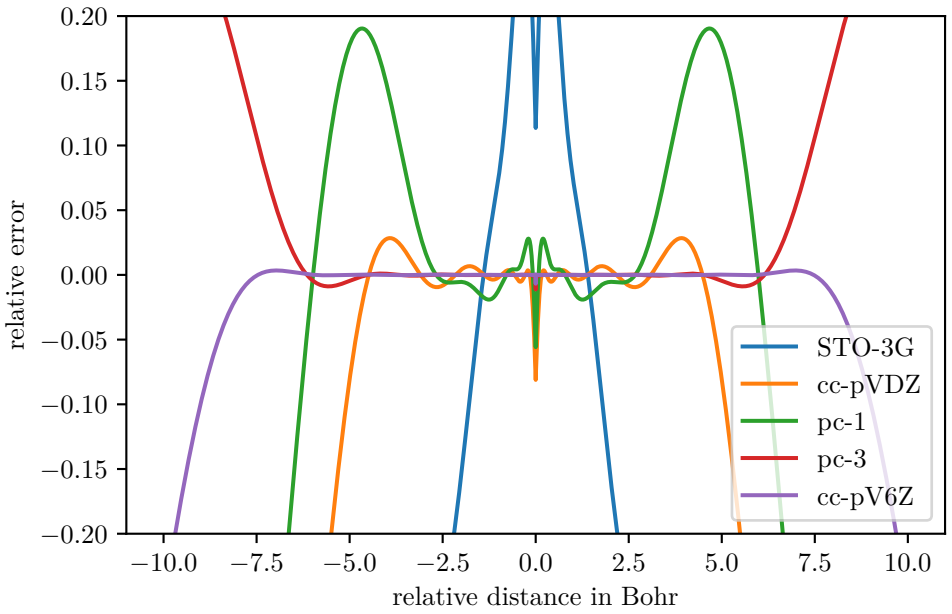


Figure 5.1: Relative error in the hydrogen ground state employing selected cGTO basis sets [4, 148, 151, 152]. The error is plotted against the relative distance of electron and proton. Notice that pc-n is a basis set at $n+1$-ζ level.

compromises between accuracy and computational cost. Nevertheless, some systematic issues related to the non-physical shape of the primitive GTOs cannot be fully accounted for, even in the largest basis sets. To illustrate this, consider figure 5.1. In this plot the relative error of the hydrogen ground state Φ_0 with respect to the exact electronic ground state Ψ_{1s} (2.48)[8]

$$\frac{\Phi_0(\underline{\boldsymbol{r}}) - \Psi_{1s}(\underline{\boldsymbol{r}})}{\Psi_{1s}(\underline{\boldsymbol{r}})}$$

is shown at various electron-proton distances. The plots for multiple standard cGTO basis sets are depicted, namely the minimal basis set STO-3G [4], the double-ζ basis sets cc-pVDZ [148] and pc-1 [151], the quadruple-ζ basis set pc-3 [151] as well as the sextuple-ζ basis set cc-pV6Z [152]. In each case the error is smallest at intermediate electron-proton distances, but increases both at the origin as well as larger distances. The former feature originates from the failure of Gaussians to represent the electron-nuclear cusp. The latter feature can be explained due to the faster fall-off of the Gaussians, $\exp(-\alpha r^2)$, compared to the exact solution, which goes as $\exp(-\zeta r)$. Larger basis sets like pc-3 or cc-pV6Z amount to recover the correct decay behaviour as well as the cusp somewhat, such that the error stays below $0.02 \equiv 2\%$ in the complete inner part of the plot up to distances of about 7.5 Bohr. Eventually all relative errors tend towards $-\infty$ as $r \to \infty$, however. Even though this cannot be seen in figure 5.1, this includes the case of pc-3, where the relative error has a local maximum around $r = 10$ and then follows a downhill slope as well. Overall the plots agree with the rule of thumb that results become more accurate at higher ζ-levels: Both the relative errors get smaller as well

[8] For hydrogen HF is equivalent to solving the full Schrödinger equation.

as the region where the wave function is well-represented becomes larger as we proceed from STO-3G to double-ζ and higher ζ levels.

In figures 5.2 and 5.3 the local energies (5.4) of the aforementioned basis sets are depicted — again as a function of relative distance. These plots not only diverge to $-\infty$ as $r \to \infty$, but at the origin as well, see particularly figure 5.3. At intermediate electron-proton distances the local energies of all basis sets fluctuate around the exact ground-state energy of 0.5 Hartree, where the amplitude of the fluctuations are lowest for cc-pV6Z and pc-3. Recall that the local energy is related to the relative residual error and that ideally it should be a constant. At intermediate distances, where the fluctuations are small, the ground state thus agrees well with the exact ground state. Unsurprisingly, the parts of figure 5.2, where $E_L(\underline{\boldsymbol{r}})$ is almost constant, agree roughly with the parts of figure 5.1 where the relative error is small. Similarly, the wrongful decay behaviour of the cGTO solutions is observed in both the plot of the relative error as well as the local energy plot. The most notable discrepancy of both error metrics is close to the nucleus, see figure 5.3. Whilst the relative error gets smaller and smaller for the larger pc-3 and cc-pV6Z basis sets close to the core as well, these show rather vivid fluctuations in $E_L(\underline{\boldsymbol{r}})$ as $r \to 0$. Eventually they diverge to $-\infty$ exactly like the result employing any other basis set. In other words, whilst these basis sets amount to produce a very good description of the ground state from distances around 0.5 Bohr up to 7.5 Bohr, they fail to do so close to the core in a rather misbehaving manner. Since the relative error is small, the issue is not that the function value of the exact ground state is missed. Much rather the culprit is the gradient of the approximated ground state.

This can be explained following [137]. The potential term in the local energy (5.4) diverges as $-Z_A/r$ close to the nucleus A, such that the kinetic energy term inside (5.4) needs to provide an equal and opposite divergence in order for the resulting local energy to be constant. Since the gradient of every cGTO basis functions is zero at the origin, so is the gradient of the final ground state, thus the local energy goes to $-\infty$. Furthermore, the gradient of each individual primitive Gaussian goes to zero at a different rate depending on its exponent $\alpha_{\mu,i}$. Overall this leads to an overcompensation of the diverging potential in the kinetic term at some points and an undercompensation at others, giving rise to the oscillatory behaviour. This oscillatory feature close to the nucleus is well-known in the quantum Monte-Carlo community [136, 137], since it can lead to problems when sampling the local energy, especially in diffusion Monte-Carlo.

In HF, DFT and Post-HF methods the failure of the cGTOs to represent the nuclear cusp or the long-range behaviour is typically only an issue if either parts of the wave functions are especially important for a particular property. The reason is that the important aspect for the modelling of chemical processes and properties is not the absolute energy of a molecule. Much rather chemistry is all about relative energies between the involved species or electronic configurations. Since changes in the electronic structure both at the nucleus as well as the region far from the nuclei are generally much less pronounced, the errors resulting from an inadequate description of these features tend to cancel one another. In other words the convergence with respect to a description of electronic properties tends to be faster than the convergence of absolute energies.

Examples for cases which require a proper representation of the nuclear cusp or the long-range behaviour of the electron density are the determination of Rydberg-like excited states, resonance processes, the computation electron affinities, the computation of X-ray absorption spectra or the computation of nuclear-magnetic resonance properties.

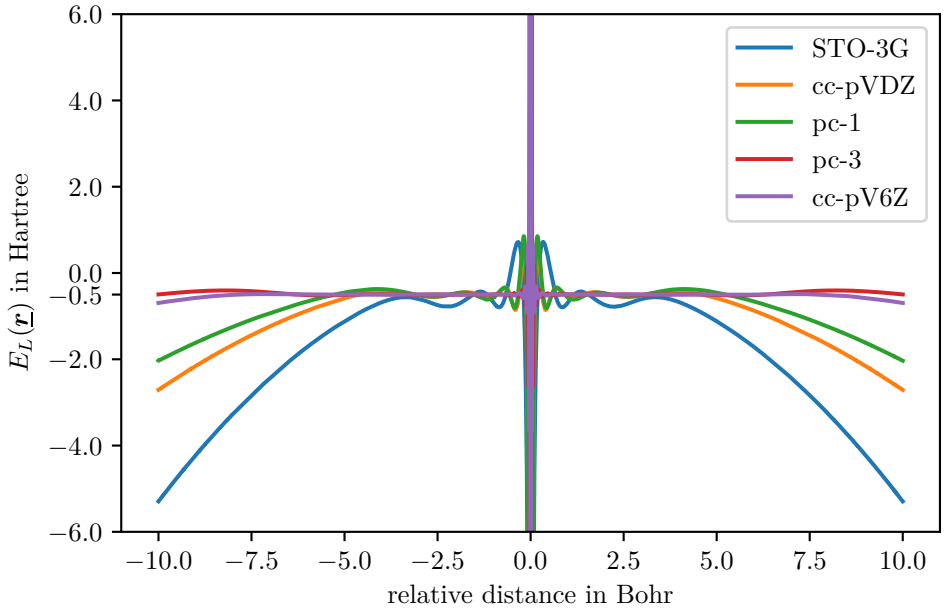


Figure 5.2: Local energy $E_L(r)$ of the hydrogen atom ground state obtained using the indicated contracted Gaussian basis sets [4, 148, 151, 152]. $E_L(r)$ is plotted against the relative distance of electron and nucleus. Notice that pc-n is a basis set at $n+1$-ζ level.

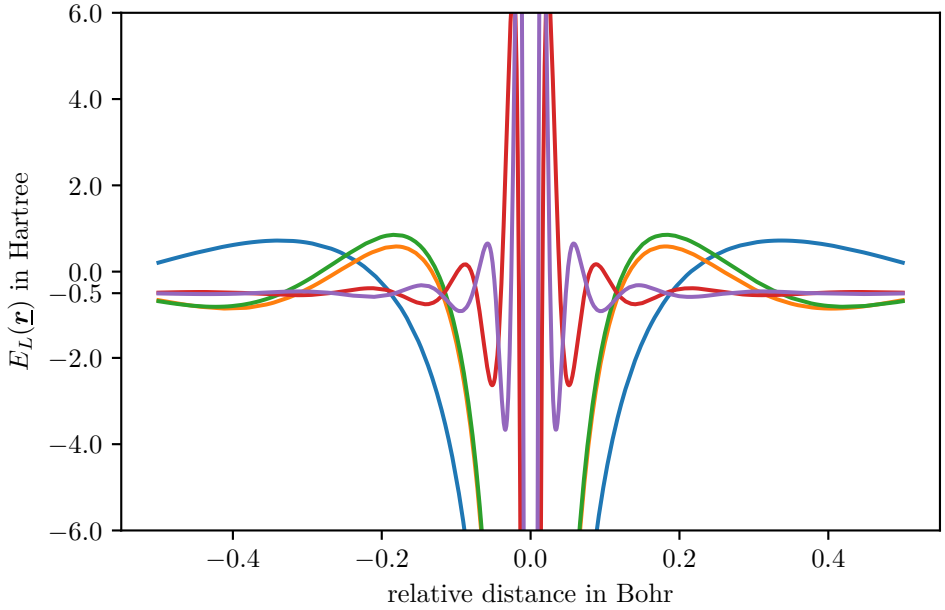


Figure 5.3: Magnified version of figure 5.2 around the origin.

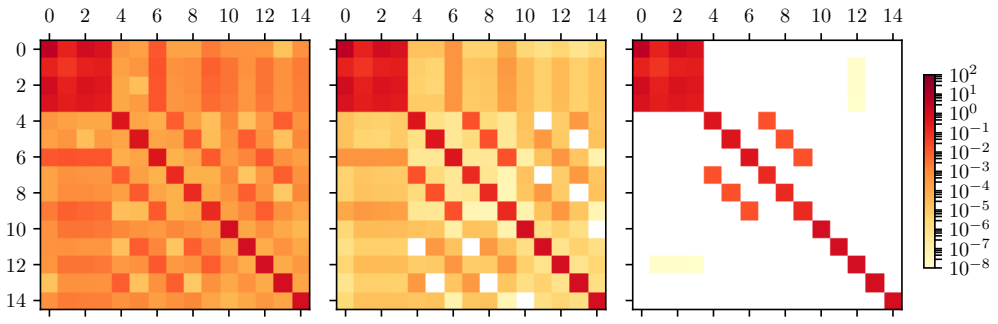


Figure 5.4: Structure of the Fock matrix for a cGTO-based Hartree-Fock calculation of the beryllium atom in a pc-2 [153] basis set. The three figures show the matrix at different convergence stages during the SCF. From left to right the Pulay error Frobenius norm is 0.18, 0.0063 and $4.1 \cdot 10^{-7}$. The colouring depends on the absolute value of the respective Fock matrix entry with white indicating entries below 10^{-8}.

For the modelling of these processes specific basis sets are required [6, 7], which include further cGTO basis functions to either sample the core region or the long-range tail more accurately. If such basis sets are not employed it may happen that the desired features are completely missed or described very inaccurately. In this sense cGTOs are not fully black-box and picking a reasonable basis set for a particular problem usually requires some idea of the electronic structure already. On the other hand, if such special basis sets are employed, one may encounter numerical instabilities. The reason is that such basis sets tend to be amended with cGTOs of either very small or very similar exponents. This implies that the basis functions φ_μ may become almost linearly dependent, yielding large off-diagonal overlap matrix elements $\langle\varphi_\mu|\varphi_\nu\rangle_1$ and a near-singular overlap matrix. This observation is typically referred to as the **overcompleteness** of the cGTO basis.

It was already mentioned that the Gaussian product theorem allows for an efficient evaluation of the integrals required for building the Fock matrix $\mathbf{F}$. Furthermore the resulting Fock matrix is comparatively small: Even for systems with hundreds of atoms one typically only needs in the order of thousands of basis functions. In other words, both building the Fock matrix as well as diagonalising it can be performed using direct methods[9]. Ignoring the basis sets suffering from overcompleteness for a second, the numerical structure of a cGTO-based Fock matrix is rather advantageous in most cases. Figure 5.4, for example, shows some Fock matrices from an SCF calculation of a beryllium atom in the pc-2 [153] basis set. The matrices are taken as snapshots during the SCF procedure. From left to right the Pulay error Frobenius norm decreases from 0.18 to 0.0063 and finally $4.1 \cdot 10^{-7}$. As the error gets smaller the matrix becomes more and more diagonal as the off-diagonal elements in the occupied-virtual block of the Fock matrix all have to vanish[10]. Already the leftmost matrix is almost diagonal-dominant

[9] A standard procedure would be to reduce the matrix to tridiagonal form using Householder reflections and then use Cuppen's divide and conquer [62] or multiple relatively robust representations [154].

[10] This is another way equivalent to (4.80) to express SCF convergence.

with 12 out of 15 rows μ satisfying the condition for **diagonal-dominance**

$$\sum_{\nu=1}^{N_{\text{bas}}} F_{\mu\nu} < 2F_{\mu\mu}.$$

For larger systems, the structure generally gets worse due to interactions between the atoms, but if a proper description of the core region or the tail is not required $\mathbf{F}$ stays numerically manageable and almost diagonal-dominant. This allows further to employ iterative eigensolver methods like Davidson's method (see section 3.2.6 on page 40) to efficiently obtain eigenpairs of the Fock matrix if only a selected part is required.

Since for most cases in chemistry the region close to the nucleus and the long-range tail are not extremely important, both obtaining and diagonalising Fock matrices from a cGTO discretisation is straightforward. Even though cGTOs are physically not the most sensible basis function type, this has historically made cGTO-based methods the most predominant approach to describe a chemical system within decent accuracy such that these methods are now implemented in countless quantum-chemistry packages. In light of this, it is remarkable, that only in 2014 error bounds were rigorously derived by Bachmayr et al. [150] for some special kinds of Gaussian basis sets and these results are not employed on a daily basis.

5.3.5 Discretisation based on finite elements

The Slater-type orbitals and Gaussian-type orbitals we introduced in the previous sections are examples for so-called atom-centred basis functions or **atom-centred orbital**s (AOs). A different ansatz in many respects are grid-based methods, where the underlying idea is to partition three-dimensional real space into smaller parts using a structured grid. The problem is then solved by grid interpolation and numerical integration instead of analytical evaluation of integrals. The example we want to consider in this work, are **finite elements**, which are specifically constructed piecewise polynomials, often employed in structural mechanics or engineering for solving partial differential equations [155]. Multiple approaches for solving the HF problem or the Kohn-Sham equations using finite-element-based discretisations have been performed over the years [17–23, 134]. This section only gives a short overview of the finite-element method in the light of the HF problem with special focus on the things I have tried during my doctoral studies. For more details the reader is referred to the literature [68, 155–157].

Construction of a FE grid

Compared to atom-centred basis functions, where a discretisation based on the complete domain $\mathbb{R}^3$ is possible, any grid-based method can only achieve this on a subset $\Omega \subset \mathbb{R}^3$. Typically Ω is taken to be open. At the boundary $\partial\Omega$ one needs to impose a boundary condition in order for the solution to be unique. Ideally we would like to model the problem as close to the complete $\mathbb{R}^3$ as possible, i.e. one would like to make the domain Ω as large as possible. In practice one needs to make a compromise, raising the question what kind of conditions to impose on the boundary. We will discuss this in more detail later in the context of solving the Poisson equation, see equations (5.20) to (5.22). For now let us assume that Ω is large enough, such that the SCF orbitals are essentially zero on $\partial\Omega$ and we can impose a homogeneous Dirichlet boundary[11]. Using this approximation

[11] This implies that the HF eigenfunctions are forced to be exactly zero at the boundary $\partial\Omega$.

as well as the inner product

$$\langle\psi|\chi\rangle_1 \equiv \int_\Omega \psi(\underline{\boldsymbol{r}})\chi(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}}$$

the spin-free, real-valued HF equations (4.53) can be adapted to read

$$\begin{aligned} \hat{\mathcal{F}}_{\Theta^0}\psi_i^0(\underline{\boldsymbol{r}}) &= \varepsilon_i\psi_i^0(\underline{\boldsymbol{r}}) \quad && \underline{\boldsymbol{r}} \in \Omega \\ \psi_i^0(\underline{\boldsymbol{r}}) &= 0 && \underline{\boldsymbol{r}} \in \partial\Omega \\ \text{where} \qquad \left\langle\psi_i^0\middle|\psi_j^0\right\rangle_1 &= \delta_{ij}, \end{aligned} \tag{5.12}$$

for $\Theta^0 = (\psi_1^0, \psi_2^0, \ldots, \psi_{N_\mathrm{elec}}^0) \in \left(H^2(\Omega,\mathbb{R})\right)^{N_\mathrm{elec}}$ being the minimiser to the HF problem (4.40). The corresponding sesquilinear form

$$a_{\Theta^0}(\psi,\chi) \equiv \int_\Omega \psi(\underline{\boldsymbol{r}})\hat{\mathcal{F}}_{\Theta^0}\chi(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}}$$

is defined in analogy to (4.75). By partial integration it can be seen that this form is defined on the domain $Q(\hat{\mathcal{F}}_{\Theta^0}) = H^1(\Omega,\mathbb{R})$. In the **finite-element method** the aim is to solve (5.12) variationally in the sense of remark 3.6 on page 34 employing a hierarchy of approximation spaces S_n. Such an attempt is of course only sensible if such spaces are more and more accurate approximations of the form domain $H^1(\Omega,\mathbb{R})$ in the sense of (3.4).

To outline the construction the spaces S_n, let us consider at first a (fictitious) one-dimensional chemical system, where $\Omega = (a,b)$ with $a,b \in \mathbb{R}$. This domain can be subdivided into N_cell parts

$$a = x_0 < x_1 < x_2 < \cdots < x_{N_\mathrm{cell}} = b,$$

which do not need to be of equal size (see figure 5.5 on the next page). The open intervals $c_j = (x_j, x_{j+1})$ for $j = 0,1,\ldots,N_\mathrm{cell}-1$ are called grid **cells** and the set $\mathcal{M}_h = \{c_j \,|\, j = 0,1,\ldots,N_\mathrm{cell}-1\}$ of all grid cells is called a **mesh** or a **triangulation**. In this set the index h stands for the maximal size of a grid cell defined as

$$h \equiv \max_{c\in\mathcal{M}_h} \big|\max(c) - \min(c)\big|.$$

Using the vector space

$$\mathbb{P}_k^1 \equiv \left\{ u \in C^\infty(\mathbb{R}) \,\middle|\, u(x) = \sum_{i=0}^{k} c_i x^i, c_i \in \mathbb{R} \right\}$$

of all real polynomials of order at most k, we can define

$$P_k(\mathcal{M}_h) \equiv \left\{ u \in C^0(\overline{\Omega}) \;\middle|\; \forall c \in \mathcal{M}_h :\; u|_{\bar{c}} \in \mathbb{P}_k^1 \right\}, \tag{5.13}$$

the set of piecewise polynomials of at most degree k. The elements of $P_k(\mathcal{M}_h)$ are at least continuous on the complete domain Ω and inside the grid cells they are completely smooth. It can be shown [68, Lemma 4.1] that this implies $P_k(\mathcal{M}_h) \subset H^1(\Omega,\mathbb{R})$. As $h \to 0$ such approximations become more exact, which make $P_k(\mathcal{M}_h)$ the desired approximation spaces of $H^1(\Omega,\mathbb{R})$ for a one-dimensional problem.

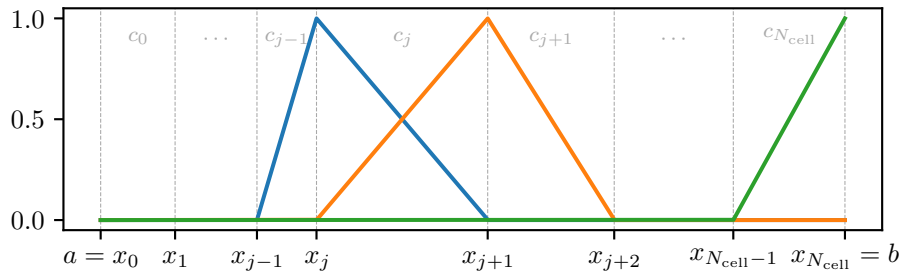


Figure 5.5: A few examples of linear finite elements on a one-dimensional grid with unevenly spread grid points x_0 to $x_{N_{\text{cell}}}$. The cells are split from another by vertical dashed grey lines and the nodal points are indicated as by ticks on the x axis.

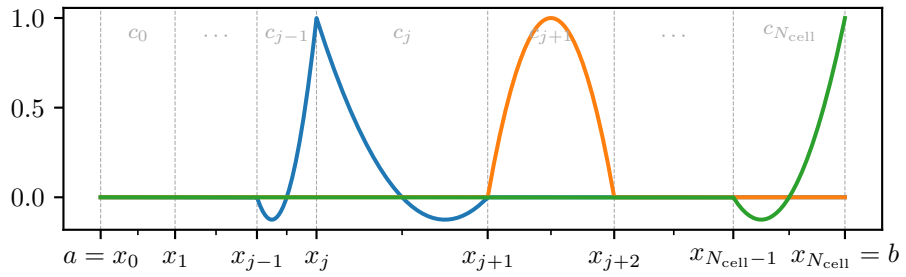


Figure 5.6: Same as figure 5.5, but showing quadratic finite elements in one dimension.

For representing $P_k(\mathcal{M}_h)$ one typically chooses a **Lagrange basis** $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas},h}}$, consisting of basis functions φ_μ with $0 \leq \mu \leq kN_{\text{cell}}$, defined as

$$\varphi_\mu \in P_k(\mathcal{M}_h), \qquad \varphi_\mu(\tilde{x}_\nu) = \delta_{\mu\nu}.$$

In this expression[12]

$$\tilde{x}_\nu = x_{\nu/k} + \frac{\nu \mod k}{k}\left(x_{(\nu/k)+1} - x_{\nu/k}\right) \tag{5.14}$$

where $0 \leq \nu \leq kN_{\text{cell}}$ are the **nodal points**. An alternative term to refer to such basis functions φ_μ is **finite element of order** k, the order k being a reference to the maximal polynomial degree inside the cells c. Examples for linear and quadratic finite elements are illustrated in figures 5.5 and 5.6, respectively. In each case the finite element functions are either 1 or 0 at the nodal points and only non-zero on a few cells, which are always direct neighbours.

In the following this construction is generalised towards a three-dimensional domain Ω. In the most general form a mesh can be defined as

Definition 5.4 (Mesh)**.** Let Ω be a domain in $\mathbb{R}^3$. A mesh is a finite set $\mathcal{M}_h = c_0, c_1, \ldots, c_{N_{\text{cell}}-1}$ of N_{cell} domains c_i with sufficiently regular boundary[13], such that

$$\overline{\Omega} = \bigcup_{i=0}^{N_{\text{cell}}-1} \overline{c_i} \qquad \text{and} \qquad c_i \cap c_j = \emptyset \quad \forall i \neq j,$$

i.e. such that these domains completely partition Ω. Furthermore we set for each $c \in \mathcal{M}_h$ the **cell diameter**

$$h(c) = \max_{\underline{\boldsymbol{x}},\underline{\boldsymbol{y}}\in\overline{c}} \left\|\underline{\boldsymbol{x}} - \underline{\boldsymbol{y}}\right\|_2$$

and call

$$h = \max_{c\in\mathcal{M}_h} h(c)$$

the **mesh size**.

Usually one only considers so-called **affine** meshes.

Definition 5.5. A mesh is called affine if a reference cell c_0 and for each cell $c_i \in \mathcal{M}_h$ affine transformations[14] τ_{c_i} exist, such that $\overline{c_i} = \tau_{c_i}(c_0)$.

In other words a mesh is affine exactly if each grid cell can be generated from the reference cell c_0 by a linear transformation followed by a shift. This work only considers **cuboidal meshes**, where the reference cell $c_0 = [0,1]^3$ is the unit cube. Similar to the construction of the grid cells, the finite elements of order k themselves can be constructed by applying the affine transformations to a set of template polynomials of the same order k. Typically one defines these so-called **shape function**s $\{e_i\}_i$ on the reference cell c_0 and uses the affine transformation τ_c to generate the finite elements on each cell via $\tau_c(e_i)$. A few examples of shape functions in one and two dimensions are illustrated in figure 5.7 and 5.8 on the facing page. Notice that the shape functions in two dimensions have

[12] In equation (5.14) ν/k denotes integer division without remainder.

[13] The boundary of the domains has to be Lipschitz.

[14] A transformation τ is affine iff $\tau(\underline{\boldsymbol{x}}) = \mathbf{A}\underline{\boldsymbol{x}} + \underline{\boldsymbol{b}}$ with $\mathbf{A}$ being a transformation matrix and $\underline{\boldsymbol{b}}$ being a constant shift vector.

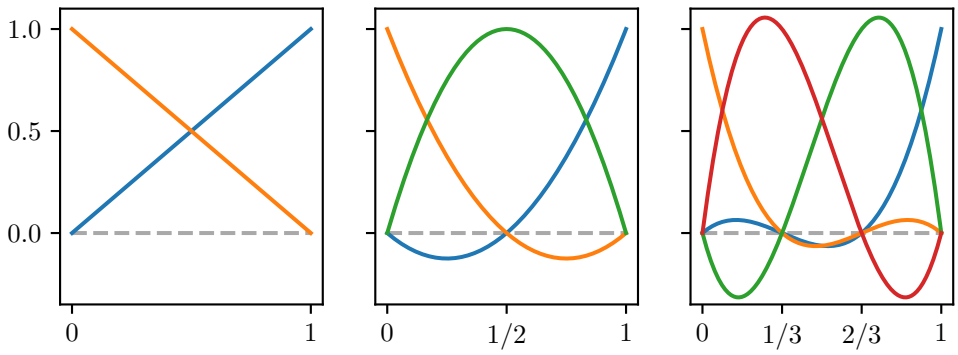


Figure 5.7: The shape functions for polynomial orders $k = 1$, $k = 2$ and $k = 3$ in one dimension.

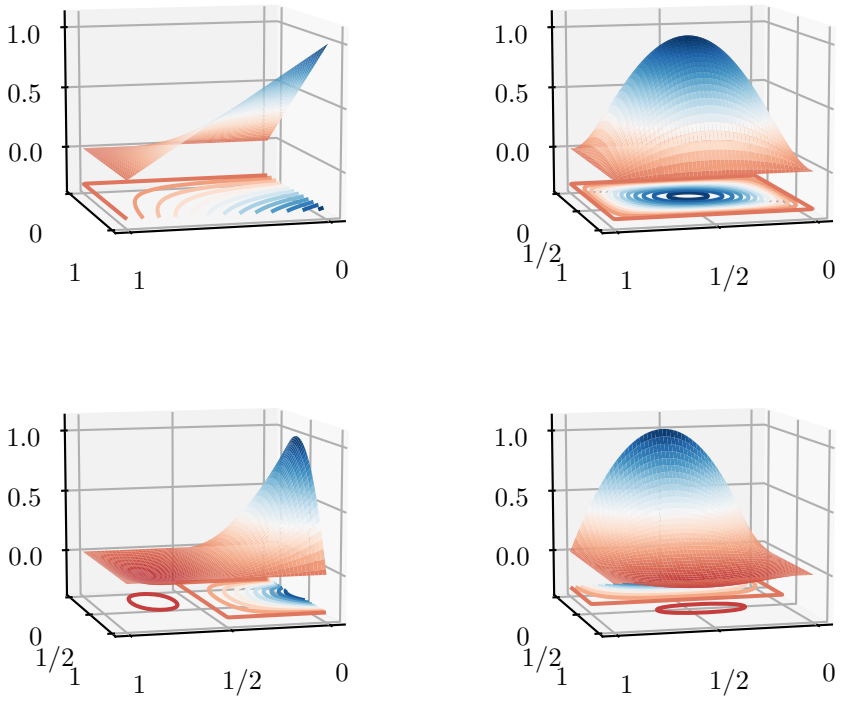


Figure 5.8: Examples for shape functions in two dimensions. The upper left shape function is for $k = 1$, all others for $k = 2$.

been constructed as tensor products from the one-dimensional ones. This construction is a special property of so-called Q_k finite elements, which are typically used in cuboidal meshes. Via the tensor product ansatz Q_k elements in three and higher dimensions can be constructed as well.

Let us denote with S_h the space spanned by all Q_k finite elements $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\mathrm{bas},h}}$ on a cuboidal mesh $\mathcal{M}_h$, which have been constructed by applying appropriate affine maps to a set of shape functions. Even though we always have $S_h \subset H^1(\Omega,\mathbb{R})$, condition (3.4) is *not* necessarily satisfied as $h \to 0$. In other words to ensure convergence of the Ritz-Galerkin procedure in three dimensions a vanishing mesh size is not sufficient. The further required conditions are that the mesh is **uniform** and **shape-regular**. Roughly speaking, these conditions ensure that all grid cells have the same size and their shape is closer to being a ball than to being a needle.

If those conditions are taken into account, an initial mesh can be refined more and more until the HF problem (5.12) is solved up to the desired accuracy. Furthermore, *a priori* and *a posteriori* error estimates can be derived for regular meshes. From these estimates, grid cells which contribute most to the estimated error, can be identified and refined — typically by splitting them into four equal-sized parts. This refinement strategy[15] is called **adaptive refinement**. Since FE grids do not need to be equally spaced, one may well start from a crude initial grid and refine the grid adaptively whilst solving the problem (5.12) until the desired accuracy is achieved. In this manner, the density of grid points is lower where the electron density does not change a lot and is higher where more grid cells are needed to represent the problem properly. Notice that such a process can be automated as well. Compared to a cGTO-based discretisation the finite-element method is therefore truly back box.

There are a couple of drawbacks to the finite-element method, which should not go unmentioned. First of all the tensor product construction of the Q_k finite elements in two and three dimensions implies that the three-dimensional FE basis $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\mathrm{bas},h}}$ is similarly only non-zero in a few cells. Therefore for a proper description of the HF orbitals on the full domain Ω *many* finite elements are required, typically 10^5 to 10^6 [22]. All approaches for solving the numerical problems arising from a FE discretisation may therefore scale at most linearly to be feasible. On the other hand the strict locality of the FE basis functions typically leads to very sparse matrices, such that this is usually no problem if appropriate algorithms are devised.

Secondly, the electron-nuclear cusp tends to be an issue for finite elements as well. For most problems the cell-wise error contains a term involving the gradient of the approximate solution. See the Kelly error estimator [158] for a very simple example. The adaptive refinement process will therefore place a larger amount of grid points — and thus a larger amount of finite-element basis functions — around the regions, where the gradient of the approximated function is large. Both the wave function as well as the HF orbitals have large gradients around the electron-nuclear cusp [5], which is furthermore the only discontinuity of these functions [69]. Even though for most applications the region around the core is not very interesting from a chemical point of view, it thus consumes a large number of FE basis functions for proper representation. In the light of the previous paragraph this is not at all ideal. As a remedy most FE-based approaches to quantum chemistry employ pseudo-potentials to represent the core region [22], leaving only the regions of smaller gradients to be represented by finite elements. Overall this

[15] In practice there is a bit more to it, since the meshes should stay uniform and shape-regular.

significantly reduces the number of finite elements required, but in turn introduces an empirical element ruining the black-box nature. For simplicity we will not consider pseudo-potentials in the remaining discussion about finite elements, but our expressions can be easily modified to incorporate such.

Evaluating the discretised Fock matrix

Let us now consider a particular cuboidal mesh $\mathcal{M}_h$ at some stage during the process of solving (5.12) up to desired accuracy. On $\mathcal{M}_h$ we can construct a set of N_{bas} Q_k finite elements $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas},h}}$ following the procedure outlined above. In a completely analogous procedure to section 4.4.1 on page 64 we use these to discretise (5.12), which results in the non-linear eigenproblem

$$\begin{aligned}\mathbf{F}\Big[\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\Big]\,\mathbf{C}_F^{(n+1)} &= \mathbf{S}\mathbf{C}_F^{(n+1)}\mathbf{E}^{(n+1)}\\ \mathbf{C}^\dagger\mathbf{S}\mathbf{C} &= \mathbf{I}_{N_{\text{elec}}},\end{aligned} \tag{5.15}$$

where

$$\begin{aligned}\mathbf{F}\Big[\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\Big] &= \mathbf{T}+\mathbf{V}_0+\mathbf{J}\Big[\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\Big]+\mathbf{K}\Big[\mathbf{C}^{(n)}\Big(\mathbf{C}^{(n)}\Big)^\dagger\Big],\\ \mathbf{E}^{(n+1)} &= \operatorname{diag}\left(\varepsilon_1^{(n+1)},\varepsilon_2^{(n+1)},\ldots,\varepsilon_{N_{\text{orb}}}^{(n+1)}\right)\in\mathbb{R}^{N_{\text{orb}}\times N_{\text{orb}}}.\end{aligned}$$

By the Aufbau principle the occupied coefficients $\mathbf{C}^{(n+1)}\in\mathbb{R}^{N_{\text{bas}}\times N_{\text{elec}}}$ are as usual the first N_{elec} columns of the full coefficient matrix $\mathbf{C}_F^{(n+1)}\in\mathbb{R}^{N_{\text{bas}}\times N_{\text{orb}}}$. The individual terms of the Fock matrix and the overlap matrix are given by expressions (4.60) to (4.64) just with the integration over $\mathbb{R}^3$ replaced by an integration over Ω. Naturally problem (5.15) can in principle be solved by the self-consistent field procedure of remark 5.1 on page 86.

For evaluating the Fock matrix $\mathbf{F}\Big[\mathbf{C}^{(n)}\big(\mathbf{C}^{(n)}\big)^\dagger\Big]$, let us first consider the terms $\mathbf{T}$ and $\mathbf{V}_0$ as well as the overlap matrix $\mathbf{S}$. This amounts to evaluating integrals

$$O_{\mu\nu}=\int_\Omega \varphi_\mu(\underline{\boldsymbol{r}})\,\hat{\mathcal{O}}\,\varphi_\nu(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}} \qquad \text{where} \quad \hat{\mathcal{O}}=\hat{\mathcal{T}},\hat{\mathcal{V}}_0 \text{ or } \mathrm{id}_{H^1(\Omega,\mathbb{R})}.$$

With reference to the grid $\mathcal{M}_h$ we can write this as a sum of cell contributions $O_{\mu\nu}^c$

$$O_{\mu\nu}=\sum_{c\in\mathcal{M}_h}O_{\mu\nu}^c \qquad \text{where} \quad O_{\mu\nu}^c=\int_c \varphi_\mu(\underline{\boldsymbol{r}})\,\hat{\mathcal{O}}\,\varphi_\nu(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}}.$$

All of the operators $\hat{\mathcal{T}}$, $\hat{\mathcal{V}}_0$ or id are so-called **local operators**, which implies

$$\forall\nu\in\mathcal{I}_{\text{bas},h}: \quad \operatorname{Supp}\left(\hat{\mathcal{O}}\,\varphi_\nu\right)\subseteq\operatorname{Supp}\left(\varphi_\nu\right),$$

where

$$\operatorname{Supp}(\chi)\equiv\{\underline{\boldsymbol{r}}\in\Omega\,|\,\chi(\underline{\boldsymbol{r}})\neq 0\}$$

denotes the **support** of a function χ. In other words $\left(\hat{\mathcal{O}}\varphi_\nu\right)(\underline{\boldsymbol{r}})$ is non-zero only if $\varphi_\nu(\underline{\boldsymbol{r}})$ is non-zero, which implies

$$c\not\subset\operatorname{Supp}(\varphi_\mu)\cap\operatorname{Supp}(\varphi_\nu) \quad\Rightarrow\quad O_{\mu\nu}^c=0.$$

Conversely, to build the matrix $\mathbf{O}$ we only need to consider those elements $O_{\mu\nu}$ where $\mathrm{Supp}(\varphi_\mu) \cap \mathrm{Supp}(\varphi_\nu) \neq \emptyset$. A particular φ_μ only has support in up to $2^3 = 8$ cells. In each cell at most $(k+1)^3$ finite elements have support, such that for a particular $\mu \in \mathcal{I}_{\mathrm{bas},h}$, $O_{\mu\nu}$ can only be non-zero for at most $8(k+1)^3$ values of $\nu \in \mathcal{I}_{\mathrm{bas},h}$. Using a clever ordering of the finite-element functions, one can determine the set of finite elements φ_ν, which couple with a given element φ_μ immediately [159], such that $\mathbf{O}$ can be evaluated by only considering a number of pairs $(\mu,\nu) \in \mathcal{I}_{\mathrm{bas},h} \times \mathcal{I}_{\mathrm{bas},h}$, which scales linearly with the number of finite elements N_{bas}. Furthermore, the cell contributions $O^c_{\mu\nu}$ to $\mathbf{O}$ are independent from one another, such that $\mathbf{O}$ can be determined by an embarrassingly parallel MapReduce step, i.e. one first distributes the computation of the $O^c_{\mu\nu}$ in batches over a number of workers (Map) and then accumulates the result of each in one place (Reduce).

By construction for each finite element φ_μ one can find a shape function e_i such that on a particular cell $c \in \mathcal{M}_h$

$$\varphi_\mu\big|_c(\underline{\boldsymbol{r}}) = e_i\left(\tau_c^{-1}(\underline{\boldsymbol{r}})\right).$$

Let similarly e_j be the shape function corresponding to φ_ν and further let $J_c(\underline{\boldsymbol{\xi}})$ denote the Jacobian matrix of the mapping $\underline{\boldsymbol{r}} = \tau_c(\underline{\boldsymbol{\xi}})$, defined as

$$\forall \alpha,\beta \in \{x,y,z\} : \left(J_c(\underline{\boldsymbol{\xi}})\right)_{\alpha\beta} = \frac{\partial\left(\tau_c(\underline{\boldsymbol{\xi}})\right)_\alpha}{\partial \xi_\beta}.$$

Then we can evaluate $O^c_{\mu\nu}$ as

$$\begin{aligned} O^c_{\mu\nu} &= \int_c \varphi_\mu(\underline{\boldsymbol{r}})\,\hat{\mathcal{O}}\,\varphi_\nu(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}} \\ &= \int_c e_i\left(\tau_c^{-1}(\underline{\boldsymbol{r}})\right)\,\hat{\mathcal{O}}\,e_j\left(\tau_c^{-1}(\underline{\boldsymbol{r}})\right) \\ &= \int_{c_0} e_i(\underline{\boldsymbol{\xi}})\,\hat{\mathcal{O}}\,e_j(\underline{\boldsymbol{\xi}})\det\left(J_c(\underline{\boldsymbol{\xi}})\right)\mathrm{d}\underline{\boldsymbol{\xi}} \\ &= \sum_{q=1}^{N_{\mathrm{quadc}}} e_i(\underline{\boldsymbol{\xi}}_q)\,\hat{\mathcal{O}}\,e_j(\underline{\boldsymbol{\xi}}_q)\det\Big(J_c(\underline{\boldsymbol{\xi}}_q)\Big)\,w_q, \end{aligned} \tag{5.16}$$

where in the last step we introduced a quadrature for the integration using N_{quadc} quadrature points $\underline{\boldsymbol{\xi}}_1, \underline{\boldsymbol{\xi}}_2, \ldots, \underline{\boldsymbol{\xi}}_{N_{\mathrm{quadc}}} \in c_0$ with quadrature weights $w_1, w_2, \ldots, w_{N_{\mathrm{quadc}}}$. Provided that the operator $\hat{\mathcal{O}}$ acting on e_j returns a polynomial[16], the numerical integration in the last step of (5.16) can be made exact using large enough N_{quadc}, since e_i and e_j are only polynomials of order k. Notice that the only quantities in the above sum depending on the cell c are the Jacobian and perhaps the operator $\hat{\mathcal{O}}$. In other words the quadrature itself only needs to be defined with respect to the reference cell and as a result the only required values of the shape function are those at the quadrature points. For a particular combination of quadrature and type of shape function, this could for example be stored in a lookup-table and used for the evaluation of many integrals. Together with the guaranteed linear scaling in the number of matrix elements $O_{\mu\nu}$ which need to be computed as well as the embarrassingly parallel procedure this makes the computation of the matrices $\mathbf{T}$, $\mathbf{V}_0$ and $\mathbf{S}$ extremely efficient despite the large number of

[16] This is the case for example for electrostatic Coulomb potentials or the kinetic energy operator.

basis functions N_{bas} for a finite-element-based discretisation. Since we know the sparsity pattern of pairs of finite elements $(\varphi_\mu, \varphi_\nu)$ with common support already *before* any computation, we can already set-up sensible storage schemes for these matrices and thus avoid storing the known zeros. This leads to linear scaling in storage with respect to the number of finite elements as well.

For the evaluation of the Coulomb term $\mathbf{J}$ and the exchange term $\mathbf{K}$ this naïve approach does not work, unfortunately, since neither $\hat{\mathcal{J}}$ nor $\hat{\mathcal{K}}$ are local operators. Let us first treat the Coulomb term. With reference to (4.50) and (4.62) we can write for all $\mu, \nu \in \mathcal{I}_{\text{bas},h}$

$$J_{\mu\nu}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right] = \int_\Omega \varphi_\mu(\underline{\boldsymbol{r}}_1)\left(\int_\Omega \frac{\rho^{(n)}(\underline{\boldsymbol{r}}_2)}{r_{12}}\,\mathrm{d}\underline{\boldsymbol{r}}_2\right)\varphi_\nu(\underline{\boldsymbol{r}}_1)\,\mathrm{d}\underline{\boldsymbol{r}}_1, \tag{5.17}$$

where we introduced the discretised electron density

$$\rho^{(n)}(\underline{\boldsymbol{r}}) \equiv \sum_{i\in\mathcal{I}_{\text{occ}}}\left|\sum_{\mu\in\mathcal{I}_{\text{bas},h}} C^{(n)}_{\mu i}\varphi_\mu(\underline{\boldsymbol{r}})\right|^2. \tag{5.18}$$

Following classical electrostatics [160] such an electron density gives rise to a potential $V_H^{(n)}(\underline{\boldsymbol{r}})$ defined by a Poisson equation

$$-\Delta V_H^{(n)}(\underline{\boldsymbol{r}}) = 4\pi\rho^{(n)}(\underline{\boldsymbol{r}}) \quad \underline{\boldsymbol{r}} \in \Omega \tag{5.19}$$

with suitable boundary condition on $\partial\Omega$ — see discussion below. In this case $V_H^{(n)}(\underline{\boldsymbol{r}})$ is called the **Hartree potential**. Assuming (5.19) can be solved, we can rewrite (5.17) to give

$$J_{\mu\nu}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right] = \int_\Omega \varphi_\mu(\underline{\boldsymbol{r}}_1)V_H^{(n)}(\underline{\boldsymbol{r}}_1)\varphi_\nu(\underline{\boldsymbol{r}}_1)\,\mathrm{d}\underline{\boldsymbol{r}}_1.$$

Since the Hartree potential $V_H^{(n)}$ is a local operator, this latter integral can be evaluated in $\mathcal{O}(N_{\text{bas}})$ time and space using the cell-wise numerical integration scheme discussed above.

Solving the Poisson equation (5.19) is a well-understood problem in numerical mathematics consisting of just solving a linear system of equations. Using a combination of multigrid preconditioning [161] and a conjugate-gradient linear solver [68], this problem can be solved in $\mathcal{O}(N_{\text{bas}})$.

Let us now address the pending question, which boundary condition to use in equations (5.15) as well as (5.19). First we note, that for non-equally spaced grids, one may take the cells close to the boundary to be rather large. Given that both the Hartree potential $V_H^{(n)}$ as well as the SCF orbitals decay asymptotically, there is less and less change in their values to be expected. In other words a coarse grid will be sufficient in these regions and we can in fact take Ω quite large, say $[-100, 100]^3$ or even larger. If we impose a homogeneous Dirichlet boundary on such a domain, this is still a sensible choice for the SCF orbitals, but it can lead to issues for the Hartree potential, which only falls off as $-1/r$. So even at distances of 10^6 Bohr from the nucleus the potential will is around 10^{-6}. The situation can be improved by approximating the density $\rho^{(n)}(\underline{\boldsymbol{r}})$ at large distances by a point charge in the sense of a multipole expansion. The solution

to the Poisson equation in this case is trivial, yielding the Coulomb potential

$$V_P(\underline{\boldsymbol{r}}) = \frac{N_{\text{elec}} - 1}{r}.$$

For the complete SCF problem (5.15) one could similarly employ a multipole approximation to yield an approximate solution at the boundary, related to what we already did in remark 5.3 on page 93. In both cases such approximate solutions can be enforced using appropriate boundary conditions. The options are

- Dirichlet boundary conditions:

$$V_H^{(n)}(\underline{\boldsymbol{r}}) = V_P(\underline{\boldsymbol{r}}) \qquad \underline{\boldsymbol{r}} \in \partial\Omega \tag{5.20}$$

- Neumann boundary conditions:

$$\partial_n V_H^{(n)}(\underline{\boldsymbol{r}}) = \partial_n V_P(\underline{\boldsymbol{r}}) \qquad \underline{\boldsymbol{r}} \in \partial\Omega, \tag{5.21}$$

  where $\partial_n V_H^{(n)}$ denotes the normal derivative at the boundary $\partial\Omega$.

- Robin boundary conditions:

$$\alpha(\underline{\boldsymbol{r}}) V_H^{(n)}(\underline{\boldsymbol{r}}) = \partial_n V_H(\underline{\boldsymbol{r}}) \qquad \underline{\boldsymbol{r}} \in \partial\Omega \tag{5.22}$$

  where $\alpha(\underline{\boldsymbol{r}})$ is determined from

$$\alpha(\underline{\boldsymbol{r}}) V_P(\underline{\boldsymbol{r}}) = \partial_n V_P(\underline{\boldsymbol{r}})$$

For the Poisson equation Robin boundary conditions (5.22) usually work best in practice, since they enforce resemblance of the gradient and the value of $V_P(\underline{\boldsymbol{r}})$ at the same time.

In theory there is no reason why one should use the same discretisation for solving the Poisson equation (5.19) and for solving the HF equations (5.12). Using different meshes is possible, but leads to complications when projecting the Hartree potential $V_H^{(n)}$ onto the grid used for solving the HF equations. The use of different polynomial orders, for example, has been investigated by Davydov et al. [22] in the context of the related Kohn-Sham equations. Their results suggest to use twice the polynomial order for solving the Poisson equation compared to the polynomials used for the HF problem. This can be rationalised by looking at the expression (5.18) for the discretised density. If φ_μ and φ_ν denote two Q_k finite elements, which are used for the discretisation of (5.12), solving the Poisson equation (5.19) requires the representation of the density $\rho^{(n)}(\underline{\boldsymbol{r}})$, which consists of products $\varphi_\mu \cdot \varphi_\nu$. These can only be represented exactly if at least Q_{2k} elements are used to discretise (5.19).

Now we consider the exchange term. Using (4.51) and (4.63) we can deduce an expression of the exchange matrix elements. For all $\mu, \nu \in \mathcal{I}_{\text{bas},h}$ we get

$$K_{\mu\nu}^{(n)} \equiv K_{\mu\nu}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right] = -\int_\Omega \int_\Omega \varphi_\mu(\underline{\boldsymbol{r}}_1) \frac{\gamma^{(n)}(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2)}{r_{12}} \varphi_\nu(\underline{\boldsymbol{r}}_2)\, \mathrm{d}\underline{\boldsymbol{r}}_2\, \mathrm{d}\underline{\boldsymbol{r}}_1, \tag{5.23}$$

where we introduced the discretised one-particle reduced density matrix

$$\gamma^{(n)}(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2) \equiv \sum_{i \in \mathcal{I}_{\text{occ}}} \sum_{\mu,\nu \in \mathcal{I}_{\text{bas},h}} C_{\mu i}^{(n)} \varphi_\mu(\underline{\boldsymbol{r}}_1)\, C_{\nu i}^{(n)} \varphi_\nu(\underline{\boldsymbol{r}}_2). \tag{5.24}$$

The double integral (5.23) can be split into a sum of contributions from each grid cell pair $(c,d) \in (\mathcal{M}_h)^2$

$$K_{\mu\nu}^{(n)} = -\sum_{c,d\in\mathcal{M}_h} \int_c \int_d \frac{\varphi_\mu(\underline{\boldsymbol{r}}_1)\,\gamma^{(n)}(\underline{\boldsymbol{r}}_1,\underline{\boldsymbol{r}}_2)\,\varphi_\nu(\underline{\boldsymbol{r}}_2)}{r_{12}}\,\mathrm{d}\underline{\boldsymbol{r}}_2\,\mathrm{d}\underline{\boldsymbol{r}}_1$$

$$= -\sum_{c,d\in\mathcal{M}_h} \int_{c_0} \int_{c_0} \frac{e_i(\underline{\boldsymbol{\xi}}_1)\,\gamma^{(n)}\Big(\tau_c(\underline{\boldsymbol{\xi}}_1),\tau_d(\underline{\boldsymbol{\xi}}_2)\Big)\;e_j(\underline{\boldsymbol{\xi}}_2)}{\Big\|\tau_c(\underline{\boldsymbol{\xi}}_1)-\tau_d(\underline{\boldsymbol{\xi}}_2)\Big\|_2}\,\mathrm{d}\underline{\boldsymbol{\xi}}_1\,\mathrm{d}\underline{\boldsymbol{\xi}}_2,$$

where e_i is the shape function corresponding to φ_μ and e_j the one corresponding to φ_ν. Introducing two quadratures $\mathcal{Q}$ and $\mathcal{Q}'$ with quadrature points $\underline{\boldsymbol{\xi}}_q$, $\underline{\boldsymbol{\xi}}'_r$ and corresponding weights w_q and w'_r yields

$$K_{\mu\nu}^{(n)} \simeq -\sum_{c,d\in\mathcal{M}_h} \sum_{q=1}^{N_{\mathrm{quadc}}} \sum_{r=1}^{N'_{\mathrm{quadc}}} \frac{e_i(\underline{\boldsymbol{\xi}}_q)\,\gamma^{(n)}\Big(\tau_c(\underline{\boldsymbol{\xi}}_q),\tau_d(\underline{\boldsymbol{\xi}}'_r)\Big)\;e_j(\underline{\boldsymbol{\xi}}'_r)}{\Big\|\tau_c(\underline{\boldsymbol{\xi}}_q)-\tau_d(\underline{\boldsymbol{\xi}}'_r)\Big\|_2} w_q w'_r.$$

Notice that in contrast to (5.16) the right-hand side expression is *not* exactly equal to the matrix element $K_{\mu\nu}^{(n)}$, since there are a couple of issues. First of all there is the $1/r_{12}$ singularity, which becomes $\Big\|\tau_c(\underline{\boldsymbol{\xi}}_q)-\tau_d(\underline{\boldsymbol{\xi}}'_r)\Big\|_2^{-1}$ after the introduction of a numerical quadrature. If the resulting matrix elements should be numerically meaningful one needs to at least make sure that the quadrature points $\underline{\boldsymbol{\xi}}_q$ and $\underline{\boldsymbol{\xi}}'_r$ are rather different for both quadratures in order to avoid divergence. More properly one needs to use a particular quadrature scheme, suitable for integrating this singularity or one needs a lot of quadrature points. Already this aspect makes the construction of $\mathbf{K}$ more challenging than the other matrices. Additionally, the non-local nature of HF exchange really comes into play as well. Unlike the previous Fock matrix terms no immediate criterion for excluding some pairs of finite element indices (μ,ν) can be found from the derived expression.

