Real-Space Finite-Difference PAW Method for Large-Scale Applications on Massively Parallel Computers

Der Fakultät für Mathematik, Informatik und Naturwissenschaften
der RWTH Aachen University genehmigte Dissertation zur Erlangung
des akademischen Grades eines Doktors der Naturwissenschaften

vorgelegt von

Dipl.-Phys. Paul Ferdinand Baumeister

aus Oelde (Kreis Warendorf)

Berichter: Universitätsprofessor Dr. Stefan Blügel
Juniorprofessor Paolo Bientinesi, Ph.D.

Tag der mündlichen Prüfung: 11. Juli 2012

This document has been created using LaTeX, Grace and Xfig.

Die Natur ist so gemacht,
daß sie verstanden werden kann. – Werner K. Heisenberg

ABSTRACT

Simulations of materials from first principles have improved drastically over the last few decades, benefitting from newly developed methods and access to increasingly large computing resources. Nevertheless, a quantum mechanical description of a solid without approximations is not feasible. In the wide field of methods for *ab initio* calculations of electronic structure, it has become apparent that density functional theory and, in particular, the local density approximation can also make simulations of large systems accessible. Density functional calculations provide insight into the processes taking place in a vast range of materials by their access to an understandable electronic structure in the framework of the Kohn-Sham single particle wave functions. A number of functionalities in the fields of electronic devices, catalytic surfaces, molecular synthesis and magnetic materials can be explained by analyzing the resulting total energies, ground state structures and Kohn-Sham spectra. However, challenging physical problems are often accompanied by calculations including a huge number of atoms in the simulation volume, mostly due to very low symmetry. The total workload of wave-function-based DFT scales at best quadratically with the number of atoms. This means that supercomputers must be used. In the present work, an implementation of DFT on real-space grids has been developed, suitable for making use of the massively parallel computing resources of modern supercomputers. Massively parallel machines are based on distributed memory and huge numbers of compute nodes, easily exceeding 100,000 parallel processes. An efficient parallelization of density functional calculations is only possible when the data can be stored process-local and the amount of inter-node communication is kept low. Our real-space grid approach with three-dimensional domain decomposition provides an intrinsic data locality and solves both the Poisson equation for the electrostatic problem and the Kohn-Sham eigenvalue problem on a uniform real-space grid. The derivative operators are approximated by finite differences leading to localized operators which only require communication with

the nearest neighbor processes. This leads to excellent parallel performance at large system sizes. Treating only valence electrons, we apply the projector-augmented wave method for accurate modeling of energy contributions and scattering properties of the atomic cores. In addition to real-space grid parallelization, we apply a distribution of the workload of different Kohn-Sham states onto parallel processes. This second parallelization level avoids the memory bottleneck for large system sizes and introduces even more parallel speedup. Calculations of systems with up to 3584 atoms of Ge, Sb and Te were performed on (up to) all 294,912 cores of JUGENE, the massively parallel supercomputer installed at Forschungszentrum Jülich.

CONTENTS

CHAPTER 1

INTRODUCTION

Modern society strongly depends on technological innovations. Technology, for its part, relies on the fundamental understanding of basic physical concepts. In this respect, the field of condensed matter research, from its very beginning, has contributed to numerous revolutions leading to technological applications which have an increasing impact onto everyday life. In particular, since the development of inexpensive electronic devices for computation, fast and extensive media storage and processing and, most important, digital communications, the way people interact has changed drastically and appears to be changing ever faster, day by day. Most of the applications in digital electronic devices are based on inventions and patents in the field of solid-state research. Examples are microfabrication for the processor layout on a silicon wafer, the giant magnetoresistance effect for magnetic hard disks, non-volatile flash memory, lithium-ion batteries as used in mobile phones, optically rewritable digital-versatile disks (DVDs) and many more. Modern society's demand for technical applications seem to be insatiable, driving the improvement of existing and the development of new digital devices to new limits.

Many discoveries in the class of novel materials are inspired by nature or found by accident. However, a new path to material design has opened up due to the vast increase and improvement in computing resources. Materials simulations starting from the most fundamental, their quantum mechanical nature, have become feasible for systems of considerable size and complexity. This development, in return, allows us to conduct fundamental research in the field of condensed matter. Beneficial for this development was the reformulation of the quantum mechanics of many electrons in terms of the density functional theory (DFT) [1, 2], the success of finding a simple but powerful approximation to the unknown energy functional of the density and the conceptualization and implementation of appropriate electronic structure methods that solve these quantum mechanical equations without

any adjustable parameters. The enormous international effort behind the development of such first-principles methods has led to numerical tools that describe the complicated interactions of electrons and atomic nuclei with great accuracy, which are, thus, now capable of making predictions about any conceivable configuration of atoms.