Each element $K_{\mu\nu}^{(n)}$ can be evaluated in $\mathcal{O}(k^6 N_{\mathrm{quadc}} N'_{\mathrm{quadc}} N_{\mathrm{elec}})$ computational time. To see this, let us first consider the evaluation of $\gamma^{(n)}\Big(\tau_c(\underline{\boldsymbol{\xi}}_q),\tau_d(\underline{\boldsymbol{\xi}}'_r)\Big)$ on one cell pair (c,d) and for one pair of quadrature points $(\underline{\boldsymbol{\xi}}_q,\underline{\boldsymbol{\xi}}'_r)$. With each cell c only $\mathcal{O}(k^3)$ finite element functions share support. Therefore, there will be at most $\mathcal{O}(N_{\mathrm{elec}}(k^3)^2)$ terms in (5.24) which are non-zero. In other words evaluating $\gamma^{(n)}\Big(\tau_c(\underline{\boldsymbol{\xi}}_q),\tau_d(\underline{\boldsymbol{\xi}}'_r)\Big)$ takes $\mathcal{O}(N_{\mathrm{elec}}k^6)$, which gives rise to a cost of $\mathcal{O}(k^6 N_{\mathrm{quadc}} N'_{\mathrm{quadc}} N_{\mathrm{elec}})$ to evaluate a single element $K_{\mu\nu}^{(n)}$. Overall the computational scaling for the naïve procedure outlined above is therefore $\mathcal{O}(N_{\mathrm{bas}}^2)$. In storage we would expect the same quadratic scaling, which is highly undesirable.

Theoretically one would expect that this can be improved by considering some distance cut-off. The physical justification for this is the exponential decay of the wave function, which causes the density matrix $\gamma^{(n)}(\underline{\boldsymbol{r}}_1,\underline{\boldsymbol{r}}_2)$ to decay exponentially with $\|\underline{\boldsymbol{r}}_1-\underline{\boldsymbol{r}}_2\|_2$ as well, provided that the discretisation is sensible for describing the problem. In combination with the additional decay of $1/r_{12}$ it should be possible to *a priori* exclude some index pairs (μ,ν) and thus reduce the scaling.

From the preliminary results I obtained, I would not expect this attempt to be beneficial by its own and that further strategies are required. To illustrate this, consider

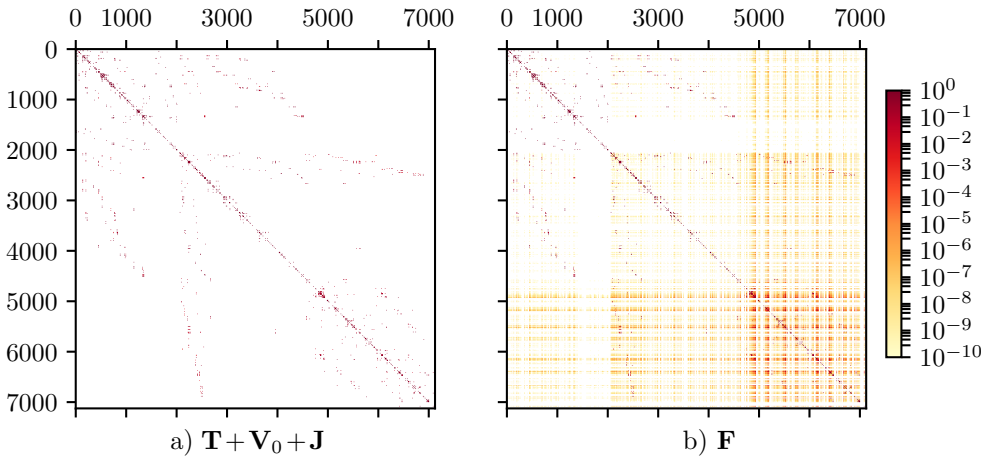


Figure 5.9: Structure of the local terms $\mathbf{T}+\mathbf{V}_0+\mathbf{J}$ and the Fock matrix $\mathbf{F}$ for a finite-element-based HF treatment of the beryllium atom in three dimensions in a small FE basis of around 7000 basis functions. The Pulay error of the depicted Fock matrix is around 0.1. The colouring depends on the absolute value of the entries with entries smaller than 10^{-10} being shown in white.

figure 5.9, where both the structure of the local terms $\mathbf{T}+\mathbf{V}_0+\mathbf{J}$ and the structure of the complete Fock matrix

$$\mathbf{F} = \mathbf{T}+\mathbf{V}_0+\mathbf{J}+\mathbf{K}$$

is depicted for a closed-shell treatment of the beryllium atom. Notice that we only used a rather small finite-element basis with around 7000 finite element functions. The colouring depends on the absolute value of the entries, where white indicates values less than 10^{-10}. From the figure it is immediately visible that the extra exchange term $\mathbf{K}$ seems to play a major role only in some blocks of the Fock matrix, but not so much in others. Whilst this probably allows for neglecting to evaluate some blocks of $\mathbf{K}$, still a large amount of elements cannot be ignored. Notice that even in the upper left corner of $\mathbf{F}$ some elements originating from $\mathbf{K}$ are larger than 10^{-10}. Compared to the structure of the local terms $\mathbf{T}+\mathbf{V}_0+\mathbf{J}$, even a clever reordering scheme will probably not improve the sparsity structure very much. I therefore believe it to be challenging if not impossible to achieve an $\mathcal{O}(N_{\text{bas}})$-scaling in the number of matrix entries $K_{\mu\nu}^{(n)}$, which have to be computed.

An alternative strategy is to avoid building and storing the matrix $\mathbf{K}$ at all and instead recompute its elements whenever needed. At first sight this does not seem to make the problem any easier, yet it even appears to lead to more computations rather than less. But as will be demonstrated for the example of the application of the exchange matrix $\mathbf{K}$ to an arbitrary vector $\underline{\boldsymbol{x}} \in \mathbb{R}^{N_{\text{bas}}}$, this is not always true. The trick is usually that changing the order of summation and integration often allows to compute the elements of a matrix like $\mathbf{K}$ in a more efficient way, reducing the overall computational scaling to $\mathcal{O}(N_{\text{bas}})$. For easier writing of the following algebra, let

$$\psi_i^{(n)}(\underline{\boldsymbol{r}}) = \sum_{\mu \in \mathcal{I}_{\text{bas},h}} C_{\mu i}^{(n)} \varphi_\mu(\underline{\boldsymbol{r}})$$

such that

$$\gamma^{(n)}(\underline{\boldsymbol{r}}_1, \underline{\boldsymbol{r}}_2) = \sum_{i \in \mathcal{I}_{\text{occ}}} \psi_i^{(n)}(\underline{\boldsymbol{r}}_1)\psi_i^{(n)}(\underline{\boldsymbol{r}}_2)$$

Using expression (5.23) we can write the application of $\mathbf{K}$ to a vector $\underline{\boldsymbol{x}}$ as:

$$\begin{aligned}
\left(\mathbf{K}^{(n)}\underline{\boldsymbol{x}}\right)_\mu &= -\sum_{\nu \in \mathcal{I}_{\text{bas},h}} x_\nu K_{\mu\nu}^{(n)} \\
&= -\sum_{\nu \in \mathcal{I}_{\text{bas},h}} x_\nu \int_\Omega \int_\Omega \varphi_\mu(\underline{\boldsymbol{r}}_1) \sum_{i \in \mathcal{I}_{\text{occ}}} \frac{\psi_i^{(n)}(\underline{\boldsymbol{r}}_1)\psi_i^{(n)}(\underline{\boldsymbol{r}}_2)}{r_{12}} \varphi_\nu(\underline{\boldsymbol{r}}_2)\,\mathrm{d}\underline{\boldsymbol{r}}_2\,\mathrm{d}\underline{\boldsymbol{r}}_1 \\
&= -\sum_{i \in \mathcal{I}_{\text{occ}}} \int_\Omega \varphi_\mu(\underline{\boldsymbol{r}}_1)\psi_i^{(n)}(\underline{\boldsymbol{r}}_1) \left(\int_\Omega \sum_{\nu \in \mathcal{I}_{\text{bas},h}} \frac{\psi_i^{(n)}(\underline{\boldsymbol{r}}_2)\, x_\nu \varphi_\nu(\underline{\boldsymbol{r}}_2)}{r_{12}}\,\mathrm{d}\underline{\boldsymbol{r}}_2 \right) \mathrm{d}\underline{\boldsymbol{r}}_1 \\
&= -\sum_{i \in \mathcal{I}_{\text{occ}}} \sum_{\kappa \in \mathcal{I}_{\text{bas},h}} \int_\Omega \varphi_\mu(\underline{\boldsymbol{r}}_1)\, C_{\kappa i}^{(n)} \varphi_\kappa(\underline{\boldsymbol{r}}_1) \left(\int_\Omega \frac{\rho_{\underline{\boldsymbol{x}},i}^{(n)}(\underline{\boldsymbol{r}}_2)}{r_{12}}\,\mathrm{d}\underline{\boldsymbol{r}}_2 \right) \mathrm{d}\underline{\boldsymbol{r}}_1,
\end{aligned} \tag{5.25}$$

where we introduced the **exchange contraction densities**

$$\rho_{\underline{\boldsymbol{x}},i}^{(n)}(\underline{\boldsymbol{r}}) = \sum_{\nu,\lambda \in \mathcal{I}_{\text{bas},h}} C_{\lambda i}^{(n)} \varphi_\lambda(\underline{\boldsymbol{r}}_2)\, x_\nu \varphi_\nu(\underline{\boldsymbol{r}}_2). \tag{5.26}$$

For each $i \in \mathcal{I}_{\text{occ}}$ we can solve a Poisson equation in analogy to (5.19)

$$-\Delta V_{\underline{\boldsymbol{x}},i}(\underline{\boldsymbol{r}}) = 4\pi \rho_{\underline{\boldsymbol{x}},i}^{(n)}(\underline{\boldsymbol{r}}) \quad \underline{\boldsymbol{r}} \in \Omega, \tag{5.27}$$

where the boundary is fixed using one of (5.20) to (5.22). Solving (5.27) defines implicitly the **exchange contraction potentials** $V_{\underline{\boldsymbol{x}},i}^{(n)}$. With these potentials (5.25) becomes

$$\left(\mathbf{K}^{(n)}\underline{\boldsymbol{x}}\right)_\mu = -\sum_{i \in \mathcal{I}_{\text{occ}}} \sum_{\kappa \in \mathcal{I}_{\text{bas},h}} \int_\Omega \varphi_\mu(\underline{\boldsymbol{r}})\, C_{\kappa i}^{(n)} \varphi_\kappa(\underline{\boldsymbol{r}}) V_{\underline{\boldsymbol{x}},i}^{(n)}(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}}. \tag{5.28}$$

Since $V_{\underline{\boldsymbol{x}},i}^{(n)}$ is a local operator, the integrals in (5.28) can be evaluated in $\mathcal{O}(k^3 N_{\text{bas}} N_{\text{quadc}})$ once the potentials $V_{\underline{\boldsymbol{x}},i}^{(n)}$ are known. Consequently, the complete expression (5.28) can be computed in $\mathcal{O}(k^3 N_{\text{quadc}} N_{\text{elec}} N_{\text{bas}})$ time.

Assuming the same grid and polynomial order are used[17] for solving the Poisson equations (5.27) and for discretising (5.12), each of the exchange contraction densities (5.26) can be evaluated on the grid in $\mathcal{O}(k^3 N_{\text{quadc}} N_{\text{bas}})$. Solving each Poisson equation (5.27) is again $\mathcal{O}(N_{\text{bas}})$, such that overall obtaining the potentials $V_{\underline{\boldsymbol{x}},i}^{(n)}$ takes $\mathcal{O}(k^3 N_{\text{quadc}} N_{\text{elec}} N_{\text{bas}})$ time. Even though computing the complete matrix $\mathbf{K}$ is quadratic in N_{bas}, the application to a vector $\underline{\boldsymbol{x}}$ can be done in $\mathcal{O}(N_{\text{bas}})$ computational time with this scheme.

Many iterative diagonalisation algorithms, like the Lanczos method (see section 3.2.5) or Davidson's method (see section 3.2.6) do not make explicit reference the elements of

[17] Similar to the results of [22] for the Poisson-solves in (5.19), it seems reasonable that one might need to go to *twice* the polynomial order for solving (5.27) as well. This does not change our analysis, however, since it just introduces a constant factor.

the matrix to be diagonalised. Much rather they only require a way to perform the matrix-vector product. In this manner obtaining a few selected eigenpairs is possible *without* having the complete matrix in memory. In the context of the finite-element method so-called **matrix-free methods** have been developed recently [162]. These follow this strategy and avoid building any finite-element-discretised form of any operator in memory. This includes local operators like $\hat{\mathcal{T}}$, $\hat{\mathcal{V}}_0$ or $\hat{\mathcal{J}}$ in our case. If properly preconditioned iterative linear solvers are used, such methods tend to perform better [162] than their traditional counterparts. For reasons which will become more clear in the next chapter, I will refer to efforts, where storing matrix data is avoided in favour of a matrix-vector contraction expression as contraction-based methods instead.

For achieving a finite-element-based HF this seems to be a very promising ansatz as well. Already lifting the requirement to build the Fock matrix $\mathbf{F}$ and instead only employ iterative diagonalisation algorithms allows to formally reduce the scaling from $\mathcal{O}(N_{\text{bas}}^2)$ to $\mathcal{O}(N_{\text{bas}})$ — in both storage and time. In our preliminary implementation of such a scheme, we were, however, not able to implement such an ansatz successfully. The biggest challenge is the application of the exchange term $\mathbf{K}$. Even though the formal scaling is linear in N_{bas}, one still needs to solve the Poisson equations (5.27) a lot of times. Already for each matrix-vector application N_{elec} Poisson-solves are required. For both Davidson and Lanczos one usually needs around 50 iterations, with a couple of hundred matrix-vector products to be computed. If we further need around 30 SCF steps, this altogether makes some 1000 Poisson-solves already for very small chemical systems. The only way this can be achieved is by proper approximations, proper preconditioning and the caching of important intermediate results. Beyond the multigrid preconditioning we already mentioned in the context of the Coulomb term (5.19), other options for preconditioning include an incomplete Choleski factorisation [163] of the discretised form of Δ or potentially even an exact sparse inversion using libraries like UMFPACK [164] Even though such approaches are comparatively costly, they are extremely good preconditioners up to the point where solving the Poisson equations (5.27) reduces to a few manageable matrix-vector products. Since Δ is essentially equivalent to $\mathbf{T}$ and occurs both is the computation of $\mathbf{J}$ as well as the equations (5.27) for applying $\mathbf{K}$ to a vector, the costs could amortise overall. If one accepts storing the discretised form of Δ as well as its sparse Choleski factorisation or its exact sparse inverse [164], one can reduce the costs even further, since these quantities only need to be computed once for each discretisation grid and not once per SCF cycle. Another problem in the proposed scheme for FE-based HF is numerical stability. In integrals like (5.28) the integrand is no longer a simple polynomial function, but could have a rather complicated functional form, such that higher quadrature orders than usual could be necessary. Furthermore, our experiments suggest, that the Poisson equations (5.27) need to be solved to high numerical accuracy in order to result in meaningful eigenpairs in the iterative diagonalisation method. Due to these challenges a practically useful implementation of the presented ansatz is still pending.

Despite these numerical challenges finite-element-based HF is a promising approach. Once clear relationships between the quadrature orders and the required accuracies between the iterative solvers are known, the error is completely controlled by the discretisation itself. An adaptively refined grid should therefore allow to solve the HF problem up to arbitrary precision. In contrast to the cGTO discretisation, where a basis set has to be selected *prior* to the calculation, the finite-element method amounts to build an appropriate basis as it solves the problem. Even though no further results from

FE-based quantum chemistry will be presented in this work, many decisions that lead to the program and algorithm design of `molsturm` (see chapter 7 on page 153) keep the numerical requirements of finite elements in mind as well.

5.3.6 Coulomb-Sturmian-type orbitals

Coulomb-Sturmians (CS) are another type of atom-centred basis functions, which so far have seen little attention in electronic structure theory. Similar to Slater-type orbitals they were introduced [24] as a generalisation to the solutions of the Schrödinger equation for hydrogen-like atoms. CS functions cannot be used for molecules, only for atoms, but closely related functions exist, which are more generally applicable. The main motivation for Shull and Löwdin [24] to look into alternative exponential functions was that they wanted to construct one-electron basis functions, which could be used to compute the spectra of many-electron atoms. From previous approaches it was known that a proper representation of the wave function required the inclusion of the continuum [29] if hydrogen-like orbital functions were used. This can be rationalised by the fact that hydrogen-like orbitals — except the $1s$ — are comparatively diffuse [165]. Their classical turning point, i.e. the distance r where they intersect with the Coulomb potential $-Z/r$, increases roughly as $\mathcal{O}(n^2)$ for the s-like functions. In other words with increasing principle quantum number n, the hydrogen-like orbitals very quickly become unsuitable for the description of bound atomic states, which are residing close to the nucleus. Increasing the basis by including more hydrogen-like functions with an even larger n allows to correct for this, but convergence will be slow as the included states become more and more continuum-like.

To avoid this dilemma, Shull and Löwdin [24] artificially modified the Schrödinger equation (2.41) for hydrogen-like atoms, such that it was on the one hand still analytically solvable, but on the other hand the spectrum of Helium could be modelled up to a rather good level of accuracy, even without explicit inclusion of the continuum. Effectively their trick was to multiply the Coulomb term in (2.41) by a prefactor

$$\beta_n = \frac{k_{\text{exp}} n}{Z} \tag{5.29}$$

with $k_{\text{exp}} \in \mathbb{R}$ arbitrary to yield

$$\left(-\frac{1}{2}\Delta - \beta_n \frac{Z}{r} - E\right) \varphi_\mu^{\text{CS}}(\underline{\boldsymbol{r}}) = 0. \tag{5.30}$$

This equation has a countably infinite number of solutions $\varphi_\mu^{\text{CS}} \in H^1(\mathbb{R}^3, \mathbb{C})$, which are the so-called **Coulomb-Sturmians**. They are **isoenergetic**, i.e. all have the identical energy eigenvalue

$$E = -\frac{k_{\text{exp}}^2}{2}, \tag{5.31}$$

such that the underlying self-adjoint operator

$$\hat{\mathcal{H}}^{\text{CS}} = -\frac{1}{2}\Delta - \frac{n k_{\text{exp}}}{r}$$

has the very simple point spectrum

$$\sigma_P(\hat{\mathcal{H}}^{\text{CS}}) = \left\{ -\frac{k_{\text{exp}}^2}{2} \right\},$$

but an empty discrete spectrum, thus $\sigma_P(\hat{\mathcal{H}}^{\mathrm{CS}}) \subset \sigma_{\mathrm{ess}}(\hat{\mathcal{H}}^{\mathrm{CS}})$.

Since (5.30) and the hydrogen-like Schrödinger equation (2.41) are very similar, we can apply the solution approach discussed in section 2.3.5 on page 28 to equation (5.30) as well. Inserting a product ansatz of radial part and spherical harmonic

$$\varphi_\mu^{\mathrm{CS}}(\underline{\boldsymbol{r}}) \equiv \varphi_{nlm}^{\mathrm{CS}}(\underline{\boldsymbol{r}}) = R_{nl}^{\mathrm{CS}}(r) Y_l^m(\hat{\underline{\boldsymbol{r}}}) \equiv R_{nl}^{\mathrm{CS}}(r) Y_l^m(\theta, \phi) \tag{5.32}$$

into (5.30) we obtain the Coulomb-Sturmian radial equation

$$\left(-\frac{1}{2r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial}{\partial r} \right) + \frac{l(l+1)}{2r^2} - \frac{n k_{\mathrm{exp}}}{r} - E \right) R_{nl}(r) = 0. \tag{5.33}$$

Its solutions have the form

$$R_{nl}^{\mathrm{CS}}(r) = N_{nl} (2k_{\mathrm{exp}} r)^l e^{-k_{\mathrm{exp}} r} \, {}_1F_1(l+1-n|2l+2|2k_{\mathrm{exp}} r) \tag{5.34}$$

with normalisation constant

$$N_{nl} = \frac{2k_{\mathrm{exp}}^{3/2}}{(2l+1)!} \sqrt{\frac{(l+n)!}{n(n-l-1)!}}$$

and ${}_1F_1(a|b|z)$ being the confluent hypergeometric function as defined in (2.45). Unsurprisingly this functional form is closely related to the radial part of the hydrogen-like orbitals (2.44). In fact the Coulomb-Sturmians can be constructed from the equivalent hydrogen-like orbitals just by replacing the factors Z/r by k_{exp}. In analogy one therefore commonly uses the spectroscopic terminology $1s$, $2s$, $2p$, ... to describe the respective triples of quantum numbers (n, l, m) for Coulomb-Sturmians as well. Originating from the same arguments as discussed in section 2.3.5 the full range of possible triples is

$$\mathcal{I}_F \equiv \left\{ (n,l,m) \,\middle|\, n,l,m \in \mathbb{Z} \quad \text{with} \quad n > 0,\ 0 \leq l < n,\ -l \leq m \leq l \right\}. \tag{5.35}$$

Given that both STOs as well as CS functions are exponential type orbitals of the form radial part times spherical harmonic, their radial parts (5.34) and (5.9) are related as well[18]. The important difference between both types of orbitals is that STO basis sets may use a different Slater exponent ζ_μ for each STO basis function, whereas all CS functions share the same exponent k_{exp} as a commonly modified parameter. Even though this difference is subtle, it is the key ingredient to derive the efficient evaluation schemes of the CS-ERI tensor discussed further down this section.

In their original work, Shull and Löwdin [24] did not yet use the term "Coulomb-Sturmians" to refer to the functions $\varphi_\mu^{\mathrm{CS}}$. This name was only introduced a few years later by Rotenberg [25, 26], who managed to find a link between the CS radial equation (5.33) and the special class of Sturm-Liouville differential equations. Sturm-Liouville equations are second order differential equation of the form

$$\left(\frac{\mathrm{d}}{\mathrm{d}r} \left(p(r) \frac{\mathrm{d}}{\mathrm{d}r} \right) + q(r) + \lambda_n w(r) \right) u_n(r) = 0, \tag{5.36}$$

where $p(r) \in C^1(\Omega, \mathbb{R})$ and $q(r), w(r) \in C^0(\Omega, \mathbb{R})$ are all positive functions and $\Omega = (a, b) \subset \mathbb{R}$ is an open interval. Provided that on a and b suitable boundary conditions

$$\begin{aligned} u_i(a) \cos\alpha - p(a) u_i'(a) \sin\alpha &= 0 \qquad & 0 < \alpha < \pi \\ u_i(b) \cos\beta - p(b) u_i'(b) \sin\beta &= 0 \qquad & 0 < \beta < \pi \end{aligned}$$

[18] In fact, some recent work [32] exploits this to evaluate STO ERI integrals with Coulomb-Sturmians.

are chosen, the eigenvalues λ_i are real and non-degenerate

$$\lambda_1 < \lambda_2 < \lambda_3 < \cdots < \lambda_n < \cdots \to \infty$$

and the eigenfunctions u_i can be normalised to satisfy the weighted orthonormality condition

$$\int_a^b u_i^*(r) w(r) u_j(r)\,\mathrm{d}r = \delta_{ij}. \tag{5.37}$$

Following Rotenberg [25, 26], one can use the ansatz

$$R_{nl}(r) = \frac{u_{nl}(r)}{r}$$

as well as (5.31) to rewrite the Coulomb-Sturmian radial equation (5.33) as

$$\left(\frac{\partial^2}{\partial r^2} - \frac{l(l+1)}{r^2} - \frac{k^2}{2} + \frac{kn}{r}\right) u_{nl} = 0,$$

which is of Sturm-Liouville form with

$$p(r) = 1, \qquad q(r) = \frac{k_{\mathrm{exp}}^2}{2} + \frac{l(l+1)}{r^2}, \qquad \lambda_n w(r) = \frac{n k_{\mathrm{exp}}}{r}$$

One consequence of this is that Coulomb-Sturmians satisfy the **potential-weighted orthonormality** condition [29]

$$\int_{\mathbb{R}^3} \left(\varphi_{nlm}^{\mathrm{CS}}(\underline{\boldsymbol{r}})\right)^* \frac{n}{r k_{\mathrm{exp}}} \varphi_{n'l'm'}^{\mathrm{CS}}(\underline{\boldsymbol{r}})\,\mathrm{d}\underline{\boldsymbol{r}} = \delta_{nn'}\delta_{ll'}\delta_{mm'}. \tag{5.38}$$

Most importantly, however, it is possible to show that the countably infinite set of all Coulomb-Sturmians $\{\varphi_\mu^{\mathrm{CS}}\}_{\mu\in\mathcal{I}_F}$ is a complete basis for $H^1(\mathbb{R}^3, \mathbb{R})$ [165, Theorem 2.3.4]. In the original context of Shull and Löwdin this implies that Coulomb-Sturmians are not only able to represent the bound states of any atomic Schrödinger operator $\hat{\mathcal{H}}_{N_{\mathrm{elec}}}$, but continuum-like states as well. When it comes to the discretised HF problem (see section 4.4.1) or the FCI problem (see remark 4.8), this makes CS basis functions rather promising, since the completeness property provides a mathematical guarantee that the exact solution can be approximated arbitrarily closely if more and more CS functions are included.

Another remarkable property of the Coulomb-Sturmians is the ability to map the set of all Coulomb-Sturmians $\{\varphi_\mu^{\mathrm{CS}}\}_{\mu\in\mathcal{I}_F}$ one-to-one onto the set of all hyperspherical harmonics, the eigenfunctions of the Laplace-Beltrami operator on the surface of a four-dimensional hypersphere. This can be achieved by applying the Fock transformation to the Fourier-transformed Coulomb-Sturmians [29]. This aspect is a key ingredient to treat multi-centre integrals involving Sturmian-type orbitals in a numerically efficient manner [31, 32, 166–169].

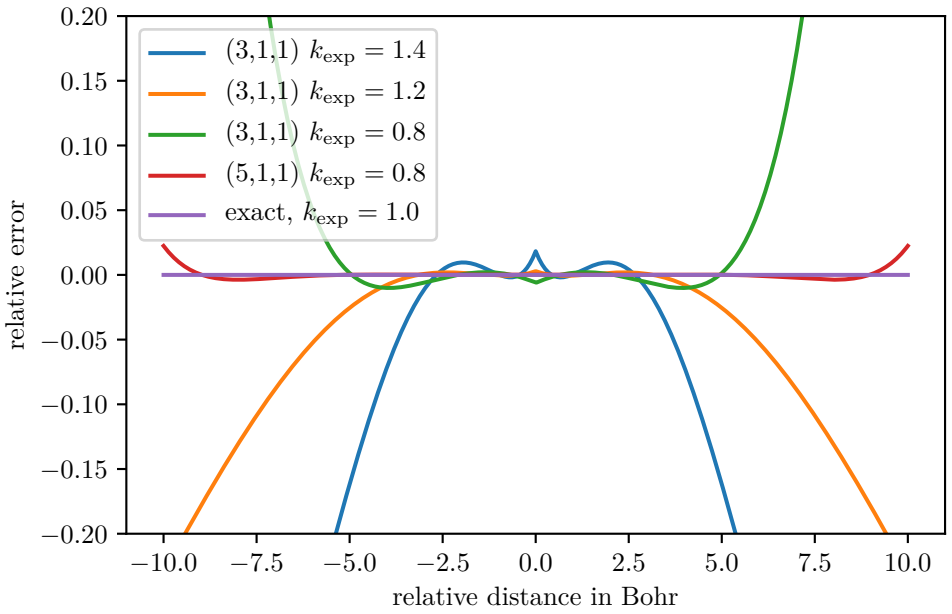


Figure 5.10: Relative error in the hydrogen ground state for selected CS basis sets. The error is plotted against the relative distance of electron and proton. The optimal value for k_{exp} for hydrogen is 1.0, which is exact.

Since CS functions contain the term

$$\exp(-k_{\text{exp}} r) = \exp\left(-\sqrt{-2E}r\right),$$

which both gives rise to a cusp at $r = 0$ as well as an energy-dependent exponential decay at $r \to \infty$, they reflect the physical properties summarised in remark 5.3 on page 93 already at the level of basis functions. As mentioned above a CS basis has exactly one exponent k_{exp}, which is used in all basis functions of the CS basis. For atomic systems other than hydrogen, where multiple electrons of deviating asymptotic decays are present, one k_{exp} therefore needs to be chosen to model all electrons of an atom and thus one needs to make a compromise. Due to the completeness of the CS basis this is not an issue, since a large enough basis will recover the errors for each k_{exp}, such that in theory *any* k_{exp} could be chosen. In practice this is not quite the case, since the rate of convergence of a CS discretisation does well depend on k_{exp}, see [32] as well as section 8.4 on page 186 for a more detailed discussion. A more suitable value for k_{exp} will thus give rise to a better representation of the physics at a smaller sized CS basis.

We will now investigate how the error in a CS discretisation changes if we move away from the optimal value for k_{exp}. Figure 5.10 shows the relative error in the hydrogen ground state versus the relative electron-nucleus distance for a few selected Coulomb-Sturmian basis sets. The labels of the plots both indicate the k_{exp} value as well as the triple $(n_{\max}, l_{\max}, m_{\max})$, which is a short hand for indicating the finite basis

$$\left\{ \varphi^{\text{CS}}_{nlm} \,\middle|\, (n,l,m) \in \mathcal{I}_F \; n \leq n_{\max},\; l \leq l_{\max},\; |m| \leq m_{\max} \right\}$$

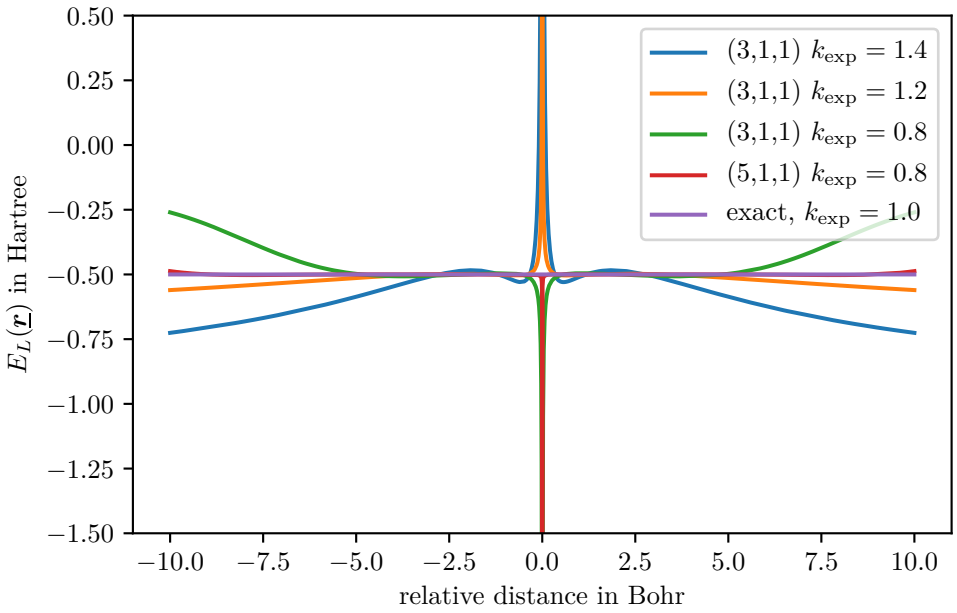


Figure 5.11: Local energy $E_L(r)$ of the hydrogen atom ground state of selected Coulomb-Sturmian basis sets. $E_L(r)$ is plotted against the relative distance of electron and nucleus. The optimal value for k_{exp} for hydrogen is 1.0, which is exact.

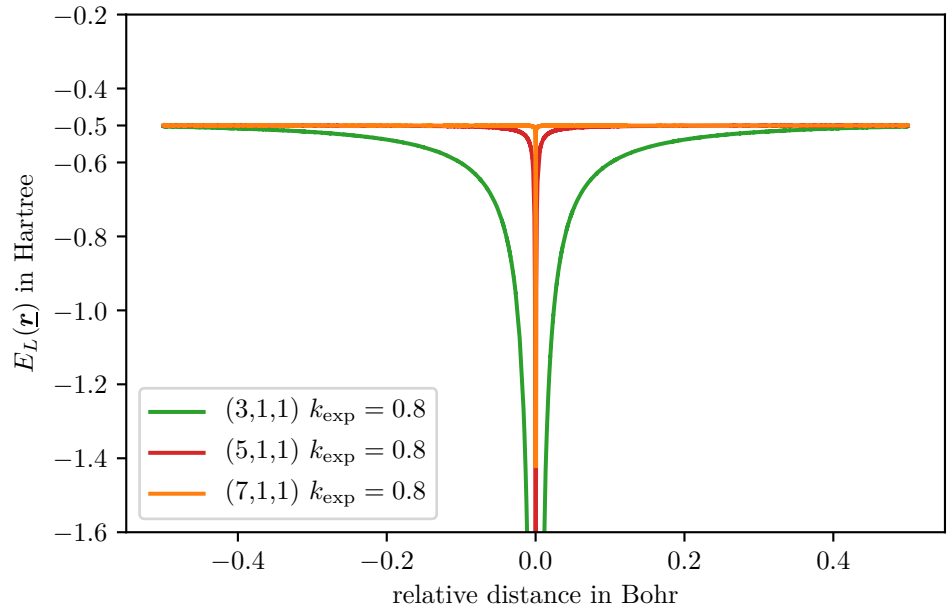


Figure 5.12: Magnified version of figure 5.11 around the origin. The orange curve theoretically goes to $-\infty$ as well, but the slope is so large that this is not visible at the resolution level of the plot.

of CS functions. For the special case of hydrogen, which is considered here, only a single electron is present in the system. One can therefore choose k_{exp} such, that the exact hydrogen ground state is obtained in the φ_{1s}^{CS} function. This is the optimal exponent for hydrogen, which is $k_{\text{exp}} = 1.0$. Figure 5.10 on page 118 shows, in agreement with our previous discussion, that both the size of the basis as well as the value for k_{exp} has an influence on the relative error. Since the slope at which the CS functions decay at infinity depends on k_{exp} — with larger values leading to faster decay — it is not surprising to find that a too large value for k_{exp} leads to a negative relative error at $r = \pm\infty$, whilst a too small value for k_{exp} leads to a positive error. Similarly, larger deviations of k_{exp} from 1.0 cause the relative error to become larger in magnitude throughout the curve: Compare the blue and the orange curve with $k_{\text{exp}} = 1.4$ and $k_{\text{exp}} = 1.2$, for example. The relative error does, however, not scale linearly with k_{exp}. Yet furthermore it is not even symmetric with respect to the direction into which k_{exp} deviates from the optimal value. In this case the orange curve is less steep as $r \to \infty$ and has a lower value at the cusp than the green one, even though both miss the best exponent by 0.2. In all systems I investigated so far, I made the similar observation that the error is more pronounced if the optimal value for k_{exp} is underestimated rather than overestimated. Compared to the effect which k_{exp} has on the error, the effect of increasing the basis is much more significant. Even though the green and the red curve both use a k_{exp} which is off by 0.2, the red curve following a $(5, 1, 1)$-basis stays below a relative error of 0.05 over the full depicted range of distances. On the other hand, the green one, a $(3, 1, 1)$-basis, starts to become rather inaccurate from distances of 7.5 Bohr and larger.

Very similar conclusions can be drawn from figure 5.11 on the preceding page, which shows the local energy versus relative distance. Comparing this plot to the local energy obtained for the cGTO discretisations in figure 5.2 on page 99, one notices how the cGTO local energy has much more wiggles and overall deviations from the exact value of 0.5. Even though the CS discretisations depicted in figure 5.11 are not perfect eigenfunctions of the hydrogen atom, the local energy is still mostly close to 0.5, thus they encode most of the physics. Even with a too small value $k_{\text{exp}} = 0.8$, the $(5, 1, 1)$ basis produces an acceptable eigenfunction over the full depicted range — except the nucleus. This is illustrated in more detail in figure 5.12, which is a close-up of the local energies of a $(3, 1, 1)$, a $(5, 1, 1)$ and a $(7, 1, 1)$ discretisation for $k_{\text{exp}} = 0.8$ around the nucleus. Whilst the $(3, 1, 1)$ and the $(5, 1, 1)$ both decay visibly to $-\infty$ at the origin, the $(7, 1, 1)$ discretisation already mostly corrects for this. Even though it still goes to $-\infty$ in theory, the resolution of the plot is no longer good enough to show this properly. From the illustrated trends it is clear that CS discretisations are able to represent both the exponential decay as well as the electron-nuclear cusp up to any desired accuracy if the basis is chosen large enough. More examples discussing the convergence behaviour of CS discretisations can be found in chapter 8 on page 171.

Apart from the ability of a basis function type to properly represent the physics of a chemical system, we also need to be able to solve the arising numerical problems in order to make it useful for practical quantum-chemical calculations. Similar to the other basis function types discussed so far, we will therefore now turn our attention to the Fock matrix $\mathbf{F}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^{\dagger}\right]$, both its structure as well as its diagonalisation. For this we first consider the computation of the integrals (4.60) to (4.64), starting with the overlap matrix. Its elements $S_{\mu\mu'}$ can be computed for any two Coulomb-Sturmians[19] φ_μ and

[19] The "CS" superscripts for basis functions and radial parts are dropped in the remainder of this

$\varphi_{\mu'}$ by treating radial and angular part separately [169]

$$\begin{aligned} S_{\mu\mu'} &= \int_{\mathbb{R}^3} \varphi_\mu^*(\underline{\boldsymbol{r}})\, \varphi_{\mu'}(\underline{\boldsymbol{r}})\, \mathrm{d}\underline{\boldsymbol{r}} \\ &= \int_0^\infty R_{nl}(r)\, R_{n'l'}(r)\, r^2\, \mathrm{d}r \cdot \int_{\mathbb{S}^2} (Y_l^m)^*(\hat{\underline{\boldsymbol{r}}})\; Y_{l'}^{m'}(\hat{\underline{\boldsymbol{r}}})\, \mathrm{d}\hat{\underline{\boldsymbol{r}}} \\ &= \delta_{mm'}\delta_{ll'} \underbrace{\int_0^\infty R_{nl}(r)\, R_{n'l}(r) r^2\, \mathrm{d}r}_{=s_{nn'}^{(l)}}. \end{aligned} \tag{5.39}$$

Normalisation implies that $s_{nn}^{(l)} = 1$ and the potential-weighted orthonormality (5.38) implies that $s_{nn'}^{(l)} = 0$ iff $|n - n'| > 1$. By following the algebra one can further show [169] that

$$s_{n,n+1}^{(l)} = s_{n+1,n}^{(l)} = -\frac{1}{2}\sqrt{\frac{(n-l)(n+l+1)}{n(n+1)}}.$$

This implies that $\mathbf{S}$ is tridiagonal in each block of identical angular momentum quantum number l, thus it has only three three non-zeros per row.

Similarly one can directly employ the potential-weighted orthonormality (5.38) to show that the nuclear attraction matrix is diagonal, namely

$$\begin{aligned} (V_0)_{\mu\mu'} &= -\int_{\mathbb{R}^3} \varphi_\mu^*(\underline{\boldsymbol{r}})\, \frac{Z}{r}\, \varphi_{\mu'}(\underline{\boldsymbol{r}})\, \mathrm{d}\underline{\boldsymbol{r}} \\ &= -\frac{Zk_{\mathrm{exp}}}{n'} \int_{\mathbb{R}^3} \varphi_\mu^*(\underline{\boldsymbol{r}})\, \frac{n'}{rk_{\mathrm{exp}}}\, \varphi_{\mu'}(\underline{\boldsymbol{r}})\, \mathrm{d}\underline{\boldsymbol{r}} \\ &= -\delta_{\mu\mu'}\frac{Zk_{\mathrm{exp}}}{n}. \end{aligned} \tag{5.40}$$

From (5.29) to (5.31) we get

$$\left(-\frac{1}{2}\Delta - \frac{nk_{\mathrm{exp}}}{r} + \frac{k_{\mathrm{exp}}^2}{2}\right)\varphi_\mu(\underline{\boldsymbol{r}}) = 0,$$

which implies for the kinetic energy matrix elements

$$\begin{aligned} T_{\mu\mu'} &= \int_{\mathbb{R}^3} \varphi_\mu^*(\underline{\boldsymbol{r}}) \left(-\frac{1}{2}\Delta\right) \varphi_{\mu'}(\underline{\boldsymbol{r}})\, \mathrm{d}\underline{\boldsymbol{r}} \\ &= \int_{\mathbb{R}^3} \varphi_\mu^*(\underline{\boldsymbol{r}}) \left(\frac{n'k_{\mathrm{exp}}}{r} - \frac{k_{\mathrm{exp}}^2}{2}\right) \varphi_{\mu'}(\underline{\boldsymbol{r}})\, \mathrm{d}\underline{\boldsymbol{r}} \\ &= k^2\left(\delta_{\mu\mu'} - \frac{1}{2}S_{\mu\mu'}\right) \\ &= k^2\delta_{ll'}\delta_{mm'}\left(\delta_{nn'} - \frac{1}{2}s_{nn'}^{(l)}\right), \end{aligned} \tag{5.41}$$

such that they follow the same advantageous sparsity pattern as the overlap matrix. The one-electron integrals thus all contain at most 3 non-zeros per row and are tridiagonal

section for simplicity.

in each block of identical angular momentum quantum number l. Due to the simplicity of the expressions of the matrix elements, storing these matrix terms in memory — even in a compressed tridiagonal form — is not needed, since recomputing the values takes a negligible number of flops.

Unsurprisingly, treating the two-electron integrals is more involved. We follow [169], which describes the treatment in a more general context and the specialised arguments presented in the documentation of `sturmint` [170]. Due to the structure of the radial part $R_{nl}(r)$ one may write the product of two Coulomb-Sturmians as a sum over Coulomb-Sturmians with twice the exponent, i.e.

$$\varphi^*_{\mu_1}(\underline{\boldsymbol{r}})\,\varphi_{\mu_2}(\underline{\boldsymbol{r}}) = \sum_{\mu} \mathcal{C}^{\mu}_{\mu_1,\mu_2}\,\varphi_{\mu}(2k_{\mathrm{exp}},\underline{\boldsymbol{r}}), \tag{5.42}$$

where $\varphi_\mu(2k_{\mathrm{exp}},\underline{\boldsymbol{r}})$ denotes a CS function with twice the exponent. This expansion looks familiar to the density-fitting approximation in the context of cGTO basis sets, but is in fact *exact* in the case of Coulomb-Sturmians. Since

$$\left(\varphi^*_{\mu_1}(\underline{\boldsymbol{r}})\varphi_{\mu_2}(\underline{\boldsymbol{r}})\right)^* = \varphi^*_{\mu_2}(\underline{\boldsymbol{r}})\varphi_{\mu_1}(\underline{\boldsymbol{r}})$$

it follows that the conjugated product requires the related expansion coefficients $\mathcal{C}^{\mu}_{\mu_2,\mu_1}$. With this the electron-repulsion integral tensor in Mulliken index (4.31) ordering may be written as the contraction

$$(\mu_1\mu_2|\mu_3\mu_4) = \sum_{\mu\mu'} \left(\mathcal{C}^{\mu}_{\mu_1,\mu_2}\right)^*\, I_{\mu\mu'}\, \mathcal{C}^{\mu'}_{\mu_3,\mu_4} = \sum_{\mu\mu'} \mathcal{C}^{\mu}_{\mu_2,\mu_1}\, I_{\mu\mu'}\, \mathcal{C}^{\mu'}_{\mu_3,\mu_4} \tag{5.43}$$

where $I_{\mu\mu'}$ is the electron-repulsion kernel in terms of the $2k_{\mathrm{exp}}$-functions

$$I_{\mu\mu'} \equiv \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} \frac{\varphi^*_{\mu'}(2k_{\mathrm{exp}},\underline{\boldsymbol{r}}_1)\,\varphi_{\mu}(2k_{\mathrm{exp}},\underline{\boldsymbol{r}}_2)}{r_{12}}\,\mathrm{d}\underline{\boldsymbol{r}}_1\,\mathrm{d}\underline{\boldsymbol{r}}_2. \tag{5.44}$$

Using the expansion of the Coulomb operator in terms of spherical harmonics [138]

$$\frac{1}{r_{12}} = \sum_{l''=0}^{\infty} \frac{r_<^{l''}}{r_>^{l''+1}} \frac{4\pi}{2l''+1} \sum_{m''=-l''}^{l''} Y^{m''}_{l''}(\hat{\underline{\boldsymbol{r}}}_1)\left(Y^{m''}_{l''}(\hat{\underline{\boldsymbol{r}}}_2)\right)^*,$$

where

$$r_< \equiv \min(r_1,r_2) \qquad\qquad r_> \equiv \max(r_1,r_2),$$

equation (5.44) may be rewritten as

$$\begin{aligned} I_{\mu\mu'} = \sum_{l''=0}^{\infty} \frac{4\pi}{2l''+1} \sum_{m''=-l''}^{l''} \int_0^\infty\int_0^\infty r_1^2 R_{nl}(2k_{\mathrm{exp}},r_1)\, r_2^2 R_{n'l'}(2k_{\mathrm{exp}},r_2) \frac{r_<^{l''}}{r_>^{l''+1}}\,\mathrm{d}r_1\,\mathrm{d}r_2 \\ \cdot \int_{\mathbb{S}^2} \underbrace{\left(Y^m_l(\hat{\underline{\boldsymbol{r}}}_1)\right)^*\, Y^{m''}_{l''}(\hat{\underline{\boldsymbol{r}}}_1)}_{=\delta_{l,l''}\delta_{m,m''}}\mathrm{d}\hat{\underline{\boldsymbol{r}}}_1 \cdot \int_{\mathbb{S}^2} \underbrace{Y^{m'}_{l'}(\hat{\underline{\boldsymbol{r}}}_2)\left(Y^{m''}_{l''}(\hat{\underline{\boldsymbol{r}}}_2)\right)^*}_{=\delta_{l',l''}\delta_{m',m''}}\mathrm{d}\hat{\underline{\boldsymbol{r}}}_2 \\ = \delta_{ll'}\delta_{mm'} I^{(l)}_{nn'}, \end{aligned} \tag{5.45}$$

where

$$I^{(l)}_{nn'} = \frac{4\pi}{2l+1} \int_0^\infty \int_0^\infty r_1^2 R_{nl}(2k_{\text{exp}}, r_1)\, r_2^2 R_{n'l}(2k_{\text{exp}}, r_2) \frac{r_<^l}{r_>^{l+1}}\, \mathrm{d}r_1\, \mathrm{d}r_2. \tag{5.46}$$

It is not immediately obvious from the form of equation (5.46), but the dependency on k_{exp} can be factored out of this expression, such that it only depends on n, n' and l. Assuming for the principle quantum number $n \leq 20$, which is rather typical, the tensor $I^{(l)}_{nn'}$ has only about $20^3 = 8000$ elements, which can be pre-evaluated and stored inside the program. In fact even more simplifications are possible if one inserts the definition of the radial parts and splits the integration kernel by powers of r_1 and r_2. For the required polynomial powers α, β the integrals

$$\int_0^\infty \int_0^\infty r_1^\alpha r_2^\beta \exp(-r_1) \exp(-r_2) \frac{r_<^l}{r_>^{l+1}}\, \mathrm{d}r_1\, \mathrm{d}r_2$$

can then be precomputed and stored as a vector, but this equation is less numerically stable than (5.46). At runtime one only needs to form the dot product of the precomputed vector with the appropriate vector of polynomial coefficients to yield the value for $I^{(l)}_{nn'}$.

Let us now return to equation (5.42), i.e.

$$\varphi^*_{\mu_1}(\underline{\boldsymbol{r}})\, \varphi_{\mu_2}(\underline{\boldsymbol{r}}) = \sum_\mu \mathcal{C}^\mu_{\mu_1,\mu_2}\, \varphi_\mu(2k_{\text{exp}}, \underline{\boldsymbol{r}}).$$

To obtain an expression for the coefficients $\mathcal{C}^\mu_{\mu_1,\mu_2}$ we multiply this equation with $\varphi^*_{\mu'}(2k_{\text{exp}}, \underline{\boldsymbol{r}})$ from the right and integrate over $\mathbb{R}^3$. Using the potential-weighted orthonormality (5.38) for the $2k_{\text{exp}}$ Coulomb-Sturmians this yields

$$\begin{aligned} \mathcal{C}^\mu_{\mu_1,\mu_2} &= \frac{n}{2k} \int_{\mathbb{R}^3} \varphi^*_{\mu_1}(\underline{\boldsymbol{r}}) \varphi_{\mu_2}(\underline{\boldsymbol{r}})\, \frac{1}{r}\, \varphi_\mu(2k_{\text{exp}}, \underline{\boldsymbol{r}})\, \mathrm{d}\underline{\boldsymbol{r}}. \\ &= \frac{n}{2k} \int_0^\infty R_{n_1,l_1}(r) R_{n_2,l_2}(r)\, R_{n,l}(2k, r) r\, \mathrm{d}r \\ &\qquad \cdot \int_{\mathbb{S}^2} \left(Y^m_l(\hat{\underline{\boldsymbol{r}}})\right)^* \left(Y^{m_1}_{l_1}(\hat{\underline{\boldsymbol{r}}})\right)^*\, Y^{m_2}_{l_2}(\hat{\underline{\boldsymbol{r}}})\, \mathrm{d}\hat{\underline{\boldsymbol{r}}}. \end{aligned} \tag{5.47}$$

The angular part of the latter expression can be written in terms of Clebsch-Gordan coefficients, which are precomputed and stored[20]. The properties of the Clebsch-Gordan coefficients imply that $\mathcal{C}^\mu_{\mu_1,\mu_2}$ can only be non-zero if

$$m = m_2 - m_1 \qquad \text{and} \qquad l \in \Big[\, |l_1 - l_2|\,, l_1 + l_2 \Big],$$

such that $\mathcal{C}^\mu_{\mu_1,\mu_2}$ is again a sparse tensor. The radial part is computed similar to (5.46), i.e. as a dot product between polynomial coefficients and precomputed kernels over polynomial powers.

[20] Due to the sparsity and symmetry properties of the Clebsch-Gordan coefficients even for a large value maximal principle quantum number like $n = 20$, no more than a few hundred thousand such coefficients need to be stored. If some recursion relations are taken into account as well, it is far less.