The procedure of downsizing, as seen in the evolution of the silicon-based transistor over various orders of magnitude, is now supplemented by the development of novel materials and functional devices on the nanoscale with a bottom-up approach. The composition of well-understood functionalized building blocks satisfies the demands of novel material design. These building blocks may be single electrons, electron spins, surface states, impurity atoms or ad-atoms, clusters, adsorbed molecules, wire structures and many more. In any case, functionalization can only take place in an environment that stabilizes the arrangement, i.e. a host crystal, a solvent or a substrate material, whose effects contribute substantially to the required functionality. A variety of phenomena have been found to be unique to a specific material so that there is a limited choice if that particular effect is required. However, in some situations, the chemical elements can easily be replaced by members of the same group in the periodic table of elements thus permitting fine tuning of the nanodevice. This substitution of a chemical element in a first-principles calculation is fairly simple compared to the modification of an experiment.

The idea of calculations from first principles occurred at a very early point. In 1929, Dirac already stated that the equations which describe all of the interactions of electrons and nuclei are far too complicated to be solved and a practical numerical method [3] is required. Since then, a variety of approximations have been introduced. The motion of the nuclei is usually separated from the electronic degrees of freedom and often treated semi-classically [4]. However, the electrons also suffer from immense complexity. The electronic many-body wave function possesses far too many degrees of freedom to be treated numerically in full detail. The number of storage element required to allow N_e electrons access to d configurations (lattice sites or predefined orbitals) is d^{N_e}. This number easily grows to unavchievable values since both d and N_e are linear in the system size. Approximations with products of single-particle wave functions lead to the class of quantum chemistry methods including the Hartree method, Hartree-Fock, coupled clusters and configuration interaction [5]. From these methods, we can learn that the electron-electron interaction introduces a correlation effect which is very difficult to treat in an exact manner.

Density functional theory assigns the degrees of freedom to the density rather than to the electronic many-body wave function [6]. This, in principle, gives rise to a computational method that scales linearly in its system size, an order-N method. However, only the total energy contributions from the external potential (of the

atom cores or applied electric and/or magnetic fields) is exact when evaluated in a straightforward way. The kinetic energy and the electron-electron interaction undergo some approximations. The exact electron-electron interaction energy is replaced by an energy functional of the density. Within the class of known functionals, the relatively crude local density approximation (LDA) has turned out to yield sufficiently accurate results, for some materials even in surprisingly good agreement with experiments. Finally, also the kinetic energy needs to be evaluated from the density. The earliest approaches by Thomas [7] and Fermi [8] and also recent investigations [9] show that with repsect to kinetic energy an explicit functional of the density exhibits a large inaccuracy without adjustable parameters. Kohn and Sham introduced an auxiliary set of non-interacting single-particle states [2]. From these, the evaluation of the kinetic energy is straightforward. The single particles move in a local effective one-particle potential that models the interaction of a single electron with the density of all electrons and includes the effect of the energy functional consistently. DFT in the framework of Kohn-Sham (KS) wave functions requires an evaluation of the N_e lowest eigenstates of the effective single-particle Hamiltonian. Giving d degrees of freedom to each KS state results in an overall scaling of this method which is between $d\,N_e$ and $d^2\,N_e$, depending on the algorithm applied for solving the KS eigenvalue problem. To describe the complete simulation volume accurately, d must be proportional to the volume and thus, roughly speaking, to the number of atoms. The number of electrons also depends linearly on the number and type of atoms. This results in an overall scaling of KS DFT which is, at least, cubic in the system size and, at best, quadratic.

The immense success of investigations based on first principles can be measured by the number of publications per year in this field. This can mainly be attributed to the strong predictive power of DFT calculations, which holds for a large variety of materials, and the insight gained by the analysis of their electronic structures and equilibrium geometries. Especially, the *ab initio* character of the DFT methodology and, compared to quantum chemistry methods, its favorable scaling enhances a fundamental understanding of novel materials. We can start our simulation simply from the atomic numbers and approximate atomic positions. However, there are some technical issues arising in implementations of DFT.