Due to the outlined sparsity of the $2k_{\text{exp}}$-kernel $I_{\mu\mu'}$ and the expansion coefficients $\mathcal{C}^{\mu}_{\mu_1,\mu_2}$ the contraction in equation (5.43) can be written more effectively as

$$\begin{aligned}
(\mu_1\mu_2|\mu_3\mu_4) &= \sum_{\mu\mu'} \mathcal{C}^{\mu}_{\mu_2,\mu_1}\, I_{\mu\mu'}\, \mathcal{C}^{\mu'}_{\mu_3,\mu_4} \\
&= \sum_{n,l,m} \sum_{n',l',m'} \mathcal{C}^{(n,l,m)}_{\mu_2,\mu_1}\, \delta_{ll'}\delta_{mm'} I^{(l)}_{nn'}\, \mathcal{C}^{(n',l',m')}_{\mu_3,\mu_4} \\
&= \sum_{n'} \sum_{n,l,m} \mathcal{C}^{(n,l,m)}_{\mu_2,\mu_1}\, I^{(l)}_{nn'}\, \mathcal{C}^{(n',l,m)}_{\mu_3,\mu_4} \\
&= \delta_{m_1-m_2,m_4-m_3} \sum_{l=l_{\min}}^{l_{\max}} \sum_{n=l+1}^{n_1+n_2-1} \sum_{n'=l+1}^{n_3+n_4-1} \mathcal{C}^{(n,l,m_2-m_1)}_{\mu_2,\mu_1}\, I^{(l)}_{nn'}\, \mathcal{C}^{(n',l,m_2-m_1)}_{\mu_3,\mu_4}
\end{aligned} \tag{5.48}$$

where

$$l_{\min} = \max(|l_1 - l_2|, |l_3 - l_4|) \qquad l_{\max} = \min(l_1 + l_2, l_3 + l_4). \tag{5.49}$$

Because of the selection rules in the quantum numbers l and m the ERI tensor is thus a sparse quantity with far less than N^4_{bas} non-zeros. When contracting it with the occupied coefficients $\mathbf{C}$ to form the Coulomb and exchange matrices, i.e. computing the elements

$$J_{\mu_3\mu_4}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^{\dagger}\right] = \sum_{i\in\mathcal{I}_{\text{occ}}} \sum_{\mu_1,\mu_2\in\mathcal{I}_{\text{bas}}} \sum_{\mu,\mu'\in\mathcal{I}_{\text{bas}}} C^{(n)}_{\mu_1 i} C^{(n)*}_{\mu_2 i}\, \mathcal{C}^{\mu}_{\mu_2,\mu_1} I_{\mu\mu'} \mathcal{C}^{\mu'}_{\mu_3,\mu_4} \tag{5.50}$$

and

$$K_{\mu_3\mu_4}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^{\dagger}\right] = \sum_{i\in\mathcal{I}_{\text{occ}}} \sum_{\mu_1,\mu_2\in\mathcal{I}_{\text{bas}}} \sum_{\mu,\mu'\in\mathcal{I}_{\text{bas}}} C^{(n)}_{\mu_1 i} C^{(n)*}_{\mu_2 i}\, \mathcal{C}^{\mu}_{\mu_2,\mu_3} I_{\mu\mu'} \mathcal{C}^{\mu'}_{\mu_1,\mu_4}, \tag{5.51}$$

the sparsity is partially lost. The reason is that the sum over the occupied orbital index i implies that each element $J_{\mu_3\mu_4}$ or $K_{\mu_3\mu_4}$ becomes a linear combination of contributions from different angular quantum number pairs (l_1, m_1) and (l_2, m_2). Thus, a Coulomb or exchange matrix element is only a known zero if *all* of the possible combinations of the indices μ_3, μ_4 with the pairs (l_1, m_1) and (l_2, m_2) are guaranteed to be zero — a much weaker selection rule. For forming the matrix-vector products of $\mathbf{J}$ and $\mathbf{K}$ with other vectors therefore most elements of $J_{\mu_3\mu_4}$ and $K_{\mu_3\mu_4}$ need to be touched. For the exchange matrix $\mathbf{K}$ in fact all elements may be non-zero, giving rise to a full quadratic scaling of a matrix-vector product in the number of basis functions. On the other hand, avoiding the storage of $(\mu_1\mu_2|\mu_3\mu_4)$ and $\mathbf{K}$ in favour of directly computing the matrix-vector product expression

$$(\mathbf{K}\underline{\boldsymbol{x}})_{\mu_3} = \sum_{i\in\mathcal{I}_{\text{occ}}} \sum_{\mu_1,\mu_2,\mu_4\in\mathcal{I}_{\text{bas}}} \sum_{\mu,\mu'\in\mathcal{I}_{\text{bas}}} C^{(n)}_{\mu_1 i} C^{(n)*}_{\mu_2 i}\, \mathcal{C}^{\mu}_{\mu_2,\mu_3} I_{\mu\mu'} \mathcal{C}^{\mu'}_{\mu_1,\mu_4}\, x_{\mu_4}, \tag{5.52}$$

whenever the contraction of $\mathbf{K}$ with a vector $\underline{\boldsymbol{x}}$ is needed, one may fully exploit all angular momentum selection rules during the evaluation. With this one may achieve the best possible scaling, certainly below quadratic. Notice that an efficient contraction scheme for computing (5.52) will carry out the contraction over occupied orbitals (index i) at the very end. In other words the improved scaling originating from (5.52) can only

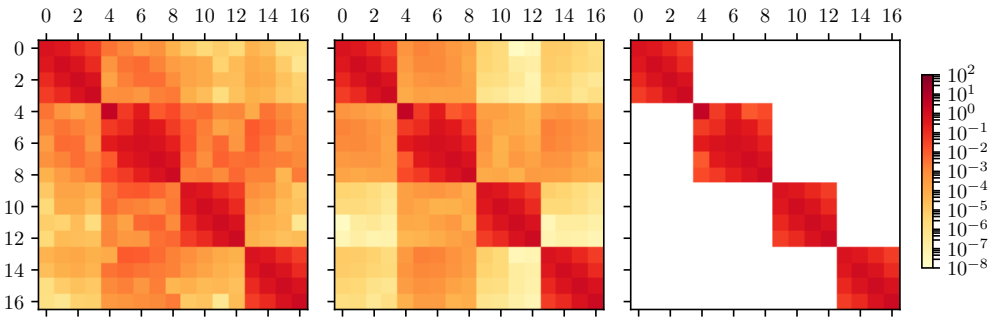


Figure 5.13: Structure of the Fock matrix for a Coulomb-Sturmian-based Hartree-Fock calculation of the beryllium atom starting from using a $(5, 1, 1)$ Coulomb-Sturmian basis in mln order and a Sturmian exponent of $k_{\text{exp}} = 1.99$. The three figures show left to right the Fock matrix at an SCF step with a Pulay error Frobenius norm of 0.13, 0.0079, $6.7 \cdot 10^{-8}$. The colouring depends on the absolute value of the respective Fock matrix entry with white indicating entries below 10^{-8}.

be achieved if $\mathbf{K}$ is not in memory and if the occupied coefficients $\mathbf{C}$ are available as separate quantities and not already contracted into a density matrix.

Both the very simple form of the one-electron matrices, given by the expressions (5.39), (5.40) and (5.41), as well as the previous discussion about the angular momentum selection rules in the case of the Coulomb and exchange matrices suggests to employ a contraction-based scheme for a Coulomb-Sturmian-based SCF. Looking at the structure of the Fock matrix $\mathbf{F}$ in figure 5.13, we notice that it is very similar to the cGTO case (figure 5.4 on page 100). Most notably it is almost diagonal dominant and of a similar size than the cGTO Fock matrix. In other words a dense diagonalisation method could in theory be employed for the Fock matrix $\mathbf{F}$ as well. The downside of a dense scheme would be the higher storage requirement as well as the larger computational scaling of the matrix-vector product. Whilst CS discretisations on the one hand do not require contraction-based methods to be feasible, they still allow for improved contraction if such methods are employed.

While Coulomb-Sturmians are not yet used for molecules due to their difficulties with respect to computing the ERI tensor in this context, a range of more generalised Sturmian-type basis functions exist [9, 30], which can be applied, for example, to molecular systems as well. Especially when it comes to evaluating the two-electron integrals, these share some of the properties of the Coulomb-Sturmians, but both the mathematical machinery as well as the numerics are more involved. CS functions can thus be seen as a first step towards these more general Sturmian-type basis functions. Generalised Sturmian-type orbitals are an active field of research [9, 21, 27–34, 138, 142, 166–169, 171–176]. Some recent works include efforts to develop schemes for the fast evaluation of the resulting ERI tensor [32, 138, 142] as well as the application of Sturmian-type functions for evaluating STO integrals more efficiently [31, 32]. Other methods include the combination of Sturmians and some numerical methods to yield ionising Sturmians to simultaneously model bound states as well as the continuum-like states in a single basis [28, 33, 34, 175, 176].

5.3.7 Other types of basis functions

The selection of basis function types discussed so far already gives a decent overview of the functions, which could be used for electronic structure theory calculations. Nevertheless there are few more basis function types, which should not go unmentioned.

For example, in the context of electronic structure calculations on extended periodic systems or systems in the solid state plane-wave and projector-augmented wave approaches[36–39] are both extremely popular as well as very well-suited. Over the years there has also been an enormous amount of development into the direction of numerical basis functions. Frediani and Sundholm [10] provide an excellent review. Such approaches include a fully numerical treatment employing clever numerical integration grids [39, 177, 178] or discretisation schemes based on finite-differences [179] or finite-elements [17, 19–23, 180, 181]. A common pattern is to only treat part of the electronic wave function numerically [18, 182, 183] and, for example, employ a factorisation of the one-particle functions into a numerical radial part and a spherical harmonic function. Last but not least one should also mention wavelet-based methods [11–16], where quite some progress has been made in recent years. To the best of my knowledge, wavelet-based electronic structure theory is the only methodology where guaranteed precision in the solution to the respective problems can be achieved.

5.3.8 Mixed bases

In theory there is no reason to stick to a single type of basis function in a discretisation. For example the projector-augmented plane-wave approaches [36–39] combine a plane-wave basis with other types of basis functions close to the atom cores. In a similar way the combination of finite elements and cGTO basis functions in one basis set has been employed for electronic structure theory calculations [184–186].

In practice, not all combinations of basis functions are feasible or sensible. This can be rationalised by looking at the cGTO or the CS discretisations, which both heavily rely on basis-specific properties for efficiently computing the electron-repulsion integrals in order to make the computation of the Fock matrix $\mathbf{F}$ or its matrix-vector product fast. In a fully mixed basis one not only needs to compute ERI integrals between the same type of basis functions, but also between all combinations of four basis functions involving different types. In a combination of the two aforementioned basis function types, this would destroy their advantageous properties. This is not meant to say that mixture basis sets involving cGTOs or CS basis functions are not possible or helpful, but that computing the electron-repulsion integrals efficiently would be a rather involved tasks.

5.3.9 Takeaway

In the previous sections we saw that different basis function types can lead to rather different numerical properties in the discretised HF problem. Just considering the three figures 5.4 on page 100, 5.9 on page 112 and 5.13 on the previous page illustrating the structures of the Fock matrices the overall differences are apparent. Whilst the number of basis functions of the atom-centred cGTO and CS discretisations depends on the number of atoms in the chemical system, FE discretisations need very similar numbers of basis functions for atoms and molecules. In contrast to AO approaches, FE-based discretisations need many more basis functions, in the order of millions compared to

hundreds for AO discretisations. For this reason only iterative methods are feasible for a FE discretisation, where a contraction-based scheme can theoretically lead to linear scaling in the number of basis functions. On the other hand the finite-element method does not rely on the intuition of the user very much. Initial grids can be easily auto-generated and while the calculation is running adaptively refined. Nevertheless, prior knowledge of the physics can be incorporated into the initial grid generation. Leaving the numerical issues aside, the FE approach in theory comes very close to the ideal basis function type we sketched in 5.3.1 on page 91.

From a practical point of view the AO approaches are less black-box, since more choices about the particular basis set need to be made before the calculation, but they are numerically much more feasible. Especially for cGTO-based methods the evaluation of the integrals is considerably less challenging compared to STOs, Coulomb-Sturmians or FEs and is well-understood by now. Coulomb-Sturmians on the other hand are physically much more sound than cGTOs such that they represent the wave function better. Contrast figures 5.2 on page 99 and 5.11 on page 119, for example. Unlike the STO based approaches, the integrals in CS discretisations can be evaluated rather efficiently due to the restriction to a single exponent k_{exp}. As discussed in section 5.3.6 on page 115 efficiency improvements are possible in a contraction-based ansatz. As figure 5.11 suggests, the convergence properties can be expected to be rather decent and predictable going to larger and larger basis sets. Originating from the completeness of the CS functions with respect to the form domain $Q(\hat{\mathcal{F}}) = H^1(\mathbb{R}^3, \mathbb{R})$, eventually both the long-range part as well as the cusp can be represented perfectly with larger and larger basis sets. The convergence properties of CS basis sets in the context of quantum-chemical calculations will be investigated in chapter 8 on page 171.

5.4 Self-consistent field algorithms

In this section we want to discuss a few standard self-consistent field algorithms in the light of the various types of basis functions we discussed in the previous section. Even though it is my hope that the selection of algorithms discussed here is representative, the vast number of methods, which has been developed over the years, makes it impossible to be exhaustive.

Most SCF algorithms are designed only with a cGTO-based discretisation of the HF and Kohn-Sham DFT problem in mind. The deviating numerical properties of the finite-element method or a CS-based discretisation therefore often call for minor modifications of the schemes. For example both finite elements as well as Coulomb-Sturmians favour contraction-based methods due to the better scaling of equations like (5.25) and (5.52) compared to building the full matrix. Therefore the Fock matrix might not be built in memory any more, which implies that a linear combination of Fock matrices cannot be computed in memory either. This does not imply that SCF schemes which form linear combinations of Fock matrices are completely ruled out, but they might become less favourable compared to other schemes.

On the other hand, in FE-based approaches all quantities which scale quadratically in N_{bas} cannot be stored in memory. This applies not only to the iterated Fock matrix $\mathbf{F}^{(n)}$, but to the density matrix $\mathbf{D}^{(n)} \in \mathbb{R}^{N_{\mathrm{bas}} \times N_{\mathrm{bas}}}$ as well. Even though clever low-rank approximation methods like hierarchical matrices [187–190] or tensor decomposition methods [191–194] could reduce the memory footprint of the density matrix, this work

will try to indicate ways by which building the density matrix in an SCF can be avoided. Naturally this implies a focus on coefficient-based SCF schemes as well, where the number of iterated parameters — the coefficient matrix $\mathbf{C} \in \mathbb{R}^{N_{\text{bas}} \times N_{\text{orb}}}$ — scales only linearly in N_{bas}. Furthermore coefficient-based SCF schemes have the advantage that iterating the density matrix destroys the possibility to follow the optimal contraction scheme for the application of $\mathbf{K}$ in CS-based methods. See equation (5.52) for details.

It was already pointed out in section 5.1 on page 85 that focusing on coefficient-based schemes is hardly a restriction in terms of the number of possible approaches, since coefficient-based and density-matrix-based schemes can be interconverted, at least approximately. For the case of the optimal damping algorithm (ODA) [195] a modification will be suggested in section 5.4.4 to bring this method to the coefficient-based setting.

Most of the SCF algorithms we will consider here only converge the HF equations (4.79) until the Pulay error (4.80) vanishes following our general description in remark 5.1 on page 86. Regarding the HF optimisation problem (4.65) this is only the necessary condition for a stationary point on the Stiefel manifold $\mathcal{C}$. Only some SCF algorithms, termed **second-order self-consistent field methods**, take at least approximate measures to ensure that the stationary point they find is a minimum. They are briefly considered in section 5.4.6.

5.4.1 Roothaan repeated diagonalisation

Roothaan's repeated diagonalisation [100] approach to the HF problem (4.79) is by far the simplest. In the formalism of remark 5.1 on page 86 this algorithm can be described by building the next Fock matrix $\tilde{\mathbf{F}}^{(n)}$ only by considering the current occupied coefficients $\mathbf{C}^{(n)}$, i.e. $\tilde{\mathbf{F}}^{(n)} = \mathbf{F}\Big[\mathbf{C}^{(n)}\big(\mathbf{C}^{(n)}\big)^\dagger\Big]$. The two-step iteration procedure of figure 5.14a on page 130results.

Even though Roothaan's algorithm already works for a few simple cases, it is far from being reliable. For example one can show [97, 196] that it either converges to a stationary point of the discretised HF problem (4.65) or alternatively it oscillates between two states, where none of them is a stationary point of (4.65). In practice it depends both on the system as well as the basis set which of these cases occurs. Furthermore there is no guarantee that the resulting stationary point found by Roothaan's algorithm is the HF ground state. All these cases can already be observed for HF calculations on atoms of the first three periods of the periodic table [97].

5.4.2 Level-shifting modification

If one uses essentially the same SCF scheme as figure 5.14a but instead diagonalises the matrix

$$\tilde{\mathbf{F}}^{(n)} = \mathbf{F}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right] - b\mathbf{S}\,\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\,\mathbf{S}$$

where $b > 0$, already a much better convergence is achieved. This modification is called **level shifting** [197, 198], where b is the **level-shifting parameter**, typically chosen in the range between 0.1 and 0.5. Effectively this modification increases the energy gap between occupied and virtual orbital energies. To see this, let us consider the converged case, where

$$\mathbf{F}\mathbf{C}_F = \mathbf{S}\mathbf{C}_F\mathbf{E}$$

exactly and let us partition the full coefficient matrix[21]

$$\mathbf{C}_F = \begin{pmatrix} \mathbf{C} & \mathbf{C}_{\text{virt}} \end{pmatrix}$$

into occupied and virtual parts. Now let $\tilde{\mathbf{F}} = \mathbf{F} - b\mathbf{S}\,\mathbf{C}\mathbf{C}^\dagger\,\mathbf{S}$ such that

$$\begin{aligned}
\tilde{\mathbf{F}}\mathbf{C}_F &= \left(\mathbf{F} - b\mathbf{S}\,\mathbf{C}\mathbf{C}^\dagger\,\mathbf{S}\right)\mathbf{C}_F \\
&= \mathbf{S}\mathbf{C}_F\mathbf{E} - b\mathbf{S}\begin{pmatrix}\mathbf{C}\mathbf{C}^\dagger\mathbf{S}\mathbf{C} & \mathbf{C}\mathbf{C}^\dagger\mathbf{S}\mathbf{C}_{\text{virt}}\end{pmatrix} \\
&= \mathbf{S}\mathbf{C}_F\mathbf{E} - b\mathbf{S}\begin{pmatrix}\mathbf{C} & 0\end{pmatrix} \\
&= \mathbf{S}\mathbf{C}_F\mathbf{E} + \mathbf{S}\mathbf{C}_F\begin{pmatrix}-b & 0\end{pmatrix} \\
&= \mathbf{S}\mathbf{C}_F\tilde{\mathbf{E}}
\end{aligned}$$

where

$$\tilde{\mathbf{E}} = \operatorname{diag}\left(\varepsilon_1 - b, \varepsilon_2 - b, \ldots, \varepsilon_{N_{\text{elec}}} - b, \varepsilon_{N_{\text{elec}}+1}, \ldots, \varepsilon_{N_{\text{orb}}}\right).$$

In other words the virtual orbitals are unaffected whereas the occupied orbitals are shifted downwards in energy by an amount b.

The effect of this is that coupling between both orbital spaces is reduced, which tends to lead to faster convergence especially if the gap between $\varepsilon_{N_{\text{elec}}}$ and $\varepsilon_{N_{\text{elec}}+1}$ is small. This empirical observation is backed up by a more sophisticated mathematical analysis by Cancès and Le Bris [196]. Their result shows that for sufficiently large b, the level-shifted Roothaan procedure is guaranteed to converge to a stationary point of the HF problem (4.65). They also provide an expression for the lower bound of b. In this manner convergence to a stationary point can be forced even for cases where the original HF equations (4.40) have no solution (like the negative ions with $N > 2Z + M$). In such a case the result is no physical ground state, however.

One can show [197] that the level-shifting modification is mathematically equivalent to another modification of Roothaan's repeated diagonalisation, called **damping**. In this procedure one chooses a **damping factor** $0 < \alpha < 1$ and sets

$$\tilde{\mathbf{F}}^{(n)} = (1-\alpha)\tilde{\mathbf{F}}^{(n-1)} + \alpha\mathbf{F}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right], \tag{5.53}$$

such that the new Fock matrix to diagonalise contains still a share of the old Fock matrix.

5.4.3 Optimal damping algorithm

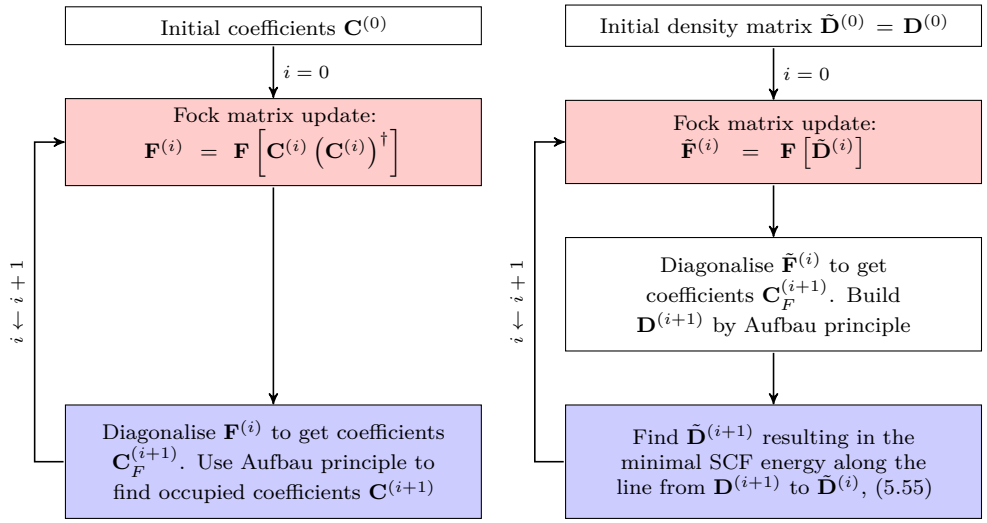
The optimal damping algorithm (ODA) was proposed by Cancès and Le Bris [195] based on their analysis of the Roothaan algorithm including the level-shifting modification.

In unmodified form [97, 195] it is a density-matrix-based SCF algorithm. Starting from an initial density $\tilde{\mathbf{D}}^{(0)} = \mathbf{D}^{(0)}$, the procedure is roughly (compare figure 5.14a) for $n = 1, 2, 3, \ldots$

- Build the Fock matrix

$$\tilde{\mathbf{F}}^{(n-1)} = \mathbf{F}\left[\tilde{\mathbf{D}}^{(n-1)}\right] \tag{5.54}$$

[21] We assume RHF here and furthermore only consider the α block. For UHF the analysis is exactly the same with the relevant equations just replicated in α and β block.



a) Roothaan repeated diagonalisation b) Optimal damping algorithm

Figure 5.14: Schematic of Roothaan repeated diagonalisation and optimal damping algorithm. The step which updates the Fock matrix is highlighted in red and the step which updates the coefficients is highlighted in blue.

and diagonalise it to obtain the new coefficient $\mathbf{C}_F^{(n)}$. Form the new density $\mathbf{D}^{(n)}$ according to the Aufbau principle from these as

$$\mathbf{D}^{(n)} = \mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^{\dagger}.$$

- Evaluate the Pulay error $\mathbf{e}^{(n)}$ (4.80) from $\mathbf{F}\left[\tilde{\mathbf{D}}^{(n)}\right]$ and $\mathbf{D}^{(n)}$. End the process if $\left\|\mathbf{e}^{(n)}\right\|_{\text{frob}} \leq \varepsilon_{\text{conv}}$.
- Solve the line search problem

$$\tilde{\mathbf{D}}^{(n+1)} = \underset{\tilde{\mathbf{D}} \in \text{Seg}\left[\tilde{\mathbf{D}}^{(n)}, \mathbf{D}^{(n+1)}\right]}{\arg\inf} \mathcal{E}_D^{\text{HF}}[\tilde{\mathbf{D}}] \tag{5.55}$$

where

$$\text{Seg}\left[\mathbf{D}_1, \mathbf{D}_2\right] = \left\{(1-\lambda)\mathbf{D}_1 + \lambda\mathbf{D}_2 \,\middle|\, \lambda \in [0,1]\right\}$$

is a line segment of density matrices and the energy functional $\mathcal{E}_D^{\text{HF}}$ is defined as in (4.70). Repeat the process thereafter.

One can show [97] that the ODA *always* converges to a local minimum of (4.69).

The remaining question to complete the picture of the ODA from a computational point of view is to find a way to obtain the minimal density $\tilde{\mathbf{D}}^{(n+1)}$. First notice that in general the density matrix segment

$$\text{Seg}\left[\mathbf{D}_1, \mathbf{D}_2\right] \not\subset \mathcal{P}$$

even if $\mathbf{D}_1, \mathbf{D}_2 \in \mathcal{P}$. Much rather this line segment is fully contained only in a superset $\tilde{\mathcal{P}} \supset \mathcal{P}$, where we relax the constraint $\mathbf{D}^2 = \mathbf{D}$ to[22] $\mathbf{D}^2 \leq \mathbf{D}$. See [97] for details. For ease of notation let us define

$$\begin{aligned} E_1\left[\mathbf{D}\right] &\equiv \mathrm{tr}\left(\mathbf{T}\mathbf{D} + \mathbf{V}_0\mathbf{D}\right), \\ \mathbf{G}\left[\mathbf{D}\right] &\equiv \mathbf{F}\left[\mathbf{D}\right] + \mathbf{K}\left[\mathbf{D}\right] \end{aligned}$$

and

$$E_2\left[\mathbf{D}\right] \equiv \frac{1}{2}\,\mathrm{tr}\left(\mathbf{D}\,\mathbf{G}\left[\mathbf{D}\right]\right).$$

For all matrices $\mathbf{D}_1, \mathbf{D}_2 \in \tilde{\mathcal{P}}$ we can show the properties [195]

$$\mathrm{tr}\left(\mathbf{D}_1\mathbf{G}\left[\mathbf{D}_2\right]\right) = \mathrm{tr}\left(\mathbf{D}_2\mathbf{G}\left[\mathbf{D}_1\right]\right) \tag{5.56}$$

$$\mathrm{tr}\left(\mathbf{F}\left[\mathbf{D}_1\right]\mathbf{D}_2\right) = E_1[\mathbf{D}_2] + \mathrm{tr}\left(\mathbf{D}_1\mathbf{G}\left[\mathbf{D}_2\right]\right) \tag{5.57}$$

These imply for E_2 and arbitrary $\alpha, \beta \in \mathbb{R}$

$$\begin{aligned} E_2[\alpha\mathbf{D}_1 + \beta\mathbf{D}_2] &= \frac{1}{2}\,\mathrm{tr}\left(\alpha^2\mathbf{D}_1\mathbf{G}\left[\mathbf{D_1}\right]\right) + \frac{1}{2}\,\mathrm{tr}\left(\alpha\beta\mathbf{D_1}\mathbf{G}\left[\mathbf{D}_2\right]\right) \\ &\quad + \frac{1}{2}\,\mathrm{tr}\left(\alpha\beta\mathbf{D}_2\mathbf{G}\left[\mathbf{D_1}\right]\right) + \frac{1}{2}\,\mathrm{tr}\left(\beta^2\mathbf{D}_2\mathbf{G}\left[\mathbf{D}_2\right]\right) \\ &\overset{(5.57)}{=} \alpha^2 E_2[\mathbf{D_1}] + \beta^2 E_2[\mathbf{D}_2] + \alpha\beta\,\mathrm{tr}\left(\mathbf{D_1}\mathbf{G}\left[\mathbf{D}_2\right]\right), \end{aligned}$$

whereas E_1 is linear

$$E_1[\alpha\mathbf{D}_1 + \beta\mathbf{D}_2] = \alpha E_1[\mathbf{D}_1] + \beta E_1[\mathbf{D}_2]. \tag{5.58}$$

These results allow to expand the HF energy for a member $\tilde{\mathbf{D}}^{(n+1)}$ of the density matrix segment $\mathrm{Seg}\left[\tilde{\mathbf{D}}^{(n)}, \mathbf{D}^{(n+1)}\right]$ as

$$\begin{aligned} \mathcal{E}_D^{\mathrm{HF}}\left[\tilde{\mathbf{D}}^{(n+1)}\right] &= \mathcal{E}_D^{\mathrm{HF}}\left[\left(1-\lambda^{(n+1)}\right)\tilde{\mathbf{D}}^{(n)} + \lambda^{(n+1)}\mathbf{D}^{(n+1)}\right] \\ &= \mathcal{E}_D^{\mathrm{HF}}\left[\tilde{\mathbf{D}}^{(n)} + \lambda^{(n+1)}\left(\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right)\right] \\ &= E_1\left[\tilde{\mathbf{D}}^{(n)} + \lambda^{(n+1)}\left(\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right)\right] \\ &\quad + E_2\left[\tilde{\mathbf{D}}^{(n)} + \lambda^{(n+1)}\left(\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right)\right] \\ &= E_1[\tilde{\mathbf{D}}^{(n)}] + \lambda^{(n+1)}E_1\left[\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right] + E_2[\tilde{\mathbf{D}}^{(n)}] \\ &\quad + \lambda^{(n+1)}\,\mathrm{tr}\left(\tilde{\mathbf{D}}^{(n)}\mathbf{G}\left[\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right]\right) \\ &\quad + \left(\lambda^{(n+1)}\right)^2 E_2[\mathbf{D}^{(n)} - \tilde{\mathbf{D}}^{(n)}] \\ &= \mathcal{E}_D^{\mathrm{HF}}\left[\tilde{\mathbf{D}}^{(n)}\right] + \lambda^{(n+1)}\underbrace{\mathrm{tr}\left(\tilde{\mathbf{D}}^{(n)}\mathbf{F}\left[\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right]\right)}_{=s} \\ &\quad + \left(\lambda^{(n+1)}\right)^2 \underbrace{E_2[\mathbf{D}^{(n)} - \tilde{\mathbf{D}}^{(n)}]}_{=c} \\ &= \mathcal{E}_D^{\mathrm{HF}}\left[\tilde{\mathbf{D}}^{(n)}\right] + \lambda^{(n+1)}s + \left(\lambda^{(n+1)}\right)^2 c \end{aligned} \tag{5.59}$$

[22] Let $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$, then $\mathbf{A} \leq \mathbf{B} \Leftrightarrow \forall \underline{\boldsymbol{x}} \in \mathbb{R}^n\ \underline{\boldsymbol{x}}^\dagger\mathbf{A}\underline{\boldsymbol{x}} \leq \underline{\boldsymbol{x}}^\dagger\mathbf{B}\underline{\boldsymbol{x}}$

The coefficients s and c can alternatively be written as

$$\begin{aligned} s &= \mathrm{tr}\left(\tilde{\mathbf{F}}^{(n)}\left(\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right)\right) \\ &= \mathrm{tr}\left(\tilde{\mathbf{F}}^{(n)}\mathbf{D}^{(n+1)}\right) - E_{\mathrm{HF}}[\tilde{\mathbf{D}}^{(n)}] - E_2[\tilde{\mathbf{D}}^{(n)}] \\ &= \mathrm{tr}\left(\tilde{\mathbf{F}}^{(n)}\mathbf{D}^{(n+1)}\right) - E_1[\tilde{\mathbf{D}}^{(n)}] - 2E_2[\tilde{\mathbf{D}}^{(n)}] \end{aligned} \tag{5.60}$$

and[23]

$$\begin{aligned} c &= E_2\left[\mathbf{D}^{(n+1)} - \tilde{\mathbf{D}}^{(n)}\right] \\ &\overset{(5.58)}{=} E_2[\mathbf{D}^{(n+1)}] - \mathrm{tr}\left(\mathbf{G}\left[\tilde{\mathbf{D}}^{(n)}\right]\mathbf{D}^{(n+1)}\right) + E_2[\tilde{\mathbf{D}}^{(n)}] \\ &= E_2[\mathbf{D}^{(n+1)}] - \mathrm{tr}\left(\tilde{\mathbf{F}}^{(n)}\mathbf{D}^{(n+1)}\right) + E_1[\mathbf{D}^{(n+1)}] + E_2[\tilde{\mathbf{D}}^{(n)}] \end{aligned} \tag{5.61}$$

Now the stationary point along the density matrix segment can be determined by differentiating (5.59) resulting in

$$\frac{\partial \mathcal{E}_D^{\mathrm{HF}}\left[\tilde{\mathbf{D}}^{(n+1)}\right]}{\partial \lambda^{(n+1)}} = s + 2\lambda^{(n+1)}c \qquad \text{and} \qquad \frac{\partial^2 \mathcal{E}_D^{\mathrm{HF}}\left[\tilde{\mathbf{D}}^{(n+1)}\right]}{\partial \left(\lambda^{(n+1)}\right)^2} = 2c$$

Due to $E_2[\mathbf{D}] \geq 0$ [195] for all $\mathbf{D} \in \tilde{\mathcal{P}}$ one easily deduces $c \geq 0$, such that the stationary point of the above expression is always a minimum. Since $\lambda^{(n+1)} \in [0,1]$ the minimiser is

$$\lambda_{\min}^{(n+1)} = \begin{cases} 1 & \text{if } 2c \leq -s \\ -\frac{s}{2c} & \text{else} \end{cases}, \tag{5.62}$$

where the cases $c = 0$ and $s = 0$ have been ignored, since they only occur at convergence. This closes the missing link and allows to implement a ODA in as a density-matrix-based SCF.

Let $\alpha, \beta \in \mathbb{R}$ and $\mathbf{D}_1, \mathbf{D}_2 \in \tilde{\mathcal{P}}$. Since $\mathbf{F}[\mathbf{D}] = \mathbf{T} + \mathbf{V}_0 + \mathbf{J}[\mathbf{D}] + \mathbf{K}[\mathbf{D}]$ and the two-electron terms are linear in the density matrix, we have

$$\mathbf{F}[\alpha\mathbf{D}_1 + \beta\mathbf{D}_2] = \alpha\mathbf{F}[\mathbf{D}_1] + \beta\mathbf{F}[\mathbf{D}_2] \tag{5.63}$$

iff $\alpha + \beta = 1$. Defining

$$\tilde{\mathbf{F}}^{(n)} \equiv \mathbf{F}\left[\tilde{\mathbf{D}}^{(n)}\right] \qquad\qquad \mathbf{F}^{(n)} \equiv \mathbf{F}\left[\mathbf{D}^{(n)}\right]$$

this allows to rewrite (5.54) as

$$\tilde{\mathbf{F}}^{(n)} = \mathbf{F}\left[\left(1-\lambda^{(n)}\right)\tilde{\mathbf{D}}^{(n-1)} + \lambda^{(n)}\mathbf{D}^{(n)}\right] = \left(1-\lambda^{(n)}\right)\tilde{\mathbf{F}}^{(n-1)} + \lambda^{(n)}\mathbf{F}^{(n)}, \tag{5.64}$$

where the "min" subscripts were dropped. Comparing with equation (5.53) one can identify with $\lambda^{(n)}$ the damping factor α. Since $\lambda^{(n)}$ is optimal in the sense of minimising the energy along the line segment spanned by $\mathbf{D}^{(n)}$ and $\tilde{\mathbf{D}}^{(n-1)}$, the ODA can be described by repetitively finding the optimal damping parameter from SCF step to SCF step. Notice that its construction guarantees that the SCF energy will always decrease.

[23] Note that the original paper [195] uses a deviating formalism which causes an extra factor of 2 to appear in their expression for c.

It is hence guaranteed to converge to a local minimum of the HF problem (4.69) [97, 195]. The ODA is only a particularly simple example from a whole family of density-matrix-based SCF algorithms called relaxed constraints algorithms, which are discussed in detail in [195].

Using (5.64) one can show by induction that

$$\tilde{\mathbf{F}}^{(n)} = \sum_{j=0}^{n} \mathbf{F}^{(j)} \lambda^{(j)} \prod_{i=j+1}^{n} \left(1 - \lambda^{(i)}\right), \tag{5.65}$$

$$\tilde{\mathbf{D}}^{(n)} = \sum_{j=0}^{n} \mathbf{D}^{(j)} \lambda^{(j)} \prod_{i=j+1}^{n} \left(1 - \lambda^{(i)}\right), \tag{5.66}$$

where we set $\lambda^{(0)} \equiv 1$. Since

$$\mathbf{F}^{(j)} = \mathbf{F}\left[\mathbf{C}^{(n)} \left(\mathbf{C}^{(n)}\right)^{\dagger}\right]$$

$$\mathbf{D}^{(j)} = \mathbf{C}^{(j)} \left(\mathbf{C}^{(j)}\right)^{\dagger}$$

these results in theory allow to express the complete ODA in terms of the coefficients such that expressions like (5.25) or (5.52) could be used for a FE-based or a CS-based discretisation respectively.

In practice this is usually not a fruitful approach for two reasons. Firstly it requires to store a growing list of coefficients, namely one for each SCF step. Especially for a FE approach this becomes increasingly costly in terms of memory. Secondly for a contraction-based ansatz we especially want to avoid storing the Fock matrices $\mathbf{F}^{(j)}$ in favour of contraction expressions like (5.25) and (5.52). In other words each application of $\tilde{\mathbf{F}}^{(n)}$ to a vector $\underline{\boldsymbol{x}}$ would need to be performed by first computing $\mathbf{F}^{(j)}\underline{\boldsymbol{x}}$ for each j and then adding the results. This procedure is roughly n times as expensive as a single apply. Even though the contraction expressions formally scale better, the increasing number of times they need to be invoked should make this ansatz rather expensive.

Overall the ODA is very suitable for cGTO and CS-based discretisations, since for these density-matrix-based SCF schemes are fine. However, this algorithm is not suitable for solving the HF problem with a FE-based discretisation without further modifications.

5.4.4 Truncated optimal damping algorithm

Let us again consider (5.65). Due to $\lambda^{(i)} \in [0, 1]$ the Fock matrix prefactor

$$\lambda^{(j)} \prod_{i=j+1}^{n} \left(1 - \lambda^{(i)}\right) \in [0, 1] \tag{5.67}$$

is a product of factors, which are all between 0 and 1. Therefore this prefactor may become rather small for small values of j as n increases. In other words in the later SCF steps the $\mathbf{F}^{(j)}$ terms which were produced at the beginning of the SCF procedure may be accompanied by a small prefactor and hence can at some point be neglected in (5.65). This is the justification for the truncated optimal damping algorithm (tODA), which

approximates the ODA by artificially restricting the number of terms in (5.65) to the m most recently obtained Fock matrices. If we define

$$j_0(n) \equiv n - m + 1$$

this allows to write the approximated sums as

$$\tilde{\mathbf{F}}^{(n)} = \frac{1}{\lambda^{(j_0(n))}} \sum_{j=j_0(n)}^{n} \mathbf{F}^{(j)} \lambda^{(j)} \prod_{i=j+1}^{n} \left(1 - \lambda^{(i)}\right), \tag{5.68}$$

and analogously for the density matrices

$$\tilde{\mathbf{D}}^{(n)} = \frac{1}{\lambda^{(j_0(n))}} \sum_{j=j_0(n)}^{n} \mathbf{D}^{(j)} \lambda^{(j)} \prod_{i=j+1}^{n} \left(1 - \lambda^{(i)}\right). \tag{5.69}$$

The factor $1/\lambda^{(j_0(n))}$ is required to make sure that the Fock matrix prefactors sum to 1, i.e. to make sure that the condition for the linear combination of Fock matrices (5.63) is fulfilled.

The simplest case of this class of approximations is $m = 1$. This implies $j_0(n) = n$ such that (5.68) and (5.69) simplify to read

$$\tilde{\mathbf{F}}^{(n)} = \mathbf{F}^{(n)} \qquad\qquad \tilde{\mathbf{D}}^{(n)} = \mathbf{D}^{(n)}$$

In other words this 2-step tODA is equivalent to an *adhoc* modification of the exact ODA where we replace $\tilde{\mathbf{D}}^{(n)}$ by $\mathbf{D}^{(n)}$, the density of the previous SCF step. Taking this into account the expressions (5.60) and (5.61) may be written as

$$\begin{aligned} s &= \mathrm{tr}\left(\tilde{\mathbf{F}}^{(n)}\mathbf{D}^{(n+1)}\right) - E_1[\tilde{\mathbf{D}}^{(n)}] - 2E_2[\tilde{\mathbf{D}}^{(n)}] \\ &= \mathrm{tr}\left(\left(\mathbf{C}^{(n+1)}\right)^\dagger \mathbf{F}^{(n)}\mathbf{C}^{(n+1)}\right) - E_1\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right] - 2E_2\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right] \end{aligned} \tag{5.70}$$

and

$$\begin{aligned} c &= E_2[\mathbf{D}^{(n+1)}] - \mathrm{tr}\left(\tilde{\mathbf{F}}^{(n)}\mathbf{D}^{(n+1)}\right) + E_1[\mathbf{D}^{(n+1)}] + E_2[\tilde{\mathbf{D}}^{(n)}] \\ &= E_2\left[\mathbf{C}^{(n+1)}\left(\mathbf{C}^{(n+1)}\right)^\dagger\right] - \mathrm{tr}\left(\left(\mathbf{C}^{(n+1)}\right)^\dagger \mathbf{F}^{(n)}\mathbf{C}^{(n+1)}\right) \\ &\qquad + E_1\left[\mathbf{C}^{(n+1)}\left(\mathbf{C}^{(n+1)}\right)^\dagger\right] + E_2\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right] \end{aligned} \tag{5.71}$$

In contrast to the exact ODA this yields a coefficient-based SCF algorithm. Starting from an initial set of coefficients $\mathbf{C}^{(0)}$ with corresponding initial Fock matrix $\tilde{\mathbf{F}}^{(0)} = \mathbf{F}\left[\mathbf{C}^{(0)}\left(\mathbf{C}^{(0)}\right)^\dagger\right]$ we proceed for $n = 1, 2, 3, \ldots$ as follows.

- Diagonalise $\tilde{\mathbf{F}}^{(n-1)}$ in order to obtain coefficients $\mathbf{C}_F^{(n)}$.
- According to the Aufbau principle select $\mathbf{C}^{(n)}$ and build $\mathbf{F}^{(n)} = \mathbf{F}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right]$.
- Evaluate the Pulay error $\mathbf{e}^{(n)}$ (4.80) and end the process if $\left\|\mathbf{e}^{(n)}\right\|_{\mathrm{frob}} \leq \varepsilon_{\mathrm{conv}}$.

- Compute s, c and $\lambda^{(n)}$ according to (5.70), (5.71) and (5.62).
- Set
$$\tilde{\mathbf{F}}^{(n)} = \left(1 - \lambda^{(n)}\right) \mathbf{F}^{(n-1)} + \lambda^{(n)} \mathbf{F}^{(n)}$$
and repeat.

In this process one only needs the history of two Fock matrices $\mathbf{F}^{(n-1)}$ and $\mathbf{F}^{(n)}$, such that $\tilde{\mathbf{F}}^{(n)}$ can be applied when needed. This in turn implies that only the coefficient matrices $\mathbf{C}^{(n-1)}$ and $\mathbf{C}^{(n)}$ are required, such that $\mathbf{F}^{(n)}$ and $\mathbf{F}^{(n-1)}$ can be applied whenever needed.

Compared to the Roothaan algorithm (see section 5.4.1) the tODA only roughly doubles the cost of each diagonalisation, since two Fock matrices need to be applied. Additionally one needs to evaluate the trace

$$\mathrm{tr}\left(\left(\mathbf{C}^{(n+1)}\right)^{\dagger} \mathbf{F}^{(n)} \mathbf{C}^{(n+1)}\right)$$

and compute the energies $E_1[\mathbf{D}^{(n)}]$ and $E_2[\mathbf{D}^{(n)}]$ in order to obtain c and s for each iteration. The former step costs about as much as a single matrix-vector product and the latter is usually done during the SCF anyways to display the progress to the user, thus representing no extra cost.

Even though about twice as expensive as the Roothaan algorithm if a contraction-based SCF is performed, the advantage of the tODA is that it automatically finds the damping coefficient $\lambda^{(n)}$, which reduces the energy at each iteration as much as possible. This amounts to break the oscillatory behaviour of the standard Roothaan repeated diagonalisation scheme in a slightly improved manner than the default damping or level-shifting modifications.

One should mention, however, that the tODA does not inherit all of the nice mathematical properties from the ODA. For example it is no longer guaranteed that the tODA converges to a minimum of the HF problem (4.65) [199]. Especially close to convergence it may for example happen, that $\lambda^{(n)} \notin [0, 1]$ since both c and s become rather small, thus $2c \leq s$ ill-defined. One can get around this by explicitly setting $\lambda^{(n)} = 1$ in the cases, where $|c|$ and $|s|$ become small. The tODA is thus best used in the initial SCF steps in order to effectively prevent the Roothaan oscillations from happening.

5.4.5 Direct inversion in the iterative subspace

In his celebrated 1982 paper Pulay not only introduced the aforementioned Pulay error (4.80), but also improved upon his previously introduced SCF convergence acceleration scheme [200]. This effort resulted in the procedure now widely known by the term **direct inversion in the iterative subspace** (DIIS). In his variant of the DIIS procedure the next Fock matrix was found as a linear combination

$$\tilde{\mathbf{F}}^{(n)} = \sum_{i=0}^{m-1} c_i \mathbf{F}^{(n-i)} \tag{5.72}$$

of Fock matrices from the m most recent SCF steps, i.e.

$$\mathbf{F}^{(j)} = \mathbf{F}\left[\mathbf{C}^{(j)}\left(\mathbf{C}^{(j)}\right)^{\dagger}\right].$$

The coefficients $\{c_i\}_{i=1,\ldots,m}$ are to be determined such that the norm of the corresponding linear combination of Pulay errors

$$f_{\mathrm{DIIS}}(c_0, c_1, \ldots, c_{m-1}) = \left\| \sum_{i=0}^{m-1} c_i \mathbf{e}^{(n-i)} \right\|_{\mathrm{frob}}^2 = \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} c_i c_j \,\mathrm{tr}\left(\mathbf{e}^{(n-i)} \mathbf{e}^{(n-j)}\right) \tag{5.73}$$

is minimal. Defining a real-symmetric matrix $\mathbf{B} \in \mathbb{R}^{m \times m}$ with elements

$$B_{ij} = \mathrm{tr}\left(\mathbf{e}^{(n-i)} \mathbf{e}^{(n-j)}\right) \tag{5.74}$$

we can alternatively write

$$f_{\mathrm{DIIS}}(\underline{\boldsymbol{c}}) = \underline{\boldsymbol{c}}^\dagger \mathbf{B} \underline{\boldsymbol{c}}. \tag{5.75}$$

In agreement with what was discussed in equation (5.63), we need to additionally impose the constraint

$$\sum_{i=0}^{m-1} c_i = 1,$$

such that the resulting Fock matrix $\tilde{\mathbf{F}}^{(n)}$ is physically sensible. In other words Pulay's DIIS scheme can be expressed as the quadratic programming problem

$$\underline{\boldsymbol{c}} = \arg\min \left\{ \underline{\boldsymbol{c}}^\dagger \mathbf{B} \underline{\boldsymbol{c}} \,\middle|\, \sum_{i=0}^{m-1} c_i = 1 \right\},$$

which has corresponding Euler-Lagrange equations

$$\begin{pmatrix} \mathbf{B} & \underline{\mathbf{1}} \\ \underline{\mathbf{1}}^\dagger & 0 \end{pmatrix} \begin{pmatrix} \underline{\boldsymbol{c}} \\ \lambda \end{pmatrix} = \begin{pmatrix} \underline{\mathbf{0}} \\ 1 \end{pmatrix}, \tag{5.76}$$

where $\underline{\mathbf{1}}, \underline{\mathbf{0}} \in \mathbb{R}^m$ are column vectors of m ones and m zeros and λ is the Lagrange multiplier corresponding to the constraint $\sum_{i=0}^{m-1} c_i = 1$. Typically one takes m between 2 and 10, such that the linear system (5.76) can be solved by an iterative method rather fast.

There are a few issues, which typically occur close to self-consistency, where the error matrices $\{\mathbf{e}^{(n-i)}\}_{i=0,\ldots,m-1}$ will be almost identical. This causes multiple rows in $\mathbf{B}$ to be extremely similar and thus gives rise to an ill-conditioned linear system (5.76). There are a couple of remedies typically used in practice [201]. For example one may drop the Fock matrices $\mathbf{F}^{(n-i)}$ where the coupling to the most recent Fock matrix $\mathbf{F}^{(n)}$, i.e. the matrix element B_{0i} is smaller than a certain threshold. Alternatively one may artificially bias the lowest-energy solution, by multiplying all other diagonal entries B_{ii} by a penalty factor slightly larger than 1. Last but not least one may always drop the oldest Fock matrix $\mathbf{F}^{(n-m+1)}$ if an ill-conditioned linear system is detected. Alternatively one can remove linear dependencies by a singular-value decomposition of $\mathbf{B}$. Another point worth noting is that the DIIS procedure is an *extrapolation* technique. In other words there is no guarantee, that the Fock matrix $\tilde{\mathbf{F}}^{(n)}$ is of any physical or mathematical significance. It could lead into totally the wrong direction causing the SCF procedure to eventually diverge.

Since there is no reason to build the density matrix in Pulay's DIIS procedure it is suitable for both density-matrix-based as well as coefficient-based SCF settings. Moreover

given that the m coefficients $\{\mathbf{C}^{(n-i)}\}_{i=0,\ldots,m-1}$ are stored, the Fock matrix terms of (5.72) can be applied using expressions like (5.25) or (5.52) without any problems, making the DIIS suitable for a contraction-based SCF as well.

Let us summarise the procedure in a contraction-based SCF. Start from an initial set of occupied coefficients $\mathbf{C}^{(0)} \in \mathcal{C}$. Set $B_{00} = 1$ and run for $n = 1, 2, 3, \ldots$

- Use the overlaps $\mathbf{B}$ to setup and solve (5.76) for the new set of DIIS coefficients $\underline{c}$.
- Build the Fock matrix $\tilde{\mathbf{F}}^{(n)}$ according to (5.72). In this process skip coefficients c_i below a certain threshold in order to save some matrix-vector products when $\tilde{\mathbf{F}}$ is contracted with trial vectors during the diagonalisation. If one entry c_i is very large, say > 10, then only keep this matrix in the expression. In all cases be sure to renormalise, such that all coefficients still sum to 1.
- Diagonalise $\tilde{\mathbf{F}}$ to obtain $\mathbf{C}_F^{(n)}$. Select $\mathbf{C}^{(n)}$ by the Aufbau principle.
- Evaluate the Pulay error $\mathbf{e}^{(n)}$ (4.80) from $\mathbf{C}^{(n)}$ and $\mathbf{F}\left[\mathbf{C}^{(n)}\left(\mathbf{C}^{(n)}\right)^\dagger\right]$. End the process if $\left\|\mathbf{e}^{(n)}\right\|_{\text{frob}} \leq \varepsilon_{\text{conv}}$.
- Calculate the new error overlaps $B_{0i} = \left\langle \mathbf{e}^{(n)} \middle| \mathbf{e}^{(n-i)} \right\rangle$. Note that only one row of $\mathbf{B}$ has to be calculated, since the others can be kept from the previous SCF iteration. Drop coefficients, which are beyond the m Fock matrices to keep and repeat the process thereafter.

The Pulay DIIS scheme outlined here is rather general and has been frequently applied to problems other than solving the HF or Kohn-Sham equations [202]. For example it can be applied to accelerate the convergence of fixed-point problems as they occur for example in coupled-cluster theory, see section 4.5.4 on page 77. In the optimisation community the DIIS technique is known as **Anderson acceleration**[203] in this context.

Last but not least one should mention that a few improvements to the DIIS have been suggested recently. This includes the **energy DIIS** [201], which effectively interpolates between densities resulting from the ODA accelerating ODA convergence whilst showing mathematically highly desirable properties. Shepard and Minkoff [139] have suggested ways to improve the DIIS by reformulating the original problem into a least squares or a linear least squares problem. The augmented Roothaan-Hall DIIS [204] is another take to yield a linear-scaling method with improved convergence. The **least-squares commutator in the Iterative Subspace** [205] approach can also be seen as a variant of the DIIS trying to correct some of its issues.

5.4.6 Second-order self-consistent-field algorithms

To conclude our review of selected SCF schemes this section will briefly touch upon so-called **second-order SCF algorithms**. Generally these methods try to go beyond incorporating gradient information of the HF minimisation problem (4.65). Next to solving the HF equations (4.74) these methods thus incorporate the Hessian of the energy functional (4.59) with respect to the parameters $\mathbf{C}$ as well. The aim is both to achieve faster convergence and to ensure that convergence is not to any stationary state on $\mathcal{C}$, but instead to a true SCF minimum. Since forming the Hessian of (4.65) can become rather expensive, many approaches use approximate Hessians instead. Typically it is observed that convergence only becomes quadratic once the SCF procedure is already reasonably close to the minimiser of (4.65).

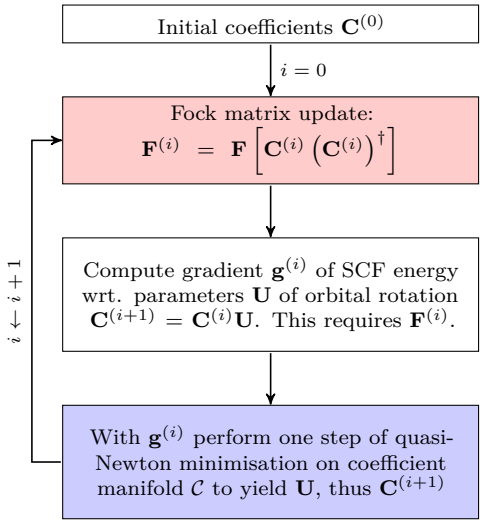


Figure 5.15: Schematic of the geometric direct minimisation algorithm. The step which updates the Fock matrix is highlighted in red and the step which updates the coefficients is highlighted in blue.

One of the first approaches, which fall in this category, is the **quadratically-convergent SCF** (QCSCF) [206]. This approach minimises the SCF energy by finding the minimum of a configuration interaction singles-doubles expansion based on the current set of SCF orbitals. Similar to most methods discussed in this section, QCSCF employs normalised basis functions $\{\varphi_\mu\}_{\mu\in\mathcal{I}_{\text{bas}}}$. These have the advantage, that the occupied SCF coefficients $\mathbf{C}^{(n)}$ can be alternatively parametrised [84] as

$$\mathbf{C}^{(n)} = \mathbf{C}^{(0)}\mathbf{U}^{(n)} = \mathbf{C}^{(0)}\exp(-\mathbf{K}^{(n)}),$$

where $\mathbf{U}^{(n)} \in \mathbb{R}^{N_{\text{orb}}\times N_{\text{orb}}}$ is a unitary rotation matrix and $\mathbf{K}^{(n)}$ is an anti-Hermitian matrix. Further details about the orbital rotation ansatz for SCF methods can be found in [84].

Related to the idea of an orbital rotation SCF is the **geometric direct minimisation** method [207], see figure 5.15. This approach tries to directly optimise the coefficients $\mathbf{C}_{(n)}$ in the sense of (4.65) on the Stiefel manifold $\mathcal{C}$. The algorithm determines the energy gradient $\mathbf{g}^{(n)}$ of the current coefficient set $\mathbf{C}^{(n)}$. Then it follows the geodesic defined by $\mathbf{g}^{(n)}$ to find a unitary rotation matrix $\mathbf{U}$ which defines the new set $\mathbf{C}^{(n+1)}\mathbf{U}$. The step size is determined by a Newton-Raphson-like step, where the Hessian is constructed approximately using the Broyden-Fletcher-Goldfarb-Shanno update scheme.

Last but not least one should mention recent linear scaling SCF approaches, for example the one by Sałek et al. [208] as well as the augmented Roothaan-Hall method [204]. Both of these can be seen as approximate second-order SCF methods, which try to directly minimise the coefficients as well.

5.4.7 Combining self-consistent field algorithms

In the previous sections we mentioned quite a few approaches to solve the HF minimisation problem using a self-consistent field ansatz. Needless to say that different

algorithms tend to perform best in different cases. For this reason in practice often a mixture of methods is employed in order to guarantee fast and reliable convergence. This section represents my own judgement of the situation and give some suggestions based on my own experience. Hardly any of this is resulted from any kind of proper scientific evaluation[24] and should therefore not be taken as a final answer, much rather as a guideline.

In the beginning of the procedure ODA or tODA work great, since they essentially direct the coefficients reliably into the right direction, breaking the oscillatory behaviour of the plain Roothaan algorithm. The energy DIIS can be seen as an accelerated improvement of those methods, which is recommendable for cGTO-based discretisations as the initial SCF algorithm in my point of view.

For the intermediate steps a Pulay DIIS shows typically a faster convergence than the energy DIIS [201]. This can be rationalised by considering the conditions on the coefficients for the linear combination of Fock matrices. In the Pulay DIIS these conditions are much laxer compared to the energy DIIS[25], making it easier to explore the SCF manifold and search for directions which lead to nearby stationary points.

Close to convergence the DIIS becomes numerically more unstable, but conversely second-order SCF schemes like QCSCF now show the fastest and most reliable convergence to the SCF minimum. These should be considered in the final SCF steps in order to obtain a highly-accurate SCF minimum after the DIIS.

5.5 Takeaway

The SCF algorithms we discussed in this section all follow the general scheme, where a **Fock-update** step and a **coefficient-update** or **density-matrix-update** step are repetitively executed. In the former step a new Fock matrix $\mathbf{F}^{(n)}$ is constructed from the present set of SCF coefficients $\mathbf{C}^{(n)}$ or the present density matrix $\mathbf{D}^{(n)}$. In the latter step this Fock matrix $\mathbf{F}^{(n)}$, perhaps with additional insight gained in previous iterations, is used in order to generate a new set of coefficients $\mathbf{C}^{(n+1)}$ and perhaps from this a new density $\mathbf{D}^{(n+1)}$. For Roothaan's repeated diagonalisation, the optimal damping algorithm and the geometric direct minimisation algorithm this sequence of steps is emphasised in figures 5.14 on page 130 and 5.15 on the preceding page, where the Fock update step is highlighted in red and the coefficient/density update step in blue in each case. Motivated by the deviating structure of the aforementioned algorithms I consider it reasonable to assume that all SCF algorithms can be thought of in such a two-step process.

Another key result in this chapter is that different basis function types give rise to different numerical structure of the quantities involved in the SCF procedure. We focused most on the Fock matrices of contracted Gaussian, finite-element and Coulomb-Sturmian discretisations, which are shown in figures 5.4 on page 100, 5.9 on page 112 and 5.13 on page 125. These matrices differ both in size as well as in sparsity. Both for FE-based as well as CS-based discretisations a contraction-based ansatz[26], where one avoids building the Fock matrix at all and instead thinks in terms of matrix-vector applications, showed

[24] Unfortunately I am not aware of a work, which properly compares the large range of SCF algorithms with another. Most papers only compare to the Pulay DIIS.

[25] The Pulay DIIS *extrapolates*, whereas the energy DIIS *interpolates*.

[26] See the next chapter for a proper introduction into the concept of contraction-based methods.

noteworthy improvements in formal computational scaling.

As we will discuss in depth in the next chapter, a contraction-based ansatz can be thought of as a generalisation of a scheme keeping the matrices in memory. This suggests targeting a contraction-based SCF scheme to achieve maximum generality of the SCF algorithm and potentially independence of the SCF code from the basis function type in a quantum chemistry program.

As mentioned before this implies to formulate the SCF in terms of coefficients to exploit the favourable computational scaling for some basis function types like the FE or the CS functions. We indicated for the ODA algorithm how approximations allow to transform this density-matrix-based SCF into the tODA scheme, which can be formulated as a contraction-based SCF. In section 5.1 on page 85 we furthermore gave more general suggestions, which allow to transform every density-matrix-based SCF into a coefficient-based SCF in theory. We therefore believe it to be possible to construct an efficient contraction-based SCF, which is independent from the type of basis function used and where one is able to switch between multiple algorithms depending on the numerical requirements of the basis functions as well as the chemical system. This in turn opens the door for achieving a single quantum-chemistry program, which is in theory compatible with every type of basis function. We will present such a program in chapter 7.

Chapter 6

Contraction-based algorithms and lazy matrices

> *There is a race between the increasing complexity of the systems we build and our ability to develop intellectual tools for understanding their complexity. If the race is won by our tools, then systems will eventually become easier to use and more reliable. If not, they will continue to become harder to use and less reliable for all but a relatively small set of common tasks. Given how hard thinking is, if those intellectual tools are to succeed, they will have to substitute calculation for thought.*
>
> — Leslie Lamport (1941–present)

Summarised in one sentence the main idea of contraction-based algorithms is to avoid storing large matrices or tensors in memory and instead employ highly optimised contraction expressions for the necessary computations. We already saw in the previous chapter that applying such a strategy to the Fock matrix resulting from a FE-based or a CS-based discretisation can lead to an improved formal computational scaling, making these methods a promising approach. Contraction-based algorithms are, however, not at all limited to SCF procedures or quantum-chemical calculations. This chapter will give a general overview of contraction-based methods, giving some examples where these methods are employed as well as discussing the potentials and some drawbacks.

Closely connected to contraction-based methods is the concept of lazy matrices, which is a direct generalisation to the conventional matrices in the form of a domain-specific language for coding contraction-based algorithms. Main goal of the lazy matrix language is to yield code, which can be used both with matrices stored in memory and additionally in a contraction-based fashion without noteworthy changes. A preliminary `C++` implementation of lazy matrices with focus on user-friendliness and flexibility is available in the `lazyten` library.

6.1 Contraction-based algorithms

The underlying idea of contraction-based methods, namely to avoid storing large matrices in favour of using matrix-vector-product expressions, is hardly new. In his paper from 1975 Davidson [66] not only describes his now famous iterative diagonalisation method (see section 3.2.6), but furthermore he suggests to use an algorithmic expression for computing the required matrix-vector products. The use case Davidson had in mind back then was the diagonalisation of the CI or full CI matrix, which is — even today — too large to keep in memory, see remark 4.9 on page 56.

Nowadays contraction-based methods are rather widespread in quantum chemistry. Even though the contraction expressions are sometimes given different names such as **working equations**, making the concept less clear. Examples are recent implementations of the algebraic diagrammatic construction (ADC) scheme [209–211], which do not build the complete ADC matrix to be diagonalised, and efficient coupled-cluster schemes [84], which similarly avoid constructing the matrix governing the CC root-finding problem explicitly. Instead both methods use appropriate tensor contractions and compute matrix-vector products on the fly during the respective iterative solves. A somewhat related take on this are the recent **matrix-free methods** [162] for solving partial differential equations in a finite-element discretisation without building the system matrix in memory at all.

From the algorithmic point of view one should notice, that especially the direct eigensolvers and linear solvers algorithms as they are implemented in LAPACK[212] do require random access into the matrix, are thus not available for a contraction-based ansatz. In practice this an acceptable restriction. Firstly because for large matrices direct methods become unfavourably expensive anyway[1]. Secondly because many diagonalisation methods and methods for solving linear systems do not need the problem matrix in memory. Instead they can be operated just like the Davidson algorithm [66], by coding an expression for delivering the required matrix-vector products. In this category practically all Krylov-subspace approaches can be found, including widely-adopted algorithms like Arnoldi, Lanczos, conjugate gradient or GMRES [62, 63, 67]. In the context of eigenproblems one should mention that such iterative methods have an additional disadvantage. It is typically very costly to obtain a large number of eigenpairs of the diagonalised matrix. Fortunately this is hardly needed for large matrices and techniques like Chebyshev filtering [213–215] or spectral transformations (see section 3.2.4 on page 38) allow to effectively direct the diagonalisation routines towards the part of the eigenspectrum one is truly interested in.

On the one hand, employing a contraction-based method thus does not really restrict the range of numerical problems, which can be tackled. On the other hand avoiding the storage of the problem matrix immediately reduces the scaling in memory from quadratic (in system size) to linear. The rationale for this is that the memory bottleneck in most subspace algorithms is storing the generated subspace, i.e. a fixed number of vectors, which take linear storage. This makes contraction-based methods especially attractive for problems where memory is a bottleneck. Therefore this concept has been introduced in a range of fields of numerics and scientific computing under different names. Terms like **apply-based** method, **matrix-free** method or phrases like using **matrix-vector product expressions** or using **matrix-vector products** overall largely describe the same concept. I personally like the term **contraction-based** best, because under the

[1] Usually exactly because they necessarily keep everything in memory.

Storage layer	Latency /ns	FLOPs
L1 cache	0.5	13
L2 cache	7	180
Main memory	100	2600
SSD read	$1.5 \cdot 10^4$	$4 \cdot 10^5$
HDD read	$1 \cdot 10^7$	$3 \cdot 10^8$

Table 6.1: Typical latency times required for random access into selected layers of storage. The right-hand side column represents the peak amount of floating point operations a Sandy Bridge CPU with 3.2 GHz clock frequency could perform in the same time assuming perfect pipelining. Data taken from [216] and [217]. Notice, that the seek time on HDDs averages out in sequential HDD reads. For example reading 1 MB from disk only takes about $2 \cdot 10^7$ ns, i.e. only twice as long as the seek by itself. For other types of storage this effect is less pronounced.

hood evaluating such matrix-vector products in many cases, that I came across, involves expressions with contractions over tensors with rank larger than 2. Consider for example the coupled-cluster doubles working equations (4.96) or the contraction expression for the exchange matrix in a CS-based discretisation of Hartree-Fock (5.52). Additionally, calling such algorithms contraction-based indicates that the idea to substitute storage by expressions is more general than the matrix-vector product. In theory one could think of similar approaches for higher-order tensor contractions as well.

6.1.1 Potentials and drawbacks

Historically the main driving force behind contraction-based methods was to circumvent the memory bottleneck. Since the amounts of memory available in the mainframes of the 70s was much more limited compared to today, the only alternative to recomputing the data as needed would have been to store the system matrices on disk. Taking a look at table 6.1 we notice that in the time needed to perform random access to HDDs in the order of millions of floating point operations can be performed. Overall it is therefore not hard to understand why people went for recomputing the data instead.

Nowadays the amounts of available main memory has increased substantially, such that for larger and larger system sizes, the required matrices can now be stored in memory. Nevertheless one should not forget that accessing main memory costs time as well, which could be equally well spent for computations. Assuming perfect pipelining on the order of 2500 floating point operations can be performed while the CPU waits for a random value to be loaded from main memory (see table 6.1). This is of course orders of magnitude lower than the corresponding value for a random read from SSD or HDD, but one should notice that this does not improve as much for sequential reads as it does for HDDs. In other words this 2500 flops of computation are in some sense lost for each memory access — whether it is sequential or completely random.

Another aspect one should take into account are the historic trends. Figure 6.1 clearly[2] shows the so-called **processor-memory performance gap**. This is meant to

[2]The original source [218] does not provide a clear description how the data points in each year were computed and what kinds of processors and chipsets were selected for each year. Nevertheless the trend is so clear, that I consider this aspect to have little influence on the conveyed picture.

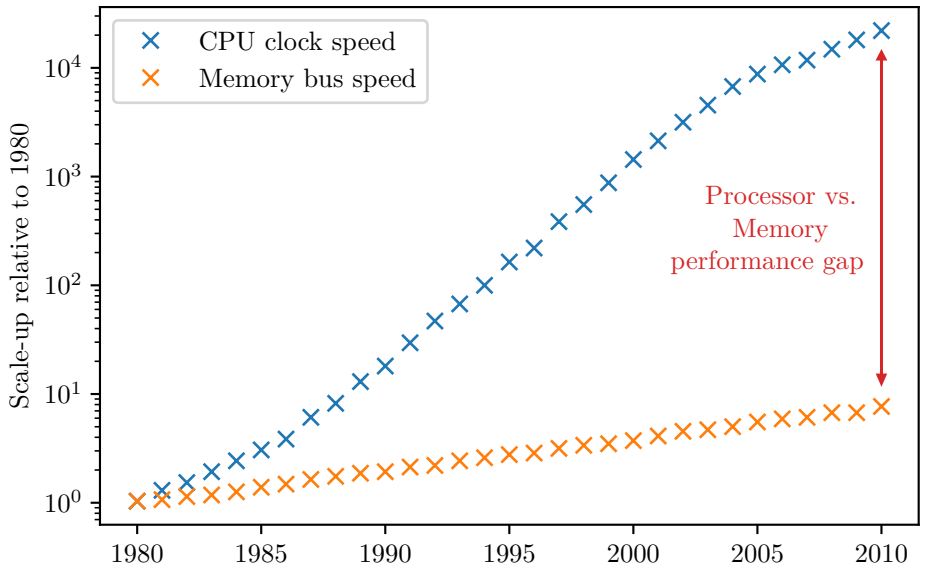


Figure 6.1: Scale-up of memory bus speed and CPU clock speed relative to 1980 for selected hardware in each year. Data taken from [218].

express that the available memory bandwidth has increased by a lesser amount relative to 1980 as the CPU clock speed. Since the number of flops per second is directly related to the CPU clock speed, one can extrapolate that the ratio of computable FLOPs per memory access will likely increase in the upcoming years. In other words contraction-based methods will become more and more attractive in the future, just because they amount to exploit the steeper increase of the CPU clock speed curve in figure 6.1 much better.

Notice, that the aforementioned advantage of contraction-based methods to exploit the emerging hardware trends is an effect which comes *on top* of the possible reduction in formal computational scaling, which we found for FE-based or CS-based discretisations of HF in the previous chapter. This reduction in scaling is not an effect which is limited to the case of an SCF routine, but can be observed in other cases as well. The underlying reason is in many cases that the delayed evaluation of the matrix elements *alongside* the contraction with an actual trial vector allows to reorder the required summations more freely. In other words giving up the storage of values implies that we do no longer need to comply with one particular storage format, allowing to more freely choose an optimal evaluation strategy. Somewhat paradoxically this implies that contraction-based method have the potential to be faster even though more computational work is done.

Let us consider an iterative diagonalisation method, like the Arnoldi or Davidson scheme in a contraction-based ansatz. All steps but the matrix-vector product expression are performed in the generated iterative subspace, which by construction has a lower dimensionality than the full system matrix. Therefore the matrix-vector product, i.e. the contraction expression is the computational bottleneck. A typical diagonalisation requires around low to mind hundreds of matrix-vector products. Implementing this

contraction expression in a highly efficient manner is key to make a contraction-based ansatz fast.

Even in the naïve manner presented in equations (5.25) and (5.52) the contraction expressions of the exchange term of the Fock matrix with a trial vector look all but simple. In a practical implementation achieving maximal performance the required procedures will most probably be more involved, since issues like the following will all need to be addressed.

- **Adoption to Hardware** and **parallelisation**: The features provided by modern hardware have of course changed a lot over the years. This includes aspects like vectorisation or the recent trend to employ general-purpose graphics cards in scientific calculations. A good algorithm takes modern features into account and shows a parallelisation scheme, which exploits the available hardware as good as possible. Notice, that in many cases the requirements can be contradictory, such that achieving best performance in all circumstances is a real challenge if not impossible.
- **Storing intermediates:** Often one can identify subexpressions of a large contraction expression, where it makes sense to store it between individual executions of the contraction.
- **Order of contractions:** For more complicated expressions involving multiple tensor contractions at once the order in which the contractions are executed can be crucial to achieve best computational scaling as well as a low memory footprint.
- **Approximations:** Especially in iterative procedures one is typically not interested in the numerically exact result of a contraction. Much rather the iterative procedure will only solve the problem up to a certain accuracy threshold, such that computing elements, which are smaller than this threshold is a waste of computational time. Sometimes this can be incorporated into a contraction expression by prescreening the elements to compute or by other approximations.

On top of that one should keep in mind that problem matrices are usually composed out of different terms with potentially different structures. In the case of a finite-element-based SCF, for example, the Fock matrix is as sum of the local one-electron terms $\mathbf{T}$ and $\mathbf{V}_0$, which can be directly computed, the Coulomb term $\mathbf{J}$, which requires the solution to a Poisson equation as well as and the exchange term $\mathbf{K}$, which requires to solve multiple equations on each single apply. It is therefore not hard to imagine that the best approach to the issues raised above might well differ from term to term. Compared to the traditional case where all these matrices are kept in memory, adding the terms $\mathbf{T}$, $\mathbf{V}_0$, $\mathbf{J}$ and $\mathbf{K}$ to form the Fock matrix $\mathbf{F}$ is thus much more involved in the contraction-based ansatz.

In passing let us note, that the parallelisation of contraction-based methods is typically both easier and more efficient than for conventional methods. The rationale is that less stored data generally implies that there is less data to manage between different cores or nodes and thus less data to communicate, because it is recomputed on each worker as required. Unlike communicating data, recomputing data is embarrassingly parallel after all.

Overall we can conclude that contraction-based methods are more flexible and amount to comply better with the current hardware trends. Since contraction expressions are the computational hot-spot, the procedures one needs to implement are, however, more

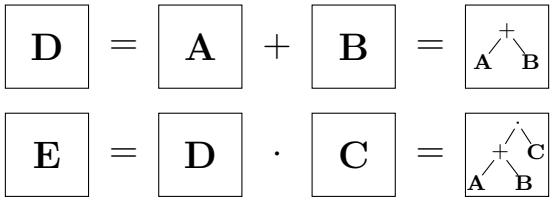


Figure 6.2: Examples for lazy matrix expression trees. The upper represents the instruction $\mathbf{D} = \mathbf{A} + \mathbf{B}$ and the lower the multiplication of the result $\mathbf{D}$ with $\mathbf{C}$.

involved and consequently the resulting code can become less intuitive and hard to modify or adapt.

6.2 Lazy matrices

The idea of lazy matrices is to encapsulate the coding contraction-based in a domain-specific language, which makes it feel as if one was dealing with actual matrices instead of contraction expressions. Even though not all complications can be hidden, the resulting syntax allows to write algorithms in a high-level manner being independent from the underlying implementation of the contraction expression. This will turn out to be the key aspect leading to the basis-type of the quantum-chemistry package `molsturm`.

For this purpose we generalise the concept of a matrix to objects we call a **lazy matrix**. Whilst a conventional or **stored matrix** is dense and has all its elements residing in a continuous chunk of memory, this restriction does no longer hold for a lazy matrix. It may for example follow a particular sparse storage scheme like compressed-row storage, but does not even need to be associated to any kind of storage at all. In the most general sense it can be thought of as an arbitrary contraction expression for computing the matrix elements, which is dressed to look like an ordinary matrix from the outside.

In other words one may still obtain individual matrix elements, add, subtract or multiply such lazy matrix objects together or apply them to a bunch of vectors or a stored matrix. Not all of these operations may be equally fast than there counterparts on stored matrices, however. Most importantly obtaining individual elements of such a matrix can become rather costly, since they might involve a computation as well and not just a lookup into memory.

On the upside one gains a much more flexible data structure where a familiar matrix-like interface can be added to more complicated objects. Most notably a lazy matrix may well be non-linear or can have a state, which may be changed by a update function in order to influence the represented values at a later point. An example where this is sensible would be the Coulomb and Exchange matrices, where the values of these matrices depend on the set of occupied coefficients, which have been obtained in the previous iterations. Other examples include the update of an accuracy threshold for a contraction expression, which might change between iterations.

All evaluations between lazy matrices like addition, subtraction or matrix-matrix multiplication is usually delayed until a contraction of the resulting expression with a vector or a stored matrix is performed and thus the represented values are unavoidably needed. This evaluation strategy is called **lazy evaluation** in programming language

theory [219], explaining the name of these data structures. To make this more clear consider the lazy matrix instructions

$$\begin{aligned} \mathbf{D} &= \mathbf{A} + \mathbf{B}, \\ \mathbf{E} &= \mathbf{D}\mathbf{C}, \\ \underline{\boldsymbol{y}} &= \mathbf{E}\underline{\boldsymbol{x}}, \end{aligned} \tag{6.1}$$

where $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are lazy matrices and $\underline{\boldsymbol{x}}$ is a vector stored in memory. The first two do not give rise to any computation being done. They only amount to build an expression tree in the returned lazy matrix $\mathbf{E}$ as illustrated in figure 6.2 on the facing page. The final line is a matrix-vector product with a stored vector, where an actual stored result should be returned in the vector $\underline{\boldsymbol{y}}$. In the lazy matrix sense this triggers the complete expression tree to be evaluated in appropriate order, leading effectively to an evaluation of the expression

$$\underline{\boldsymbol{y}} = (\mathbf{A} + \mathbf{B})\,\mathbf{C}\underline{\boldsymbol{x}} \tag{6.2}$$

at once at this very instance. (6.2) can be evaluated entirely only using matrix-vector contraction expressions. For example one could first form the product $\underline{\tilde{\boldsymbol{x}}} \equiv \mathbf{C}\underline{\boldsymbol{x}}$ using the matrix-vector-product expression of the lazy matrix $\mathbf{C}$. Afterwards one would form $\mathbf{A}\underline{\tilde{\boldsymbol{x}}}$ and $\mathbf{B}\underline{\tilde{\boldsymbol{x}}}$ again by appropriate contraction expressions and finally add the result to give $\underline{\boldsymbol{y}}$. This is just one way to perform the evaluation. An implementation of the lazy matrix language is free to choose a different route for evaluating (6.2) by reordering the expression if it considers this useful. If $\mathbf{C}$ for example was made up of a sum $\mathbf{F} + \mathbf{G}$, it could use distributivity to write

$$(\mathbf{A} + \mathbf{B})\,(\mathbf{F} + \mathbf{G})\,\underline{\boldsymbol{x}} = \mathbf{A}\,(\mathbf{F}\underline{\boldsymbol{x}}) + \mathbf{A}\,(\mathbf{G}\underline{\boldsymbol{x}}) + \mathbf{B}\,(\mathbf{F}\underline{\boldsymbol{x}}) + \mathbf{B}\,(\mathbf{G}\underline{\boldsymbol{x}})\,.$$

Which of these routes is best differs very much on the structure of the lazy matrices being part of the expression to evaluate. But other factors like the operating system or hardware on which the program code is executed are not unimportant either. Since the evaluation is delayed until the call to $\mathbf{E}\underline{\boldsymbol{x}}$ gets executed at the actual program runtime, all of this can in theory be taken into account for deciding which route to take. Naturally the design of an appropriate cost function is not easy as previous works have shown [220–226]

In either case such decision happen in the evaluation back end and are well-abstracted by the lazy matrix language from the instructions (6.1), which stay intelligible and understandable. Furthermore if the structure of the matrices changes, since for example the discretisation scheme changes, the evaluation route will automatically adapt given that the cost function is sensibly chosen.

6.3 Lazy matrix library `lazyten`

An initial implementation of the lazy matrix language has been achieved in the `C++` library `lazyten` [227]. Not all aspects of lazy matrix concept are yet covered, however. For example many opportunities to achieve performance improvements by reordering the lazy matrix expression tree are currently missing. On the other hand `lazyten` goes a bit beyond the lazy matrix specification in the sense that it has become a full abstraction layer for linear algebra. As depicted in figure 6.3 on the next page the goal of `lazyten` is to provide a common interface for contraction-based methods with access to different linear algebra back ends or solver implementations. Not everything has been achieved

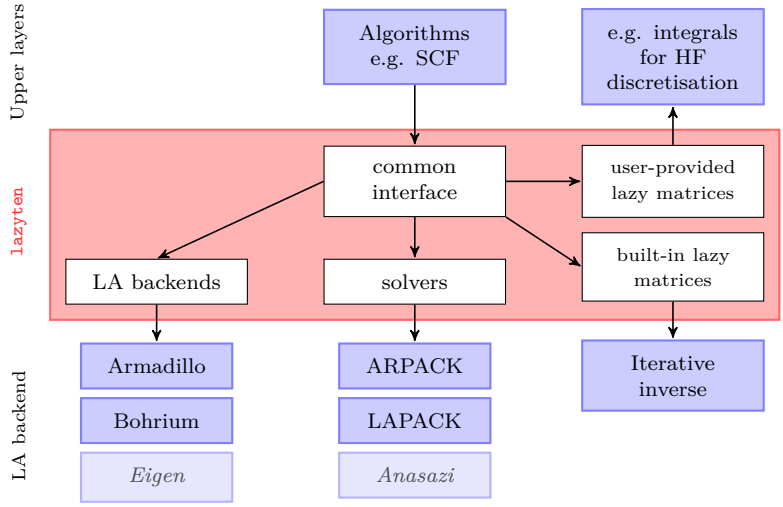


Figure 6.3: Structure of the `lazyten` lazy matrix library [227] and its interfaces to the 3rd party codes armadillo [228], Bohrium [225, 226], LAPACK [212] and ARPACK [229]. Support for Eigen [230] and Anasazi [231] is planned.

as planned, but nevertheless `lazyten` is already used in production by the `molsturm` quantum-chemistry framework discussed in the next section.

`lazyten` is open-source software licensed under the GNU General Public License. Its source code can be obtained from `https://lazyten.org` free of charge. As of December 2017 `lazyten` amounts around 22500 source lines of code excluding comments and blanks, but including the helper library `krims` [232] as well as examples and tests.

Inside the framework of `lazyten` combining custom lazy matrices as well as built-in structures, like a lazy matrix representing the inverse of a matrix, can be achieved transparently. Even a combination with stored matrices in any of these expressions is possible. In this manner code working on `lazyten` matrix objects will continue to work if the type of one of the involved objects is changed. In other words replacing a plain stored matrix by an involved lazy matrix, which exploits the sparsity properties of the represented quantity much better, can typically be done without changing any of the code operating on such a matrix.

This is possible, since the interface of `lazyten` provides high-level routines to perform linear solves and to access eigensolvers, where the call passes through a branching layer, which mediates between the available back ends depending on the structure of the problem matrix. By providing appropriate parameters to the high-level function a user of the implemented code may still overwrite which solver implementation is chosen and what precise setup parameters are passed to it. Currently a selection of methods from the LAPACK [212] linear algebra library as well as the ARPACK [229] package for Arnoldi diagonalisation methods is available from `lazyten`. The selection mechanism between the different algorithms for one particular task is not yet extremely sophisticated. Generally it will for example favour direct diagonalisation methods from LAPACK [212] if many eigenpairs are requested or if the supplied system matrix is already dense. On the other hand Arnoldi methods are selected for lazy matrices and if only very few

eigenpairs are desired.

Whenever an operation like a product of a lazy matrix with a stored vector unavoidably requires computation, `lazyten` addresses the employed LA back end through an abstracted interface, such that switching behaviour on this layer is possible as well. At the present stage armadillo [228] as a LAPACK-based back end as well as Bohrium [225, 226] as an array-operation-based back end are currently available. Rather inconveniently switching the back end right now requires to recompile `lazyten` with the appropriate configure options.

For evaluating a lazy matrix contraction expression the LA back end is typically not extremely important, since it is only required for very few operations. Consider for example the third line of (6.1) above, where evaluation of the product $\mathbf{E}\underline{\boldsymbol{x}}$ is required. Most work is done by the matrix-vector contraction expressions of the lazy matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$. Only for the final sum of the vectors $\mathbf{A}\underline{\tilde{\boldsymbol{x}}}$ and $\mathbf{B}\underline{\tilde{\boldsymbol{x}}}$ `lazyten` passes on to the LA back end. The impact of changing the back end is naturally larger for operations between stored matrices or vectors, where it is used to evaluate all arising expressions.

6.3.1 Examples

To finish off this section, we demonstrate the high-level syntax of `lazyten` in two example cases. First consider a general linear problem $\mathbf{A}\underline{\boldsymbol{x}} = \underline{\boldsymbol{b}}$ with known right-hand side $\underline{\boldsymbol{b}}$ and unknown $\underline{\boldsymbol{x}}$. The system matrix $\mathbf{A}$ shall be represented by the `lazyten` matrix object `A` and the right-hand side $\underline{\boldsymbol{b}}$ by the object `b`, which is taken to be a simple stored vector of type `SmallVector<double>`. In `lazyten` there are two absolutely equivalent ways to solve this problem. First

```
SmallVector<double> x(b.size());
solve(A, x, b);
```

or equivalently

```
auto invA = inverse(A);
auto x = invA * b;
```

i.e. quite literally coding the application of the inverse. In both cases the last line will cause the problem to be passed to a linear solver algorithm in order to solve it iteratively or by direct methods. The user may provide extra parameters to the calls of `solve` or `inverse` in order to influence the selected eigensolver algorithm or provide some means of preconditioning the problem matrix.

The second example is more relevant to the scope of this work and brings us back to the end of section 5.5 on page 139, where we discussed the possibility of an SCF routine, which is independent from the type of basis function used. Figure 6.4 on the next page shows a code fragment from a very simple Roothaan repeated diagonalisation SCF routine (see section 5.4.1 on page 128) for closed-shell systems coded in the syntax of `lazyten`.

Before the depicted code segment is executed, the integral library is given information about the chemical system and the desired discretisation and returns the objects `Tbb`, `Vbb`, `Jbb`, `Kbb` and `Sbb`, which represent the matrices $\mathbf{T}$, $\mathbf{V}_0$, $\mathbf{J}$, $\mathbf{K}$ and $\mathbf{S}$ as they are defined in (4.60) to (4.64). Additionally parameters appearing in the code include `n_alpha`, the number of alpha electrons and `n_orb`, the number of SCF orbitals to compute in each step.

```
// Obtain a core Hamiltonian guess
const auto hcorebb       = Tbb + Vbb;
const auto eigensolution = eigensystem_hermitian(hcorebb, Sbb,
                                                 n_orbs);

// View current occupied coefficients in convenient data structure
const auto cocc = eigensolution.evectors().subview({0, n_alpha});

// Initialise two-electron terms with guess coefficients
Jbb.update({{"coefficients_occupied", cocc}});
Kbb.update({{"coefficients_occupied", cocc}});

// Start Roothaan repeated diagonalisation
double oldene = 0;
std::cout << "Iter      etot      echange" << std::endl;
for (size_t i = 0; i < max_iter; ++i) {
  // Obtain new eigenpairs ...
  const auto Fbb           = hcorebb + (2 * Jbb - Kbb);
  const auto eigensolution = eigensystem_hermitian(Fbb, Sbb,
                                                   n_orbs);

  // ... and a new view to the occupied coefficients
  const auto cocc = eigensolution.evectors().subview({0, n_alpha});

  // Compute HF energies:
  //   Coulomb energy is 2 * tr(C^T J C),
  //   where 2 appears, since we only consider alpha block,
  //   but both alpha and beta coefficients would contribute.
  double ene_coulomb  = 2 * trace(outer_prod_sum(cocc,
                                                 Jbb * cocc));
  double ene_exchge   = -trace(outer_prod_sum(cocc,
                                              Kbb * cocc));
  double ene_one_elec = trace(outer_prod_sum(cocc,
                                             hcorebb * cocc));
  double energy       = 2 * (ene_one_elec + 0.5 * ene_coulomb +
                             0.5 * ene_exchge);

  // Display current iteration
  double energy_diff = energy - oldene;
  std::cout << i << " " << energy << " " << energy_diff << ↙
    ↪std::endl;
  oldene = energy;

  // Check for convergence
  if (fabs(energy_diff) < 1e-6) break;

  // Update the two-electron integrals,
  // before coefficients go out of scope
  Jbb.update({{"coefficients_occupied", cocc}});
```

Figure 6.4: Code fragment of a simple basis-type independent Hartree-Fock procedure implemented with `lazyten`. The procedure follows the Roothaan repeated diagonalisation algorithm in the specialisation for closed-shell system (see section 5.4.1).

Alongside the comments the code should largely be self-explanatory. In lines 1 to 6 a core Hamiltonian guess is obtained by diagonalising $\mathbf{T} + \mathbf{V}_0$ (see 5.2.1). Then the Coulomb and exchange lazy matrices are updated to the guess coefficients in lines 9 and 10. Depending on the implementation of these lazy matrices, this might already involve the computation of the matrices (4.62) and (4.63), but for others this might just update an internal reference to the current set of coefficients and apart from that do nothing. From what we discussed in the previous chapter it should be clear that the former is better-suited for a cGTO discretisation and the latter from a FE-based discretisation for example.

Afterwards the main loop starts, where first the Fock matrix expression is built in line 17 and then diagonalised in line 21. Then the current energies are computed in lines 27 to 30 following (4.59) making vivid use of the `outer_prod_sum` routine. Right now this routine is required in such a case originating from the unfortunate decision to represent a matrix and a set of vectors as two inherently different data structures. Effectively it computes products such as $\mathbf{C}^T(\mathbf{JC})$ from the matrices $\mathbf{C}$ and $\mathbf{JC}$ represented as a list of vectors. The remaining lines 32 to 43 print the current iteration to the user, check for convergence and update the state of $\mathbf{J}$ and $\mathbf{K}$ for the next iteration.

Despite its simplicity the depicted code is independent of the type of basis function used to discretise the problem as `lazyten` automatically adapts the executed eigensolver routines for the calls in lines 6 and 18 to the structure of the Fock matrix. Indirectly it is thus the structure of the matrices `Tbb`, `Vbb`, `Jbb`, `Kbb` and `Sbb` and usually[3] not the code depicted in figure 6.4 which decides, which eigensolver algorithms are chosen. Given that the basic heuristics currently implemented, the depicted code would for example perform a contraction-based SCF for a CS-based discretisation and use direct eigensolves for a cGTO-based discretisation. In the light of this `lazyten` becomes a very effective abstraction layer between the details of the lazy matrix implementation, i.e. the integral evaluation, and the SCF algorithm.

In the SCF depicted in figure 6.4 many expressions like lines 17 and 18 or the energy computation are designed to resemble the equivalent equations one would derive on paper up to a very large extent. Nevertheless the matrices like `Tbb`, `Vbb`, `Jbb`, `Kbb` and `Sbb` could be stored or lazy for the code to work. Adding an extra term to the Fock matrix expression in line 17 can be done by a simple addition of another matrix object irrespective whether the added object is lazy or stored. In either case its structure would be taken into account during the following diagonalisation without explicit user interaction. Still the user could influence the behaviour of the called solver by providing appropriate parameters explicitly. For this reason we believe `lazyten` to be very suitable for teaching or experimentation with novel methods, since many details are abstracted and one may at first concentrate on the algorithm and not on numerics.

Overall `lazyten` therefore amounts to yield a very intuitive syntax for contraction-based methods in the form of lazy matrices, where algorithms can be written at a high level. By means of changing the implementation behind the employed lazy matrix objects the code can be fixed but still flexible to changes in the available hardware or if novel types of basis functions with unusual matrix structures become available.

[3] Since the automatic selection methods are not yet extremely advanced, it is necessary to overwrite the automatic choice from user code from time to time.

Chapter 7

The `molsturm` method development framework

> *[C has] the power of assembly language and the convenience of ... assembly language.*
>
> — Dennis Ritchie (1941–2011)

> *Keep it simple, make it general, and make it intelligible.*
>
> — Doug McIlroy (1932–present)

When the `molsturm` project [40] was initiated a couple of years back, the original idea was to support *mol*ecular calculations with *Sturm*ian-type basis functions in a formulation similar to the contraction-based ansatz mentioned in section 5.3.6, which explains the name. Following my unsuccessful attempts on a finite-element-based Hartree-Fock scheme, where I mostly used the approaches known to me from a cGTO setting, we expanded the goals of `molsturm`. Now, the primary project goal has become to yield a quantum-chemistry method development framework, which supports the implementation and evaluation of discretisations based on novel types of basis functions. We have achieved this by building largely on the conclusions about the common structure of SCF algorithms and the generality of the lazy matrix syntax of `lazyten`, which were discussed in the previous chapters. Additionally `molsturm` has been equipped with a powerful `python` interface, where it is easy to obtain, archive and analyse results. Even implementing completely new features on top of `molsturm`'s SCF procedure often takes comparatively little development time as will be demonstrated. The arguments and examples of this chapter follow closely our publication [233].

All components of the `molsturm` program package are open-source software, licensed under the GNU General Public License. To obtain a copy of the code go to `https://molsturm.org`.

7.1 Related quantum-chemical software packages

Apart from molsturm I am unaware of another quantum-chemistry package which has achieved a similar flexibility with respect to the type of basis functions in its SCF procedure. Many packages still have related goals towards flexibility or generality of their codes and should therefore not go completely unmentioned here.

When it comes to flexibility of a program package a key ingredient is a versatile interface. This allows to invoke or extend the methods already available elsewhere. Recently the scripting language python has become very popular for achieving this. Even even meta-projects like ASE [234], which aim at extending existing packages by a common python front end, have emerged. Other packages like HORTON [235], pyscf [236], pyQuante [237] and GPAW [38, 39] are written almost exclusively in python and only employ low-level C or C++ code for the computational hot spots to various extents. Starting from the opposite direction Psi4 [238] has gradually introduced a more and more powerful python interface on top of their existing C++ core over the years.

The popularity of the combination of FORTRAN or a C-like language in the core and python as the high-level interface language can be understood by considering the recent publication of Sun et al. [236] about the pyscf package. They rationalise the choice of python as follows:

- There is no need to learn a particular domain-specific input format.
- All language elements from python are immediately available to e.g. automatise repetitive calculations with loops or similar.
- The code is easily extensible beyond what is available inside pyscf, for example to facilitate plotting or other kinds of analysis.
- Computations can be done interactively, which is helpful for testing or debugging.

Additionally one should mention, that python as a high-productivity, multi-paradigm language often permits to achieve even complicated tasks with few lines of code, whilst still staying surprisingly easy to read. In the context of quantum chemistry this has the pleasant side effect that a python script used for performing calculations and subsequent analysis is typically not overly lengthy, but still documents the exact procedure which is followed in a readable manner. All this comes at pretty much no downside if python is combined with carefully optimised low-level C or FORTRAN code in the computational hot spots. Sun et al. [236] for example claim that pyscf is as fast as any other existing quantum chemistry packages written solely in C or FORTRAN.

Another common feature between pyscf and Psi4 is their modular design inside the package. They vividly facilitate well-established open standards like HDF5 [239] or numpy arrays [240] for data exchange, such that linking their codes to external projects is easily feasible. Psi4 for example managed to integrate more than 15 external packages into their framework. This includes three completely different back ends for computing integrals. In the case of pyscf it only took us about a day to link our molsturm to the FCI algorithms of pyscf via an interface based on numpy. Nevertheless the numerical requirements of Gaussian-type orbitals are currently hard-coded inside the optimised C or C++ parts of both these projects, such that extending them by other types of basis functions could still be involved.

With respect to supporting a large range of basis function types, especially the

quantum Monte Carlo packages CASINO [241] and QMCPACK [242] should be mentioned. They both allow to start a quantum Monte Carlo calculation from discretisations of the trial wave function in terms of cGTOs, STOs, plane-waves or numerical orbitals like splines. Similarly the packages `CP2K` [243], `ASE` [234] and `GPAW` can be employed to perform and post-process computations using more than one type of basis function. Both `GPAW` and `CP2K` even allow to perform calculations employing hybrid basis sets with a mixture of Gaussian-type orbitals and plane waves. To the best of our knowledge, the design of these packages is, however, very specific to the particular combinations of basis function type and method they support.

7.2 Design of the `molsturm` package

As mentioned above the main target for the current design of `molsturm` is to yield a framework, which supports notions towards novel quantum-chemical methods, including methods where unusual discretisation approaches or new types of basis functions are employed. Ideally, adding new basis functions becomes kind of plug-and-play, such that one only implements a minimal interface linking an integral library and the rest of `molsturm`. Thereafter the SCF and the Post-HF methods would just work. Many details regarding the numerical and algorithmic treatment will most likely still need to be optimised thereafter, such that high-level access to influence all kinds of parameters of the SCF scheme or the diagonalisation algorithms are absolutely key. Additionally such a new method would need to be tested and evaluated towards their overall usefulness in standard problems of quantum-chemical modelling. These tasks are usually both highly repetitive and again require access into many layers of a quantum-chemistry program to obtain the quantities to compare. This motivates the following overall design goals for the `molsturm` project.

Enable rapid development: In the early stages of developing new quantum-chemical algorithms, it is often not clear how well these algorithms perform or if they even meet the expected requirements. In other words before worrying about making an algorithm fast, one first wants to know whether it even works. A light-weight framework, which possesses the flexibility to quickly combine or amend what is already implemented is very important for this. The syntax of the resulting code should be high-level and intuitive, resembling the physical formulae as much as possible. To make the initial implementation easy for people not entirely familiar with all tricks of the trade, the details regarding numerics and linear algebra should be mostly hidden in the code. Especially in highly interdisciplinary subjects such as quantum chemistry, too often PhD students are rather unfamiliar to coding and numerics and thus spend half a year to understand the clunky programs, a year to implement the method, just to find that it did not quite work that well. Still influencing the algorithmic details will in many cases be required at a later step. Ideally this can be done by the means of changing mere parameters directly from the user interface and without changing the code very much, such that it stays nice and clean for the next feature to be implemented. A careful reader should have noticed that the `lazyten` lazy matrix library, described in the previous chapter, covers many of the aspects mentioned here. This is of course no surprise and explains why `lazyten` has become one of the key ingredients to `molsturm`. For some more details see 7.2.2 on page 159.

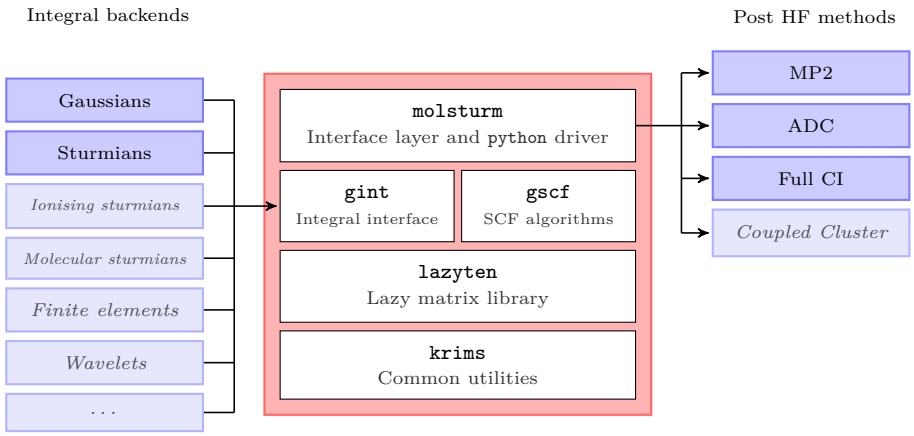


Figure 7.1: Structure of the `molsturm` package. Shown are the five major modules of the package, along with the set of integrals accessible from `gint` and the set of post-HF method, which can be used together with `molsturm`. Only the modules inside the red box are part of `molsturm`. The blue boxes are all external packages. The greyed-out boxes are planned, but not yet implemented.

Modular design with low code complexity: The aspired flexibility of `molsturm` as well as the intended strong separation between high-level code describing algorithms and low-level code dealing with the numerics, necessarily calls for proper modularisation. Even though our current design takes our experiences with many types of basis functions into account, it is very likely that we missed certain aspects, which will make major restructuring unavoidable in the future. To proactively account for this `molsturm` consists of five small modules, which are designed as layers, see figure 7.1. The top layer, "`molsturm`", defines the application programming interface (API), by which other programs may control the flow of `molsturm` or exchange data. Unlike the other layers, which are mostly implemented in `C++`, the `molsturm` layer is mainly `python`. `gint`, the *general integral library*, provides a single link to multiplex between the supported integral calculation back ends. `gscf` implements a couple of contraction-based SCF schemes following the general two-step Fock-update, coefficient/density-matrix-update structure, we mentioned in section 5.5. `gscf` is written on top of `lazyten`, both to make use of its linear algebra abstraction as well as the generality of the lazy matrix formalism to work on dense, sparse and contraction-based Fock matrices under the same syntax. Finally `krims` is the library for common utility *Krims*krams[1]. In our design we made sure that dependencies are only downwards, never sideways or upwards, to make it easy to replace libraries at a later point.

Plug-and-play implementation of new discretisations: When attempting to implement a new basis function type or an unusual discretisation technique in existing quantum-chemistry packages, there is one rather significant obstacle: Implicit assumptions about the numerical properties of the employed basis are scattered around the sometimes rather large codes. Using the lazy matrix language of `lazyten`, we have achieved to centralise the basis function specifics as much as possible at a single place,

[1] German for "odds and ends"

namely the implementation of the contraction expressions of the integral lazy matrices. This takes place in our integral interface `gint`, where all properties of the basis function type as well as the precise back end implementation regarding symmetries, selection rules, sparsity properties or evaluation schemes are known and can be fully exploited. In line with the final example presented in section 6.3.1 on page 149, the SCF and post-HF methods only need to care about the integral lazy matrix objects, being completely independent from the precise nature of the contraction expression. The result of these efforts is that switching from one implemented basis function type to another can be achieved by merely passing the corresponding string parameter. All that effectively changes is the collection of lazy matrices, which is exposed to the SCF and all methods building on top of the SCF results. Trying a new basis function type or a new computational back end for the integral values thus becomes really plug and play: One implements the respective collection of lazy matrices and selects it using the appropriate parameter. For more details regarding this aspect see section 7.2.1.

Easy interfacing with existing code: The evaluation and assessment of new quantum-chemical methods necessarily implies that one needs to test their performance towards standard problems of quantum chemistry. This is especially true for new discretisation methods. Implementing everything from scratch, however, is a rather daunting task. For this reason it is explicitly *not* our goal to create yet another fully-fledged quantum chemistry package as well as the surrounding ecosystem around it. Instead `molsturm` is designed as a small package, where both the full package as well as the individual modules can be readily incorporated into other infrastructures or used on their own for teaching and experimentation. Overall our goal here is not to force a particular "`molsturm`-way" upon our users, much rather provide well-documented and open interfaces, with which it is easy to for them to interact with `molsturm` exactly how it fits there workflow best. In this notion we want our `molsturm` interface to be flexible enough such that interacting with third-party packages can be easily achieved, thus building on the hundreds of man-years, which went into the development of already existing quantum-chemistry codes, and extend `molsturm` beyond the scope originally intended. For details see section 7.2.2 on page 159 as well as the examples in section 7.3 on page 161.

The following sections discuss some of the aforementioned design aspects in more detail.

7.2.1 Self-consistent field methods and integral interface

In chapter 5 on page 85 we looked at various ways to solve the HF equations, both with respect to different types of basis functions for the discretisation as well as different SCF algorithms. One conclusion was that all SCF schemes can be condensed into a similar structure, namely a two-step process, where a Fock-update step and a coefficient-update or density-matrix-update step repeat each other until convergence. We already saw in the last example of section 6.3.1 on page 149, that `lazyten` is well-suited to support this. In the example the Fock-update step could be expressed by a call to the update-function of the exchange and Coulomb lazy matrices and the coefficient-update was a diagonalisation using `eigensystem_hermitian`. For more complicated SCF schemes, where the coefficient-update is more involved, a contraction-based ansatz still allows to express this latter step only in terms of calls to the contraction expression. Taking this

idea one step further the SCF schemes of `gscf` do not need to see the individual terms of the Fock matrix, but access to the update function and contraction expression of a lazy matrix object representing the complete Fock matrix is sufficient. This makes the algorithms of `gscf` self-contained and applicable to *any* non-linear eigenproblem with a structure similar to the HF minimisation problem (4.40). This is rather desirable, because there are plenty of methods in electronic structure theory, which can be thought of as modifications of the HF problem. Examples include the Kohn-Sham matrix (see section 4.6 on page 81) arising in standard treatments of density-functional theory (DFT) or additional terms in the problem matrix arising from a modelling of an external field, a density embedding or a polarisable embedding.

Overall the precise self-consistency problem to be solved by `gscf` is thus defined by the lazy Fock matrix object, which is passed downwards from the upper layer `molsturm`. For building this lazy matrix `molsturm` inspects the method, which is selected by the user, and appropriately combines the integral lazy matrices representing the required one-electron and two-electron terms, i.e. kinetic, nuclear attraction, Coulomb and exchange for HF. Appropriately one would take an exchange-correlation term for DFT and additional terms for embedding theories. Both are not currently available, however.

The integral lazy matrices are obtained from `gint`, which acts as broker presenting a common interface for all basis function types and third-party integral back end libraries towards the rest of the `molsturm` ecosystem. On calculation start `molsturm` will take the discretisation parameters supplied by the user and hand them over to `gint`, which — based on these parameters — sets up the selected integral back end library and returns a collection of lazy matrix integral objects. For each basis type and back end the interface of the returned objects will thus look alike, since they are all lazy matrices. On call to their respective contraction expressions, however, the required computations will be invoked in the previously selected integral back end. In other words `gint` itself does not implement any routine for computing integral values at all, but it just transparently redirects the requests. Notice, that the precise kind of parameters needed by `gint` to setup the back end depends very much on the selected basis function type. For example a Coulomb-Sturmian basis requires the Sturmian exponent k_{exp} and the selection of (n, l, m) triples of the basis functions whereas a contracted Gaussian basis requires the list of angular momentum, exponents and contraction coefficients.

Right now contracted Gaussian and Coulomb-Sturmians are in fact the only basis function types supported for discretisation in `gint`. For each of these at least two different implementations is available, however, and adding more back end libraries or basis function types is rather easy. Essentially one only needs to implement a collection of lazy matrices, where the contraction expressions initiates the appropriate integral computations in the back end. This collection then needs to be registered as a valid basis type to `gint` to make it available to the upper layers. Adding preliminary support for the contracted Gaussian library `libcint`, for example, was added in just two days of work. Notice, that the design of `gint` would even allow all of this to be achieved without changing a single line of code inside `gint` itself, since the call to the registration function could happen dynamically at runtime. So one could implement a new integral back end in a separate shared library and add it as needed in a plug-in fashion without recompiling `molsturm`.

To summarise, the responsibility for the HF problem has thus been effectively split between three different, well-abstracted modules by the means of the lazy matrices

of lazyten. gint provides the interface to the integrals, depending on the supplied discretisation parameters, molsturm builds the Fock matrix expression according to the method selected by the user and hands it over to gscf to get the SCF problem solved.

7.2.2 python interface

The topmost layer of the molsturm quantum-chemistry package is our interface layer. It provides helper functionality to define the calculation parameters like the chemical system or the basis type and basis set choices and finally receives those from the user to setup the calculation. That is, it obtains the integral lazy matrices from gint, builds the Fock matrix as described above and starts the actual SCF calculation. The converged results from gscf are returned in a convenient data structure afterwards.

We chose to implement most of this interface layer and especially the interface itself in the scripting language python. Our reasoning for this is similar to the arguments of the pyscf authors already outlined in section 7.1. By providing the required functionality to setup and drive a molsturm calculation from python, we avoid to define yet another "input format" and "output format" for quantum-chemical calculations. Instead, calculations can be initiated cleanly and flexibly directly from a host script. This can afterwards be used to post-process the results as well, such that any kind of output parsing is not needed. The whole calculation can be performed in a single script from setup to analysis, which makes the complete procedure much more transparent. Moreover, rather repetitive processes like benchmarking a new method can be easily automated in this way.

To give the user full control over the complete molsturm ecosystem *all* parameters for gint, the SCF algorithms of gscf as well as the employed linear solvers from lazyten are forwarded to the python interface of molsturm, where they can all be directly accessed and changed. These parameters include ways to influence which algorithms are chosen by lazyten for diagonalisation or how gscf switches between the implemented SCF solvers. For returning the SCF results back to the host environment molsturm heavily relies on numpy arrays, which have become the *de facto* standard for storing and manipulating matrices or tensors in python. A large range of standard python packages, which are commonly used for plotting or data analysis, like matplotlib [244], scipy [240, 245], or pandas [246], similarly employ numpy arrays as their primary interface. Additionally by the means of interface generators like SWIG [247] numpy arrays can be automatically converted to plain C arrays for calling more low-level C++, C or FORTRAN code as well.

A python interface comes in handy in the context of method development as well. For example we explicitly support running molsturm from an interactive IPython [248] shell, such that one can immediately interfere with the progress of a calculation, check assertions about intermediate results or visualise such graphically with matplotlib [244] This greatly reduces the feedback loop for small calculations, e.g. during debugging. If one uses molsturm from a Jupyter notebook [249], one can even perform calculations and view plots interactively from within a web browser.