The various scales of the effective local potential impose a challenge in representing potentials and KS wave functions. The local effective potential is very deep in the core region and shows a singularity at the nuclear site, but shallow and slowly varying in the region between the atoms, usually called the interstitial or valence region. The $\frac{1}{r}$-singularity of the atomic core potentials has no proper length scale, i.e. its range is infinite, and thus cannot be treated properly in a finite basis set or on a real-space grid with constant grid spacing. In fact, it requires an infinitely high density of basis functions[1] to resolve it accurately. This causes a disadvantageous

[1]Here, we treat grid points as if they were basis functions. A more detailed discussion of their

convergence behavior with respect to the number of basis functions. The deep, attractive potential leads to the formation of strongly bound core states, which are localized in the vicinity of the nuclei. States higher in energy and, in particular valence states, are orthogonal to the lower ones and thus exhibit rapid spatial oscillations in the core region. The representation of the local potential and the KS wave functions can only become sufficiently accurate within a limited, reasonable number of degrees of freedom if we increase the number of basis functions per unit volume in the core region compared to the valence region.

Various ways of overcoming the aforementioned challenges of representing potentials and KS wave functions have been developed over the past decades [6, 10]. The class of muffin-tin (MT) methods including orthogonalized plane wave [11], augmented plane wave (APW) [12], linearized muffin-tin orbital (LMTO) [13] and linearized augmented plane wave (LAPW) methods [14, 15, 16], introduce an atomic sphere and a sharp cut-off at the atomic sphere boundary, where a matching of the numerically given radial solution inside the sphere to the analytically given (smooth) basis function outside the sphere is essential. The various implementations of LAPW [17, 18, 19, 20, 21] are considered to be the method representing the *gold standard* of DFT results for solids due to their accuracy.

However, we apply the projector augmented wave (PAW) method in our work, which was invented by P. E. Blöchl [22]. The class of methods that applies the concept of the PAW transformation, introduced in Chapter 3, also allows the smooth basis functions to cover the atomic spheres leading to corrections inside each sphere. The norm-conserving pseudopotentials [23, 24, 25, 26, 27, 28, 29] or the ultrasoft pseudopotentials [30], which loosen the contraint of a norm-conservation, can be considered as PAW transformations with some approximations in addition to the frozen core states. A large variety of implementations are based on these [31, 32]. The most common implementations represent the smooth PAW transformed quantities in plane waves [33, 34, 35, 36]. Generalizing the PAW method, even the LAPW implementation of Soler and Williams [37] can be incorporated into this class. These methods show a reasonably good convergence with respect to the total number of smooth basis functions whereas the number of smooth basis functions is constant at each point in space. Furthermore, the smooth basis functions do not depend on the atomic positions. This is a prerequisite for an unbiased description of the forces acting on the atoms. Accurate forces are needed for structural relaxation, i.e. the determination of the ground-state geometry, prerequisite for molecular dynamics simulations [38].

In order to reduce the computational cost for the self-consistent calculation of the electronic structure, combinations of the pseudopotential methods and smooth basis functions that depend on the atomic positions have been realized. Within

equivalence is found in Appendix B.0.6.

these implementations, large inner or outer regions of vacuum are sampled with fewer basis functions than the atom-rich portions of space. This applies to adaptive coordinates real-space grid methods [39] and local orbital method, for example using linear combinations of atomic orbitals (LCAO) [40, 41, 42]. This class of DFT implementations based on local orbitals has also proved to be very successful. A representation in numerical atomic orbitals [43] allows an all-electron description with favorable scaling behavior. The local-orbital-based approach also allows an order-N treatment in the framework of density matrix formulations. The density matrix decays exponentially with the distance in real-space if no band crosses the Fermi energy. Hence, these methods cannot treat metals with order-N and are thus less generally applicable.

In the early days of density functional calculations, the computing resources were, even compared to modern mobile phones, smaller by several orders of magnitude. However, calculations of millions of atoms were performed in perfect crystal symmetry. Exploiting Bloch's theorem and restricting calculations to the irreducible part of the Brillouin zone enabled the determination of band structures, lattice constants and bulk moduli for pure materials and alloys. As long as the real-space unit cell is small, the diagonalization of the KS Hamiltonian is relatively cheap. However, high symmetry is seldom found in nature. Experimentalists struggle to obtain perfect material purity, best vacuum and to come closer to 0 Kelvin in their laboratories whereas *ab initio* calculations approach the problem from the other end. The pure crystal in high symmetry can be computed inexpensively, space without atoms represents a perfect vacuum at zero pressure and the ground-state character of DFT is associated with zero temperature. Modern questions of physics focus on broken symmetries. Here, we may think of amorphous solids, glasses, interchanged stacking orders, lattice mismatches or Jahn-Teller distortions, partial disorder, core-hole calculations, surface relaxations, dilute impurity concentrations in otherwise pure host crystals, heterojunctions, epitaxial growth, grain boundaries, surface reconstructions and entire nanoparticles or functionalized molecules on substrates.

A more than quadratic scaling with the size of the systems real-space cell leads to an enormous workload of large-scale DFT calculations that cannot be managed on a workstation. In order to investigate material properties, we need to be able to find answers, at least within several days and if possible within hours. A single calculation lasting several weeks or even months cannot keep pace with the speed of modern science in the age of electronically accessible information. The need for high-performance computing is therefore ubiquitous and can only be satisfied by modern supercomputers. Although these machines help us to illuminate many of the secrets of physics, their usage also raises some, mostly technical, challenges dealing with the distribution of compute tasks, the limitation of communication and the modest memory size per node. In summary, the most intriguing question

is: Which of the above mentioned methods is most suitable for large-scale DFT calculations on supercomputers?

The uniform real-space grid-based approach for DFT calculations allows efficient parallelization of the real-space grid [44, 45, 46, 32, 47, 48, 49, 50] and is thus suited for massively parallel machines. The PAW method models the correct scattering of the KS wave functions at the atomic cores and provides accurate all-electron energy contributions of the atoms [51], although the density of grid points is not refined in the vicinity of the core region [52, 53]. The locality of the KS equation in real-space formulation is can be exploited as data locality, i.e. each compute node acts mostly on data stored in local memory. In contrast, non-locality requires communication. This thesis reveals a way of maximally exploiting the locality using a uniform real-space grid and incorporating long-range interactions by iterative invocation of finite-range operations. Communication in parallel computing is, compared to the same calculation in serial, an overhead. The parallel scaling behavior of an algorithm depends crucially on the dependence of the overhead times on the degree of parallelization.

Over the last two decades, the performance of the fastest computers in the world has grown by six orders of magnitude and reached the petascale some years ago. This was, at the beginning, mostly due to an increased clock rate of the processors. Moore's law about the number of transistors on a chip growing by a factor of two every two years has proven right for about 35 years, much longer than Moore himself expected. However, the paradigms have shifted. Increasing the clock frequency of the central processing unit (CPU) is accompanied by a higher power dissipation. The largest fraction of the CPUs power consumption then produces heat. An extremely high density of circuits and increased clock rates lead to enormous heat accumulation on the chip, and the cooling of the chip represents a technical challenge. Passive air cooling is no longer feasible. Therefore, the total power consumption increases by additional cooling units based on fans or liquid coolants. This technical challenge forces the development of faster computers into the plurality of compute nodes, each one equipped with not-too-fast processors. Massively parallel machines are the answer to further increasing the total computing performance. Production costs forced the design of some massively parallel machines to work with a relatively small memory per compute node, as for example `JUGENE` [54]. The main memory is local to each compute node. Remote memory access thus requires explicit communication operations, in which both the sending node and the receiving node are actively involved. It becomes clear that the network connecting the assembly of vast numbers of compute nodes plays a crucial role. The communication operations introduce synchronization of the processes. The challenge in obtaining parallel efficiency is thus to minimize the number of synchronizations and avoid waiting times by distributing the compute task to the parallel processes according to an estimated load balance. Libraries implementing

two-sided communication are the parallel virtual machine and the message passing interface (MPI). MPI has been used in the present work and is discussed in detail in Chapter 5.

Another change of paradigms that has been observed over the last decade is the development from a single core CPU to multi core chipsets. Several compute cores are located on the same chip and share their access to the main memory of the compute node and sometimes the level 3 (L3) cache. L2, L1 and registers, however, are private to the compute core. In the last few years, many-core ($\geqslant 16$) architectures have also been designed. Access of the parallel threads running on different cores to a common main memory enables us to parallelize communicating tasks differently than in the distributed memory scheme. Shared-memory parallelization has been strongly automated, as for example compilers supporting OpenMP [55]. Its performance depends on environment variables that can be controlled during runtime.