For larger calculations molsturm is able to archive the complete calculation, including *all* SCF results in the widely adopted YAML [250] or HDF5 [239] formats. In this way large calculations can be performed in advance on a cluster or bigger computer, then archived and transferred to a workstation machine for analysis. This can be again done in an interactive shell with full access to the state of the calculation as if it would have been

performed locally. Additionally this allows to restore an archived state and continue in a different direction at a later point building on results obtained previously and without redoing everything from scratch. On top of that, `molsturm`'s archived state contains the precise set[2] of input parameters which were used to obtain the stored results. This not only makes the archive self-documenting, but additionally restarting the identical calculation or a slightly amended calculation becomes very easy.

Our aforementioned `numpy` interface has already proven to be very helpful to link to other third-party quantum chemistry codes to `molsturm`. For example `molsturm`'s SCF can be used to start a full configuration interaction (FCI) calculation in `pyscf` [236]. Recently support for computing excited state energies at ADC(2), ADC(2)-x and ADC(3) [118, 119] level with `adcman` [210] was added. Both these links make use of the `python-numpy` interfaces these third-party packages already offer and were realised in only a couple of days. Notice that these interfaces are general enough to work both for CS and cGTO discretisations and theoretically all basis function types which are implemented in `gint`.

The aforementioned aspects will be demonstrated in the context of practical examples in section 7.3 on the facing page.

7.2.3 Test suite

A very important subsidiary to a good software design is a good testing framework. A test suite which is simple to execute, fast and easy to expand not only allows to verify that the current status of a piece of software is correct, but it also allows to verify that all future changes do not break anything. This naturally includes a potential adoption of the design towards future requirements. A good test suite generally aids with any code refactoring, since all changes can be performed in a sequence of many small steps, verifying the correctness of the software on the way.

For this reason `molsturm` comes with an extensive test framework with roughly four types of tests. Firstly there are **unit tests**, which test the functionality of a single function or code unit in a couple of hard-coded examples. Further we have **functionality tests**, which test a larger portion of code and are meant to ensure that the results of `molsturm` agree between different versions. Thirdly the **reference tests** compare the results of `molsturm` to other quantum-chemistry packages. Especially in `lazyten` we furthermore make use of yet another type of tests, namely so-called **property-based tests**. This testing technique uses the expected properties of a code unit to randomly generate test cases and to verify the result as well. On failure the generated test cases are simplified until the most simple, failing test case is found. This is very helpful to reduce the human aspect in testing and finding the actual issue during debugging.

Next to combining various different types of tests, it is very important that running the tests by itself is hassle-free. Only in this way they are actually used. In `molsturm` we therefore employ the **continuous integration** service offered free of charge by Travis CI GmbH for open source projects: Whenever a new commit is made to our github repository, a set of virtual machines start automatically in order to checkout and build `molsturm` completely from scratch in a few typical build configurations. Afterwards the test suite is automatically executed and all output generated by the test suite displayed.

[2] That is *not* the parameters provided by the user, but the post-processed parameters which were actually used by the lower layers, amended, for example, by default values.

In this way even people, who are unfamiliar with all details of `molsturm` can test their changes thoroughly, which encourages code contributions. Additionally this gives us `molsturm` developers the chance to easily make sure that no untested code enters the stable source branch, since only if the continuous integration testing passes, a commit to the stable branch is allowed.

One might argue that such a continuous integration system only achieves its purpose if users committing new code furthermore write the accompanying tests to check its validity. For this reason we try to make it simple to add new tests by providing tooling to automate this process as well. For example in order to add a new reference test for a new feature or a corner case, where a bug was discovered, one only needs to add a small configuration file to a special directory. Afterwards one only needs to call a `python` script, which picks up the configuration file, reads the desired test case and calls the external third-party reference quantum-chemistry program (currently ORCA [130]) to compute the required data. From the next time the `molsturm` test suite is executed, this additional reference test will be part of the test suite as well.

Another way we employ to make sure that most of `molsturm`'s code is indeed tested is a measure called **coverage analysis**. Roughly speaking this method inserts special checkpoints during the compilation of a program, which allow to retrace which code paths have been executed and which have not. In combination with our automatic continuous integration builds, we use this to check automatically which parts of the code have been touched when the test suite was executed, i.e. which parts of `molsturm` are covered by the test suite. Ideally one would keep test coverage close to 100%, implying that literally every line of `molsturm` was tested. In practice in most modules of `molsturm` we currently achieve between 80% and 90% coverage. Notice that coverage analysis is more powerful than just detecting untested code paths. For example it allows to find hot spots, i.e. parts of the code executed very often, or dead code, which is never executed. Similar to a pass of the continuous integration builds, we currently enforce that `molsturm`'s coverage may not decrease by more that 0.5% each time code is merged into the stable branch. This makes sure that most of `molsturm`'s code really gets tested each time new features are added.

7.3 Examples

In this section we present a few examples, which demonstrate how the `python` interface of `molsturm` could be combined with existing features of `python` in order to analyse results or to extend the capabilities of `molsturm`. In all cases shown the computations are done using contracted Gaussian basis sets. It should be noted, however, that due to the basis-function-independent nature of `molsturm` the procedures outlined in the scripts could be easily used with other types of basis functions as well.

7.3.1 Fitting a dissociation curve

Many day-to-day tasks in quantum chemistry boil down to performing a multitude of similar calculations, followed by a subsequent graphical analysis by plotting or fitting. Here, we want to consider the computation of a dissociation curve followed by the fit of a Morse potential.

Even though many traditional quantum-chemistry programs have developed function-

```
1  from matplotlib import pyplot as plt
2  import molsturm
3  import molsturm.posthf
4  import numpy as np
5  from scipy.optimize import curve_fit
6
7
8  def compute_curve(atom, basis_args, conv_tol=1e-6,
9                    zrange=(0.65, 7.15),
10                   n_points=30):
11   distances = np.linspace(zrange[1], zrange[0],
12                           n_points)
13   energies = np.empty_like(distances)
14   previous_hf = None
15
16   for i, z in enumerate(distances):
17     system = molsturm.System(
18       atoms=[atom, atom],
19       coords=[(0, 0, 0), (0, 0, z)],
20     )
21
22     # Run a UHF and subsequent UMP2 calculation
23     # using the basis parameters. If a previous
24     # result exists (i.e. this is not the first
25     # HF calculation we do along the curve)
26     # use it as a guess.
27     guess = "random" if not previous_hf \
28       else previous_hf
29     state = molsturm.hartree_fock(
30       system, conv_tol=conv_tol, guess=guess,
31       restricted=False, **basis_args
32     )
33     mp2 = molsturm.posthf.mp2(state)
34     energies[i] = mp2["energy_ground_state"]
35     previous_hf = state
36   return distances, energies
37
38
39 def plot_morse_fit(dist, ene):
40   # First fit Morse potential:
41   def morse(x, de, a, xeq, off):
42     return de * (1 - np.exp(-a * (x - xeq)))**2 + off
43   popt, pcov = curve_fit(morse, dist, ene)
44
45   # Plot data and Morse using 100 sampling points:
46   x = np.linspace(np.min(dist), np.max(dist), 100)
47   plt.plot(dist, ene, "+", label="UMP2")
48   plt.plot(x, morse(x, *popt),
49            label="Morse potential")
50
51   plt.xlabel("Bond distance in Bohr")
52   plt.ylabel("Energy in Hartree")
53   plt.legend()
54
55
56 def main():
57   # Compute the H2 dissociation using a particular
58   # basis type, backend and basis set:
59   basis_args = {
60     "basis_type": "gaussian",
61     "backend": "libint",
62     "basis_set_name": "def2-svp"
63   }
64   distances, energies = compute_curve("H", basis_args)
65
66   plot_morse_fit(distances, energies)
67   plt.show()
68
69
70 if __name__ == "__main__":
71   main()
```

Figure 7.2: Script for computing a H_2 dissociation curve and fitting a Morse potential to it. The decision about the basis function type, the integral back end as well as the basis set are only made in line 64, where the `compute_curve` is called with the chosen set of parameters.

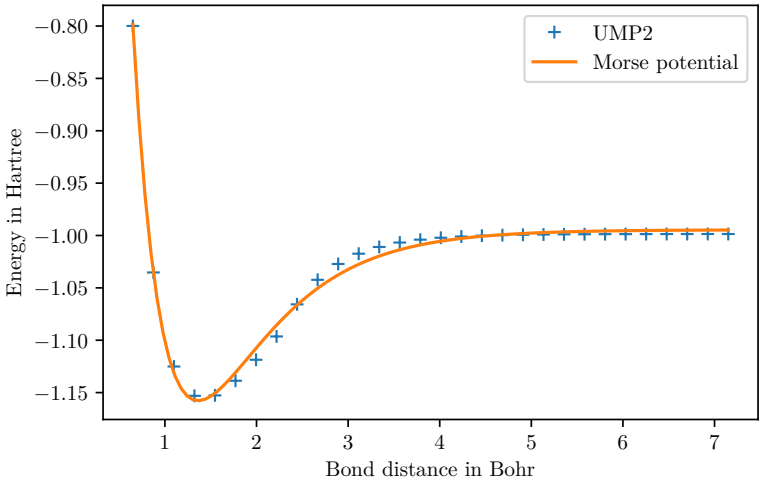


Figure 7.3: Plot resulting from computing the H_2 dissociation curve in a def2-SVP basis set [251] at UMP2 level, employing the script of figure 7.2. Shown are both the UMP2 energy values as well as the fitted Morse potential.

ality for automatising simple energy versus geometry scans, the vast number of possible post-processing methodologies makes it impossible to cover everything. In other words in many cases one is required to write a script to parse the program's output and then feed it to potentially yet another program for doing the fitting and the plotting.

This has the disadvantage that skill in at least two different settings is required: The domain-specific language the quantum-chemistry program uses in its input file as well as the scripting language to parse the results. For people who are new to the field, this can become quite an obstacle. More subtly, the output formats of quantum-chemistry programs change from time to time breaking the parser scripts or — even worse — producing wrong results without any notice. This is a common problem in the practice of computational chemistry.

Contrast this with the approach taken by packages like `molsturm`, which can be controlled solely by a scripting language interface. The `python` script shown in figure 7.2 performs exactly what has been discussed above: First, in the function `compute_curve`, it computes the energy of the H_2 molecule at various bond distances using spin-unrestricted second-order Møller-Plesset perturbation theory (UMP2) [252, 253] Then it calls the function `plot_morse_fit` to plot the resulting data points of energies vs. distances and to fit a Morse potential through them, see figure 7.3. Note how the error-prone parsing step is replaced by just line 34 of figure 7.2, where UMP2 ground-state energy is requested from the dictionary of results returned by the calculation.

Since we are able to orchestrate the full computational procedure from a single script, all parameters influencing the computation, the plotting or the fitting are denoted in a single location. The script therefore serves as automatic documentation of the precise computational procedure. On top of that if our efforts are to be reproduced by someone else, all it takes is to re-run the script.

It should be noted that the present script only makes a single reference to the

parameters used for the selection of the discretisation basis, namely in lines 59–63. By a trivial extension one may hence utilise this script as a building block for a systematic study investigating the effect a change of basis set, integral implementation or basis function type might have on the UMP2 description of the H_2-dissociation. Additionally the basis-function independent design of `molsturm` assures that if a new basis type or a new integral back end becomes available in `gint`, only the appropriate keyword needs to be replaced in lines 59–63 in order to employ it for the UMP2 calculations instead.

7.3.2 Coupled-cluster doubles

In this example we want to show how the high-level `python` interface of `molsturm` may be used in combination with standard functionality from the `python` package `numpy` [240, 245] to quickly extend `molsturm` by novel methods.

Even though `molsturm` right now neither offers any coupled-cluster method nor an interface to any third-party coupled-cluster code, we managed to implement a simple, working coupled-cluster doubles (CCD) (see section 4.5.4 on page 77) algorithm in only about 100 lines of code and about two days of work. The most relevant part of the implementation, namely computing the CCD residual for the current guess of the T_2 amplitudes t_{ij}^{ab}, is shown on the right of figure 7.4, side-by-side with the expression of the CCD residual (compare (4.96)). The full CCD code is available as an example in the file `examples/state_interface/coupled_cluster_doubles.py` of the `molsturm` repository [40]. We follow the standard procedure of employing a quasi-Newton minimisation of the CCD residual with respect to the T_2 amplitudes using the orbital energy differences as an approximate Jacobian [84, 113]. The guess for the T_2 amplitudes is taken from second order Møller-Plesset perturbation theory.

The `python` implementation (right-hand side of figure 7.4) computes the CCD residual following equation (4.96), which was introduced in section 4.5.4. For this it employs the data structures `molsturm` provides in the `state` object, which is returned by the SCF procedure. Since our code uses chemists' indexing convention in the electron-repulsion integrals object `state.eri` and we do not store the antisymmetrised tensor, the first two lines of the code of figure 7.4 need to be executed once to generate the antisymmetrised electron-repulsion integrals $\langle mn||ef\rangle$ in physicists' indexing convention inside the `eri` tensor object. All subsequent lines compute the residual tensor `res` by contracting the relevant blocks of the Fock matrix `state.fock`, the `eri` object and the T_2 amplitudes contained in `t2` and are executed once per CCD iteration. For this the code makes heavy use of the `einsum` method from `numpy`, which performs tensor contractions expressed in the form of Einstein's summation convention. Note, how the interplay of `numpy` with the data structures `molsturm` results in a strikingly close resemblance of implementation and actual equation.

The `state` object provides access to more quantities from the SCF procedure than just the Fock matrix and the repulsion integrals. Individual terms of the Fock matrix or quantities like the overlap matrix in terms of the underlying discretisation basis functions may be obtained as well. We provide this data either as actual `numpy` arrays or by the means of structures, which are based on `numpy` arrays, such that the user can employ the SCF results freely and flexibly within the `python` ecosystem. Coupled with the basis-function independence of `molsturm`'s SCF this allows for rapid development and systematic investigation of Post-HF methods based on arbitrary basis functions.

$$
\begin{aligned}
r_{ij}^{ab} &= \langle ab||ij\rangle \\
&+ \sum_{e} f_{ae}\, t_{ij}^{eb} - \sum_{e} f_{be}\, t_{ij}^{ea} - \sum_{m} f_{mi}\, t_{mj}^{ab} + \sum_{m} f_{mj}\, t_{mi}^{ab} \\
&+ \frac{1}{2}\sum_{mn} \langle mn||ij\rangle\, t_{mn}^{ab} + \frac{1}{2}\sum_{ef} \langle ab||ef\rangle\, t_{ij}^{ef} \\
&+ \sum_{me} \langle mb||ej\rangle\, t_{im}^{ae} - \sum_{me} \langle mb||ei\rangle\, t_{jm}^{ae} \\
&- \sum_{me} \langle ma||ej\rangle\, t_{im}^{be} + \sum_{me} \langle ma||ei\rangle\, t_{jm}^{be} \\
&- \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{mn}^{af}\, t_{ij}^{eb} + \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{mn}^{bf}\, t_{ij}^{ea} \\
&- \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{in}^{ef}\, t_{mj}^{ab} + \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{jn}^{ef}\, t_{mi}^{ab} \\
&+ \frac{1}{4}\sum_{mnef} \langle mn||ef\rangle\, t_{mn}^{ab}\, t_{ij}^{ef} + \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{im}^{ae}\, t_{jn}^{bf} \\
&- \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{jm}^{ae}\, t_{in}^{bf} - \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{im}^{be}\, t_{jn}^{af} \\
&+ \frac{1}{2}\sum_{mnef} \langle mn||ef\rangle\, t_{jm}^{be}\, t_{in}^{af}
\end{aligned}
$$

```
eri_phys = state.eri.transpose((0, 2, 1, 3))
eri = eri_phys - eri_phys.transpose((1, 0, 2, 3))
...

oooo = eri.block("oooo"); vvvv = eri.block("vvvv")
oovv = eri.block("oovv"); ovvo = eri.block("ovvo")
res = (
    + np.einsum("abij->iajb", eri.block("vvoo"))
    + np.einsum("ae,iejb->iajb", state.fock.block("vv"), t2)
    - np.einsum("be,ieja->iajb", state.fock.block("vv"), t2)
    - np.einsum("mi,majb->iajb", state.fock.block("oo"), t2)
    + np.einsum("mj,maib->iajb", state.fock.block("oo"), t2)

    + 0.5 * np.einsum("mnij,manb->iajb", oooo, t2)
    + 0.5 * np.einsum("abef,iejf->iajb", vvvv, t2)
    + np.einsum("mbej,iame->iajb", ovvo, t2)
    - np.einsum("mbei,jame->iajb", ovvo, t2)
    - np.einsum("maej,ibme->iajb", ovvo, t2)
    + np.einsum("maei,jbme->iajb", ovvo, t2)

    - 0.5  * np.einsum("mnef,manf,iejb->iajb", oovv, t2, t2)
    + 0.5  * np.einsum("mnef,mbnf,ieja->iajb", oovv, t2, t2)
    - 0.5  * np.einsum("mnef,ienf,majb->iajb", oovv, t2, t2)
    + 0.5  * np.einsum("mnef,jenf,maib->iajb", oovv, t2, t2)
    + 0.25 * np.einsum("mnef,manb,iejf->iajb", oovv, t2, t2)
    + 0.5  * np.einsum("mnef,iame,jbnf->iajb", oovv, t2, t2)
    - 0.5  * np.einsum("mnef,jame,ibnf->iajb", oovv, t2, t2)
    - 0.5  * np.einsum("mnef,ibme,janf->iajb", oovv, t2, t2)
    + 0.5  * np.einsum("mnef,jbme,ianf->iajb", oovv, t2, t2)
)
```

Figure 7.4: Equation for the coupled-cluster doubles (CCD) residual (4.96) on the left and excerpt of a CCD implementation using `molsturm` and `numpy` on the right. Equivalent quantities are highlighted in the same colour. The first two lines of code show the computation of the antisymmetrised electron-repulsion integrals from the `state.eri` object obtainable from `molsturm`, which is carried out once at the beginning of the algorithm. The remaining lines compute the residual for a particular T_2 amplitude stored in the tensor object `t2`. We follow the same index convention used in section 4.5 on page 73.

```python
import numpy as np
import scipy.optimize
import molsturm


def optimize_h2o(rHO_guess, angHO_guess, conv_tol,
                 **params):
  # Function which computes the cartesian geometry
  # from r and theta.
  def geometry(r, theta):
    rad = theta / 180 * np.pi
    pos = (r * np.cos(rad), r * np.sin(rad), 0)
    return molsturm.System(["O", "H", "H"],
                           [(0, 0, 0), (r, 0, 0), pos])

  def objective_function(args):
    system = geometry(*args)
    ret = molsturm.hartree_fock(
        system, conv_tol=conv_tol/100, **params,
    )
    return ret["energy_ground_state"]

  guess = (rHO_guess, angHO_guess)
  res = scipy.optimize.minimize(
      objective_function, guess, tol=conv_tol,
      method="Powell"
  )
  return res.x[0], res.x[1]

def main():
  r = 2.0         # O-H    radius guess (in au)
  theta = 120     # H-O-H  angle guess

  # First a crude optimisation with sto-3g
  r, theta = optimize_h2o(r, theta, conv_tol=5e-4,
                          basis_type="gaussian",
                          basis_set_name="sto-3g")

  # Then a more fine optimisation with def2-sv(p)
  r, theta = optimize_h2o(r, theta, conv_tol=1e-5,
                          basis_type="gaussian",
                          basis_set_name="def2-sv(p)")
  print("optimal H-O bond length (au): ", r)
  print("optimal H-O-H bond angle:     ", theta)

if __name__ == "__main__":
  main()
```

Figure 7.5: Example `python` script for performing a gradient-free optimisation using Powell's method [255, 256] and `molsturm`.

At the moment we make no efforts to employ symmetry or parallelise the computation of the tensor contractions shown in the script of figure 7.4. For this reason such implementations are all but suitable for real-world applications. Nevertheless, the script presented in figure 7.4 may be used for CCD calculations of small molecules with small basis sets. For example an O_2 6-31G [254] calculation on a recent laptop took about an hour to converge up to a residual l_∞-norm of 10^{-4}. For investigating new methods on top of the `molsturm` framework or to provide a flexible playground for teaching Post-HF methods to students such scripts are therefore still well-suited.

7.3.3 Gradient-free geometry optimisation

In order to make a novel basis function type properly accessible to the full range of quantum-chemical methods a daunting amount of integral routines and computational procedures need to be implemented. For assessing the usefulness of a new discretisation

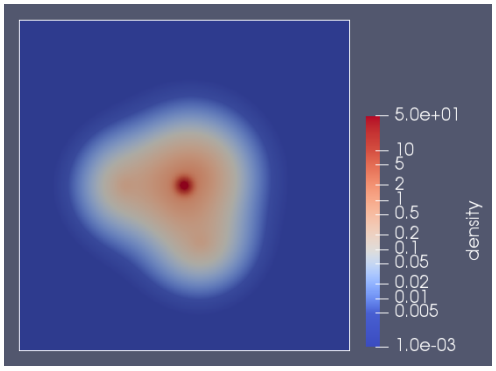


Figure 7.6: Density plot of the final optimised H_2O Hartree-Fock geometry with a O−H bond length of 0.95046 Å and a H−O−H bond angle of 106.35°. A geometry optimisation in ORCA [130] employing the same basis set agrees with this result within the convergence tolerance of 10^{-5}.

method it is, however, important to be able to quickly investigate its performance with respect to as many problems as possible. Undoubtedly a very important application of computational chemistry is structure prediction, i.e. geometry optimisation. For performing such calculations, the implementation of appropriate integral derivatives inside the integral library is required. Since doing so can become as difficult as implementing the integrals required for the SCF scheme itself, one would much rather skip this step and concentrate only on what is required for the SCF at first.

In this example we will demonstrate how the flexible design of `molsturm` enables us to incorporate readily available building blocks of `python` such that a decent gradient-free geometry-optimisation scheme results. This effectively works around the lack of nuclear derivatives on the side of the integral library and allows to perform simple structure optimisations even without nuclear gradients — neither analytical nor numerical.

The script shown in figure 7.5 performs a geometry optimisation of a water based on Powell's gradient-free optimisation algorithm [255, 256] as implemented in the `scipy` library [240, 245]. The optimal structure is found in a two-step procedure. First, a cheap STO-3G [4] basis set is used to obtain a reasonable guess. Then, the final geometry is found by minimising to a lower convergence threshold in the more costly def2-SV(P) [251] basis.

Similar to the CCD example the time required to code the script was rather little, about 30 minutes. Nevertheless the script is able to converge in a couple of minutes to the equilibrium geometry shown in figure 7.6. A novel basis function type, for which one just implemented the SCF integrals in `gint`, can be used with the script of figure 7.5 by only altering the parameters in lines 36 and 41. This makes the script very suitable for giving such basis functions a try in the context of geometry optimisation.

7.4 Current state of molsturm

After about two years of development, molsturm has become a light-weight quantum-chemistry package of about 45000 lines of C++ and python code[3], which tries to explicitly keep the requirements of more than one basis function in mind. The key ingredient to reach the necessary flexibility is a contraction-based self-consistent field scheme, where the details of the Fock matrix contraction expression are varied according to the numerical properties of the basis function type by our integral back end library gint. In contrast the code describing the SCF algorithm is well-separated from the details of the contraction expression using the high-level lazy matrix language of lazyten. As such it becomes independent from the type of basis function used for the discretisation. Even if changes to the SCF scheme or some back end library are needed in the future our modular design will assure that the other parts of the molsturm package will stay unaffected. Once the SCF orbitals have been obtained, the remainder of a calculation, e.g. a Post-HF method, can be typically be formulated entirely in the SCF orbital basis and without any reference to the underlying basis functions. Because of this molsturm can be thought of as a mediator to produce SCF results in a very general fashion on top of which one may stick any Post-HF method.

In molsturm the SCF procedure can be started from either a core Hamiltonian guess, a completely random guess or any other arbitrary set of initial coefficients supplied by the user via a numpy array. For some cases, e.g. if Coulomb-Sturmians are employed, molsturm offers means to interpolate a guess from the converged state of a previous calculation. During the SCF molsturm automatically switches between the implemented SCF schemes. This is necessary since the plain Roothaan repeated diagonalisation [100] as well as the truncated optimal damping algorithm schemes are cheaper, but typically do not converge as efficiently as a coefficient-based Pulay DIIS. Switching the algorithms allows us to balance this. As demonstrated in the examples in section 7.3, the algorithmic details of the SCF procedure can be fully controlled from python, such that one often only needs to change a single keyword in order to switch to a different solver algorithm or to employ a different basis function type.

Once an SCF computation has finished the obtained results can be archived in either in YAML [250] plain text or in HDF5 [239] binary files. Such an archive not only contains the full final state of the calculation but also the precise set parameters which were used to start the SCF. A file therefore becomes self-explanatory and reproducible without any special measures taken from the user. On top of that an archived calculation can be continued or analysed at a later point or on a different machine with ease by just restoring the state.

On top the SCF a range of interfaces for performing Post-HF calculations or further analysis are available. Right now molsturm only implements second order Møller-Plesset perturbation theory (see section 4.5.3) and some utility functions for plotting. Some selected methods from third-party libraries can be easily invoked on any SCF result using third party libraries. For example full configuration interaction is available via pyscf [236] and the excited states methods ADC(1), ADC(2), ADC(2)-x [118] and ADC(3) [119] based on the algebraic diagrammatic construction scheme are available via adcman [210]. Calculations in molsturm may be performed based on contracted Gaussians [4] — using the integral libraries libint [257, 258] or libcint [259] — and

[3]The number includes the code from the dependencies gint, gscf, lazyten, krims as well as all examples and tests, but excludes comment lines and blanks.

based on Coulomb-Sturmians [24, 30] — using `sturmint` [170]. Implementing further types of basis functions takes nothing more than providing appropriate interface classes in our integral interface library `gint`. Thereafter such basis functions are available for the full `molsturm` ecosystem including the Post-HF methods provided by the third-party libraries mentioned above.

In section 7.3 the abilities of `molsturm` have been demonstrated by three practical examples. We put particular emphasis on how our `python` interface integrates with existing third-party `python` packages such that additional functionality can be quickly combined with `molsturm`, potentially to extend `molsturm` in ways we as the authors would have never thought of. In the examples it was further shown how to aid repetitive calculations, implement novel quantum-chemical methods or rapidly amend functionality in a preliminary way, where a proper implementation would be much more involved. We hinted how systematic comparisons with established basis functions as well as subsequent graphical analysis is convenient to perform by the means of our readily scriptable interface. We hope that in this manner `molsturm` will be a useful package to rapidly try novel basis function types and get a feeling for their range of applicability.

Chapter 8

Coulomb-Sturmian-based quantum chemistry

> *The real problem is that programmers have spent far too much time worrying about efficiency in the wrong places and at the wrong times; premature optimization is the root of all evil (or at least most of it) in programming.*
>
> — Donald Knuth (1928–present)

In this chapter we will discuss preliminary computational results obtained from quantum-chemical calculations using Coulomb-Sturmian basis functions inside the `molsturm` framework. The focus will be on discussing and understanding the convergence properties of CS-based calculations on atoms of the second and the third period of the periodic table, mostly at Hartree-Fock level. Some exemplary FCI and MP2 calculations have been performed to get an idea how the picture changes if correlation effects are taken into account as well.

Based on these results we will discuss some preliminary guidelines for selecting CS basis set parameters, like the angular momentum restrictions or the Sturmian exponent k_{exp}, where the overall aim is to yield rapid convergence of the ground state energies at HF or correlated level. Finally we present the first results of a CS-based excited states calculation at ADC(2) level.

For the calculations in this chapter we employed `molsturm` (version 0.0.3) [40] as well as its interfaces to `pyscf` (version 1.4.0) [236] and `adcman` (version 2.5-core_valence) [210].

8.1 Denoting Coulomb-Sturmian basis sets

In section 5.3.6 on page 115 we denoted a Coulomb-Sturmian basis function as the product

$$\varphi_{nlm}(\underline{\boldsymbol{r}}) = R_{nl}(r) Y_l^m(\hat{\underline{\boldsymbol{r}}}) \tag{8.1}$$

of radial part

$$R_{nl}(r) = N_{nl} \left(2k_{\text{exp}} r\right)^l e^{-k_{\text{exp}} r} \, {}_1F_1 \left(l + 1 - n | 2l + 2 | 2k_{\text{exp}} r\right) \tag{8.2}$$

and spherical harmonic, compare equation (5.32). It is uniquely defined by specifying both the CS exponent k_{exp} as well as the quantum number triple $(n, l, m) \in \mathcal{I}_F$, where $\mathcal{I}_F$ is the index set defined in (5.35). Since all basis functions share the same exponent k_{exp} a truncated CS basis is thus uniquely defined by specifying the common exponent k_{exp} as well as the set of all (n, l, m) triples of all basis functions $\varphi_{nlm}^{\text{CS}}$.

Theoretically any selection of triples (n, l, m) can be used to form a CS basis. From the similarity of the CS functions to the hydrogen-like orbital functions as well as the shape of the orbitals of other atoms one would, however, expect Coulomb-Sturmians with smaller values of n to be the most important. In this work we have therefore restricted ourselves to CS basis sets of the form

$$\left\{ \varphi_{nlm} \,\middle|\, (n, l, m) \in \mathcal{I}_F,\ n \leq n_{\text{max}},\ l \leq l_{\text{max}},\ -m_{\text{max}} \leq m \leq m_{\text{max}} \right\},$$

i.e. where all three quantum numbers are bound from above. We will sometimes refer to such a CS basis set by the triple $(n_{\text{max}}, l_{\text{max}}, m_{\text{max}})$ of the three maximal quantum numbers itself. In other words a $(3, 2, 2)$-basis set shall denote a basis set with $n_{\text{max}} = 3$, $l_{\text{max}} = 2$ and $m_{\text{max}} = 2$. One should mention that a restriction to basis sets of this form is entirely arbitrary and mainly done for the sake of reducing the search space at hand for an initial investigation.

In existing literature about Coulomb-Sturmians the existing terminology to denote atomic orbitals as well as sets of atomic orbitals is often carried forward to the CS context as well. For examples the spectroscopic terms $1s, 2s, 2p_{-1}$ and so on are often used to denote the Coulomb-Sturmian functions φ_{100}, φ_{200}, $\varphi_{2,1,-1}$.

8.2 Convergence at Hartree-Fock level

In chapter 4 on page 43 we mentioned that Hartree-Fock is typically the first step for a quantum-chemical simulation with many accurate Post-HF methods building on top of the HF result. Because of this as well as its simplicity it is a very good starting point for our investigation of the convergence of Coulomb-Sturmian-based discretisations in quantum-chemical calculations. To reduce the complexity further, we will not yet consider variations of the CS exponent k_{exp} in this as well as the next few sections. Instead we will only discuss the effect of changing the maximal quantum numbers n_{max}, l_{max} and — to a lesser extend — m_{max}. The reason for this is twofold. First of all already our initial discussion about the relative error and local energies of CS discretisations in section 5.3.6 on page 115 showed that the effect of varying the maximal quantum numbers is much more pronounced compared to changing k_{exp}. Secondly the completeness property of the Coulomb-Sturmians is satisfied regardless of the value of k_{exp} and thus any error resulting from a less ideal value of k_{exp} can be corrected with larger basis sets. Notice,

however, that the rate of convergence with increasing the basis size does well depend on the choice of k_{exp} as previous work suggests [32] and our discussion in section 8.4 confirms. In other words a sensible value for k_{exp} does need to be chosen for our analysis nonetheless.

The net effect of tuning the maximal quantum numbers $n_{\max}$, $l_{\max}$ and $m_{\max}$ is that one effectively selects which part of the set of all radial functions $\{R_{nl}\}_{nl}$ and which part of the set of all angular functions $\{Y_l^m\}_{lm}$ is available for modelling the wave function. The completeness property of the Coulomb-Sturmians implies that both $\{R_{nl}\}_{nl}$ as well as $\{Y_l^m\}_{lm}$ are complete bases. Whilst the latter is furthermore a well-known property of the spherical harmonics [138], the former is also apparent from the connection of the CS radial equation to Sturm-Liouville theory (see section 5.3.6 on page 115), where one key result is that the eigenfunctions of a Sturm-Liouville differential equation form a complete basis for a weighted L^2-space [29]. Since the angular momentum quantum number l is a parameter in the CS radial equation (5.33), not only the set $\{R_{nl}\}_{nl}$ of *all* radial parts is complete, but also the set $\{R_{nl'}\}_n$, where the angular momentum l' is held fixed. This allows to express each function R_{nl} with $l > 0$ as a linear combination of the functions $\{R_{n',0}\}_{n'}$. A careful analysis using the recurrence relations between the confluent hypergeometric functions shows, that employing those radial parts with $l = 0$ and $n' \leq n$ is sufficient to express R_{nl}. In other words convergence in the radial part can be completely controlled by the range of available principle quantum numbers n, i.e. by tuning $n_{\max}$. Conversely $l_{\max}$ and $m_{\max}$ control the convergence with respect to the angular part in agreement with the physical interpretation given to the quantum numbers l and m.

Related to this aspect is the scaling of CS basis set size with the maximal quantum numbers $n_{\max}$, $l_{\max}$ and $m_{\max}$. For example a CS basis consisting of complete shells with principle quantum numbers up to and including $n_{\max}$ consists of

$$\begin{aligned} N_{\text{bas}}(n_{\max}) &= \sum_{n=1}^{n_{\max}} \sum_{l=0}^{n-1} \sum_{m=-l}^{l} 1 = \sum_{n=1}^{n_{\max}} \sum_{l=0}^{n-1} 2l+1 \\ &= \sum_{n=1}^{n_{\max}} n^2 = \frac{(2n_{\max}+1)(n_{\max}+1)n_{\max}}{6} \in \mathcal{O}(n_{\max}^3), \end{aligned} \tag{8.3}$$

basis functions, i.e. scales cubically with $n_{\max}$. In contrast the size of a basis set, which is limited both by $n_{\max}$ as well as the maximal angular momentum $l_{\max}$ scales as

$$\begin{aligned} N_{\text{bas}}(n_{\max}) &= \sum_{n=1}^{n_{\max}} \sum_{l=0}^{\min(l_{\max},n-1)} 2l+1 \\ &= \sum_{n=1}^{n_{\max}} \Big(\min(l_{\max}+1, n) \Big)^2 \\ &\leq \sum_{n=1}^{n_{\max}} (l_{\max}+1)^2 = (n_{\max}-1)(l_{\max}+1)^2 \in \mathcal{O}(n_{\max} l_{\max}^2). \end{aligned} \tag{8.4}$$

In other words if we manage to find a sensible upper bound for $l_{\max}$, which captures all of the angular part of the HF wave function, we can converge the radial part thereafter by just increasing the basis set size linearly. A key aspect of the next few sections will therefore be to find a suitable upper bound for $l_{\max}$ for a particular chemical system.

system	E_{HF}	system	E_{HF}
Li	$-7.4327376^{a,U}$	Na	$-161.8589459^{a,U}$
Be	$-14.57302317^{b,R}$	Mg	$-199.61463642^{b,R}$
B	$-24.5334831^{a,U}$	Al	$-241.8808503^{c,U}$
C	$-37.6937751^{c,U}$	Si	$-288.8589476^{c,U}$
N	$-54.4046409^{c,U}$	P	$-340.7192829^{c,U}$
O	$-74.8192096^{c,U}$	S	$-397.5133666^{c,U}$
F	$-99.4166858^{c,U}$	Cl	$-459.4899302^{c,U}$
Ne	$-128.54709811^{b,R}$	Ar	$-526.8175128^{b,R}$

^{U} unrestricted HF
^{R} restricted HF
^{a} CBS extrapolation using cc-pVDZ to cc-pv5Z [148, 261]
^{b} Morgon et al. [260]
^{c} CBS extrapolation using cc-pVTZ to cc-pv6Z [148, 152, 261–263]

Table 8.1: Reference values used for comparison of the CS-based results and for estimating errors in the CS values. The CBS extrapolation was done with a builtin routine provided by `molsturm` following Jensen [149].

Notice, that the completeness of the radial part of the CS functions implies that this upper bound for $l_{\max}$ is not specific to CS functions, but can be applied to *any* basis function type, which is of the product form radial part times angular part.

To estimate errors and judge the quality of our CS-based HF results we compare to the reference values given in table 8.1. For the closed-shell atoms we use the very accurate numerical RHF energies obtained by Morgon et al. [260]. For open-shell atoms as well as the other systems we employ the method of Jensen [149] to extrapolate the UHF complete basis set (CBS) limit from UHF calculations using the Dunning cc-pVnZ family of cGTO basis sets.

8.2.1 Basis sets without limiting angular momentum

Without truncating the maximal angular momentum by limiting $l_{\max}$ the CS basis set effectively consists only of full shells of principle quantum numbers n, ranging from 1 to $n_{\max}$. Since the CS functions are complete, increasing $n_{\max}$ is guaranteed to reduce the error due to the Courant-Fischer theorem 3.3 on page 33. Figure 8.1 on the facing page shows this for the atoms of the second period by plotting the absolute error in the HF energy versus the number of basis functions. For each calculation of a particular atom the same value of k_{exp} was used, which was taken to be close to the optimal exponent of this atom at $(6, 5, 5)$ level to exclude any influence on the behaviour originating from a very unsuitable exponent. Whilst we notice a clear convergence with increasing basis set size, it is furthermore visible that the convergence rate drops for larger values of $n_{\max}$.

The question is now whether all employed basis functions are actually required in order to represent the HF wave function properly. From a physical point of view, we would not expect all angular momentum to be equally important for the description of the electronic ground state of a particular atom. In beryllium, for example, only the $1s$ and $2s$ atomic orbitals are occupied, such that we would expect, that only angular momentum up to $l = 0$ is required. In light of our discussion in the previous section, we would therefore propose that a basis with $l_{\max} = 0$ is sufficient to converge the angular

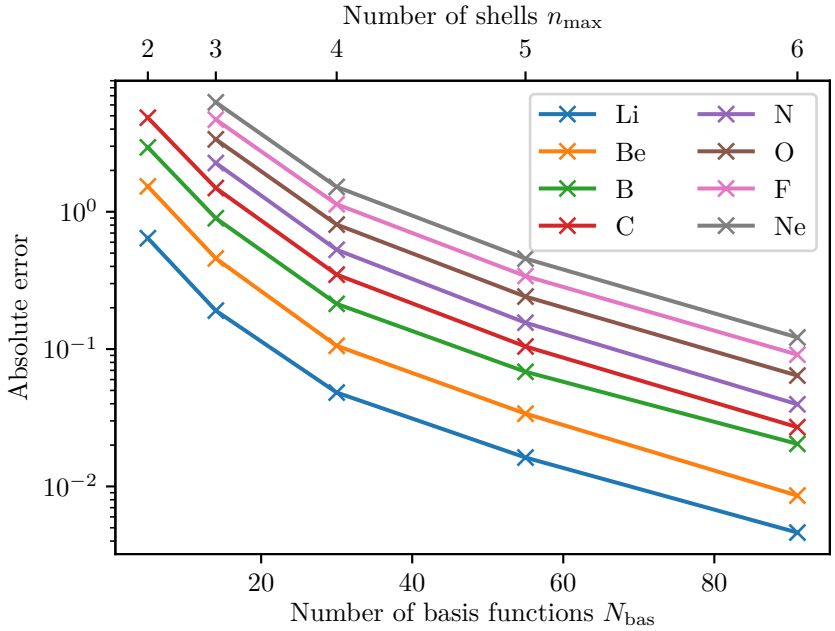


Figure 8.1: Plot of the absolute error in the HF energy versus the number of basis functions in a CS basis containing complete shells up to and including principle quantum number $n_{\max}$. For the closed-shell atoms Be and Ne the restricted HF procedure was used, whereas for the other systems UHF was employed. The errors were computed against the reference values from table 8.1 on the facing page.

part of the beryllium ground state. Conversely we would expect all CS functions with $l > 0$ to contribute only very little to the increase in accuracy as we go to larger basis sets in figure 8.1 on the previous page. To test this hypothesis, let us introduce the **root mean square occupied coefficient** per angular momentum l, formally defined as follows.

Definition 8.1. The root mean square (RMS) occupied coefficient per angular momentum l is the quantity

$$\mathrm{RMSO}_l = \sqrt{\sum_{(n,l,m)\in\mathcal{I}_{\mathrm{bas}}} \sum_{i\in\mathcal{I}^{\alpha}_{\mathrm{occ}}} \frac{1}{N^{\alpha}_{\mathrm{elec}}\, N_{\mathrm{bas},l}} \left(C^{\alpha}_{nlm,i}\right)^2 + \sum_{i\in\mathcal{I}^{\beta}_{\mathrm{occ}}} \frac{1}{N^{\beta}_{\mathrm{elec}}\, N_{\mathrm{bas},l}} \left(C^{\beta}_{nlm,i}\right)^2} \tag{8.5}$$

where $C^{\alpha}_{\mu i}$ and $C^{\beta}_{\mu i}$ are the orbital coefficients of the α and β orbitals (see (4.58)) and

$$N_{\mathrm{bas},l} := \left| \left\{ (n',l',m') \,\middle|\, (n',l',m') \in \mathcal{I}_{\mathrm{bas}} \text{ and } l' = l \right\} \right|$$

is the number of basis functions in the CS basis which has angular momentum quantum number l.

By construction RMSO_l is the RMS-averaged coefficient for a particular angular momentum quantum number l in the occupied SCF orbitals. It therefore provides a measure of which angular momentum quantum numbers l are required in the current basis set for describing a state properly. Conversely values of RMSO_l below the convergence threshold $\varepsilon_{\mathrm{conv}}$ of the SCF procedure indicates that all CS basis functions of this angular momentum value l can be safely removed from the CS basis set without influencing the accuracy of the HF calculation above this level. In many cases this property of RMSO_l can assist in finding a good value of l_{max} for truncating the orbital angular momentum.

For example let us consider figures 8.2 and 8.3 on the facing page, which show the variation of RMSO_l vs. l for the HF ground state for the atoms of the second and third period if a $(6,5,5)$ Coulomb-Sturmian basis is employed. In the plot two kinds of behaviour can be identified. The first kind applies to those atoms which are either closed-shell like Be, Ne, Mg or Ar or which have a half-filled valence sub-shell like Li, N, Na or P. For these a very pronounced drop in RMSO_l occurs once a particular angular momentum value l has been reached. For Li and Be, where only s-functions are occupied in the ground state, this happens from $l = 0$ to $l = 1$ and for the other mentioned atoms from $l = 1$ to $l = 2$, which in both cases is in perfect agreement with the behaviour expected from the physical point of view. For these atoms truncating at $l_{\mathrm{max}} = 0$ or $l_{\mathrm{max}} = 1$, respectively, will not introduce a noticeable error as we will see in the next section. In contrast to this the other atoms B, C, O, F, Al, Si, S and Cl vary in a decreasing staircase pattern. Much rather their RMSO_l value decreases only very moderately over the range of angular momentum quantum numbers.

Figures 8.4, 8.5 on page 178 and 8.6 on page 179 provide a good hint to understand this behaviour. These show the RMS-averaged value of those orbital coefficients that share the same angular momentum quantum number l in the corresponding basis function. For the modelling of the atoms in each case a $(6,5,5)$ Coulomb-Sturmian basis with a near-optimal value of k_{exp} is employed. Whilst for nitrogen (figure 8.4) the $2s$ function mostly has significant coefficient values associated to basis functions with $l = 0$, both for carbon (figure 8.5) as well as oxygen (figure 8.6) the basis functions with $l = 2$ and $l = 4$

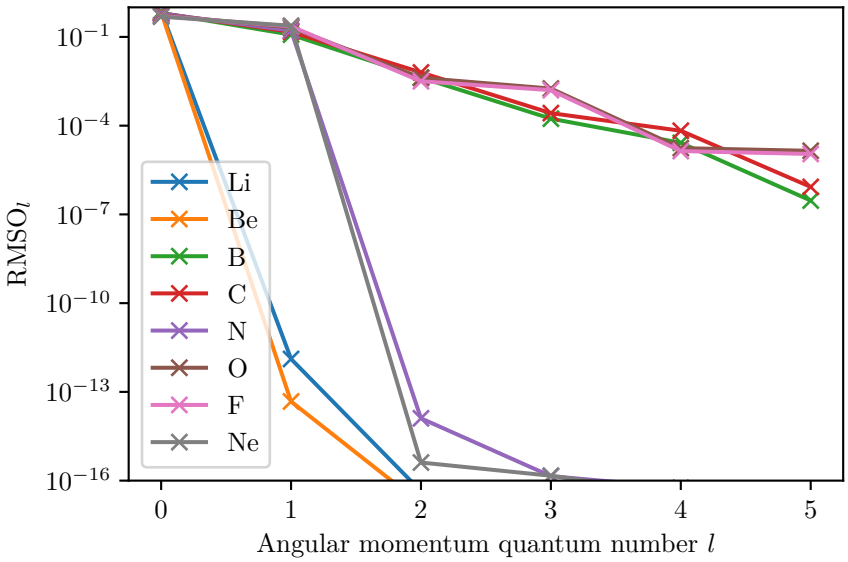


Figure 8.2: Plot RMSO$_l$ vs. l for the HF ground state of the atoms of the second period if a $(6, 5, 5)$ CS basis is employed. In each case k_{exp} was taken close to the optimal value. For Be and Ne a RHF procedure was used, for the other cases UHF.

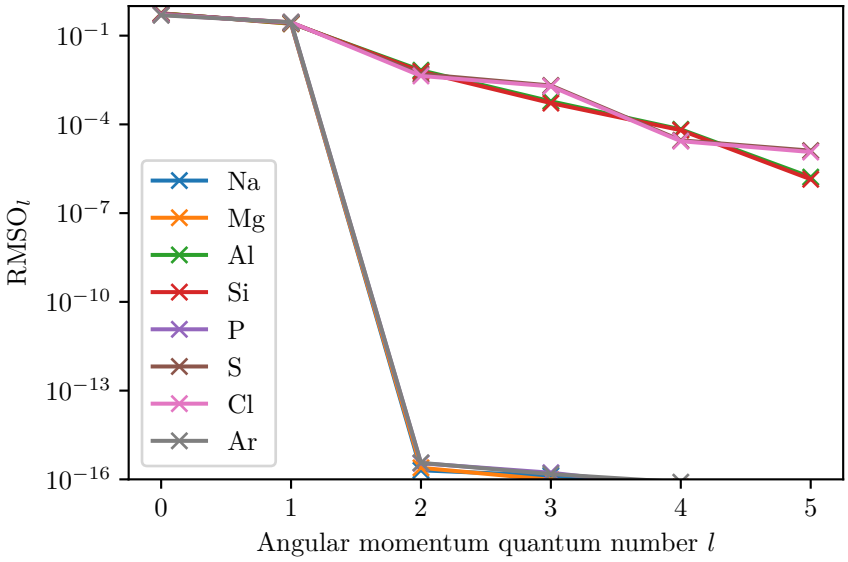


Figure 8.3: Plot RMSO$_l$ vs. l for the HF ground state of the atoms of the third period if a $(6, 5, 5)$ CS basis is employed. In each case k_{exp} was taken close to the optimal value. For Mg and Ar a RHF procedure was used, for the other cases UHF.

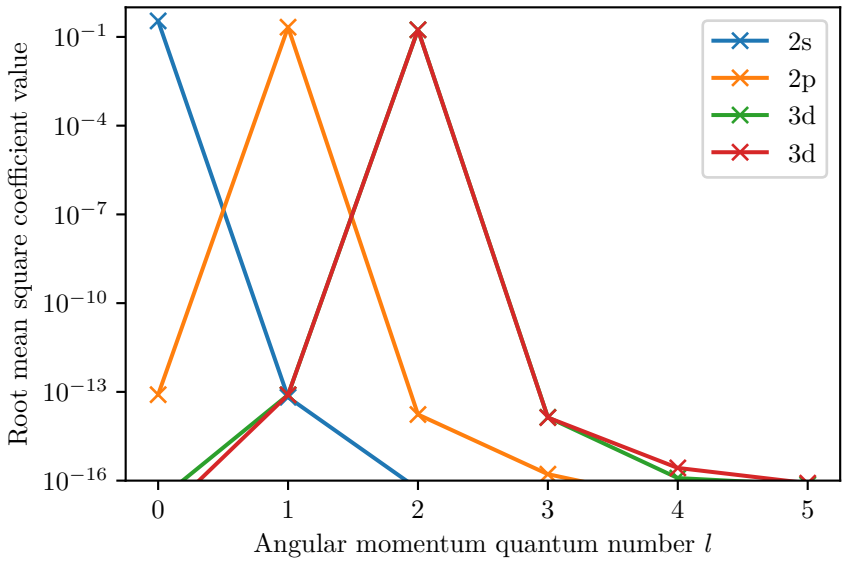


Figure 8.4: Root mean square coefficient value per basis function angular momentum quantum number l for selected orbitals of nitrogen. The atom is modelled in a $(6, 5, 5)$ CS basis using UHF.

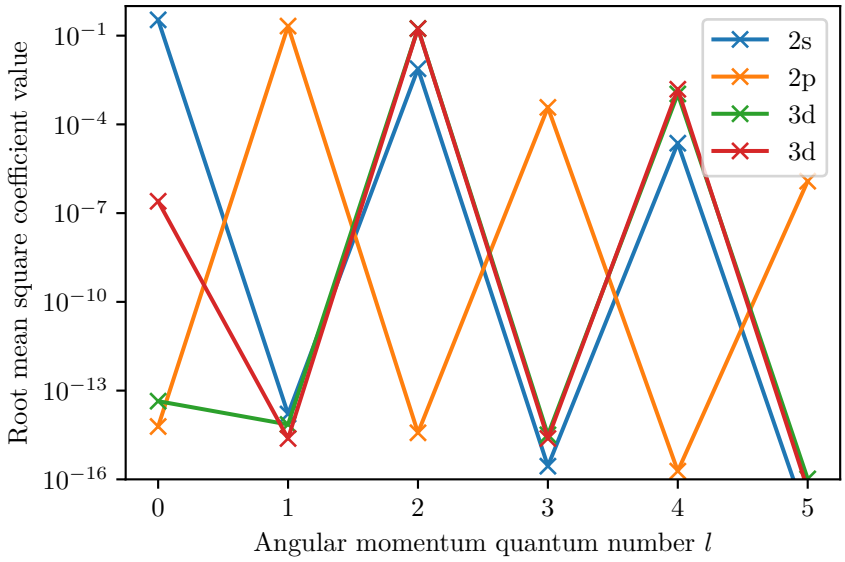


Figure 8.5: Root mean square coefficient value per basis function angular momentum quantum number l for selected orbitals of carbon. The atom is modelled in a $(6, 5, 5)$ CS basis using UHF.

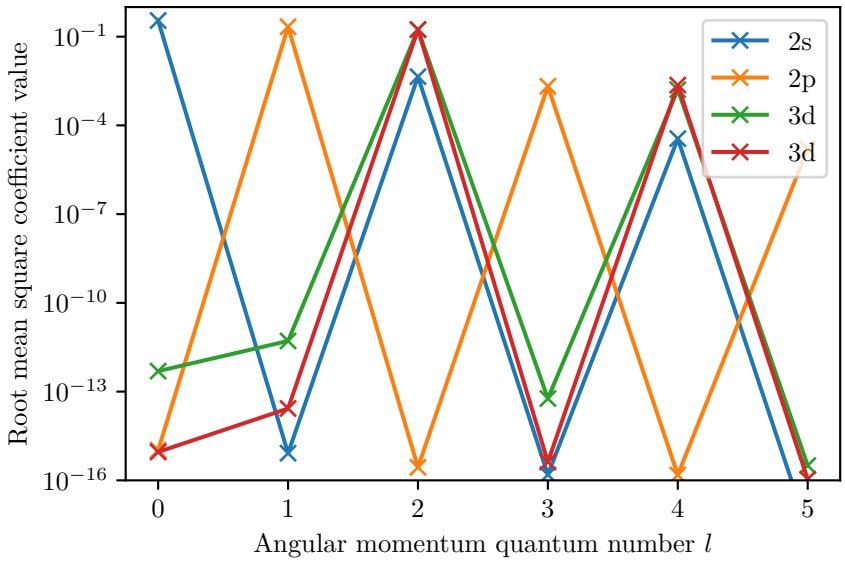


Figure 8.6: Root mean square coefficient value per basis function angular momentum quantum number l for selected orbitals of oxygen. The atom is modelled in a $(6, 5, 5)$ CS basis using UHF.

are important as well. Similar observations can be made for the $2p$ functions, which for C and O have significant coefficients at angular momenta $l = 1, 3, 5$ and the $3d$ functions, which require $l = 2$ and $l = 4$, sometimes even $l = 0$, for a proper description. This explains why RMSO_l plots for carbon and oxygen do not show the expected drop from $l = 1$ to $l = 2$, since the higher angular momenta play a role for the occupied s-type and p-type SCF orbitals as well. Equivalent plots to figures 8.5 and 8.6 for the other atoms, which are not closed-shell or have a half-filled valence shell, show very similar features, which overall explains the slow decrease in the RMSO_l plots for such atoms.