A further trend is the incorporation of accelerator units into a compute node. Many scientific implementations now exploit the strong parallelism in graphical processing units (GPUs) that originated from the need for accelerated rendering in graphics applications, above all, for video games. The combination of one general-purpose GPU with two CPUs in a compute node have reached a world record of 2.5 petaflops in 2010 [56]. In terms of shared memory parallelization or GPU acceleration, DFT implementations can benefit strongly from node-internal speed gain.

Despite the different aspects impacting the speed of a single compute node, future supercomputer architectures will always be based on an array of compute nodes linked by a fast network. Therefore, the aspect of scalability for large problem sizes on the distributed memory parallelization is key to the fast solution of large problems and has been addressed in this work with a focus on MPI parallelism. The road to hybrid parallelization in terms of MPI+OpenMP is discussed.

This thesis is structured as follows. The following chapter explains the foundations of DFT and the motivation for the effective local potential. The third chapter provides an introduction to the PAW formalism with the approximations that lead to lower computational cost and defines the basic quantities and operations appearing in the implementation of the application. Chapter 4 contains a detailed discussion of the Cartesian real-space grid and the approximations made to the kinetic energy. Furthermore, a special technique is illustrated that smoothens the interface between localized functions and wave functions extending in full space. The algorithms used to find the eigenstates of the KS Hamiltonian in real space are outlined at the end of Chapter 4, followed by their parallelization with MPI described in Chapter 5. Two levels of parallelization for distributed memory supercomputer systems are introduced, the real-space grid parallelization and a parallelization over KS states as a response to the quadratic scaling behavior of the DFT method.

The scaling behavior of the DFT method in total, the scaling of each code component and their efficiencies are discussed in detail. Chapter 6 shows the results of large-scale calculations. The implementation has been applied to investigate the electronic structure of disordered alloys of GeSbTe, a phase-change material used in industry for optically rewritable storage media [57, 58, 59, 60], before and after structural relaxation. Statistical methods help to evaluate the vast amount of results of calculations in the large-scale limit. Finally, a conclusion on the results of this work is given in the last chapter.

Input Syntax Explanations

A large part of this thesis describes implementation details of `jüRS`, the Jülich Real-Space DFT code. Sections that explicitly describe a functionality of the code which may be controlled via the input file contain a syntax box explaining code usage, e.g. the line

```
spacing 0.24 Ang
```

in the input file sets the grid spacing to this value and the number of grid points for the representation is chosen such that the grid spacings $h_i \leqslant 0.24$ Å for all three directions $i \in \{x, y, z\}$. Then, the input syntax explanation reads:

`jüRS` **Input Syntax:** `spacing 0.24 Ang`

Unit System

Throughout this document we use the Hartree atomic unit (au) system. These are most convenient for non-relativistic quantum mechanics since they are based on the electron mass m_e=1 and the electron charge e=+1. Electrons are accounted positive whereas protons possess charge −1. Furthermore, Planck's (reduced) quantum of action is unity $\hbar$=1 and ϵ_0=$\frac{1}{4\pi}$ such that the energy scale is fixed to 1 Ha=27.21 eV and lengths are measured in Bohr where 1 Bohr=0.529 Å. For relativistic effects, as emerging in heavy atoms close to the nucleus, we also need the speed of light c_{light}=$\frac{1}{\alpha}$=137.036 Ha Bohr where α is the fine structure constant.

Further quantities to be used are the approximate nucleon mass that is 1 u (1 Dalton) = 1822.89 m_e and Boltzmann's constant $k_B = 3.16683 \cdot 10^{-6} \frac{\text{Ha}}{\text{K}}$ for the conversion of temperatures to energies.

For output of the application, unit systems other than atomic units may be used. A usual choice are eVolt (eV,`eV`) and Ångström (Å,`Ang`). During input, predefined multipliers as `eV`, `Kel` or `Ang` simplify the usage.