The pending question is now why angular momenta higher than the expected $l = 0$ and $l = 1$ are needed for modelling the s-like and p-like orbitals in some atoms in the first place. Since very similar RMSO_l and RMS orbital coefficient plots are observed if cGTO discretisations are used, see appendix B on page 211, this effect cannot be due to the CS discretisation we employ. Much rather it is an artefact of our UHF treatment of the open-shell systems. For example Cook [264] described a similar behaviour for a UHF modelling of carbon and fluorine based on cGTO discretisations. He noticed that the s-type and p-type SCF orbitals for both these systems were not only composed of cGTO basis functions with $l = 0$ and $l = 1$, but much rather were linear combinations of basis functions with angular momentum quantum numbers in steps of 2 apart. So for s-like SCF orbitals s, d, g, ... basis functions were combined in his calculations — exactly what we observe in figure 8.5. Later it was found that the occurrence of higher angular momenta in the ground state is a general issue of UHF [91, 92, 265]. Fukutome [91] provides a very detailed analysis of the underlying mechanisms including a discussion of the effect of spin symmetry breaking and HF instabilities in UHF and GUHF.

Let us try to understand the occurrence of this behaviour in the context of our

calculations. For the case with an unevenly occupied electronic configuration, like a single or two unpaired electrons, a very naïve guess for UHF, that is without using fractional occupation numbers, is not spherically but *axially* symmetric [92]. This implies that the obtained SCF orbitals no longer represent a spherically symmetric density, but an axially symmetric density instead. This is carried forward over the iterations, such that for the final UHF ground state spherical symmetry is broken. If we were to use fractional occupations, then numerical error, which accumulates over the SCF procedure, could still lead to axially symmetric SCF orbitals. The reason is the same mechanism leading to the Jahn-Teller effect [266], namely that breaking the symmetry allows to change the geometry, such that overall the energy of the occupied SCF orbitals is net lowered, and the energy of the virtual orbitals overall raised. As soon as numerical error amounts to break the spherical symmetry once, the effect will amplify over the UHF iterations and thus lead to the same axially symmetric UHF ground state.

On the level of the SCF orbitals, the broken spherical symmetry allows for CS basis functions of different angular momentum to be combined inside a single orbital, such that these are no longer of pure s, p, d, ... character. Let us give a few examples for this. If a spherically symmetric s orbital is amended with a fraction of d_{z^2}, then this effectively causes a stretching of the orbital along the z axis, which makes it axially symmetric. Similarly the p_x, p_y and p_z orbitals may be amended with f_{xz^2}, f_{yz^2} and f_{z^3} to cause the same stretching along the z-axis in each of these. Even if all p orbitals of this p-shell are evenly occupied in the final HF ground state, the wave function is then axially symmetric.

Since such a linear combination of angular momenta lowers the total SCF energy, the UHF procedure for unevenly occupied electron configurations may well explore these. In contrast, for evenly occupied valence shells, like the half-filled valence shell atoms N and P, such a symmetry breaking does not help to lower the energy since all p orbitals are occupied to the same level, thus it does not occur and the pure angular momentum character of each of the orbitals is kept even in an UHF treatment. In other words the observed slow decrease in the RMSO_l plots in figures 8.2 and 8.3 for the atoms with one or two unpaired electrons is not due to the CS discretisation not being able to represent the system, but much rather due to a known issue of UHF. A treatment of the atoms with ROHF should give more consistent results for all atoms.

Conversely our discussion shows that RMSO_l is a good diagnostic measure for understanding which angular momentum quantum numbers are required for an accurate quantum-chemical modelling. Since its value for a particular quantum number l indicates the RMS-averaged coefficient value it even provides a quantitative measure for the error, which is introduced if the range of available angular momentum in a CS basis set is truncated to angular momenta below this value.

8.2.2 Basis sets with truncated angular momentum

From the discussion of the previous section it becomes clear that at least in some cases it makes sense to limit not only $n_{\max}$, but on top of that $l_{\max}$ as well. For example for beryllium the clear drop in RMSO_l in figure 8.2 suggests that limiting the angular momentum quantum numbers to $l_{\max} = 0$ is reasonable. Similar arguments for N and P suggest taking $l_{\max} = 1$ for these atoms. In other words we would expect the discretisation of the angular part of the HF wave function to be already well-converged for CS discretisations with $l_{\max} = 0$ or $l_{\max} = 1$, respectively. Consequently we will only

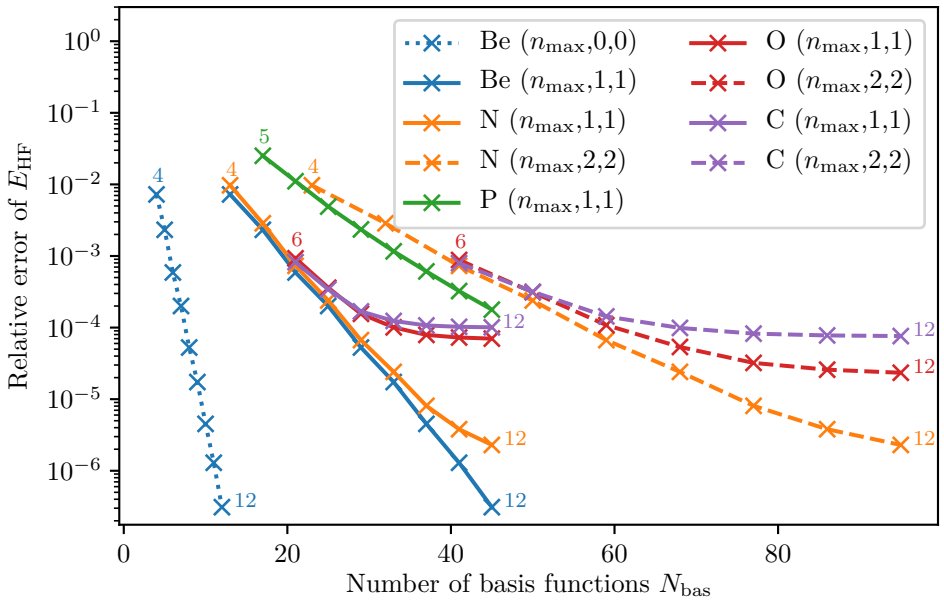


Figure 8.7: Relative error in E_{HF} versus the number of basis functions for selected CS basis sets of the form $(n_{\max}, l_{\max}, l_{\max})$. The connected points show basis set progressions in which the maximum principle quantum number $n_{\max}$ is varied in steps of one and the maximum $l_{\max}$ is fixed. The first and last value for $n_{\max}$ are indicated as small numbers next to the plot. The same line type is used for all progressions of the same $l_{\max}$ and the same colour for all progressions of the same atom.

need to increase $n_{\max}$ further and further in order to converge the remaining radial part of the wave function as well. Since fixing $l_{\max}$ reduces the scaling of the basis set size from cubic in $n_{\max}$ (see (8.3)) to linear (see (8.4)), we would expect to obtain a much faster convergence rate.

To test our hypothesis figure 8.7 shows some example calculations of beryllium, nitrogen, carbon, oxygen and phosphorus using progressions of CS basis sets, where $l_{\max}$ is limited to either 0, 1 or 2, but $n_{\max}$ is ranged between 4 and 12. In each case the relative error of the HF energy with respect to the reference values in table 8.1 are plotted against the size of the CS basis and those error values corresponding to the same atom and the same $l_{\max}$, but different $n_{\max}$, are connected by lines. We will refer to such a sequence of connected error values by the term **progression** in the following. As usual $k_{\exp}$ is fixed to a sensible value for all calculations of the same atom and for beryllium we used RHF, for the other atoms UHF.

Even though the basis sets are now additionally truncated in angular momentum quantum numbers, the HF energies for beryllium still converge steadily. This applies both to the cases $l_{\max} = 0$ as well as $l_{\max} = 1$. Compared to figure 8.1 one notices, however, a massive improvement in convergence rate. For the $l_{\max} = 1$ progressions of nitrogen and phosphorus the same holds true. Choosing $l_{\max} = 2$ for nitrogen does not improve the obtained values very much in accordance with the RMSO_l plot (figure 8.2). Since

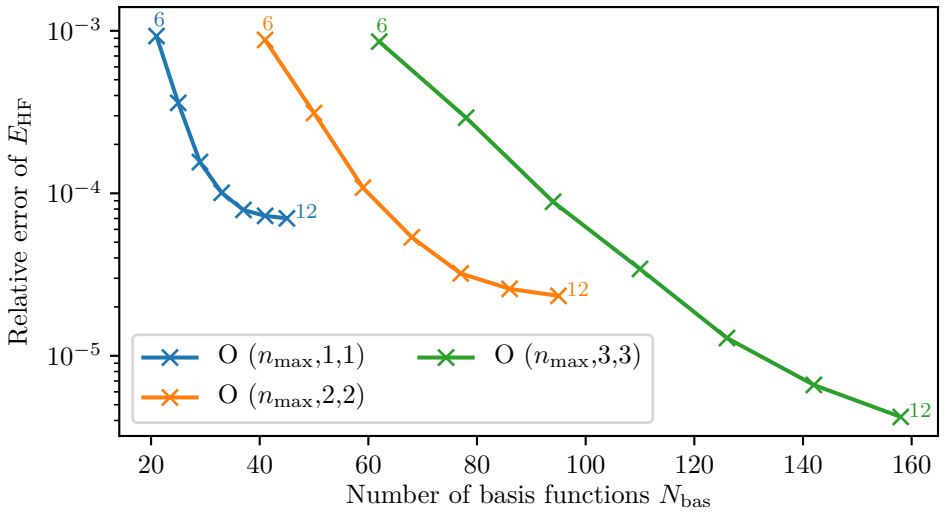


Figure 8.8: Relative error in E_{HF} versus the number of basis functions for oxygen. The plot shows the oxygen progressions of figure 8.7 amended with further progressions using larger basis sets,

the basis now grows faster as $n_{\max}$ increases, the convergence rate is slower, however. For oxygen and carbon the angular momentum values $l > 2$ are important for a proper modelling of the ground state as well. As such it is no surprise that the convergence of the HF energy for these two cases stagnates visibly for the $n_{\max}$-progressions with $l_{\max} = 1$ and $l_{\max} = 2$. Even though the convergence is initially linear as well, the curves bend off at some point. The reason for this is that the truncation of the available set of angular momentum quantum numbers to at most $l_{\max}$ makes some of the true solution of the angular part not accessible to the CS discretisation. At some point the resulting error completely dominates, such that improving the radial part by increasing $n_{\max}$ does not improve the relative error by much any more.

The results of our investigations on the oxygen atom are summarised in figure 8.8, which shows the $(n_{\max}, 1, 1)$ and $(n_{\max}, 2, 2)$ progressions already depicted above as well as one using $(n_{\max}, 3, 3)$ CS basis sets. The effect of truncating the angular momentum is clearly visible. The blue curve with $l_{\max} = 1$ is not able to converge to relative errors below around $7 \cdot 10^{-5}$, whilst the orange curve with $l_{\max} = 2$ can take the error down to $2 \cdot 10^{-5}$. The green curve with $l_{\max} = 3$ on the other hand converges almost linearly over the full range of $n_{\max}$ considered. This can be explained if we take another look at the RMSO_l plot of oxygen in figure 8.2. Comparing these findings with the RMSO_l plot of oxygen in figure 8.2 indicates some noteworthy agreements. Whilst RMSO_2 and RMSO_3 are of a similar size, there is a larger decrease, about 2 orders of magnitude, going from RMSO_3 to RMSO_4. In other words selecting $l_{\max} = 2$ instead of $l_{\max} = 1$ does not improve the error in the angular part as much as going to $l_{\max} = 3$ does. This is reflected by the fact that for $l_{\max} = 3$ an almost linear convergence up to $n_{\max} = 12$ is obtained, whilst for $l_{\max} < 3$, the error in the angular part starts to dominate from around $n_{\max} = 10$, such that the convergence slows down.

For constructing CS basis sets, which converge rapidly at HF level, a good balance between the error remaining in the discretisation of the angular part as well as the error in the discretisation of the radial part is required. As discussed the former aspect is controlled by selecting $l_{\max}$, the latter by selecting $n_{\max}$. We found in the case of oxygen that the magnitude of the error in the discretisation of the angular part and the trends in the RMSO_l versus l plots are related. We believe this finding to be general in the sense that a significant drop of RMSO_l from l to $l+1$ indicates that the CS basis set progression with $l_{\max} = l$ will allow convergence to a lower error in the HF energy than $l_{\max} = l - 1$ would be able to. Once a large enough value for $l_{\max}$ is chosen to converge the angular part, the convergence in the radial part with increasing $n_{\max}$ is initially linear. For large values of $n_{\max}$ the remaining error in the angular part will start to dominate and yet a larger $l_{\max}$ has to be chosen to make further progress.

From the examples considered in this section we would expect to require around $n_{\max} = 10$ to reach a target accuracy of 5 digits, which equals a relative error of below 10^{-5}. For Li and Be, where $l_{\max} = 0$ is sufficient this equals a $(10, 0, 0)$ basis consisting of only 10 CS basis functions. For N, Ne, Na, Mg, P and Ar a $(10, 1, 1)$ basis would be required, which has 37 basis functions. For the difficult cases like O, C and most other atoms with a single or 2 unpaired electrons at least $l_{\max} = 3$ is required. A $(10, 3, 3)$ CS basis has the enormous number of 126 basis functions, which still only reaches 5 digits of accuracy in the HF energy.

8.3 Convergence at correlated level

In the previous section we took a first look at the convergence properties of Coulomb-Sturmian-based discretisations at Hartree-Fock level. In practical quantum-chemical calculations Hartree-Fock is typically not the final answer, but only a first step, such that our discussion of convergence should really not focus on Hartree-Fock alone. In this spirit the aim of this section is to take our preliminary guidelines for sensible basis sets at HF level and describe some adaptions, which could help to construct sensible CS basis sets for correlated quantum-chemical methods. In line with what we discussed before, we will ignore the dependency of the CS basis on the exponent k_{exp} and keep a sensibly chosen, fixed exponent value for each atom throughout. So far we have not carried out many calculations for investigating the dependency of the correlated ground-state energy with respect to altering the maximal quantum numbers $n_{\max}$, $l_{\max}$ and $m_{\max}$. Our results in this section are therefore just exemplary and should not be taken to be general for Coulomb-Sturmian discretisations at correlated level. Moreover we have only considered two Post-HF approaches, namely MP2 and to a much lesser extent full CI, such that the behaviour might well deviate in other methods.

A first impression regarding the dependency of the correlation energy on the CS basis set provides figure 8.9 on the following page. It shows the fraction of the beryllium atom correlation energy, which is recovered by selected CS basis sets and at FCI and MP2 level, plotted against the size of the basis. As the reference, i.e. 100% correlation energy, we take the value obtained in a FCI calculation employing the rather large $(10, 2, 2)$ CS basis. When it comes to interpreting this figure one has to be a little careful. First of all the ground-state energy at HF level is not necessarily constant for all of the basis sets employed. In the depicted cases the changes in E_{HF} are, however, only very little and orders of magnitude smaller than the changes in correlation energy. The reason for this is that the selected basis sets only differ in the maximal angular momentum quantum

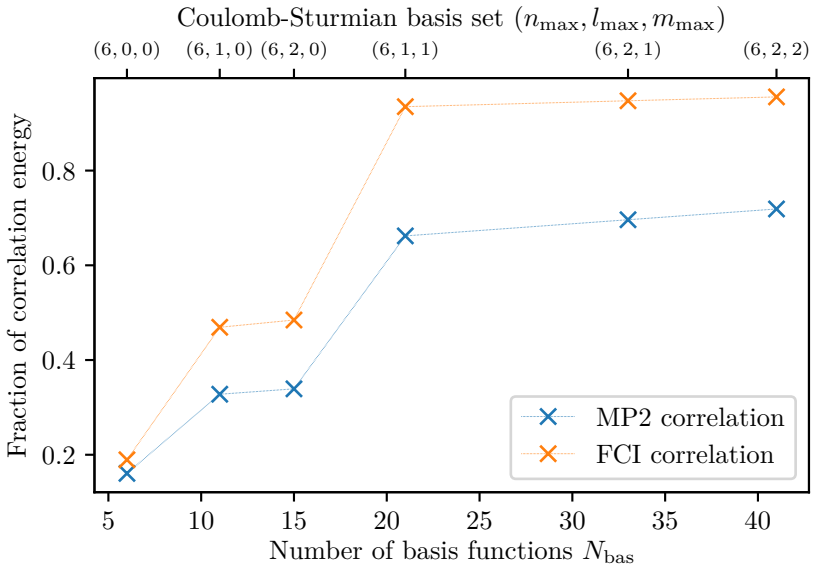


Figure 8.9: Fraction of beryllium correlation energy recovered relative to a FCI reference calculation with a $(10, 2, 2)$ CS basis. All calculations employ an exponent of $k_{\text{exp}} = 2.1$.

numbers $l_{\max}$ and $m_{\max}$, whilst the beryllium HF wave function is already converged very well in the angular part for $l_{\max} = 0$. The second issue with this plot is that the blue curve somewhat compares apples and pears, namely a variationally obtained reference correlation energy at FCI level with a perturbatively obtained correlation energy using MP2. Ignoring this fact for a moment, we find that the FCI and the MP2 correlation energy curves follow very similar trends. Most notable are the two strong increases in the amount of correlation energy recovered going from $(6, 0, 0)$ to $(6, 1, 0)$ and from $(6, 2, 0)$ to $(6, 1, 1)$. Interestingly another increase of $l_{\max}$, namely the transition $(6, 1, 1)$ to $(6, 2, 1)$ does not have such a pronounced effect. In line with the arguments presented in the context of HF it seems that the angular momentum discretisation of the MP2 or FCI ground-state wave functions are largely converged as soon as $l_{\max} = m_{\max} = 1$, such that further increases of angular momentum have much smaller effects.

Since the FCI calculations on large basis sets such as $(10, 2, 2)$ become extremely costly for larger atoms with more electrons, we did not perform such calculations except for beryllium. The only other correlation method, which is currently available from `molsturm` is MP2. Thus somewhat pragmatically we limited our investigation of the convergence at correlated level for the other atoms of the second and third period to MP2 only, arguing that at least for the case of beryllium we got the same trends. The results are presented in the tables of appendix C on page 213 and graphically in figures 8.10 and 8.11 on page 186. These show that the fraction of *total* MP2 energy which is missed by a particular basis set compared to the most accurate result we obtained in our calculations for a particular atom. Again this value is plotted against the size of the basis set. Similar to the case of beryllium sketched above, we concentrate on capturing the effect of converging the discretisation of the angular part of the MP2 wave function by varying $l_{\max}$ and $m_{\max}$. In figure 8.10 for the half-filled and filled valence shells, a

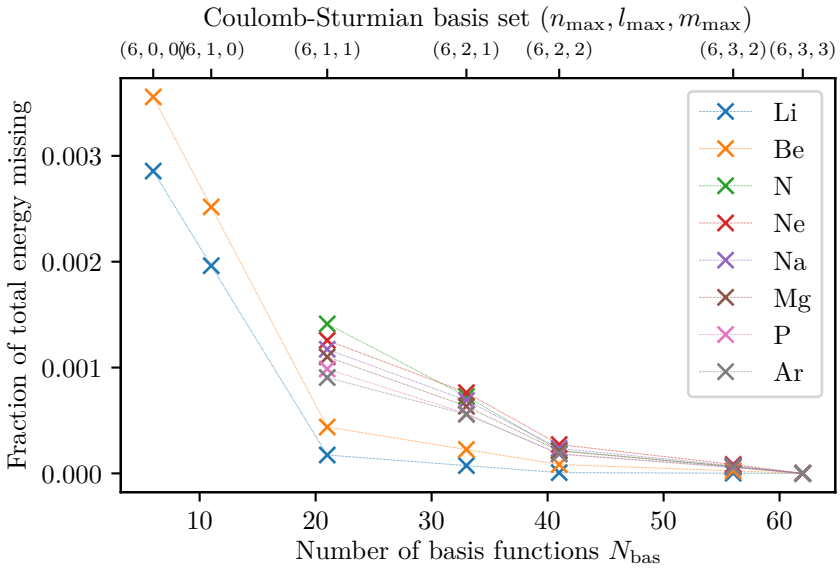


Figure 8.10: Plot of the missing fraction of total MP2 energy compared to a calculation employing a $(6, 3, 3)$ CS basis versus the basis size. Shown are the atoms of the second and third period with a full or half-full valence shell.

convergence is visible. For Li and Be, where $l_{\max} = m_{\max} = 0$ converges the angular part of the HF ground state, $l_{\max} = m_{\max} = 1$ does so pretty much for the MP2 ground state. For the other atoms shown in figure 8.10 $l_{\max} = m_{\max} = 2$ is at least required. In other words compared to converging the HF ground state we roughly speaking need one extra angular momentum. For the cases of one and two unpaired electrons, which are shown in figure 8.11, the picture is not so conclusive. Since already the HF ground state requires $l_{\max} = 3$ for a decent modelling of the angular part, this is of course at least required for the MP2 wave function as well. But figure 8.11 seems to suggest that $l_{\max} = 4$ is important as well since the change from $(6, 3, 3)$ to $(6, 4, 4)$ is much steeper compared to the change from $(6, 2, 2)$ to $(6, 3, 3)$ in figure 8.10. Whether even larger angular momentum is required cannot be said with the currently available results.

Following our discussion above it is probably a little far fetched to assume that one can properly judge how well a CS basis set is able to capture correlation effects just by looking at MP2 correlation energies. Nevertheless given how well the trends of MP2 and FCI agree for beryllium, it seems likely that at least for the well-behaving cases with closed or half-filled valence shell the rule to take one extra angular momentum for the correlated calculation captures the predominant effects. On top of the few investigations towards converging the angular part of the correlated wave function, no further investigation regarding $n_{\max}$ and the convergence of the radial part was attempted so far.

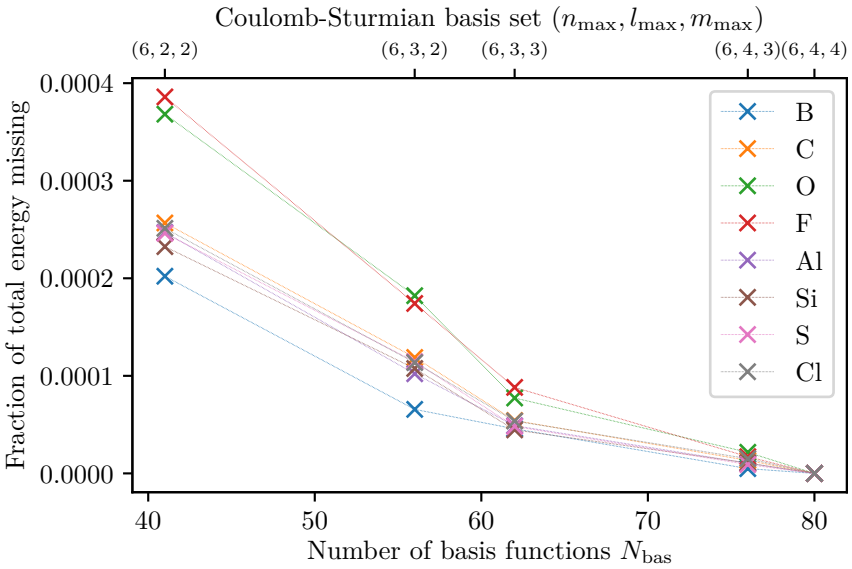


Figure 8.11: Plot of the missing fraction of total MP2 energy compared to a calculation employing a $(6, 4, 4)$ CS basis versus the basis size. Shown are the atoms of the second and third period with one or two unpaired electrons.

8.4 The effect of the Coulomb-Sturmian exponent

In our discussion about the properties of Coulomb-Sturmian basis sets in this chapter, we have neglected the effect of the CS exponent k_{exp} so far. Our main argument was that a CS basis is complete regardless of the value of k_{exp}, such that for large enough CS basis sets the result will not depend on k_{exp} anyway. In practice the aim is of course to yield a sensible discretisation of the wave function in the smallest basis possible and furthermore to obtain the best rate of convergence as the basis is increased. As we will discuss in this section the value of k_{exp} has an influence on these matters and can therefore not be chosen completely arbitrarily. We will subsequently develop an algorithm for obtaining the optimal exponent k_{opt} with respect to minimising the HF energy and present some results for the atoms of the first two periods of the periodic table.

Notice that for the case of a CS-based FCI a reformulation of the FCI problem exists, which allows to find the optimal exponent k_{exp} alongside the FCI energies [29]. In fact this reformulation yields to an eigenproblem in which the obtained eigenvalues are not energies, but the optimal CS exponents for each FCI state. Via the relationship

$$E = -\frac{N_{\text{elec}} k_{\text{exp}}^2}{2}$$

the energy of each state can be found thereafter. This is highly advantageous, because the aforementioned explicit optimisation of the exponent can be avoided. To the best of my knowledge a related approach for the HF problem, has not been found, however, such that at the level of HF and Post-HF (excluding FCI) a CS exponent k_{exp} needs to be specified explicitly for performing a calculation.

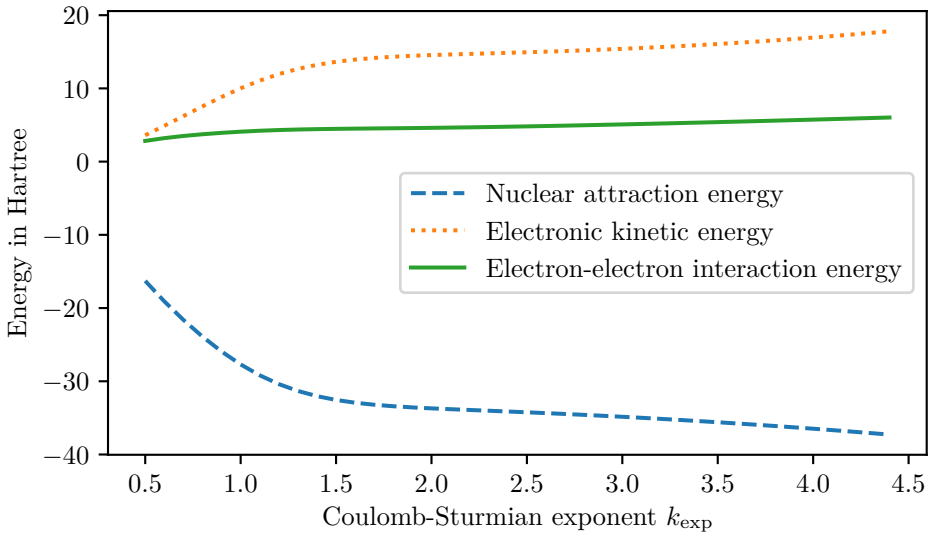


Figure 8.12: Plot of the HF energy contributions of the beryllium atom versus the Coulomb-Sturmian exponent k_{exp}. All calculations are done in a $(5, 1, 1)$ CS basis.

In the CS basis functions k_{exp} only occurs in the radial part (8.2). In the form of the exponential term $\exp(-k_{\mathrm{exp}}r)$ it influences how quickly the basis functions decay asymptotically and in the form of the polynomial prefactor it determines the curvature of the radial functions as they oscillate between the radial nodes. Keeping this in mind let us consider figure 8.12, which shows the changes to individual energy contributions of the HF ground-state energy as k_{exp} is altered. The largest changes are apparent for the nuclear attraction energy, which decreases — initially rather steeply — as k_{exp} is increased. This can be easily understood from a physical point of view: Since larger values of k_{exp} imply a more rapid decay of the basis functions, the electron density on average stays closer to the nucleus, which in turn leads to a lower (more negative) interaction energy between electrons and nucleus. The converse effect happens for smaller values of k_{exp}, where the electron density is more expanded and thus on average further away from the nucleus. On the other hand the kinetic energy is related to the curvature of the wave function, which — as described above — increases for larger k_{exp}. In other words the trends of nuclear attraction energy and electronic kinetic energy oppose each other, with the kinetic energy being somewhat less effected. On the scale depicted in figure 8.12 the variation of the electron-electron interaction, i.e. both classical Coulomb repulsion as well as the exchange interaction combined, is much less pronounced. Only a very minor increase with k_{exp} can be observed. The physical mechanism is again similar to the nuclear attraction energy term, namely that larger k_{exp} compresses the wave function and thus leads to the electrons reside more closely to another, which increases the Coulomb repulsion between them. The exchange interaction is effected as well, but the changes are smaller and thus not visible.

Summing up all energy contributions leads to the blue curve in figure 8.13, which shows the total Hartree-Fock energy versus the Coulomb-Sturmian exponent k_{exp}. From our discussion of the individual terms it is apparent that at small values for k_{exp} the

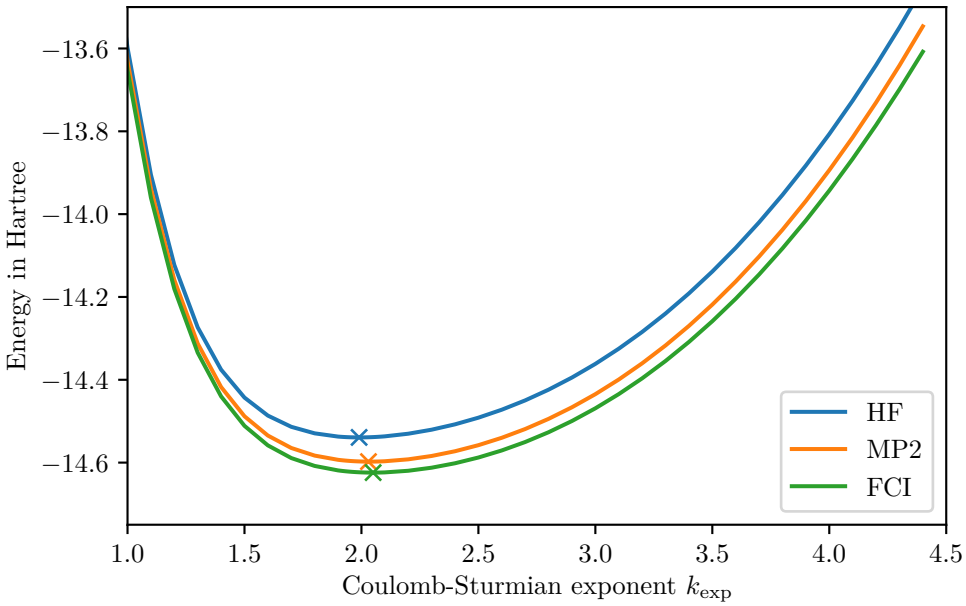


Figure 8.13: Plot of the HF, MP2 and FCI ground state energies of beryllium versus the Coulomb-Sturmian exponent k_{exp}. The optimal exponent k_{opt} for each method is marked by a cross. All calculations are done in a $(5, 1, 1)$ CS basis.

increase in nuclear attraction energy dominates, such that the HF energy increases rapidly. At large distances the kinetic energy and electron-electron interaction terms win, such that a convex curve for the plot E_{HF} versus k_{exp} results. Adding correlation effects by a treatment of the atom at MP2 or FCI level, does not change this overall behaviour much. Up to a large extent the curves are just shifted downwards by the correlation energy term. The shift is, however, not completely uniform. This can be seen if we consider the optimal CS exponent k_{opt}, which is denoted by a cross in each of the plots of figure 8.13. This exponent of minimal energy shifts to slightly larger values going from HF to MP2 and finally to FCI indicating that the amount of correlation energy is somewhat larger at exponents slightly above k_{opt} for Hartree-Fock. Notice that k_{opt} not only depends on the method used for modelling a particular state, but it well depends on the state as well. For example for modelling the first $T1$ excited state of beryllium a smaller value for k_{opt} is obtained than the FCI k_{opt} of the depicted $S0$ ground state.

Since k_{exp} only occurs in the radial part of the CS basis functions the effect of its variation depends on the maximal principle quantum number n_{max} of the basis set. As larger and larger values of n_{max} are used, the discretisation of the radial part of the wave function becomes more and more complete, such that the choice of k_{exp} in turn becomes less important. Figure 8.14 on the facing page shows this for the ground-state energy of the carbon atom at unrestricted HF and MP2 level. Whilst a $(4, 2, 2)$ CS basis reproduces largely the shape of the plots in figure 8.14, for $(5, 2, 2)$ and $(6, 2, 2)$ the energy versus exponent curves become visibly flatter close to the optimal exponent (around $k_{\mathrm{exp}} = 2.8$). The influence of increasing n_{max} is not the same for all values of k_{exp}. Instead the curves seem to bend down in the range $k_{\mathrm{exp}} > 3$, indicating a faster rate of convergence in this

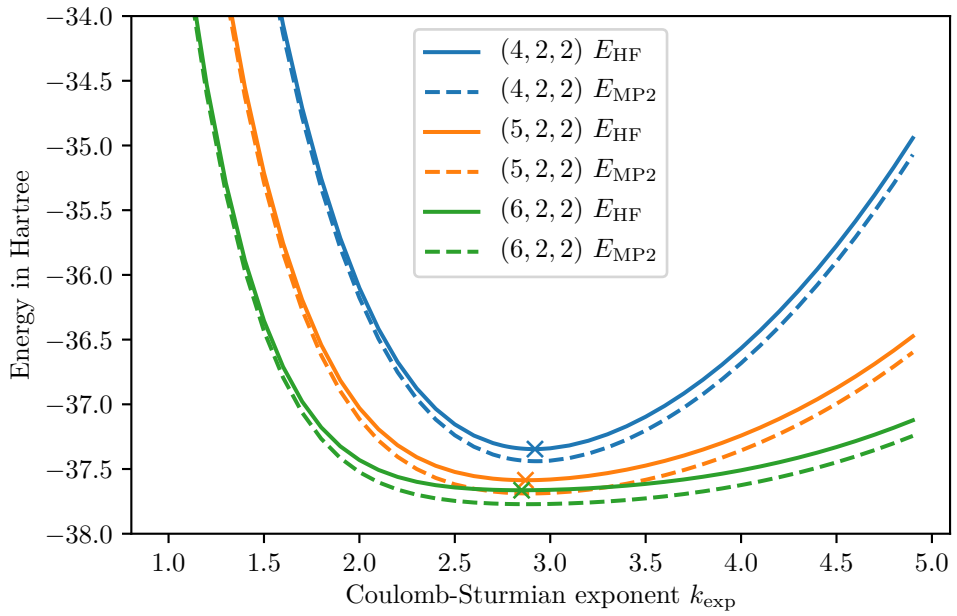


Figure 8.14: Plot of the unrestricted HF and MP2 ground state energies of carbon versus the Coulomb-Sturmian exponent k_{exp} in the $(4,2,2)$, $(5,2,2)$ and $(6,2,2)$ basis sets. The optimal exponent k_{opt} at HF level for each basis set is marked by a cross.

region compared to the range $k_{\mathrm{exp}} < 2.5$. In other words choosing a CS exponent larger than k_{opt} will generally speaking lead to better convergence, thus a smaller error than choosing a too small exponent[1]. Another conclusion we can draw from figure 8.14 is that the optimal value for the exponent k_{opt} depends on n_{max} as well as larger basis sets give rise to smaller values for k_{opt}. We can rationalise by taking the plots of the energy terms in figure 8.12 on page 187 into account. We already noticed above that the nuclear attraction energy is influenced by k_{exp} most strongly. Additionally it is (by magnitude) the largest contribution to the HF energy. In order to yield the minimal ground-state energy in a small basis the dominating effect is therefore to minimise the nuclear attraction energy as much as possible. As a result the optimal exponent k_{opt} takes comparatively large values. As the basis becomes larger a balanced description of the complete physics becomes possible, such that the electron repulsion and kinetic energy terms are described better as well and thus smaller values for k_{opt} result.

Due to the structure of the energy versus exponent curves, like the ones shown in figure 8.14, one hardly ever needs to know k_{opt} very accurately. As long as one uses a reasonable guess, which is constructed to overestimate k_{opt} rather than underestimate it, one is usually safe. If a highly accurate treatment of a particular system is required, then increasing n_{max} has both a much larger effect and is computationally cheaper than finding the optimal exponent in the smaller basis. See the next section for details.

[1] We already noted this aspect in the context of discussing the local energy plots in section 5.3.6.

8.4.1 Determining the optimal exponent $k_{\mathbf{opt}}$

For variational quantum-chemical methods finding the best Coulomb-Sturmian exponent k_{opt} for the ground state is equivalent to minimising the ground state energy with respect to k_{exp}. Since such energy curves are convex (compare figures 8.13 and 8.14) and only scalar functions of a single parameter, this minimisation can be performed quite effectively by a gradient-free optimisation algorithm. The procedure implemented in `molsturm` for finding k_{opt} uses Brent's method [267]. Starting from a reasonable guess for k_{opt} convergence to the minimum is usually achieved in around 10 iterations. For achieving this Brent's method will require a similar number of energy computations using the chosen quantum-chemical method and the chosen CS basis.

With respect to the basis, which is used for such a procedure, there are two things to note. Firstly we already mentioned in our previous discussion that k_{exp} is a parameter, which only affects the radial part. In other words for obtaining a situation in which the individual calculations of the energies are not dominated by the error in the angular discretisation, but the current value of k_{exp}, large enough values for $l_{\max}$ and $m_{\max}$ should be chosen. Too large values of $l_{\max}$ will, however, lead to large basis sets, thus long run times for the energy calculations. In practice a compromise between accuracy and runtime needs to be found. Our investigations (see tables 8.2 and 8.3 on page 195) seem to suggest that one can find reasonable values for k_{opt} already for basis sets where $l_{\max}$ is chosen smaller than the value suggested by the RMSO_l plots. Secondly one should keep in mind that too large values of $n_{\max}$ will cause the energy-vs-k_{exp} curves to become flat around k_{opt}, which slows down convergence of the optimisation procedure. Keeping in mind that typically getting *roughly* the right value for k_{opt} is good enough, it is sometimes more sensible to find k_{opt} in a smaller basis set, where the energy-vs-k_{exp} is more steep and calculations are faster, and use this value for larger basis sets as well. For the reasons we discussed in the previous section such a k_{opt} from a smaller basis will always be an *overestimation* of the actual k_{opt}, which is favourable.

Our investigations have so far only considered obtaining optimal exponents k_{opt} at HF level. The most challenging aspect for doing so is in fact the stability of the SCF procedure itself. Especially at the beginning of the iteration, when the Coulomb-Sturmian exponent k_{exp} is still relatively far off the optimal value, the core Hamiltonian guess[2] we employ by default is not very good and frequently fails to lead to the true SCF minimum in our SCF scheme. Much rather another stationary point on the SCF Stiefel manifold is found. If we now continue to use the resulting wrongfully converged SCF coefficients as the guess for the next iteration of Brent's method, we will typically manage to find a k_{opt}, but this might not be the k_{opt} of the true SCF minimum, i.e. the true HF ground state. On the other hand if we start from the core Hamiltonian guess each time, it can happen that the SCF iterations for different values of k_{exp} lead to different stationary points on the Stiefel manifold. This violates a fundamental assumption of Brent's method, namely the continuity of the objective function. In other words the optimisation procedure is likely to find a wrong value for k_{opt} in this case.

Our remedy is to first make very sure we obtain a reliable guess for starting the SCF procedures called during the optimisation before starting the optimisation procedure energy versus k_{exp} at all. In order to do so we first perform 5 SCFs starting from totally random guesses for the input value of k_{exp} supplied by the user. From the lowest-energy

[2] So far only random guesses, guesses from previous SCF cycles and core Hamiltonian guesses are implemented in `molsturm`.

result of these we then take the orbital energies ε_i and use them to estimate a second value for k_{exp}, namely

$$k_{\text{exp}} \simeq \sqrt{\frac{-2}{N_{\text{elec}}} \sum_{i \in \mathcal{I}_{\text{occ}}} \varepsilon_i}. \tag{8.6}$$

The rationale for this heuristic formula is the energy-dependent decay of the exact wave function [69], which — assuming HF to be exact — would manifest as well in an energy-dependent decay of the orbitals by themselves. Applying the formula

$$\varepsilon_i = -\frac{1}{2}k_i^2 \qquad \Leftrightarrow \qquad k_i = \sqrt{-2\varepsilon_i}$$

to yield the best exponent k_i for describing orbital i and taking the average over all k_i results in (8.6). The result from applying (8.6) is typically not extremely good, but in the cases we considered it is at least in the same order of magnitude as the final k_{opt}, such that this estimate is easy to compute and corrects for the cases, where the user's guess was very far off. For this second k_{exp} we perform another 5 SCF iterations starting completely from random guesses. From all 10 obtained SCF ground states, both the 5 with the k_{exp} supplied by the user and the 5 with the k_{exp} from (8.6), we only keep the solution, which has the lowest HF energy. For *all* SCFs which are started during the subsequent energy versus k_{exp} optimisation this solution is used as the initial guess. In this way all inner SCFs approach the SCF procedure from the same reliable guess, which largely avoids discontinuities in the HF energies and thus directs Brent's method to a sensible value for k_{opt}.

This algorithm for finding k_{opt} is not cheap, since around 20 to 30 complete SCFs are required for convergence. It is, however, reliable and allowed us to obtain optimal exponents for a range of basis sets for all atoms of the second and third period. These results are shown in tables 8.2 on page 194 and 8.3 on page 195. For convenience this procedure is implemented in `molsturm` and can be called from `python` using the function `find_kopt` from the module `molsturm.sturmian.cs`. `molsturm` also offers the function `empirical_kopt` as a cheaper empirical estimate for k_{opt}. It is based on interpolations using the values from tables 8.2 and 8.3, can thus only be used for atoms of the second and third period.

8.4.2 Relationship to the effective nuclear charge

In his 1930 paper Slater [3] proposed simple guidelines for approximating the orbitals of atoms. For this he introduced for each orbital a shielding parameter σ, which was supposed to indicate how much of the nuclear charge is screened away by the electrons closer to the core. He then proceeded to describe the functional form of the atomic orbitals by the simple analytic expression

$$\chi_{n^*,\sigma} = r^{n^*-1} \exp\left(-\frac{(Z-\sigma)r}{n^*}\right) \equiv r^{n^*-1} \exp\left(-\zeta r\right), \tag{8.7}$$

along with empirical rules to find n^* and $Z - \sigma$, the **effective nuclear charge**. We already met functions like (8.7) as basis functions for solving the Hartree-Fock problem when we discussed Slater-type orbitals in section 5.3.3 on page 93. In the same chapter we mentioned the close relationship between the Coulomb-Sturmians and the Slater-type orbitals in the sense that the CS exponent k_{exp} plays the role of the Slater exponent

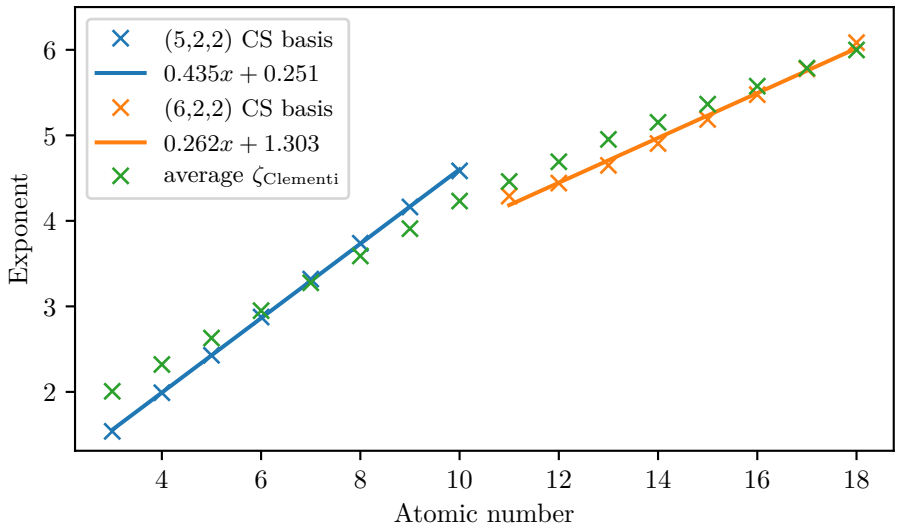


Figure 8.15: Plot of the atomic number versus the optimal Coulomb-Sturmian exponent k_{opt} for the neutral atoms of the second and the third period. For comparison the occupation-averaged value of the Clementi and Raimondi [268] optimal Slater exponent ζ_{Clementi} are shown as well.

ζ with the subtle difference that for CS basis sets all functions need to carry the same exponent.

The rough results obtained by Slater's rules were later refined by Clementi and Raimondi [268], who determined optimal values for ζ by performing HF calculations. In turn they used these values to define a new set of shielding parameters and thus a new set of effective nuclear charges. Their optimisation procedure was very similar to the procedure we followed to find k_{opt}, namely they optimised the energy variationally with respect to the Slater exponents ζ. Both the similarity of the form of both types of functions as well as the similarity of the procedures followed indicates that our k_{opt} and the optimal exponents ζ_{Clementi} from Clementi and Raimondi should bear some resemblance.

As a first attempt to characterise this similarity we propose to compare k_{exp} to the average value of ζ_{Clementi} taken in all occupied orbitals of a particular atom. A plot of these values across the second and third period of the periodic table is shown in figure 8.15. Over the full depicted range the magnitude of k_{opt} and ζ_{Clementi} stays similar. Furthermore except the sharp drop going from atom number 10 to 11 the roughly linear increase of ζ_{Clementi} is reproduced by k_{opt}. One reason why the diverging feature between atom number 10 and 11 is observed is that we chose to use a different, larger CS basis set for determining k_{opt} in the third period. In our discussion related to figure 8.14 we already mentioned that larger basis sets tend to yield a lower value of k_{opt}. The observed drop in figure 8.15 is, however, much larger than any lowering induced by increasing the basis we observed in our calculations (see tables 8.2 and 8.3). One possible additional explanation could be the reduction of information, which is implied by taking the average of all ζ_{Clementi}. For example when changes in the physics of the

electronic structure of the atom cause relative adjustments of the exponents ζ_{Clementi}, this is not captured by the average ζ_{Clementi}. Especially when going to a new shell, i.e. when adding a new, more expanded orbital with only a single electron in it, the structure of the electron density does indeed change more compared to the previous atom as in other cases. Whilst the Slater-type orbital basis has more degrees of freedom in form of the multiple exponents to adapt to this, the CS basis needs to balance the errors, which could lead to the observed deviation from the trend in the previous period.

Overall figure 8.15 suggests that there is some connection between k_{opt} and the average ζ_{Clementi}. Considering the relationship between ζ_{Clementi} and the effective nuclear charge in turn, we could think of k_{opt} as a measure for the average effective nuclear charge, which is felt by the individual orbitals.

system	CS basis	k_{opt}	N_{bas}	E_{HF}	relative error
Li	$(4,1,1)$	1.562	13	-7.38483^{U}	$6.4 \cdot 10^{-03}$
Li	$(5,1,1)$	1.539	17	-7.41652^{U}	$2.2 \cdot 10^{-03}$
Li	$(6,1,1)$	1.533	21	-7.42812^{U}	$6.2 \cdot 10^{-04}$
Be	$(4,1,1)$	2.017	13	-14.46796^{R}	$7.2 \cdot 10^{-03}$
Be	$(5,1,1)$	1.990	17	-14.53916^{R}	$2.3 \cdot 10^{-03}$
Be	$(6,1,1)$	1.988	21	-14.56445^{R}	$5.9 \cdot 10^{-04}$
B	$(3,2,2)$	2.480	14	-23.64847^{U}	$3.6 \cdot 10^{-02}$
B	$(4,1,1)$	2.464	13	-24.32117^{U}	$8.7 \cdot 10^{-03}$
B	$(4,2,2)$	2.466	23	-24.32594^{U}	$8.5 \cdot 10^{-03}$
B	$(5,1,1)$	2.426	17	-24.46411^{U}	$2.8 \cdot 10^{-03}$
B	$(5,2,2)$	2.428	32	-24.46852^{U}	$2.6 \cdot 10^{-03}$
B	$(6,1,1)$	2.409	21	-24.51204^{U}	$8.7 \cdot 10^{-04}$
B	$(7,2,2)$	2.402	50	-24.52731^{U}	$2.5 \cdot 10^{-04}$
C	$(4,1,1)$	2.916	13	-37.34490^{U}	$9.3 \cdot 10^{-03}$
C	$(5,1,1)$	2.873	17	-37.58533^{U}	$2.9 \cdot 10^{-03}$
C	$(6,1,1)$	2.849	21	-37.66284^{U}	$8.2 \cdot 10^{-04}$
N	$(4,1,1)$	3.364	13	-53.88221^{U}	$9.6 \cdot 10^{-03}$
N	$(5,1,1)$	3.320	17	-54.24940^{U}	$2.9 \cdot 10^{-03}$
N	$(6,1,1)$	3.287	21	-54.36501^{U}	$7.3 \cdot 10^{-04}$
O	$(5,2,2)$	3.738	32	-74.57763^{U}	$3.2 \cdot 10^{-03}$
O	$(6,1,1)$	3.685	21	-74.74979^{U}	$9.3 \cdot 10^{-04}$
O	$(7,2,2)$	3.638	50	-74.79613^{U}	$3.1 \cdot 10^{-04}$
F	$(5,2,2)$	4.162	32	-99.07686^{U}	$3.4 \cdot 10^{-03}$
F	$(6,1,1)$	4.099	21	-99.32043^{U}	$9.7 \cdot 10^{-04}$
F	$(7,2,2)$	4.038	50	-99.38482^{U}	$3.2 \cdot 10^{-04}$
Ne	$(4,1,1)$	4.637	13	$-127.0528\ ^{R}$	$1.2 \cdot 10^{-02}$
Ne	$(5,1,1)$	4.585	17	$-128.0943\ ^{R}$	$3.5 \cdot 10^{-03}$
Ne	$(6,1,1)$	4.512	21	$-128.4255\ ^{R}$	$9.5 \cdot 10^{-04}$

^{U} unrestricted HF
^{R} restricted HF

Table 8.2: Optimal CS exponent for the 2nd period of the periodic table at HF level. Relative errors are given with respect to the reference energies of table 8.1 on page 174.

system	CS basis	k_{opt}	N_{bas}	E_{HF}	relative error
Na	$(5,1,1)$	4.449	17	-159.8132^{U}	$1.3 \cdot 10^{-02}$
Na	$(6,1,1)$	4.286	21	-160.9291^{U}	$5.7 \cdot 10^{-03}$
Na	$(7,1,1)$	4.096	25	-161.4028^{U}	$2.8 \cdot 10^{-03}$
Na	$(8,1,1)$	3.917	29	-161.6200^{U}	$1.5 \cdot 10^{-03}$
Mg	$(5,1,1)$	4.583	17	-196.1362^{R}	$1.7 \cdot 10^{-02}$
Mg	$(6,1,1)$	4.442	21	-198.0276^{R}	$8.0 \cdot 10^{-03}$
Mg	$(7,1,1)$	4.267	25	-198.8705^{R}	$3.7 \cdot 10^{-03}$
Mg	$(8,1,1)$	4.107	29	-199.2445^{R}	$1.9 \cdot 10^{-03}$
Al	$(6,2,2)$	4.649	41	-239.5138^{U}	$9.8 \cdot 10^{-03}$
Al	$(7,1,1)$	4.485	25	-240.7812^{U}	$4.5 \cdot 10^{-03}$
Al	$(7,2,2)$	4.485	50	-240.7885^{U}	$4.5 \cdot 10^{-03}$
Si	$(4,2,2)$	5.009	23	-271.6163^{U}	$6.0 \cdot 10^{-02}$
Si	$(5,2,2)$	5.009	32	-282.0009^{U}	$2.4 \cdot 10^{-02}$
Si	$(6,2,2)$	4.904	41	-285.7755^{U}	$1.1 \cdot 10^{-02}$
Si	$(7,1,1)$	4.754	25	-287.4682^{U}	$4.8 \cdot 10^{-03}$
Si	$(7,2,2)$	4.755	50	-287.4751^{U}	$4.8 \cdot 10^{-03}$
Si	$(8,1,1)$	4.616	29	-288.1995^{U}	$2.3 \cdot 10^{-03}$
P	$(6,1,1)$	5.186	21	-336.9464^{U}	$1.1 \cdot 10^{-02}$
P	$(7,1,1)$	5.049	25	-339.0724^{U}	$4.8 \cdot 10^{-03}$
P	$(8,1,1)$	4.922	29	-339.9651^{U}	$2.2 \cdot 10^{-03}$
S	$(4,2,2)$	5.451	23	-370.8869^{U}	$6.7 \cdot 10^{-02}$
S	$(5,2,2)$	5.540	32	-387.1635^{U}	$2.6 \cdot 10^{-02}$
S	$(6,2,2)$	5.476	41	-392.9687^{U}	$1.1 \cdot 10^{-02}$
Cl	$(5,2,2)$	5.821	32	-447.2744^{U}	$2.7 \cdot 10^{-02}$
Cl	$(6,2,2)$	5.777	41	-454.1715^{U}	$1.2 \cdot 10^{-02}$
Cl	$(7,2,2)$	5.656	50	-457.2387^{U}	$4.9 \cdot 10^{-03}$
Ar	$(5,1,1)$	6.109	17	-512.6726^{R}	$2.7 \cdot 10^{-02}$
Ar	$(6,1,1)$	6.084	21	-520.7125^{R}	$1.2 \cdot 10^{-02}$
Ar	$(7,1,1)$	5.970	25	-524.2770^{R}	$4.8 \cdot 10^{-03}$
Ar	$(8,1,1)$	5.862	29	-525.7054^{R}	$2.1 \cdot 10^{-03}$

^{U} unrestricted HF
^{R} restricted HF

Table 8.3: Optimal CS exponent for the 3rd period of the periodic table at HF level. Relative errors are given with respect to the reference energies of table 8.1 on page 174.