`jüRS` **Input Syntax:** `units Ang eV`

Notation

Meaning of mathematical symbols if not specified otherwise:

- $\mathbf{r}$ Real-space vector
- $\mathbf{R}^a$ Position of the atomic nucleus of atom a
- $\mathbf{r}^a = \mathbf{r} - \mathbf{R}^a$
- $r^a = |\mathbf{r}^a|$

- $\mathcal{S}^a$ Atomic sphere centered at $\mathbf{R}^a$
- r^a_{aug} Augmentation radius of $\mathcal{S}^a$
- n Principal quantum number, $n > 0$ (as in $\tilde{\phi}_{n\ell}$)
- ℓ Angular momentum quantum number, $0 \leqslant \ell < n$
- m Magnetic quantum number, $|m| \leqslant \ell$
- L Combined set ℓm of angular momentum quantum numbers
- n Projector- and partial wave index (as in $\tilde{p}_{\ell n m}$)
- i Short projector index stands for $\ell n m$
- $\mathbf{k}$ Bloch vector, element of the Brillouin zone sampling
- σ Collinear spin index, $\sigma \in \{\downarrow, \uparrow\}$
- n Band index (as in $n\sigma\mathbf{k}$)
- $|\Psi\rangle$ Single particle state
- $\Psi(\mathbf{r})$ Single particle wave function, $\Psi(\mathbf{r}) = \langle \mathbf{r}|\Psi\rangle$
- $n^\sigma(\mathbf{r})$ Electron spin density
- $n(\mathbf{r})$ Total electron density (spin integrated)
- $\rho(\mathbf{r})$ Generalized density (electrons and protons)
- $\tilde{\rho}(\mathbf{r})$ Smooth generalized density (with compensators)
- $\tilde{\Psi}(\mathbf{r})$ Smooth representation of $\Psi(\mathbf{r})$
- $\tilde{n}^\sigma(\mathbf{r})$ Smooth electron spin density
- $\phi(r)$ True partial wave
- $\tilde{\phi}(r)$ Smooth partial wave
- $\tilde{p}(\mathbf{r})$ Localized projector function
- $\hat{g}_L(\mathbf{r})$ Localized and normalized compensator function
- $f_{n\sigma\mathbf{k}}$ Occupation numbers

Abbreviations

PAW	Projector augmented wave
DFT	Density functional theory
KS	Kohn-Sham
DOS	Density of states
pDOS	Projected density of states
xc	exchange-correlation
es	electrostatic
kin	kinetic
LDA	Local density approximation
MPI	Message passing interface
LCAO	Linear combination of atomic orbitals

CHAPTER 2

DENSITY FUNCTIONAL THEORY

2.1. The Many-Body System

Quantum mechanics fully describe the phenomena of the nano world. Most observed properties of solid materials can be explained by solving the full (non-relativistic) many-body Schrödinger equation

$$i\frac{\partial}{\partial t}|\Psi_{\text{full}}\rangle = \hat{H}_{\text{full}}|\Psi_{\text{full}}\rangle \tag{2.1}$$

with the full many-body Hamiltonian for electrons and nuclei

$$\begin{aligned}\hat{H}_{\text{full}} = &\sum_{i}^{N_e}\frac{(\hat{\mathbf{p}}_i)^2}{2} + \sum_{a}^{N_a}\frac{(\mathbf{P}^a)^2}{2m^a} - \sum_{i}^{N_e}\sum_{a}^{N_a}\frac{Z^a}{|\hat{\mathbf{r}}_i - \mathbf{R}^a|} \\ &+ \sum_{i}^{N_e}\sum_{i<j}\frac{1}{|\hat{\mathbf{r}}_j - \hat{\mathbf{r}}_i|} + \sum_{a}^{N_a}\sum_{a'<a}\frac{Z^a Z^{a'}}{|\mathbf{R}^a - \mathbf{R}^{a'}|}.\end{aligned} \tag{2.2}$$

Here, the Hamiltonian $\hat{H}_{\text{full}}$ contains (ordered as in the equation) the kinetic energy of the electrons, the kinetic energy of the atomic nuclei and the interactions of the electrons with the atomic cores as well as the interaction among electrons and the interaction among cores, respectively.