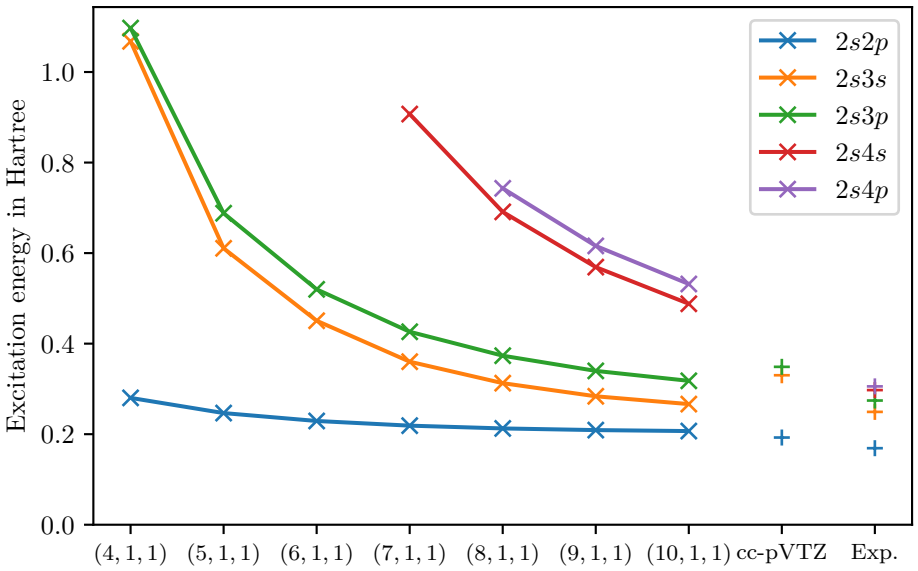


Figure 8.16: Convergence of a CS-based ADC(2) [118] calculation of beryllium. Plotted are the singlet excitation energies going from the ground state $2s2s$ to the denoted excited state. We show the results from a progression of CS calculations with exponent $k_{\text{exp}} = 2.0$ as well as bases sets of the form $(n_{\text{max}}, 1, 1)$. For comparison the last two data points show the results from a cGTO-based calculation using cc-pVTZ [263] as well as the experimental values from Moore [269].

8.5 Coulomb-Sturmian-based excited states calculations

This section provides an outlook towards excited states calculations employing Coulomb-Sturmians as the underlying basis functions. As mentioned in section 7.4 on page 168 the `python` interface of `molsturm` allowed us to link it to multiple third-party packages. One of these is `adcman` [210], which in this manner can be employed to perform excited states calculations based on the algebraic diagrammatic construction scheme at ADC(1), ADC(2), ADC(2)-x [118] and ADC(3) [119] level based on any basis function type supported by `molsturm`.

This section reports the first successful ADC(2) calculation using Coulomb-Sturmians for the discretisation. In figure 8.16 we show the singlet excitation energies of the beryllium atom as a progression with increasing CS basis set size from $(4, 1, 1)$ to $(10, 1, 1)$. For comparison the figure further indicates an equivalent calculation using cc-pVTZ [263] as well as the experimental values [269]. Within the CS basis set progression the results converge from above as expected. Judging from the plots a maximum principle quantum number around $n_{\text{max}} = 10$ seems to be at least required to converge the radial part. This agrees with our findings for the ground state, see for example figures 8.7 on page 181 and 8.8 on page 182,

Comparing the computed excitation energies to the experimental values the $(10, 1, 1)$

basis set performs worse than cc-pVTZ at the first excited state $2s2p$, but better for the $2s3s$ and the $2s3p$ states. This result is, however, a little misleading for two reasons. First the cc-pVTZ basis set and the $(10, 1, 1)$ CS basis are not exactly comparable, since they have a deviating structure. Whilst cc-pVTZ contains 10 contracted Gaussian functions with angular momentum up to $l = 4$, $(10, 1, 1)$ contains 37 uncontracted Coulomb-Sturmians with angular momentum at most $l = 1$. Second the CS basis has not really been optimised at all with respect to ADC(2) as a method or with respect to the excited states of beryllium. For example the employed CS exponent of 2.0 is a good value for describing the ground state of beryllium, but it is certainly not an optimal value for describing the excited states. Further there is some indications from example calculations that at least angular momentum $l = 2$ is required for a proper description of the $2s2p$ excited state. In figure 8.16 this amounts to explain, why the observed convergence to a *higher* excitation energy than the cGTO result or experiment is observed.

Keeping both these aspects in mind it is therefore not yet possible to directly compare the CS and the cGTO results. But given than no attempts to optimise the CS basis towards the ADC(2) excited states setting have been made, it is still remarkable to find the observed convergence. A further, more systematic investigation could easily lead to a clarification of the picture and allow to contrast the different properties of both discretisations with respect to computing atomic spectra.

Chapter 9

Conclusions

> *We must include in any language with which we hope to describe complex data-processing situations the capability for describing data.*
>
> — Grace Hopper (1906–1992)

The present thesis devised a self-consistent field (SCF) scheme for solving the Hartree-Fock (HF) problem based on matrix-vector contraction expressions. It was subsequently utilised in order to design and implement the quantum-chemical method development framework `molsturm`, where novel methods can be readily implemented and tested. Furthermore `molsturm` was used to investigate the convergence properties of quantum-chemical calculations based on Coulomb-Sturmians, a basis function type which got little attention so far. Initial results of Coulomb-Sturmian-based excited states calculation employing the algebraic-diagrammatic construction scheme for the polarisation propagator were reported as an outlook to future developments.

Chapter 1 provided an introduction into the setting of the thesis. Chapter 2 reviewed the mathematical background of quantum mechanics and sketched important results of functional analysis and spectral theory. In chapter 3 the Ritz-Galerkin ansatz for numerically treating spectral problems was discussed, followed by the ideas of common algorithms to solve the arising eigenproblems. The emphasis was put on discussing this established mathematical material from a quantum-chemical perspective, while indicating the often overlooked peculiarities, which occur when transforming from the infinite-dimensional regime of functional analysis to the finite-dimensional regime of linear algebra.

In the light of this section 4.2.1 discussed the spectral properties of the electronic Schrödinger equation and described common quantum-chemical methods for solving this equation numerically. The mathematical formulation of both full configuration interaction (FCI), in section 4.3, as well as HF, in section 4.4, were discussed. In section 4.4.1 multiple formulations of HF were given and their numerical properties were compared. In remark 4.18 the usual SCF procedure as a scheme to solve the HF problem was introduced. The physical aspects missing in an HF treatment of the electronic structure were mentioned in section 4.5.1 and common Post-HF methods to correct for these were reviewed in sections 4.5.2 to 4.5.5.

A detailed discussion of the basis function types, which can be employed to discretise the HF problem was given in section 5.3. Section 5.3.1 and 5.3.2 first provided a summary of the desirable properties of such a discretisation, namely feasible resulting numerical problems on the one hand and a good description of the physical features of the wave function on the other. With respect to this four basis function types were evaluated in particular.

First the well-known properties of the Slater-type orbitals and the Gaussian-type orbitals were reviewed in sections 5.3.3 and 5.3.4. It was mentioned that Slater-type orbitals lead to challenging integrals when discretising HF in such a basis, whereas discretisations employing Gaussian-type orbitals give up a physical functional form in the basis functions in order to gain feasible integrals. The well-known conclusion that suitable Gaussian basis sets need to be used to correctly describe electronic structures was emphasised.

In contrast to the first two, both finite elements, as well as Coulomb-Sturmians, were discussed. It was demonstrated how both of these basis functions have the possibility to represent all physical features of the wave function properly, such that they are promising alternatives. In contrast to Gaussian-type and Slater-type orbitals their discretisations, however, gave rise to unusual numerical demands. For finite elements, for example, the matrix representation of the exchange matrix term of the HF equations was shown to be rather expensive to compute. Building on the idea of matrix-free methods [162], a novel, contraction-based ansatz for HF was introduced to compensate for this. In this approach the difference is that storing the Fock matrix in memory is avoided and instead only matrix-vector product applications are performed. An analysis of the computational complexity for the exchange term in the context of finite elements was presented, which showed that a contraction-based scheme reduces the formal computational scaling from $\mathcal{O}(N_{\text{bas}}^2)$ to $\mathcal{O}(N_{\text{bas}})$ with N_{bas} being the number of finite elements. For Coulomb-Sturmians such a contraction-based SCF ansatz allowed to exploit the available selection rules in the integral kernels to a further extent, thus similarly improving performance. Even though both Coulomb-Sturmians as well as Slater-type orbitals are exponential basis functions of related functional form, it was found that the integral expressions of Coulomb-Sturmians are much simpler and fit very well into the context of a contraction-based SCF.

Section 5.3.9 summarised our discussion of the basis function types and section 5.4 reviewed common SCF algorithms with respect to their ability to support the contraction-based SCF. For the case of the optimal damping algorithm [195] section 5.4.4 gave an approximate modification to carry some advantageous properties of the complete scheme to the contraction-based setting.

In chapter 6 contraction-based methods were formally introduced and in section 6.1.1 their potentials and drawbacks were evaluated. The trend of an increasing gap between processor and memory performance was outlined and used to emphasise that recomputing data can sometimes be advantageous, even over storing it in main memory. A typical challenge with contraction-based methods, namely their tendency to lead to more involved and harder-to-read code, was identified and lazy matrices were introduced in section 6.2 as a data structure to tackle this problem. It was discussed how lazy matrices, as a generalisation of conventional matrices, allow to encapsulate arbitrary contraction expressions, but still maintain the high-level interface of matrices. This was achieved by employing lazy evaluation, which means that operations on lazy matrices

are only evaluated when needed and otherwise cached inside an expression tree for later evaluation. Whilst the primary application for lazy matrices in this thesis was the quantum-chemical program package `molsturm`, lazy matrices are more general and could be used for other problems of physics and chemistry as well. For this reason an implementation of lazy matrices was carried out in the `lazyten` library and its applicability demonstrated in section 6.3. An example showing a simple SCF scheme coded in the language of `lazyten` was given in section 6.3.1.

In section 7.2 the design of the `molsturm` program package was discussed. In particular the interplay between the contraction-based SCF scheme and the integral library was detailed in section 7.2.1. It was emphasised how the lazy matrix language of `lazyten` on the one hand enables to write SCF algorithms without making explicit reference to the basis function type, whilst on the other hand still allowing the integral back end full control over the way integral data is produced and consumed. Thus the code describing the SCF algorithms has become independent from the code dealing with the discretisation details. On top of that a suitable integral abstraction layer has made implementing additional integral back ends or basis function types very easy. In this way a connection from `molsturm` to `libint` [257, 258] and `libcint` [259] for Gaussian-type integrals and to `sturmint` [170] for Coulomb-Sturmian-type integrals has been achieved.

The test suite and the testing strategy of `molsturm` were outlined in section 7.2.3. Together with the modularised design of the program this ensures that even if changes to the SCF scheme were needed in the future, code could be amended in steps and the correctness of `molsturm` verified in each of these steps.

The key aspects of the design of the `python` interface of `molsturm` were to enable full control over the algorithmic details via a detailed set of parameters on the one hand and to return computed SCF results in a readily usable data structure on the other. It was discussed how in this way many aspects of the SCF as well as the linear algebra back end, like the employed diagonalisation algorithms, can be altered directly from the interface and without changing any code. It was pointed out this is of significance when developing methods based on novel basis functions, since the best numerical approach might not be clear in the beginning. In such cases `molsturm` allows for experimenting directly from `python` scripts or even interactively.

By means of three examples, (1) fitting a H_2 dissociation curve in section 7.3.1, (2) implementing a coupled-cluster doubles on top of the SCF of `molsturm` in section 7.3.2 and (3) a gradient-free optimisation in section 7.3.3, the usefulness of the `python` interface for automating calculations, analysing results and implementing novel methods has been demonstrated. Furthermore the `python` interface has been used to establish links to selected methods from `pyscf` [236] and `adcman` [210], such that these may be used in combination with any of the basis function types and integral back ends implemented in `molsturm`. As a result one can think of `molsturm` as a mediator between integral libraries and Post-HF methods. The current features of `molsturm` were summarised in section 7.4.

In chapter 8 the link of `molsturm` to the Coulomb-Sturmian integral library `sturmint` was used in order to perform an initial investigation of the convergence properties of Coulomb-Sturmian-based quantum-chemical calculations. The main focus was on HF calculations of atoms of the second and third period of the periodic table. In section 8.2 a detailed analysis based on the root mean square values of the occupied coefficients per angular momentum (RMSO_l) allowed to suggest that a maximal angular momentum

quantum number of $l_{\max} = 0$ is sufficient for Li and Be, whereas $l_{\max} = 1$ is required for N, Ne, Na, Mg, P and Ar. It further allowed to understand that a known fundamental issue of the unrestricted HF procedure was responsible for the slow convergence observed for the atoms with one or two unpaired electrons. At correlated level some full configuration interaction and MP2 calculations were performed, which suggested that increasing the angular momentum quantum number by one is sufficient to capture most correlation effects for Li, Be, N, Ne, Na, Mg, P and Ar.

Furthermore the effect of modifying the Coulomb-Sturmian exponent on the resulting HF energies was analysed in section 8.4. Both an ansatz for estimating the optimal exponent k_{opt}, i.e. the exponent leading to the minimal energy, as well as an algorithm for finding the value of k_{opt} systematically, were developed in section 8.4.1. Following the relationship of Coulomb-Sturmians and Slater-type orbitals an analogy between the optimal exponent and the effective nuclear charge was indicated in section 8.4.2 and from this context the observed linear relationship of k_{opt} with the atomic number explained.

In section 8.5 the connection from `molsturm` to `adcman` via `python` was employed to perform the first excited states calculation based on the algebraic-diagrammatic construction scheme for the polarisation propagator. Initial results were reported, which looked promising and motivating for future research.

Chapter 10

Prospects and future work

> *Humanity needs practical men, who get the most out of their work, and, without forgetting the general good, safeguard their own interests. But humanity also needs dreamers, for whom the disinterested development of an enterprise is so captivating that it becomes impossible for them to devote their care to their own material profit.*
>
> — Marie Skłodowska Curie (1867–1934)

With the `molsturm` program package in its current state, a flexible research tool for the development and the investigation of novel quantum-chemical methods has become available. The possibility to easily extend the present functionality by linking to existing third-party packages, opens the door to rapidly test novel combinations of basis function types and existing quantum-chemical methods. In this way one could seek to find alternatives for cases, which are challenging to describe using Gaussian-type basis functions. Examples are the description of extended states, potentially embedded in the continuum, like autoionising or resonance states [61, 270–272] or the computation of properties involving a description of the wave function close to the nuclei, like nuclear magnetic resonance properties [8, 9].

From this respect interesting candidates are Sturmian-type basis functions, which are exponential-type orbitals, obtained as analytic solutions to Schrödinger-like equations. Such functions are able to properly represent the physical features of the wave function, see section 5.3.6, and they lead to feasible integrals in the Hartree-Fock (HF) self-consistent field (SCF) procedure. Furthermore they are complete and thus able to model continuum-like states as well. See section 10.2 for a more detailed discussion with respect to quantum-chemical calculations employing Sturmian-type orbitals.

Additionally, `molsturm` is a framework with an interface in which a novel method only needs to be implemented once and can subsequently be used with different types of basis functions. This was already discussed in section 7.3.2 where a coupled-cluster doubles (CCD) code building on top of `molsturm` was shown. Already at the present state, such a user code can directly utilise all basis function types available in `molsturm`, i.e. Coulomb-Sturmians as well as Gaussian-type orbitals, but the performance is far from optimal. Section 10.1 provides some direction how performance could be improved.

Another aspect of such a framework is that links to different libraries and programs providing the *same* quantum-chemical methods can be achieved. In this way implementations with deviating algorithmic details can be compared directly. One application of this would be to verify the correctness of integral back ends, see section 10.3 for details.

10.1 `molsturm` program package

After two years of development, `molsturm` is in a state, where calculations based on contracted Gaussian basis sets and Coulomb-Sturmian basis sets can be performed. Furthermore, as mentioned in section 7.4 and demonstrated in chapter 8, not only HF, but full configuration interaction (FCI) and methods based on the algebraic-diagrammatic construction scheme (ADC) are available via interfaces to `pyscf` [236] and `adcman` [210]. For employing these Post-HF methods with more than around a hundred basis functions, a drawback at the moment is performance inside the SCF procedure of `molsturm`.

One reason for this is that the SCF scheme, which is currently used in `molsturm`, only consists of a few rather simple algorithms, namely the Roothaan repeated diagonalisation [100], the direct inversion in the iterative subspace (DIIS) algorithm [99] and the truncated optimal damping algorithm (see section 5.4.4). More sophisticated schemes like the energy DIIS [201] or a second-order SCF scheme [204, 208] would be more efficient as well as more reliable. For this the published schemes need to be adapted, such that they fit into the contraction-based setting of `molsturm`, where the Fock matrix is not stored in memory, but only employed in the form of a matrix-vector product, see section 5.4. As discussed in section 5.1 such modifications are always possible in theory, but in practice one needs to be careful that the introduced changes keep the advantageous mathematical properties of such algorithms with respect to convergence and stability.

Another aspect for improvement is the lazy matrix library `lazyten`, which is a key component inside the contraction-based self-consistent field (SCF) of `molsturm`, see section 7.2. Whenever the Fock matrix is applied to a trial vector inside the SCF or the diagonalisation algorithm employed by the SCF, a contraction expression is evaluated. This proceeds by working on the expression tree, which represents the Fock matrix, see section 6.2. In `lazyten` this is currently neither parallelised, nor is the expression tree optimised before the computation begins. Both these aspects, i.e. automatic parallelisation of linear algebra expressions as well as finding optimal evaluation schemes, is ongoing research, where, both in the context of quantum-chemical calculations as well as a more general setting, enormous progress has been made in recent years [220–226, 273, 274]. By linking to such libraries these advances could be incorporated or reused inside `lazyten` leading to performance improvements in the SCF of `molsturm`.

Additionally the modular structure of `molsturm` makes it fast to interface to further quantum-chemistry libraries or program packages. An effort worth pursuing is the `libxc` [275] library, which offers a range of exchange-correlation functionals for density-functional theory. Implementing an interface to this library inside appropriate lazy matrix objects would allow to construct the Kohn-Sham matrix inside `molsturm`'s SCF without any further changes, such that density-functional theory calculations would become available. Furthermore a better link to `pyscf` [236] would allow to perform configuration-interaction and coupled-cluster calculations as well as calculations employing the density matrix renormalisation group approach directly from `molsturm`. The prospect is to try

other basis functions in the context of such methods and investigate their applicability with respect to quantum-chemical calculations.

10.2 Investigation of Sturmian-type discretisations

The original purpose of the `molsturm` package has been to devise a program which could be used for quantum-chemical calculations employing Sturmian-type basis functions. The package has outgrown this purpose in the current design, but investigating the properties of Sturmian-type basis functions is still of interest as discussed above.

10.2.1 Convergence properties of Coulomb-Sturmian basis sets

The simplest example for Sturmian-type basis functions are Coulomb-Sturmians. An initial investigation of such basis functions in chapter 8 looked overall promising, but the obtained results were not yet sufficient to provide definitive construction schemes for Coulomb-Sturmian (CS) basis sets or general estimates for the overall accuracy. There are three ways this could be improved.

Firstly so far only main group elements of the second and third period of the periodic table were considered. Originating from the involvement of the d-orbitals the properties of the electronic structures of the transition metals do, however, differ compared to the main group elements. An analysis with respect to the fourth period and beyond is therefore required to reach more general conclusions with respect to sensible CS basis sets.

Secondly, most of the presented investigation has concentrated on the HF level with some minor modifications suggested mostly based on second-order Møller-Plesset perturbation theory (MP2). Whilst these two methods are both used in electronic structure theory, more methods should be added to reach a representative set. Most notably MP2 as a perturbative approach for modelling electron correlation effects is very different from configuration-interaction-based or coupled-cluster-based approaches — both with respect to the way the physics is described as well as the numerics. Further convergence studies employing, for example, the latter kind of methods would be required.

Thirdly, the full flexibility towards constructing CS basis sets has not yet been exploited in the discussion in section 8.2. There is no reason why one should define a basis set by limiting the angular quantum numbers l and m to the same maximum for all principle quantum numbers n. As mentioned in section 8.1 a CS basis set can be any combination of the quantum number triples (n, l, m). Since alternative construction schemes might allow to reduce the required basis set size, they are worth considering. With respect to this it would also be interesting to compare the observed convergence properties with the typical construction schemes employed for contracted Gaussian (cGTO) basis sets [6, 7]. Potentially the schemes employed in the cGTO setting are applicable to CS basis sets and vice versa.

In general a detailed comparison of the results obtained from employing Coulomb-Sturmians as well as cGTO discretisations seems appropriate. An interesting question is, for example, the required basis sizes for both discretisation types to reach a certain accuracy in the description of the ground or excited-state energies at various levels of theory.

10.2.2 Coulomb-Sturmian-based excited states calculations

In section 8.5 first results for computing excited state energies of atoms based on ADC were already presented. A more detailed analysis of the convergence properties could help to proceed with the application of CS-based ADC calculations in order to compute the spectra of atoms. Due to the completeness of the Coulomb-Sturmians and their possibility to describe both core region as well as the exponential decay, a range of interesting applications for the modelling of excitation processes come to mind. Three of them are (1) methods where continuum-like states needs to be modelled, like Fano-Stieltjes [270–272], (2) cases where modelling both the core and the valence shell is required, like core-valence excitations [276, 277] as well as (3) the modelling of expanded bound states, like the determination of Rydberg-like states [61, 278].

10.2.3 Avoiding the Coulomb-Sturmian exponent as a parameter

An unfavourable aspect of the CS basis sets employed in chapter 8 is the Coulomb-Sturmian exponent k_{exp}. As was discussed in section 8.4, this parameter has indeed an influence on the results obtained. For example the obtained SCF minimum could be unphysical, i.e. with occupied orbitals of positive energies, if k_{exp} is not chosen in the vicinity of the optimal exponent, which is the one yielding the lowest possible HF ground state energy. Furthermore, there is some indication that the ordering of excited states in ADC calculations depends on k_{exp}. Using the algorithm described in section 8.4.1, a route for finding an optimal value at HF level has been sketched. With respect to excited states methods like ADC, it is not immediately obvious how to determine the most optimal exponent, since the value for describing the ground state with the least error will differ from the value for representing a particular excited state best. In turn each excited state will have a different k_{exp} to give the best description in a particular CS basis. Which value or which combination of the values should be taken is not directly clear.

An equivalent problem at FCI level can be avoided. The reason is that the relationship

$$E = -\frac{N_{\text{elec}} k_{\text{exp}}^2}{2},$$

between the CS exponent k_{exp}, the number of electrons N_{elec} and the energy E of a particular state, can be employed to re-formulate FCI in terms of the Coulomb-Sturmian exponents [29]. In other words, instead of solving for the energy of a state, one solves for the k_{exp} for each state and uses this value both to find the corresponding energy as well as the exponent of the basis functions, when properties for such a state are to be computed. A similar reformulation should be possible for HF and potentially even for some other Post-HF methods as well, even though this is uncertain at the moment. If this could be achieved both the determination of an optimal k_{exp} would become obsolete at HF level and for excited states k_{exp} would adapt automatically to the required state.

10.2.4 Molecular Sturmians

The implementation of a CS-based SCF scheme was always intended to be only the first step with respect to the exploration of Sturmian-based quantum-chemical calculations. More general and at the same time more challenging types of Sturmian basis functions exist, which are able to describe molecular systems, for example. Building on recent

advances in the calculation of the ERI integrals (4.31) for such generalised Sturmian-type orbitals [21, 29–34] a Sturmian-based HF suitable for molecular calculations is within reach and could be implemented within `sturmint` [170] building on already existing infrastructure required for the Coulomb-Sturmians. This would allow for performing calculations based on Sturmian-type basis functions for molecules as well.

10.3 Fuzzing of integral back ends

The common interface, which `molsturm` provides for accessing the implemented integral libraries, allows to test the correctness of the algorithms employed inside these libraries by comparing the results of random or semi-random input against each other. Due to the `python` interface of `molsturm` this process could even be completely automated. Such fuzzing approaches have already been applied with huge success in the context of hardening security-critical software [279]. With respect to quantum-chemical software a similar work by Knizia et al. [280], which tested the hardness of quantum-chemical software with respect to numerical instabilities using random noise, lead to the discovery of unexpected bugs in the integral evaluation scheme of `Molpro` [281], justifying a closer look at this subject.

Appendix A

Symmetry properties of the electron-repulsion integrals

In accordance with (4.31) of remark 4.9 on page 56 we define the ERIs in Mulliken notation as well as the physicist's indexing convention

$$\begin{aligned}(ij|kl) &= \int_\Omega \int_\Omega \psi_i^* \left(\underline{\boldsymbol{r}}_1\right) \psi_j^* \left(\underline{\boldsymbol{r}}_1\right) \frac{1}{r_{12}} \psi_k \left(\underline{\boldsymbol{r}}_2\right) \psi_l \left(\underline{\boldsymbol{r}}_2\right) \mathrm{d}\underline{\boldsymbol{r}}_1 \, \mathrm{d}\underline{\boldsymbol{r}}_2 \\ &= \langle ik|jl\rangle\end{aligned}$$

and the antisymmetrised repulsion integrals

$$\langle ij||kl\rangle = \langle ij|kl\rangle - \langle ji|kl\rangle = (ik|jl) - (jk|il)\,.$$

where $\psi_i, \psi_j, \psi_k, \psi_l \in H^1(\mathbb{R}^3, \mathbb{C})$. There is 4-fold symmetry in Mulliken convention

$$\begin{aligned}(ij|kl) &= (kl|ij) && \text{Swap shell pairs} && \text{(A.1)}\\ &= (ji|lk)^* && \text{Swap inside } \textit{both} \text{ shell pairs} && \text{(A.2)}\\ &= (lk|ji)^* && \text{Both the above}\end{aligned}$$

and 4-fold symmetry in physicist's convention as well

$$\begin{aligned}\langle ik|jl\rangle &= \langle ki|lj\rangle && \text{Swap shell pairs} && \text{(A.3)}\\ &= \langle jl|ik\rangle^* && \text{Swap inside } \textit{both} \text{ shell pairs} && \text{(A.4)}\\ &= \langle lj|ki\rangle^* && \text{Both the above.}\end{aligned}$$

The asymmetric integrals, however, have 8-fold symmetry

$$\begin{aligned}\langle ik||jl\rangle &= \langle ki||lj\rangle && \text{Swap shell pairs} && \text{(A.5)}\\ &= \langle jl||ik\rangle^* && \text{Swap inside } \textit{both} \text{ shell pairs} && \text{(A.6)}\\ &= \langle lj||ki\rangle^* && \text{Both the above}\\ &= -\langle ki||jl\rangle && \text{Antisymmetry} && \text{(A.7)}\\ &= -\langle ik||lj\rangle && \text{(A.7) and (A.5)}\\ &= -\langle lj||ik\rangle^* && \text{(A.7) and (A.6)}\\ &= -\langle jl||ki\rangle^* && \text{(A.7) and (A.6).}\end{aligned}$$

For **real-valued** functions, i.e. $\psi_i, \psi_j, \psi_k, \psi_l \in H^1(\mathbb{R}^3, \mathbb{R})$, the ERI tensor shows an 8-fold symmetry as well:

$$
\begin{aligned}
(ij|kl) &= (kl|ij) && \text{Swap shell pairs} && \text{(A.8)}\\
&= (ji|kl) && \text{Swap inside } \textit{first} \text{ shell pair} && \text{(A.9)}\\
&= (ij|lk) && \text{Swap inside } \textit{second} \text{ shell pair} && \text{(A.10)}\\
&= (ji|lk) && \text{Swap inside } \textit{both} \text{ shell pairs} && \text{(A.11)}\\
&= (lk|ij) && \text{(A.8) and (A.9)}\\
&= (kl|ji) && \text{(A.8) and (A.10)}\\
&= (lk|ji) && \text{(A.8) and (A.11)}
\end{aligned}
$$

Similarly for **real functions** and physicist's convention:

$$
\begin{aligned}
\langle ik|jl\rangle &= \langle ki|lj\rangle && \text{Swap shell pairs} && \text{(A.12)}\\
&= \langle jk|il\rangle && \text{Swap inside } \textit{first} \text{ shell pair} && \text{(A.13)}\\
&= \langle il|jk\rangle && \text{Swap inside } \textit{second} \text{ shell pair} && \text{(A.14)}\\
&= \langle jl|ik\rangle && \text{Swap inside } \textit{both} \text{ shell pairs} && \text{(A.15)}\\
&= \langle li|kj\rangle && \text{(A.12) and (A.13)}\\
&= \langle kj|li\rangle && \text{(A.12) and (A.14)}\\
&= \langle lj|ki\rangle && \text{(A.12) and (A.15)}
\end{aligned}
$$

For the antisymmetrised ERI tensor and **real functions** we *still* get only an 8-fold symmetry:

$$
\begin{aligned}
\langle ik||jl\rangle &= \langle ki||lj\rangle && \text{Swap shell pairs} && \text{(A.16)}\\
&= \langle jl||ik\rangle && \text{Swap inside } \textit{both} \text{ shell pairs} && \text{(A.17)}\\
&= \langle lj||ki\rangle && \text{Both the above}\\
&= -\langle ki||jl\rangle && \text{Antisymmetry} && \text{(A.18)}\\
&= -\langle ik||lj\rangle && \text{(A.18) and (A.16)}\\
&= -\langle lj||ik\rangle && \text{(A.18) and (A.17)}\\
&= -\langle jl||ki\rangle && \text{(A.18) and (A.17).}
\end{aligned}
$$

Appendix B

RMSO_l plots for Dunning basis sets

This appendix shows RMSO_l (see definition 8.1 on page 176) plots for the cGTO basis sets cc-pV5Z and cc-pV6Z [148, 152, 261–263] similar to the ones depicted for CS discretisations in section 8.2 on page 172. The values were computed `molsturm` [40, 233] using `libint` [257] as a back end for the integral values. The most notable differences compared to figures 8.2 and 8.3 on page 177 is that the RMSO_l of lithium and phosphorus decreases much slower and that beryllium has some pronounced spikes of larger RMSO_l values for $l = 2$ and $l = 4$.

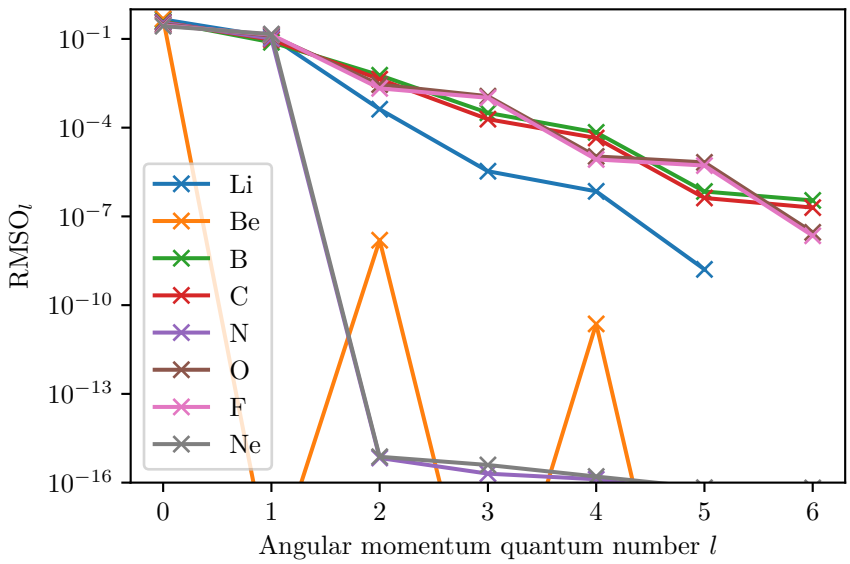


Figure B.1: Plot RMSO_l vs. l for the HF ground state of the atoms of the second period. All calculations done with the cc-pV6Z basis set, except Li and Be for which at cc-pV5Z was used. For Be and Ne a RHF procedure was used, for the other cases UHF.

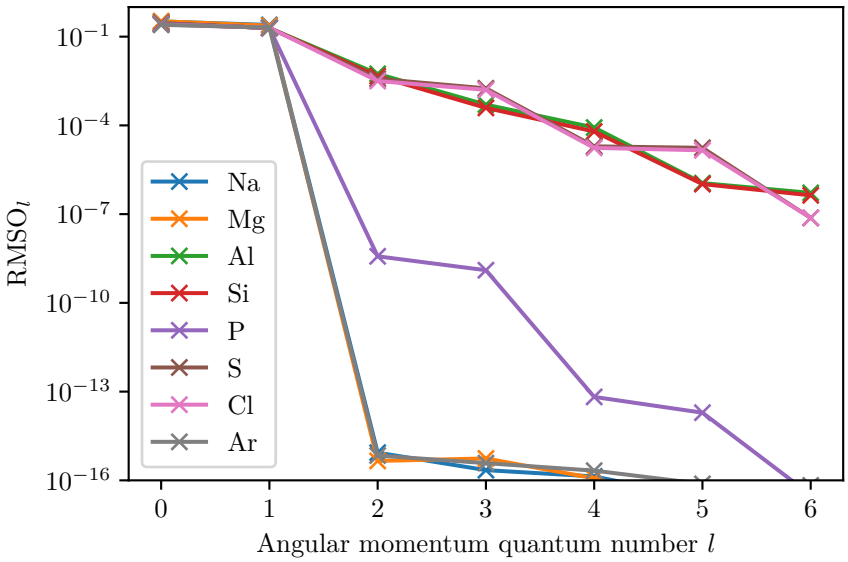


Figure B.2: Plot RMSO$_l$ vs. l for the HF ground state of the atoms of the third period. All calculations done with the cc-pV6Z basis set, except Na and Mg for which at cc-pV5Z was used. For Mg and Ar a RHF procedure was used, for the other cases UHF.

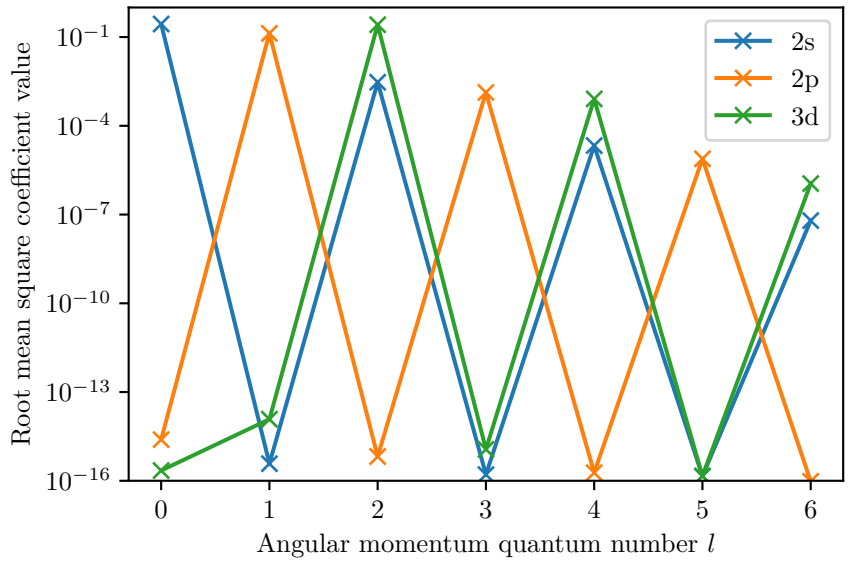


Figure B.3: Root mean square coefficient value per basis function angular momentum quantum number l for selected orbitals of oxygen. The atom is modelled in a cc-pV6Z basis using UHF.

Appendix C

Coulomb-Sturmian-based MP2 ground state energies

The tables presented in this appendix, show computed ground state energies for the atoms of the second and third period of the periodic table at Hartree-Fock and second-order Møller-Plesset perturbation theory level. For each atom only one CS exponent k_{exp}, but a range of CS basis sets is employed. Table C.1 continues on the next page.

atom		Coulomb-Sturmian basis set			
k_{exp}		$(6,0,0)$	$(6,1,0)$	$(6,1,1)$	$(6,2,1)$
	UHF	−7.42812	−7.42812	−7.42812	−7.42812
Li	UMP2	−7.44091	−7.44758	−7.46092	−7.46167
1.532	UMP2 corr.	−0.01278	−0.01945	−0.03280	−0.03355
	RHF	−14.56445	−14.56445	−14.56445	−14.56445
Be	MP2	−14.57906	−14.59427	−14.62469	−14.62780
1.988	MP2 corr.	−0.01461	−0.02983	−0.06025	−0.06335
	UHF			−54.36501	−54.36501
N	UMP2			−54.43401	−54.47133
3.287	UMP2 corr.			−0.06900	−0.10632
	RHF			−128.42549	−128.42549
Ne	MP2			−128.59497	−128.65853
4.512	MP2 corr.			−0.16948	−0.23304

Table C.1: Hartree-Fock and MP2 ground state energies as well as MP2 correlation energies (MP2 corr.) for the atoms of the 2nd period and a range of CS basis sets.

atom		Coulomb-Sturmian basis set				
k_{exp}		$(6,2,2)$	$(6,3,2)$	$(6,3,3)$	$(6,4,3)$	$(6,4,4)$
	UHF	−7.42812	−7.42812	−7.42812		
Li	UMP2	−7.46217	−7.46220	−7.46222		
1.532	UMP2 corr.	−0.03404	−0.03408	−0.03410		
	RHF	−14.56445	−14.56445	−14.56445		
Be	MP2	−14.62987	−14.63075	−14.63110		
1.988	MP2 corr.	−0.06542	−0.06630	−0.06666		
	UHF	−24.51614	−24.51616	−24.51616	−24.51616	−24.51616
B	UMP2	−24.59788	−24.60123	−24.60172	−24.60273	−24.60285
2.409	UMP2 corr.	−0.08174	−0.08508	−0.08557	−0.08657	−0.08669
	UHF	−37.66382	−37.66383	−37.66383	−37.66383	−37.66383
C	UMP2	−37.77146	−37.77667	−37.77911	−37.78066	−37.78116
2.849	UMP2 corr.	−0.10764	−0.11285	−0.11529	−0.11683	−0.11733
	UHF	−54.36501	−54.36501	−54.36501		
N	UMP2	−54.49917	−54.50751	−54.51100		
3.287	UMP2 corr.	−0.13416	−0.14250	−0.14599		
	UHF	−74.75348	−74.75421	−74.75504	−74.75504	−74.75504
O	UMP2	−74.93218	−74.94613	−74.95399	−74.95816	−74.95979
3.685	UMP2 corr.	−0.17870	−0.19192	−0.19895	−0.20312	−0.20474
	UHF	−99.32407	−99.32478	−99.32560	−99.32560	−99.32560
F	UMP2	−99.55722	−99.57831	−99.58688	−99.59396	−99.59566
4.099	UMP2 corr.	−0.23316	−0.25353	−0.26128	−0.26836	−0.27006
	RHF	−128.42549	−128.42549	−128.42549		
Ne	MP2	−128.72163	−128.74585	−128.75676		
4.512	MP2 corr.	−0.29614	−0.32036	−0.33127		

Table C.1: Hartree-Fock and MP2 ground state energies as well as MP2 correlation energies (MP2 corr.) for the atoms of the 2nd period and a range of CS basis sets. (continued)

atom		Coulomb-Sturmian basis set						
k_{exp}		$(6,1,1)$	$(6,2,1)$	$(6,2,2)$	$(6,3,2)$	$(6,3,3)$	$(6,4,3)$	$(6,4,4)$
	UHF	−160.92907	−160.92907	−160.92907	−160.92907	−160.92907		
Na	UMP2	−161.09552	−161.17343	−161.24656	−161.27291	−161.28465		
4.289	UMP2 corr.	−0.16645	−0.24436	−0.31749	−0.34385	−0.35558		
	RHF	−198.02758	−198.02758	−198.02758	−198.02758	−198.02758		
Mg	MP2	−198.21545	−198.30812	−198.39191	−198.42111	−198.43398		
4.442	MP2 corr.	−0.18786	−0.28054	−0.36433	−0.39353	−0.40640		
	UHF			−239.51378	−239.51413	−239.51413	−239.51414	−239.51414
Al	UMP2			−239.88342	−239.91818	−239.93094	−239.94018	−239.94268
4.649	UMP2 corr.			−0.36963	−0.40404	−0.41681	−0.42605	−0.42855
	UHF			−285.77553	−285.77583	−285.77583	−285.77584	−285.77584
Si	UMP2			−286.16187	−286.19764	−286.21565	−286.22535	−286.22841
4.904	UMP2 corr.			−0.38634	−0.42180	−0.43982	−0.44951	−0.45258
	UHF	−336.94639	−336.94639	−336.94639	−336.94639	−336.94639		
P	UMP2	−337.08603	−337.22801	−337.35742	−337.39998	−337.41864		
5.186	UMP2 corr.	−0.13965	−0.28162	−0.41103	−0.45359	−0.47225		
	UHF			−392.96867	−392.97038	−392.97235	−392.97235	−392.97235
S	UMP2			−393.40463	−393.45640	−393.48253	−393.49790	−393.50147
5.476	UMP2 corr.			−0.43596	−0.48602	−0.51018	−0.52554	−0.52912
	UHF			−454.17153	−454.17321	−454.17514	−454.17514	−454.17514
Cl	UMP2			−454.64351	−454.70610	−454.73332	−454.75087	−454.75775
5.776	UMP2 corr.			−0.47198	−0.53289	−0.55819	−0.57574	−0.58261
	RHF	−520.71255	−520.71255	−520.71255	−520.71255	−520.71255		
Ar	MP2	−520.85353	−521.03471	−521.22884	−521.29531	−521.32548		
6.084	MP2 corr.	−0.14099	−0.32217	−0.51630	−0.58276	−0.61293		

Table C.2: Hartree-Fock and MP2 ground state energies as well as MP2 correlation energies (MP2 corr.) for the atoms of the 3rd period and a range of CS basis sets.

Bibliography

[1] P. A. M. Dirac. *Quantum mechanics of many-electron systems.* Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, **123**, 714 (1929).

[2] S. F. Boys. *Electronic wave functions - I. A general method of calculation for the stationary states of any molecular system.* Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, **200**, 542 (1950).

[3] J. C. Slater. *Atomic Shielding Constants.* Physical Review, **36**, 57 (1930).

[4] W. J. Hehre, R. F. Stewart and J. A. Pople. *Self-Consistent Molecular-Orbital Methods. I. Use of Gaussian Expansions of Slater-Type Atomic Orbitals.* The Journal of Chemical Physics, **51**, 2657 (1969).

[5] T. Kato. *On the eigenfunctions of many-particle systems in quantum mechanics.* Communications on Pure and Applied Mathematics, **10**, 151 (1957).

[6] F. Jensen. *Atomic orbital basis sets.* Wiley Interdisciplinary Reviews: Computational Molecular Science, **3**, 273 (2013).

[7] J. G. Hill. *Gaussian basis sets for molecular applications.* International Journal of Quantum Chemistry, **113**, 21 (2013).

[8] M. Güell, J. M. Luis, M. Solà and M. Swart. *Importance of the Basis Set for the Spin-State Energetics of Iron Complexes.* The Journal of Physical Chemistry A, **112**, 6384 (2008).

[9] P. E. Hoggan. *How Exponential Type Orbitals Recently Became a Viable Basis Set Choice in Molecular Electronic Structure Work and When to Use Them*, 199–219. Springer-Verlag, Dordrecht (2009).

[10] L. Frediani and D. Sundholm. *Real-space numerical grid methods in quantum chemistry.* Physical Chemistry Chemical Physics, **17**, 31357 (2015).

[11] F. A. Bischoff and E. F. Valeev. *Low-order tensor approximations for electronic wave functions: Hartree–Fock method with guaranteed precision.* The Journal of Chemical Physics, **134**, 104104 (2011).

[12] F. A. Bischoff, R. J. Harrison and E. F. Valeev. *Computing many-body wave functions with guaranteed precision: The first-order Møller-Plesset wave function for the ground state of helium atom.* The Journal of Chemical Physics, **137**, 104103 (2012).

[13] F. A. Bischoff and E. F. Valeev. *Computing molecular correlation energies with*

guaranteed precision. The Journal of Chemical Physics, **139**, 114106 (2013).

[14] F. A. Bischoff. *Regularizing the molecular potential in electronic structure calculations. II. Many-body methods.* The Journal of Chemical Physics, **141**, 184106 (2014).

[15] F. A. Bischoff. *Regularizing the molecular potential in electronic structure calculations. I. SCF methods.* The Journal of Chemical Physics, **141**, 184105 (2014).

[16] F. A. Bischoff. *Analytic second nuclear derivatives of Hartree-Fock and DFT using multi-resolution analysis.* The Journal of Chemical Physics, **146**, 124126 (2017).

[17] E. Tsuchida and M. Tsukada. *Electronic-structure calculations based on the finite-element method.* Physical Review B, **52**, 5573 (1995).

[18] J. M. Soler, E. Artacho, J. D. Gale, A. García, J. Junquera, P. Ordejón and D. Sánchez-Portal. *The SIESTA method for ab initio order-N materials simulation.* Journal of Physics: Condensed Matter, **14**, 2745 (2002).

[19] L. Lehtovaara, V. Havu and M. Puska. *All-electron density functional theory and time-dependent density functional theory with high-order finite elements.* The Journal of Chemical Physics, **131**, 054103 (2009).

[20] R. Alizadegan, K. J. Hsia and T. J. Martínez. *A divide and conquer real space finite-element Hartree–Fock method.* The Journal of Chemical Physics, **132**, 034101 (2010).

[21] J. E. Avery. *New Computational Methods in the Quantum Theory of Nano-Structures.* Ph.D. thesis, University of Copenhagen (2011).

[22] D. Davydov, T. D. Young and P. Steinmann. *On the adaptive finite element analysis of the Kohn–Sham equations: methods, algorithms, and implementation.* International Journal for Numerical Methods in Engineering, **106**, 863 (2015).

[23] N. M. Boffi, M. Jain and A. Natan. *Efficient Computation of the Hartree–Fock Exchange in Real-Space with Projection Operators.* Journal of Chemical Theory and Computation, **12**, 3614 (2016).

[24] H. Shull and P.-O. Löwdin. *Superposition of Configurations and Natural Spin Orbitals. Applications to the He Problem.* The Journal of Chemical Physics, **30**, 617 (1959).

[25] M. Rotenberg. *Application of sturmian functions to the Schroedinger three-body problem: Elastic e^+-H scattering.* Annals of Physics, **19**, 262 (1962).

[26] M. Rotenberg. *Theory and Application of Sturmian Functions.* vol. 6 of *Advances in Atomic and Molecular Physics*, 233 – 268. Academic Press (1970).

[27] P. F. Gruzdev, G. S. Solov'eva and A. I. Sherstyuk. *Calculation of the coupling constants of many-electron atoms with external fields using expansions in a discrete basis of sturmian-type virtual orbitals.* Soviet Physics Journal, **33**, 685 (1990).

[28] J. M. Randazzo, L. U. Ancarani, G. Gasaneo, A. L. Frapiccini and F. D. Colavecchia. *Generating optimal Sturmian basis functions for atomic problems.* Physical Review A, **81**, 042520 (2010).

[29] J. E. Avery and J. S. Avery. *Generalized Sturmians and Atomic Spectra.* World Scientific (2006).

[30] J. S. Avery, S. Rettrup and J. E. Avery. *Symmetry-Adapted Basis Sets: Automatic Generation for Problems in Chemistry and Physics*. World Scientific (2011).

[31] D. A. Morales. *On the evaluation of integrals with Coulomb Sturmian radial functions*. Journal of Mathematical Chemistry, **54**, 682 (2016).

[32] J. E. Avery and J. S. Avery. *4-Center STO Interelectron Repulsion Integrals With Coulomb Sturmians*. Advances in Quantum Chemistry. Academic Press (2017).

[33] J. M. Randazzo, D. Mitnik, G. Gasaneo, L. U. Ancarani and F. D. Colavecchia. *Double photoionization of helium: a generalized Sturmian approach*. The European Physical Journal D, **69**, 189 (2015).

[34] C. M. Granados-Castro, L. U. Ancarani, G. Gasaneo and D. M. Mitnik. *A Sturmian Approach to Photoionization of Molecules*. In P. E. Hoggan and T. Ozdogan (Eds.), *Electron Correlation in Molecules – ab initio Beyond Gaussian Quantum Chemistry*, vol. 73 of *Advances in Quantum Chemistry*, 3 – 57. Academic Press (2016).

[35] L. Kong, F. A. Bischoff and E. F. Valeev. *Explicitly Correlated R12/F12 Methods for Electronic Structure*. Chemical Reviews, **112**, 75 (2012).

[36] G. Kresse and J. Furthmüller. *Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set*. Physical Review B, **54**, 11169 (1996).

[37] G. Kresse and D. Joubert. *From ultrasoft pseudopotentials to the projector augmented-wave method*. Physical Review B, **59**, 1758 (1999).

[38] J. J. Mortensen, L. B. Hansen and K. W. Jacobsen. *Real-space grid implementation of the projector augmented wave method*. Physical Review B, **71**, 035109 (2005).

[39] J. Enkovaara, C. Rostgaard, J. J. Mortensen, J. Chen, M. Dułak, L. Ferrighi, J. Gavnholt, C. Glinsvad, V. Haikola, H. A. Hansen, H. H. Kristoffersen, M. Kuisma, A. H. Larsen, L. Lehtovaara, M. Ljungberg, O. Lopez-Acevedo, P. G. Moses, J. Ojanen, T. Olsen, V. Petzold, N. A. Romero, J. Stausholm-Møller, M. Strange, G. A. Tritsaris, M. Vanin, M. Walter, B. Hammer, H. Häkkinen, G. K. H. Madsen, R. M. Nieminen, J. K. Nørskov, M. Puska, T. T. Rantala, J. Schiøtz, K. S. Thygesen and K. W. Jacobsen. *Electronic structure calculations with GPAW: a real-space implementation of the projector augmented-wave method*. Journal of Physics: Condensed Matter, **22**, 253202 (2010).

[40] M. F. Herbst and J. E. Avery. *A modular electronic structure theory code*. `https://molsturm.org`. Accessed on 13th March 2019.

[41] R. Shankar. *Principles of Quantum Mechanics*. Springer-Verlag (1994).

[42] V. F. Müller. *Quantenmechanik*. Oldenbourg (2000).

[43] B. Helffer. *Spectral Theory and its Applications*. Cambridge University Press (2013).

[44] J. v. Neumann. *Die Eindeutigkeit der Schrödingerschen Operatoren*. Mathematische Annalen, **104**, 570 (1931).

[45] J. v. Neumann. *Über Einen Satz Von Herrn M. H. Stone*. Annals of Mathematics, **33**, 567 (1932).

[46] M. H. Stone. *On One-Parameter Unitary Groups in Hilbert Space*. Annals of Mathematics, **33**, 643 (1932).

[47] M. Quack and F. Merkt (Eds.). *Handbook of High-resolution Spectroscopy*. John Wiley & Sons (2011).

[48] P. J. Mohr, D. B. Newell and B. N. Taylor. *CODATA Recommended Values of the Fundamental Physical Constants: 2014* (2015).

[49] M. F. Herbst. *The Mathematical Concept of Dirac Notation*. `https://michael-herbst.com/talks/2014.07.22_Mathematical_Concept_Dirac_Notation.pdf` (2014).

[50] R. A. Adams. *Sobolev spaces*. Academic Press, Amsterdam Boston (2003).

[51] D. Werner. *Funktionalanalysis*. Springer-Verlag, 7. edn. (2011).

[52] Y. Last. *Quantum Dynamics and Decompositions of Singular Continuous Spectra*. Journal of Functional Analysis, **142**, 406 (1996).

[53] E. B. Davies. *Linear Operators and their Spectra*. Cambridge University Press (2007).

[54] G. Teschl. *Mathematical Methods in Quantum Mechanics — With Applications to Schrödinger Operators*, vol. 157 of *Graduate Studies in Mathematics*. American Mathematical Society, 2nd edn. (2014).

[55] G. Zhislin. *A characteristic of the spectrum of Schrödinger's operator for systems of molecular type.* Doklady Akademii Nauk SSSR, **128**, 231 (1959).

[56] G. M. Zhislin. *A study of the spectrum of the Schrödinger operator for a system of several particles*. Trudy Moskovskogo Matematicheskogo Obshchestva, **9**, 81 (1960).

[57] M. Reed and B. Simon. *Methods of Modern Mathematical Physics IV: Analysis of Operators*, vol. 4 of *Methods of Modern Mathematical Physics*. Academic Press (1978).

[58] M. Reed and B. Simon. *Methods of Modern Mathematical Physics I: Functional Analysis*, vol. 1 of *Methods of Modern Mathematical Physics*. Academic Press, revised and enlarged edn. (1980).

[59] Z. Bacic and J. Simons. *Resonance energies and lifetimes from stabilization-based methods*. The Journal of Physical Chemistry, **86**, 1192 (1982).

[60] A. U. Hazi and H. S. Taylor. *Stabilization Method of Calculating Resonance Energies: Model Problem*. Physical Review A, **1**, 1109 (1970).

[61] U. V. Riss and H.-D. Meyer. *Calculation of resonance energies and widths using the complex absorbing potential method*. Journal of Physics B, **26**, 4503 (1993).

[62] P. Arbenz. *Lecture Notes on Solving Large Scale Eigenvalue Problems*. Lecture Notes (ETH Zürich) (2010). Available from `https://people.inf.ethz.ch/arbenz/ewp/Lnotes/lsevp.pdf`.

[63] Y. Saad. *Numerical methods for large eigenvalue problems*. SIAM Publishing, 2nd edn. (2011).

[64] W. E. Arnoldi. *The principle of minimized iterations in the solution of the matrix eigenvalue problem*. Quarterly of Applied Mathematics, **9**, 17 (1951).

[65] C. Lanczos. *An iteration method for the solution of the eigenvalue problem of linear*

differential and integral operators. Journal of Research of the National Bureau of Standards, **45**, 255 (1950).

[66] E. R. Davidson. *The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices.* Journal of Computational Physics, **17**, 87 (1975).

[67] Y. Saad. *Iterative Methods for Sparse Linear Systems.* SIAM Publishing, 2nd edn. (2003).

[68] C. Großmann and H.-G. Roos. *Numerik partieller Differentialgleichungen.* Teubner Studienbücher Mathematik. Vieweg+Teubner Verlag, 2nd edn. (1992).

[69] T. Kato. *Fundamental properties of Hamiltonian operators of Schrödinger type.* Transactions of the American Mathematical Society, **70**, 195 (1951).

[70] E. A. Hylleraas. *Neue Berechnung der Energie des Heliums im Grundzustande, sowie des tiefsten Terms von Ortho-Helium.* Zeitschrift für Physik, **54**, 347 (1929).

[71] M. Baer. *Beyond Born-Oppenheimer: Electronic Nonadiabatic Coupling Terms and Conical Intersections.* John Wiley & Sons (2006).

[72] *Wikipedia on Born-Oppenheimer approximation.* `https://en.wikipedia.org/wiki/Born%E2%80%93Oppenheimer_approximation` (2018). Accessed on 10th January 2018.

[73] M. Born and R. Oppenheimer. *Zur Quantentheorie der Molekeln.* Annalen der Physik, **389**, 457 (1927).

[74] W. Gerlach and O. Stern. *Der experimentelle Nachweis des magnetischen Moments des Silberatoms.* Zeitschrift für Physik, **8**, 110 (1922).

[75] W. Gerlach and O. Stern. *Der experimentelle Nachweis der Richtungsquantelung im Magnetfeld.* Zeitschrift für Physik, **9**, 349 (1922).

[76] W. Gerlach and O. Stern. *Das magnetische Moment des Silberatoms.* Zeitschrift für Physik, **9**, 353 (1922).

[77] W. Pauli. *Über den Zusammenhang des Abschlusses der Elektronengruppen im Atom mit der Komplexstruktur der Spektren.* Zeitschrift für Physik, **31**, 765 (1925).

[78] P. A. M. Dirac. *On the theory of quantum mechanics.* Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, **112**, 661 (1926).

[79] N. Straumann. *The Role of the Exclusion Principle for Atoms to Stars: A Historical Account.*

[80] V. Fock. *Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems.* Zeitschrift für Physik, **61**, 126 (1930).

[81] J. C. Slater. *The Theory of Complex Spectra.* Physical Review, **34**, 1293 (1929).

[82] J. C. Slater. *Note on Hartree's Method.* Physical Review, **35**, 210 (1930).

[83] A. Szabo and N. S. Ostlund. *Modern Quantum Chemistry.* Dover Publications, 1st edn. (1996).

[84] T. Helgaker, J. Olsen and P. Jørgensen. *Molecular Electronic-Structure Theory.* John Wiley & Sons, 1st edn. (2013).

[85] G. M. Zhislin. *On the nodes of the eigenfunctions of the Schrödinger operator.* Uspekhi Matematicheskikh Nauk, **16**, 149 (1961).

[86] D. R. Yafaev. *On the point spectrum in the quantum-mechanical many-body problem.* Mathematics of the USSR-Izvestiya, **10**, 861 (1976).

[87] A. Schäfer, H. Horn and R. Ahlrichs. *Fully optimized contracted Gaussian basis sets for atoms Li to Kr.* The Journal of Chemical Physics, **97**, 2571 (1992).

[88] E. Rossi, G. L. Bendazzoli, S. Evangelisti and D. Maynau. *A full-configuration benchmark for the N2 molecule.* Chemical Physics Letters, **310**, 530 (1999).

[89] L. K. Sørensen, S. Bauch and L. B. Madsen. *The Integral Screened Configuration Interaction Method.*

[90] D. R. Hartree. *The Wave Mechanics of an Atom with a Non-Coulomb Central Field. Part I. Theory and Methods.* Mathematical Proceedings of the Cambridge Philosophical Society, **24**, 89–110 (1928).

[91] H. Fukutome. *Unrestricted Hartree–Fock theory and its applications to molecules and chemical reactions.* International Journal of Quantum Chemistry, **20**, 955 (1981).

[92] R. McWeeny and B. Sutcliffe. *Fundamentals of Self-Consistent-Field (SCF), Hartree-Fock (HF), Multi-Configuration (MC)SCF and Configuration Interaction (CI) schemes.* Computer Physics Reports, **2**, 219 (1985).

[93] E. H. Lieb and B. Simon. *The Hartree-Fock theory for Coulomb systems.* Communications in Mathematical Physics, **53**, 185 (1977).

[94] P.-L. Lions. *Solutions of Hartree-Fock equations for Coulomb systems.* Communications in Mathematical Physics, **109**, 33 (1987).

[95] P. Lions. *On Hartree and Hartree-Fock equations in atomic and nuclear physics.* Computer Methods in Applied Mechanics and Engineering, **75**, 53 (1989).

[96] V. Bach, E. H. Lieb, M. Loss and J. P. Solovej. *There are no unfilled shells in unrestricted Hartree-Fock theory.* Physical Review Letters, **72**, 2981 (1994).

[97] B. Sutcliffe, E. Cancès, M. Caffarel, R. Assaraf, G. Turinici, I. Catto, P.-L. Lions, C. L. Bris, O. Bokanowski, B. Grébert, N. J. Mauser, X. Blanc, M. Defranceschi, V. Louis-Achille, B. Mennucci, J. Dolbeault, M. J. Esteban, E. Séré, T. Saue and H. J. A. Jensen. *Mathematical Models and Methods for Ab Initio Quantum Chemistry*, vol. 74 of *Lecture Notes in Chemistry.* Springer-Verlag (2000).

[98] J. A. Pople and R. K. Nesbet. *Self-Consistent Orbitals for Radicals.* The Journal of Chemical Physics, **22**, 571 (1954).

[99] P. Pulay. *Improved SCF convergence acceleration.* Journal of Computational Chemistry, **3**, 556 (1982).

[100] C. C. J. Roothaan. *New Developments in Molecular Orbital Theory.* Reviews of Modern Physics, **23**, 69 (1951).

[101] C. C. J. Roothaan. *Self-Consistent Field Theory for Open Shells of Electronic Systems.* Reviews of Modern Physics, **32**, 179 (1960).

[102] B. J. Alder. *Methods in Computational Physics: Advances in Research and Applications.* Academic Press. (1963).

[103] K. R. Glaesemann and M. W. Schmidt. *On the Ordering of Orbital Energies in High-Spin ROHF.* The Journal of Physical Chemistry A, **114**, 8772 (2010).

[104] F. Jensen. *Introduction to Computational Chemistry.* John Wiley & Sons (2007).

[105] C. F. Fischer. *Self-consistent-field (SCF) and multiconfiguration (MC) Hartree-Fock (HF) methods in atomic calculations: Numerical integration approaches.* Computer Physics Reports, **3**, 274 (1986).

[106] C. D. Sherrill and H. F. Schaefer. *The Configuration Interaction Method: Advances in Highly Correlated Approaches.* vol. 34 of *Advances in Quantum Chemistry*, 143 – 269. Academic Press (1999).

[107] K. R. Shamasundar, G. Knizia and H.-J. Werner. *A new internally contracted multi-reference configuration interaction method.* The Journal of Chemical Physics, **135**, 054101 (2011).

[108] C. Møller and M. S. Plesset. *Note on an Approximation Treatment for Many-Electron Systems.* Physical Review, **46**, 618 (1934).

[109] R. Zalesny, M. G. Papadopoulos, P. G. Mezey and J. Leszczynski (Eds.). *Linear-Scaling Techniques in Computational Chemistry and Physics.* Springer-Verlag (2011).

[110] T. D. Crawford and H. F. Schaefer. *An Introduction to Coupled Cluster Theory for Computational Chemists*, 33–136. John Wiley & Sons (2007).

[111] M. Hodecker. *Employing the Coupled Cluster Doubles Ground State within the Algebraic Diagrammatic Construction Scheme for the Polarisation Propagator.* Master's thesis, Universität Heidelberg (2016).

[112] A. C. Hurley. *Electron correlation in small molecules.* Academic Press, London; New York (1976).

[113] R. J. Bartlett and G. D. Purvis. *Many-body perturbation theory, coupled-pair many-electron theory, and the importance of quadruple excitations for the correlation problem.* International Journal of Quantum Chemistry, **14**, 561 (1978).

[114] C. Riplinger, B. Sandhoefer, A. Hansen and F. Neese. *Natural triple excitations in local coupled cluster calculations with pair natural orbitals.* The Journal of Chemical Physics, **139**, 134101 (2013).

[115] A. Dreuw and M. Head-Gordon. *Single-Reference ab Initio Methods for the Calculation of Excited States of Large Molecules.* Chemical Reviews, **105**, 4009 (2005).

[116] J. Schirmer and F. Mertins. *Review of biorthogonal coupled cluster representations for electronic excitation.* Theoretical Chemistry Accounts, **125**, 145 (2010).

[117] H. Sekino and R. J. Bartlett. *A linear response, coupled-cluster theory for excitation energy.* International Journal of Quantum Chemistry, **26**, 255 (1984).

[118] J. Schirmer. *Beyond the random-phase approximation: A new approximation scheme for the polarization propagator.* Physical Review A, **26**, 2395 (1982).

[119] A. B. Trofimov, G. Stelter and J. Schirmer. *A consistent third-order propagator method for electronic excitation.* The Journal of Chemical Physics, **111**, 9982 (1999).

[120] P. Hohenberg and W. Kohn. *Inhomogeneous Electron Gas.* Physical Review, **136**,

B864 (1964).

[121] M. Levy. *Universal variational functionals of electron densities, first-order density matrices, and natural spin-orbitals and solution of the v-representability problem.* Proceedings of the National Academy of Sciences, **76**, 6062 (1979).

[122] W. Kohn and L. J. Sham. *Self-Consistent Equations Including Exchange and Correlation Effects.* Physical Review, **140**, A1133 (1965).

[123] T. Tsuneda and K. Hirao. *Long-range correction for density functional theory.* Wiley Interdisciplinary Reviews: Computational Molecular Science, **4**, 375 (2014).

[124] S. Grimme. *Density functional theory with London dispersion corrections.* Wiley Interdisciplinary Reviews: Computational Molecular Science, **1**, 211 (2011).

[125] J. P. Perdew, A. Ruzsinszky, J. Tao, V. N. Staroverov, G. E. Scuseria and G. I. Csonka. *Prescription for the design and selection of density functional approximations: More constraint satisfaction with fewer fits.* The Journal of Chemical Physics, **123**, 062201 (2005).

[126] J. P. Perdew, K. Burke and M. Ernzerhof. *Generalized Gradient Approximation Made Simple.* Physical Review Letters, **77**, 3865 (1996).

[127] A. D. Becke. *Density-functional thermochemistry. III. The role of exact exchange.* The Journal of Chemical Physics, **98**, 5648 (1993).

[128] C. Lee, W. Yang and R. G. Parr. *Development of the Colle-Salvetti correlation-energy formula into a functional of the electron density.* Physical Review B, **37**, 785 (1988).

[129] R. Polly, H.-J. Werner, F. R. Manby and P. J. Knowles. *Fast Hartree–Fock theory using local density fitting approximations.* Molecular Physics, **102**, 2311 (2004).

[130] F. Neese. *The ORCA program system.* Wiley Interdisciplinary Reviews: Computational Molecular Science, **2**, 73 (2012).

[131] R. Hoffmann. *An Extended Hückel Theory. I. Hydrocarbons.* The Journal of Chemical Physics, **39**, 1397 (1963).

[132] M. Wolfsberg and L. Helmholz. *The Spectra and Electronic Structure of the Tetrahedral Ions MnO_4^-, CrO_4^{2-}, and ClO_4^-.* The Journal of Chemical Physics, **20**, 837 (1952).

[133] G. Landrum. *YAeHMOP: Yet Another Extended Hückel Molecular Orbital Package.* `https://github.com/greglandrum/yaehmop` (2018). Accessed on 23th January 2018.

[134] M. Lee, K. Leiter, C. Eisner, J. Crone and J. Knap. *Extended Hückel and Slater's rule initial guess for real space grid-based density functional theory.* Computational and Theoretical Chemistry, **1062**, 24 (2015).

[135] J. H. van Lenthe, R. Zwaans, H. J. J. Van Dam and M. F. Guest. *Starting SCF calculations by superposition of atomic densities.* Journal of Computational Chemistry, **27**, 926 (2006).

[136] W. M. C. Foulkes, L. Mitas, R. J. Needs and G. Rajagopal. *Quantum Monte Carlo simulations of solids.* Reviews of Modern Physics, **73**, 33 (2001).

[137] A. Ma, M. D. Towler, N. D. Drummond and R. J. Needs. *Scheme for adding*

electron–nucleus cusps to Gaussian orbitals. The Journal of Chemical Physics, **122**, 224322 (2005).

[138] J. E. Avery and J. S. Avery. *Hyperspherical Harmonics and Their Physical Applications.* World Scientific (2018).

[139] R. Shepard and M. Minkoff. *Some comments on the DIIS method.* Molecular Physics, **105**, 2839 (2007).

[140] P. Hoggan, M. B. Ruiz Ruiz and T. Ozdogan. *Molecular Integrals over Slater-type Orbitals. From pioneers to recent progress.* Quantum Frontiers of Atoms and Molecules in Physics, Chemistry, and Biology, 63–90 (2011).

[141] E. J. Weniger and E. O. Steinborn. *The Fourier transforms of some exponential-type basis functions and their relevance to multicenter problems.* The Journal of Chemical Physics, **78**, 6121 (1983).

[142] J. E. Avery. *Fast Electron Repulsion Integrals for Molecular Coulomb Sturmians.* In P. E. Hoggan (Ed.), *Exponential Type Orbitals for Molecular Electronic Structure Theory*, vol. 67 of *Advances in Quantum Chemistry*, 129 – 151. Academic Press (2013).

[143] A. Bouferguene, M. Fares and P. E. Hoggan. *STOP: A slater-type orbital package for molecular electronic structure determination.* International Journal of Quantum Chemistry, **57**, 801 (1996).

[144] J. Fernández Rico, R. López, A. Aguado, I. Ema and G. Ramírez. *New program for molecular calculations with Slater-type orbitals.* International Journal of Quantum Chemistry, **81**, 148 (2001).

[145] G. te Velde, F. M. Bickelhaupt, E. J. Baerends, C. Fonseca Guerra, S. J. A. van Gisbergen, J. G. Snijders and T. Ziegler. *Chemistry with ADF.* Journal of Computational Chemistry, **22**, 931 (2001).

[146] E. Besalú and R. Carbó-Dorca. *The general Gaussian product theorem.* Journal of Mathematical Chemistry, **49**, 1769 (2011).

[147] P. M. Gill. *Molecular integrals Over Gaussian Basis Functions.* vol. 25 of *Advances in Quantum Chemistry*, 141 – 205. Academic Press (1994).

[148] T. H. Dunning. *Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen.* The Journal of Chemical Physics, **90**, 1007 (1989).

[149] F. Jensen. *Estimating the Hartree—Fock limit from finite basis set calculations.* Theoretical Chemistry Accounts, **113**, 267 (2005).

[150] M. Bachmayr, H. Chen and R. Schneider. *Error estimates for Hermite and even-tempered Gaussian approximations in quantum chemistry.* Numerische Mathematik, **128**, 137 (2014).

[151] F. Jensen. *Polarization consistent basis sets: Principles.* The Journal of Chemical Physics, **115**, 9113 (2001).

[152] A. K. Wilson, T. van Mourik and T. H. Dunning. *Gaussian basis sets for use in correlated molecular calculations. VI. Sextuple zeta correlation consistent basis sets for boron through neon.* Journal of Molecular Structure: THEOCHEM, **388**, 339 (1996).

[153] F. Jensen. *Polarization Consistent Basis Sets. 4: The Elements He, Li, Be, B, Ne, Na, Mg, Al, and Ar.* The Journal of Physical Chemistry A, **111**, 11198 (2007).

[154] I. S. Dhillon, B. N. Parlett and C. Vömel. *The Design and Implementation of the MRRR Algorithm.* ACM Transactions on Mathematical Software, **32**, 533 (2006).

[155] C. Johnson. *Numerical solution of partial differential equations by the finite element method.* Cambridge University Press (1987).

[156] W. Bangerth and R. Rannacher. *Adaptive Finite Element Methods for Differential Equations.* Birkhäuser Verlag (2003).

[157] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods.* Springer-Verlag, 3rd edn. (2008).

[158] D. W. Kelly, J. P. De S. R. Gago, O. C. Zienkiewicz and I. Babuška. *A posteriori error analysis and adaptive processes in the finite element method: Part I–Error Analysis.* International Journal for Numerical Methods in Engineering, **19**, 1593 (1983).

[159] E. Cuthill and J. McKee. *Reducing the Bandwidth of Sparse Symmetric Matrices.* In *Proceedings of the 1969 24th National Conference*, ACM '69, 157–172. ACM, New York, NY, USA (1969).

[160] J. D. Jackson. *Classical electrodynamics.* John Wiley & Sons, New York, NY, 3rd edn. (1999).

[161] W. Hackbusch. *Multi-Grid Methods and Applications*, vol. 4 of *Springer Series in Computational Mathematics.* Springer-Verlag (1985).

[162] M. Kronbichler and K. Kormann. *A generic interface for parallel cell-based finite element operator application.* Computers & Fluids, **63**, 135 (2012).

[163] C.-J. Lin and J. J. Moré. *Incomplete Cholesky Factorizations with Limited Memory.* SIAM Journal on Scientific Computing, **21**, 24 (1999).

[164] T. A. Davis. *Algorithm 832: UMFPACK V4.3—an Unsymmetric-pattern Multifrontal Method.* ACM Transactions on Mathematical Software, **30**, 196 (2004).

[165] J. E. Avery. *The generalised Sturmian method.* Master's thesis, University of Copenhagen (2008).

[166] V. Aquilanti, S. Cavalli, C. Coletti and G. Grossi. *Alternative Sturmian bases and momentum space orbitals: an application to the hydrogen molecular ion.* Chemical Physics, **209**, 405 (1996).

[167] V. Aquilanti, S. Cavalli and C. Coletti. *The d-dimensional hydrogen atom: hyperspherical harmonics as momentum space orbitals and alternative Sturmian basis sets.* Chemical Physics, **214**, 1 (1997).

[168] V. Aquilanti, S. Cavalli, C. Coletti, D. D. Domenico and G. Grossi. *Hyperspherical harmonics as Sturmian orbitals in momentum space: A systematic approach to the few-body Coulomb problem.* International Reviews in Physical Chemistry, **20**, 673 (2001).

[169] J. E. Avery and J. S. Avery. *Molecular Integrals for Exponential-Type Orbitals Using Hyperspherical Harmonics.* vol. 70 of *Advances in Quantum Chemistry*, 265 – 324. Academic Press (2015).

[170] J. E. Avery and M. F. Herbst. *Integral library for Coloumb-Sturmian-type orbitals.* `https://molsturm.org/sturmint`. Accessed on 13th March 2019.

[171] V. Aquilanti, S. Cavalli and C. Coletti. *Hyperspherical Symmetry of Hydrogenic Orbitals and Recoupling Coefficients among Alternative Bases.* Physical Review Letters, **80**, 3209 (1998).

[172] V. Aquilanti, A. Caligiana and S. Cavalli. *Hydrogenic elliptic orbitals, Coulomb Sturmian sets, and recoupling coefficients among alternative bases.* International Journal of Quantum Chemistry, **92**, 99 (2003).

[173] J. S. Avery and J. E. Avery. *Kramers Pairs in Configuration Interaction.* vol. 43 of *Advances in Quantum Chemistry.* Academic Press (2003).

[174] J. S. Avery, J. E. Avery, V. Aquilanti and A. Caligiana. *Atomic Densities, Polarizabilities, and Natural Orbitals Derived from Generalized Sturmian Calculations.* Advances in Quantum Chemistry, **47**, 157 (2004).

[175] D. M. Mitnik, F. D. Colavecchia, G. Gasaneo and J. M. Randazzo. *Computational methods for Generalized Sturmians basis.* Computer Physics Communications, **182**, 1145 (2011).

[176] A. Abdouraman, A. Frapiccini, A. Hamido, F. Mota-Furtado, P. O'Mahony, D. Mitnik, G. Gasaneo and B. Piraux. *Sturmian bases for two-electron systems in hyperspherical coordinates.* Journal of Physics B: Atomic, Molecular and Optical Physics, **49**, 235005 (2016).

[177] S. A. Losilla and D. Sundholm. *A divide and conquer real-space approach for all-electron molecular electrostatic potentials and interaction energies.* The Journal of Chemical Physics, **136**, 214104 (2012).

[178] E. A. Toivanen, S. A. Losilla and D. Sundholm. *The grid-based fast multipole method - a massively parallel numerical scheme for calculating two-electron interaction energies.* Physical Chemistry Chemical Physics, **17**, 31480 (2015).

[179] J. Kobus. *A finite difference Hartree–Fock program for atoms and diatomic molecules.* Computer Physics Communications, **184**, 799 (2013).

[180] E. L. Briggs, D. J. Sullivan and J. Bernholc. *Real-space multigrid-based approach to large-scale electronic structure calculations.* Physical Review B, **54**, 14362 (1996).

[181] J. E. Pask and P. A. Sterne. *Finite element methods in ab initio electronic structure calculations.* Modelling and Simulation in Materials Science and Engineering, **13**, R71 (2005).

[182] C. F. Fischer. *Numerical solution of general Hartree-Fock equations for atoms.* Journal of Computational Physics, **27**, 221 (1978).

[183] J. Olsen and D. Sundholm. *LUCAS, an atomic program.*

[184] S. Yamakawa and S.-a. Hyodo. *Gaussian finite-element mixed-basis method for electronic structure calculations.* Physical Review B, **71**, 035113 (2005).

[185] Y. Kurashige, T. Nakajima and K. Hirao. *Gaussian and finite-element Coulomb method for the fast evaluation of Coulomb integrals.* The Journal of Chemical Physics, **126**, 144106 (2007).

[186] M. A. Watson, Y. Kurashige, T. Nakajima and K. Hirao. *Linear-scaling multipole-*

accelerated Gaussian and finite-element Coulomb method. The Journal of Chemical Physics, **128**, 054105 (2008).

[187] W. Hackbusch. *A Sparse Matrix Arithmetic Based on $\mathcal{H}$-Matrices. Part I: Introduction to $\mathcal{H}$-Matrices.* Computing, **62**, 89 (1999).

[188] W. Hackbusch and S. Börm. *$\mathcal{H}^2$-matrix approximation of integral operators by interpolation.* Applied Numerical Mathematics, **43**, 129 (2002).

[189] L. Grasedyck and W. Hackbusch. *Construction and Arithmetics of $\mathcal{H}$-Matrices.* Computing, **70**, 295 (2003).

[190] W. Hackbusch. *Hierarchical Matrices: Algorithms and Analysis.* Springer-Verlag (2015).

[191] T. G. Kolda and B. W. Bader. *Tensor Decompositions and Applications.* SIAM Review, **51**, 455 (2009).

[192] I. V. Oseledets. *Tensor-Train Decomposition.* SIAM Journal on Scientific Computing, **33**, 2295 (2011).

[193] U. Schollwöck. *The density-matrix renormalization group in the age of matrix product states.* Annals of Physics, **326**, 96 (2011).

[194] V. Khoromskaia and B. N. Khoromskij. *Tensor numerical methods in quantum chemistry: from Hartree-Fock to excitation energies.* Physical Chemistry Chemical Physics, **17**, 31491 (2015).

[195] E. Cancès and C. Le Bris. *Can we outperform the DIIS approach for electronic structure calculations?* International Journal of Quantum Chemistry, **79**, 82 (2000).

[196] E. Cancès and C. Le Bris. *On the convergence of SCF algorithms for the Hartree-Fock equations.* ESAIM: M2AN, **34**, 749 (2000).

[197] V. R. Saunders and I. H. Hillier. *A "Level–Shifting" method for converging closed shell Hartree–Fock wave functions.* International Journal of Quantum Chemistry, **7**, 699 (1973).

[198] M. Guest and V. R. Saunders. *On methods for converging open-shell Hartree-Fock wave-functions.* Molecular Physics, **28**, 819 (1974).

[199] E. Cancès. Private communication (2017).

[200] P. Pulay. *Convergence acceleration of iterative sequences. the case of scf iteration.* Chemical Physics Letters, **73**, 393 (1980).

[201] K. N. Kudin, G. E. Scuseria and E. Cancès. *A black-box self-consistent field convergence algorithm: One step closer.* The Journal of Chemical Physics, **116**, 8255 (2002).

[202] T. P. Hamilton and P. Pulay. *Direct inversion in the iterative subspace (DIIS) optimization of open-shell, excited-state, and small multiconfiguration SCF wave functions.* The Journal of Chemical Physics, **84**, 5728 (1986).

[203] D. G. Anderson. *Iterative Procedures for Nonlinear Integral Equations.* Journal of the ACM, **12**, 547 (1965).

[204] S. Høst, J. Olsen, B. Jansík, L. Thøgersen, P. Jørgensen and T. Helgaker. *The*

augmented Roothaan–Hall method for optimizing Hartree–Fock and Kohn–Sham density matrices. The Journal of Chemical Physics, **129**, 124106 (2008).

[205] H. Li and D. J. Yaron. *A Least-Squares Commutator in the Iterative Subspace Method for Accelerating Self-Consistent Field Convergence.* Journal of Chemical Theory and Computation, **12**, 5322 (2016).

[206] G. B. Bacskay. *A quadratically convergent Hartree-Fock (QC-SCF) method. Application to closed shell systems.* Chemical Physics, **61**, 385 (1981).

[207] T. van Voorhis and M. Head-Gordon. *A geometric approach to direct minimization.* Molecular Physics, **100**, 1713 (2002).

[208] P. Sałek, S. Høst, L. Thøgersen, P. Jørgensen, P. Manninen, J. Olsen, B. Jansík, S. Reine, F. Pawłowski, E. Tellgren, T. Helgaker and S. Coriani. *Linear-scaling implementation of molecular electronic self-consistent field theory.* The Journal of Chemical Physics, **126**, 114110 (2007).

[209] M. Wormit. *Development and Application of Reliable Methods for the Calculation of Excited States: From Light-Harvesting Complexes to Medium-Sized Molecules.* Ph.D. thesis, Universität Frankfurt (2009).

[210] M. Wormit, D. R. Rehn, P. H. Harbach, J. Wenzel, C. M. Krauter, E. Epifanovsky and A. Dreuw. *Investigating excited electronic states using the algebraic diagrammatic construction (ADC) approach of the polarisation propagator.* Molecular Physics, **112**, 774 (2014).

[211] A. Dreuw and M. Wormit. *The algebraic diagrammatic construction scheme for the polarization propagator for the calculation of excited states.* Wiley Interdisciplinary Reviews: Computational Molecular Science, **5**, 82 (2014).

[212] 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.* Society for Industrial and Applied Mathematics, 3rd edn. (1999).

[213] Z. Teng, Y. Zhou and R.-C. Li. *A block Chebyshev-Davidson method for linear response eigenvalue problems.* Advances in Computational Mathematics, **42**, 1103 (2016).

[214] A. Pieper, M. Kreutzer, A. Alvermann, M. Galgon, H. Fehske, G. Hager, B. Lang and G. Wellein. *High-performance implementation of Chebyshev filter diagonalization for interior eigenvalue computations.* Journal of Computational Physics, **325**, 226 (2016).

[215] Y. Zhou, J. R. Chelikowsky and Y. Saad. *Chebyshev-filtered subspace iteration method free of sparse diagonalization for solving the Kohn–Sham equation.* Journal of Computational Physics, **274**, 770 (2014).

[216] *Latency numbers every programmer should know.* `https://gist.github.com/hellerbarde/2843375`. Accessed on 08th Februrary 2018.

[217] C. Scott. *Latency numbers every programmer should know.* `https://people.eecs.berkeley.edu/~rcs/research/interactive_latency.html`. Accessed on 08th Februrary 2018.

[218] D. Cheney. *Five things that make Go fast.* Presentation at Gocon2014, Tokio, Japan (2014).

[219] P. Hudak. *Conception, Evolution, and Application of Functional Programming Languages*. ACM Computing Surveys, **21**, 359 (1989).

[220] G. Baumgartner, A. Auer, D. Bernholdt, A. Bibireata, V. Choppella, D. Cociorva, X, R. H. Gao, S. Hirata, S. Krishnamoorthy, S. Krishnan, C. Lam, Q. Lu, M. Nooijen, R. Pitzer, J. Ramanujam, P. Sadayappan and A. Sibiryakov. *Synthesis of High-Performance Parallel Programs for a Class of Ab Initio Quantum Chemistry Models*. In *Proceedings of the IEEE*, vol. 93, 276–292 (2005).

[221] E. Solomonik, D. Matthews, J. Hammond, J. Stanton and J. Demmel. *A massively parallel tensor contraction framework for coupled-cluster computations*. Journal of Parallel and Distributed Computing, **74**, 3176 (2014).

[222] E. Peise, D. Fabregat-Traver and P. Bientinesi. *On the Performance Prediction of BLAS-based Tensor Contractions*. In S. A. Jarvis, S. A. Wright and S. D. Hammond (Eds.), *High Performance Computing Systems. Performance Modeling, Benchmarking, and Simulation*, 193–212. Springer-Verlag, Cham (2015).

[223] J. A. Calvin, C. A. Lewis and E. F. Valeev. *Scalable Task-based Algorithm for Multiplication of Block-rank-sparse Matrices*. In *Proceedings of the 5th Workshop on Irregular Applications: Architectures and Algorithms*, IA3 '15, 4:1–4:8. ACM, New York, NY, USA (2015).

[224] J. A. Calvin and E. F. Valeev. *Task-Based Algorithm for Matrix Multiplication: A Step Towards Block-Sparse Tensor Computing*. **arXiv:1504.05046** (2015).

[225] M. R. Kristensen, S. A. Lund, T. Blum and J. Avery. *Fusion of parallel array operations*. In *Proceedings of the 2016 International Conference on Parallel Architectures and Compilation*, PACT 16, 71–85. ACM (2016).

[226] M. Kristensen, J. Avery, T. Blum, S. Lund and B. Vinter. *Battling memory requirements of array programming through streaming*. In *International Conference on High Performance Computing*, vol. 9945 (2016).

[227] M. F. Herbst and J. E. Avery. *The lazyten lazy matrix library*. `https://lazyten.org`. Accessed on 13th March 2019.

[228] C. Sanderson and R. Curtin. *Armadillo: a template-based C++ library for linear algebra*. Journal of Open Source Software, **1**, 26 (2016).

[229] R. Lehoucq, D. Sorensen and C. Yang. *ARPACK Users' Guide: Solution of Large-scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*. Software, Environments, Tools. Society for Industrial and Applied Mathematics (1998).

[230] G. Guennebaud, B. Jacob et al. *Eigen v3*. `http://eigen.tuxfamily.org` (2010). Accessed on 10th March 2018.

[231] C. G. Baker, U. L. Hetmaniuk, R. B. Lehoucq and H. K. Thornquist. *Anasazi Software for the Numerical Solution of Large-scale Eigenvalue Problems*. ACM Transactions on Mathematical Software, **36**, 13:1 (2009).

[232] M. F. Herbst and J. E. Avery. *The bucket of Krimskrams every C or C++ project needs*. `https://lazyten.org/krims`. Accessed on 13th March 2019.

[233] M. F. Herbst, A. Dreuw and J. E. Avery. *`molsturm`: A light-weight quantum-chemistry framework for rapid method development not restricted to a particular type of basis function*. In preparation.

[234] A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. B. Jensen, J. Kermode, J. R. Kitchin, E. L. Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng and K. W. Jacobsen. *The atomic simulation environment—a Python library for working with atoms.* Journal of Physics: Condensed Matter, **29**, 273002 (2017).

[235] T. Verstraelen, P. Tecmer, F. Heidar-Zadeh, C. E. González-Espinoza, M. Chan, T. D. Kim, K. Boguslawski, S. Fias, S. Vandenbrande, D. Berrocal and P. W. Ayers. *HORTON 2.1.0* (2017).

[236] Q. Sun, T. C. Berkelbach, N. S. Blunt, G. H. Booth, S. Guo, Z. Li, J. Liu, J. McClain, E. R. Sayfutyarova, S. Sharma, S. Wouters and G. K.-L. Chan. *The Python-based Simulations of Chemistry Framework (PySCF).* Wiley Interdisciplinary Reviews: Computational Molecular Science (2017).

[237] R. Muller. *PyQuante: Python Quantum Chemistry.* `http://pyquante.sourceforge.net`. Accessed on 26th November 2017.

[238] R. M. Parrish, L. A. Burns, D. G. A. Smith, A. C. Simmonett, A. E. DePrince, E. G. Hohenstein, U. Bozkaya, A. Y. Sokolov, R. Di Remigio, R. M. Richard, J. F. Gonthier, A. M. James, H. R. McAlexander, A. Kumar, M. Saitow, X. Wang, B. P. Pritchard, P. Verma, H. F. Schaefer, K. Patkowski, R. A. King, E. F. Valeev, F. A. Evangelista, J. M. Turney, T. D. Crawford and C. D. Sherrill. *Psi4 1.1: An Open-Source Electronic Structure Program Emphasizing Automation, Advanced Libraries, and Interoperability.* Journal of Chemical Theory and Computation, **13**, 3185 (2017).

[239] *HDF5 Reference Manual.* The HDF Group (2011). Release 1.8.8.

[240] S. van der Walt, S. C. Colbert and G. Varoquaux. *The NumPy Array: A Structure for Efficient Numerical Computation.* Computing in Science & Engineering, **13**, 22 (2011).

[241] R. J. Needs, M. D. Towler, N. D. Drummond and P. L. Ríos. *Continuum variational and diffusion quantum Monte Carlo calculations.* Journal of Physics: Condensed Matter, **22**, 023201 (2010).

[242] J. Kim, K. P. Esler, J. McMinis, M. A. Morales, B. K. Clark, L. Shulenburger and D. M. Ceperley. *Hybrid algorithms in quantum Monte Carlo.* Journal of Physics: Conference Series, **402**, 012008 (2012).

[243] J. Hutter, M. Iannuzzi, F. Schiffmann and J. VandeVondele. *cp2k: atomistic simulations of condensed matter systems.* Wiley Interdisciplinary Reviews: Computational Molecular Science, **4**, 15 (2014).

[244] J. D. Hunter. *Matplotlib: A 2D graphics environment.* Computing In Science & Engineering, **9**, 90 (2007).

[245] E. Jones, T. Oliphant, P. Peterson et al. *SciPy: Open source scientific tools for Python* (2001–). Accessed on 09th March 2017.

[246] W. McKinney, J. Reback et al. *pandas: Python Data Analysis Library* (2008–). Accessed on 09th March 2017.

[247] D. Beazley, L. Ballabio, W. Fulton, M. Gossage, M. Köppe, J. Lenz, M. Matus, J. Stewart, A. Yerkes, S. Yoshiki, S. Singhi, X. Delacour, O. Betts and D. Z. Gang. *SWIG: Simplified Wrapper and Interface Generator*. Accessed on 09th March 2018.

[248] F. Pérez and B. E. Granger. *IPython: a System for Interactive Scientific Computing*. Computing in Science and Engineering, **9**, 21 (2007).

[249] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, C. Willing and Jupyter Development Team. *Jupyter Notebooks - a publishing format for reproducible computational workflows*. `https://jupyter.org/` (2016).

[250] O. Ben-Kiki, C. Evans and I. döt Net. *YAML Ain't Markup Language (YAML™) Version 1.2*. `http://www.yaml.org/spec/1.2/spec.html` (2009). Accessed on 03rd December 2017.

[251] F. Weigend and R. Ahlrichs. *Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy*. Physical Chemistry Chemical Physics, **7**, 3297 (2005).

[252] N. C. Handy, P. J. Knowles and K. Somasundram. *On the convergence of the Møller-Plesset perturbation series*. Theoretica chimica acta, **68**, 87 (1985).

[253] P. Knowles, K. Somasundram, N. Handy and K. Hirao. *The calculation of higher-order energies in the many-body perturbation theory series*. Chemical Physics Letters, **113**, 8 (1985).

[254] W. J. Hehre, R. Ditchfield and J. A. Pople. *Self-Consistent Molecular Orbital Methods. XII. Further Extensions of Gaussian—Type Basis Sets for Use in Molecular Orbital Studies of Organic Molecules*. The Journal of Chemical Physics, **56**, 2257 (1972).

[255] M. J. D. Powell. *An efficient method for finding the minimum of a function of several variables without calculating derivatives*. The Computer Journal, **7**, 155 (1964).

[256] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery. *Numerical Recipes*. Cambridge University Press (1992).

[257] E. Valeyev, J. Calvin, D. Lewis, J. Dullea, C. Peng, K. Nishimra, jfermann, jdwhitfield, O. Čertík, M. F. Herbst, S. Y. Willow and D. Williams-Young. *evaleev/libint: 2.3.1* (2017).

[258] E. F. Valeev. *Libint: A library for the evaluation of molecular integrals of many-body operators over Gaussian functions*. http://libint.valeyev.net/ (2017). Version 2.3.1.

[259] Q. Sun. *Libcint: An efficient general integral library for Gaussian basis functions*. Journal of Computational Chemistry, **36**, 1664 (2015).

[260] N. H. Morgon, R. Custodio and J. Mohallem. *A method for the determination of the Hartree-Fock limit: application to closed-shell atoms*. Journal of Molecular Structure: THEOCHEM, **394**, 95 (1997).

[261] D. E. Woon and T. H. Dunning. *Gaussian basis sets for use in correlated molecular calculations. III. The atoms aluminum through argon*. The Journal of Chemical Physics, **98**, 1358 (1993).

[262] T. Van Mourik and T. H. Dunning. *Gaussian basis sets for use in correlated molecular calculations. VIII. Standard and augmented sextuple zeta correlation consistent basis sets for aluminum through argon.* International Journal of Quantum Chemistry, **76**, 205 (2000).

[263] B. P. Prascher, D. E. Woon, K. A. Peterson, T. H. Dunning and A. K. Wilson. *Gaussian basis sets for use in correlated molecular calculations. VII. Valence, core-valence, and scalar relativistic basis sets for Li, Be, Na, and Mg.* Theoretical Chemistry Accounts, **128**, 69 (2011).

[264] D. B. Cook. *Unrestricted Hartree-Fock functions for the carbon and fluorine atoms.* Theoretica chimica acta, **58**, 155 (1981).

[265] D. B. Cook. *Symmetry constraints on the Hartree-Fock model.* Molecular Physics, **53**, 631 (1984).

[266] H. A. Jahn and E. Teller. *Stability of polyatomic molecules in degenerate electronic states - I—Orbital degeneracy.* Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, **161**, 220 (1937).

[267] R. P. Brent. *Algorithms for minimization without derivatives*, chap. 3-4. Prentice-Hall (1972).

[268] E. Clementi and D. L. Raimondi. *Atomic Screening Constants from SCF Functions.* The Journal of Chemical Physics, **38**, 2686 (1963).

[269] C. E. Moore. *Atomic Energy Levels.* National Bureau of Standards, Washington DC (1949).

[270] H. Feshbach. *Unified theory of nuclear reactions.* Annals of Physics, **5**, 357 (1958).

[271] H. Feshbach. *A unified theory of nuclear reactions. II.* Annals of Physics, **19**, 287 (1962).

[272] R. Santra and L. S. Cederbaum. *Non-Hermitian electronic theory and applications to clusters.* Physics Reports, **368**, 1 (2002).

[273] B. Huber and S. Wolf. *Xerus - A General Purpose Tensor Library.* `https://libxerus.org/` (2014–2017). Accessed on 10th March 2018.

[274] E. Epifanovsky, M. Wormit, T. Kuś, A. Landau, D. Zuev, K. Khistyaev, P. Manohar, I. Kaliman, A. Dreuw and A. I. Krylov. *New implementation of high-level correlated methods using a general block tensor library for high-performance electronic structure calculations.* Journal of Computational Chemistry, **34**, 2293 (2013).

[275] S. Lehtola, C. Steigemann, M. J. Oliveira and M. A. Marques. *Recent developments in libxc — A comprehensive library of functionals for density functional theory.* SoftwareX, **7**, 1 (2018).

[276] J. Wenzel, M. Wormit and A. Dreuw. *Calculating core-level excitations and x-ray absorption spectra of medium-sized closed-shell molecules with the algebraic-diagrammatic construction scheme for the polarization propagator.* Journal of Computational Chemistry, **35**, 1900 (2014).

[277] J. Wenzel. *Development and Implementation of Theoretical Methods for the Descriptions of Electronically Core-Excited States.* Ph.D. thesis, Universität Heidelberg (2016).

[278] K. Kaufmann, W. Baumeister and M. Jungen. *Universal Gaussian basis sets for an optimum representation of Rydberg and continuum wavefunctions.* Journal of Physics B, **22**, 2223 (1989).

[279] H. Böck. *The Fuzzing Project.* `https://fuzzing-project.org/`. Accessed on 05th March 2018.

[280] G. Knizia, W. Li, S. Simon and H.-J. Werner. *Determining the Numerical Stability of Quantum Chemistry Algorithms.* Journal of Chemical Theory and Computation, **7**, 2387 (2011).

[281] H.-J. Werner, P. J. Knowles, G. Knizia, F. R. Manby, M. Schütz, P. Celani, W. Györffy, D. Kats, T. Korona, R. Lindh, A. Mitrushenkov, G. Rauhut, K. R. Shamasundar, T. B. Adler, R. D. Amos, A. Bernhardsson, A. Berning, D. L. Cooper, M. J. O. Deegan, A. J. Dobbyn, F. Eckert, E. Goll, C. Hampel, A. Hesselmann, G. Hetzer, T. Hrenar, G. Jansen, C. Köppl, Y. Liu, A. W. Lloyd, R. A. Mata, A. J. May, S. J. McNicholas, W. Meyer, M. E. Mura, A. Nicklass, D. P. O'Neill, P. Palmieri, D. Peng, K. Pflüger, R. Pitzer, M. Reiher, T. Shiozaki, H. Stoll, A. J. Stone, R. Tarroni, T. Thorsteinsson and M. Wang. *MOLPRO, version 2015.1, a package of ab initio programs* (2015).

Publications

Articles in preparation

- M. F. Herbst, A. Dreuw and J. E. Avery. *`molsturm`: A light-weight quantum-chemistry framework for rapid method development not restricted to a particular type of basis function.* In preparation.
- M. F. Herbst, J. E. Avery and A. Dreuw. *Judging the angular momentum requirements for the representation of a wave function from Coulomb-Sturmian convergence progressions.* In preparation.
- M. F. Herbst, J. E. Avery and A. Dreuw. *Coulomb-Sturmian-based excited electronic states using the algebraic diagrammatic construction scheme of the polarisation propagator.* In preparation.
- M. F. Herbst, M. R. B. Kristensen, J. E. Avery, B. Vinter, A. Dreuw. *Lazyten and bohrium: Combining efforts for an automatically parallelising readable code.* In preparation.

Scientific software

This section gives the list of scientific software projects, which I started in collaboration with others and which are released or will be released soon.

`molsturm` modular and flexible quantum chemistry package

- M. F. Herbst and J. E. Avery. *`molsturm` 0.0.3.* `https://molsturm.org` (2018)
- M. F. Herbst and J. E. Avery. *`gscf` 0.1.0.* `https://molsturm.org/gscf` (2017)
- M. F. Herbst and J. E. Avery. *`gint` 0.0.0.* `https://molsturm.org/gint` (2017)
- M. F. Herbst and J. E. Avery. *`lazyten` 0.4.1.* `https://lazyten.org` (2017)
- M. F. Herbst and J. E. Avery. *`krims` 0.2.1.* `https://lazyten.org/krims` (2017)
- J. E. Avery and M. F. Herbst. *`sturmint` 0.1.0.* To be released.

`look4bas`: Interactive tool to search for Gaussian basis sets online

- M. F. Herbst. *`look4bas`: First release.* (2018) DOI 10.5281/zenodo.1188992

Lecture notes

The following lecture notes were written and published during my graduate studies.

- M. F. Herbst. *Advanced Bash Scripting 2015.* (2015) DOI 10.5281/zenodo.1038526
- M. F. Herbst. *Introduction to awk programming 2016.* (2016) DOI 10.5281/zenodo.1038522
- M. F. Herbst. *Advanced Bash Scripting 2017.* (2017) DOI 10.5281/zenodo.1045332

Acknowledgements

First and foremost I would like to thank my supervisor Prof. Andreas Dreuw, my co-supervisor Prof. Guido Kanschat as well as my advisor Dr. James Avery. They provided me with a constant source of support and helpful guidelines throughout. Andreas, thanks for allowing me scientific freedom, for supporting the collaboration with the wonderful people in Copenhagen in every aspect and for giving me the chance to work on such a highly interdisciplinary project in the first place. Guido, thanks for the enlightening discussions about numerics as well as the occasional criticism to make me invest a second thought. James, thanks for introducing me to Coulomb-Sturmians, for the great time I had whenever I visited you in Copenhagen and of course for showing me Heisenberg's bath tub.

Along the same line I wish to commemorate Dr. Michael Wormit, with whom I had the pleasure to work on the topic of finite elements, albeit for too short. It brings me a consoling thought that some of his ideas managed to bear fruit in this thesis.

An aspect of quantum chemistry which has always fascinated me, is the interdisciplinarity of the field. It is absolutely astonishing to find that the current status could only be achieved by a combination of developments guided by chemical as well as physical intuition on the one hand and a careful study of the beautiful underlying mathematical structures as well as the many dirty tricks of numerical linear algebra on the other. In this respect I am very grateful to all the impulses and ideas from every scientific field and research topic I was exposed to. I wish to express my thanks for all the discussions with (in random order) Maximilian Scheurer, Fabian Klein, Prof. Reinhold Schneider, Jie Han, Dr. Tim Stauch, Adrian Dempwolff, Dr. Thomas Fransson, Jan Christoph Peters, Dr. Katharina Kormann, Dr. Mads Kristensen, Manuel Hodecker, Dr. Jenny Wagner, Dr. Klaus Birkelund, Lucas Fabian Hackl, Henrik Larsson, Jan Janßen, Dr. Tobias Setzer, Dr. Denis Davydov, Prof. Eric Cancès and everyone I forgot to mention in this list.

In the same respect I would like to acknowledge the curious and thoughtful people I met at the Chaos Communication Congress and various other hacker events I attended. The many night of talking about science, society, ethics and programming often brought a different perspective to my own work. A special thanks goes to (again in random order) supaake, cherti, hauro, kungi, rami, janx2 and all the other guys from the NoName e.V. Similarly I acknowledge my fellow jazzers Susanne, Rudolf, Jürgen, Walter und Uli. Thanks for taking my mind off science for a while.

I gratefully acknowledge the University of Copenhagen, the KTH Royal Institute of Technology in Stockholm, and above all Heidelberg University for hosting me. Further I thank the Heidelberg Graduate School for the Mathematical and Computational Methods

for the Sciences for providing me with financial support and for the opportunity to take action myself in the form of organising conferences or teaching fellow students.

I thank the past and present members of the Dreuw group, for the relaxed atmosphere with constant running gags about stuffed animals and awful singers. Thanks for being more than just colleagues, for all the fun at our retreats in Kleinwalsertal, our trips to Wurstmarkt and the beers and wines after work. In the same way I wish to thank the group of Prof. Dr. Brian Vinter at the Niels Bohr Institute for the open and welcoming atmosphere. There is no question I immediately felt part and truly enjoyed the time I spent with you guys. I am already looking forward to coming back for another visit!

I gratefully acknowledge the computational time I was provided at the computing facilities of the Vinter group, as well as all the excellent support from Jonas Bardino with respect to getting my code to run. I thank Manfred Trunk for having an open door for all requests related to our cluster in Heidelberg and for providing almost instant answers for all emails I sent. Thanks to Ellen Vogel for taking care of the bureaucracy.

For proof-reading earlier drafts of the thesis I thank (in random order) Manuel Hodecker, Marvin Hoffmann, Carine Dengler, Adrian Dempwolff, Maximilian Scheurer, Dr. Jenny Wagner, Fabian Faulstich, Dr. James Avery, Henrik Larsson, Reena Sen, Andreas Fink and Dr. Thomas Fransson, who provided helpful comments and detailed feedback. Furthermore I wish to express my thanks to the many people developing the software I used during the preparation of the presented work, namely `bash`, `clang`, `git`, `gcc`, `i3`, LaTeX, `numpy`, `python`, TeX, `vim` and all the other tiny utilities I use without thinking about them. Especially I want to acknowledge all the people involved with making Debian — the truly best GNU/LinuX distribution in the world.

I express my thanks to Henrik Larsson for a friendship, which has lead to many weekends of activities accompanied by fruitful debate and new insights into my own subject. I like that we do not always agree and that the only way to convince you is scientific evidence. Thanks for teaching another Deutsche Schülerakademie course with me this summer. In the same way I say thanks to my old friends from the days in Grünstadt, Jan Janßen, Jan Christoph Peters, Alexandra Schulte, and Peter Schwalb, which have accompanied me for many years by now and have always found the time for a spontaneous visit in one direction or the other. Thanks to all the new friends I made in Cambridge, in Heidelberg and the rest of Germany. I know we have met far too rarely in recent months, but rest assured I am really grateful for every minute we have spent so far and I look forward for the time to come.

E große Dank sei ach mei ganze Familie vun de Palz. Ganz bsonders will ich danke Sven, Oliver, Michael, Markus un Jun, als ach mei Unkel Klaus un Hans, mei Tante Gerda, Edith un Änne. Mit eich ebbes zu unnernemme oder efach blos zu bable des war oft e Quell vun Kraft fer die mehr anstrengende Tage. E grousse Merci dem Sonja an dem Jim fir oppen Oure an d'Härzlëchkeet. Danewe dank ich meine Eltre Ingrid un Ortwin, bei dene efache Worte net genuch sin, um mei Dank im entferndeschde zu fasse.

Finally and above all I thank my beloved fiancée Carine for every minute we have been, are and will be together. Thanks for the continuous support and being exactly the way you are.

Eidesstattliche Versicherung gemäß §8 der Promotionsordnung der Naturwissenschaftlich-Mathematischen Gesamtfakultät der Universität Heidelberg

1. Bei der eingereichten Dissertation zu dem Thema

   "Development of a modular quantum-chemistry framework for the investigation of novel basis functions"

   handelt es sich um meine eigenständig erbrachte Leistung.

2. Ich habe nur die angegebenen Quellen und Hilfsmittel benutzt und mich keiner unzulässigen Hilfe Dritter bedient. Insbesondere habe ich wörtlich oder sinngemäß aus anderen Werken übernommene Inhalte als solche kenntlich gemacht.

3. Die Arbeit oder Teile davon habe ich bislang nicht an einer Hochschule des In- oder Auslands als Bestandteil einer Prüfungs- oder Qualifikationsleistung vorgelegt.

4. Die Richtigkeit der vorstehenden Erklärungen bestätige ich.

5. Die Bedeutung der eidesstattlichen Versicherung und die strafrechtlichen Folgen einer unrichtigen oder unvollständigen eidesstattlichen Versicherung sind mir bekannt.

Ich versichere an Eides statt, dass ich nach bestem Wissen die reine Wahrheit erklärt und nichts verschwiegen habe.

Ort / Datum

Unterschrift

Index