Already in 1929 Dirac [3] stated that the Equations (2.1) and (2.2) describe "a large part of physics and the whole of chemistry"but "are too complicated to be soluble"analytically for more than two particles. However, the very different scale of masses, $m^a \approx 3676\, Z^a\, m_e$, justifies the separation of motions according to the different time scales, the so called *Born-Oppenheimer* approximation [4], has shown to hold in most cases. It fails only in very special scenarios where *vibronic* (combined vibrational and electronic) states play a central role. It is possible to treat

only the electronic part of the Hamiltonian quantum mechanically assuming the atomic positions $\mathbf{R}^a$ to be parameters. The atomic motion can be treated classically neglecting the quantum nature of their vibrational motion.

The many-body (MB) Schrödinger equation in the Born-Oppenheimer approximation forms the basis for a (non-relativistic) quantum mechanical description of the interacting electron system at zero temperature, i.e. no atomic movements are considered. It is given by

$$i\frac{\partial}{\partial t}|\Psi_{MB}\rangle = \hat{H}_{MB}|\Psi_{MB}\rangle \tag{2.3}$$

with the electronic Hamiltonian

$$\hat{H}_{MB}\{\mathbf{R}^{a_1}, \mathbf{R}^{a_2}, \dots\} = \sum_i^{N_e} \frac{\hat{\mathbf{p}}_i^2}{2} + \sum_i^{N_e}\sum_a^{N_a} \frac{Z^a}{|\hat{\mathbf{r}}_i - \mathbf{R}^a|} + \sum_i^{N_e}\sum_{j<i} \frac{1}{|\hat{\mathbf{r}}_j - \hat{\mathbf{r}}_i|}, \tag{2.4}$$

where the atomic positions $\{\mathbf{R}^{a_1}, \mathbf{R}^{a_2}, \dots\}$ enter as parameters rather than degrees of freedom.

The challenge arising from this equation is the dimensionality of the underlying Hilbert space. Considering for example a system where each particle has two accessible eigenstates, e.g. a spin-$\frac{1}{2}$ system, the many-body Hilbert space is 2^N-dimensional. The exponential growth makes it problematic to find eigensolutions of the Schrödinger equation, since conventional numerical solvers for eigenvalue problems scale with the third power of the dimension. This leads to a total workload proportional to $\propto 2^{3N}$ for finding the exact solutions of a spin system with N particles.

The problem becomes even more challenging giving more degrees of freedom to the electrons. Quantum mechanics tells us to express position and momentum of the electrons by a continuous distribution function. So we have to consider an entire function space to represent their state. Most approximations are based on truncating these function spaces to subspaces of a finite number of dimensions. However, there is a constant tradeoff: On the one hand the number of basis functions has to be kept large to preserve the accuracy of results. On the other hand the number of basis functions needs to be kept small at the same time since using d basis requires a storage of d^N numbers and the number of computation operations scales proportional to $\propto d^{3N}$ which makes this practically impossible already for systems with a few electrons.

In the following we consider the electronic Hamiltonian of the many body sys-

tem of interacting electrons

$$\hat{H}_{MB} = \sum_{i}^{N_e} \frac{\hat{\mathbf{p}}_i^2}{2} + \sum_{i}^{N_e} V_{ext}(\hat{\mathbf{r}}_i) + \sum_{i}^{N_e} \sum_{j<i}^{N_e} \frac{1}{|\hat{\mathbf{r}}_j - \hat{\mathbf{r}}_i|} \tag{2.5}$$

where $V_{ext}(\hat{\mathbf{r}})$ stands for the one-particle external potential of the atom cores like the electron-core interaction in Equation (2.2). Optionally, additional electric or magnetic fields can be included into the external potential.

The MB wave function for the electrons needs to be antisymmetric under exchange of any two particles since electrons are fermions. Consider a two-particle wave function $\Psi_{MB}(\mathbf{r}_1, s_1; \mathbf{r}_2, s_2)$ with the spatial coordinate $\mathbf{r}_i$ and the spin state s_i of the i-th particle. The Pauli exclusion principle demands that no two fermions occupy the same state. The consequence is a wave function that is antisymmetric under the exchange of the particles, i.e. simultaneously interchanging $\mathbf{r}_1 \leftrightarrow \mathbf{r}_2$ and $s_1 \leftrightarrow s_2$ produces a factor -1 such that

$$\Psi_{MB}(\mathbf{r}_2, s_2; \mathbf{r}_1, s_1) = -\Psi_{MB}(\mathbf{r}_1, s_1; \mathbf{r}_2, s_2). \tag{2.6}$$

Now, if $\mathbf{r}_1 = \mathbf{r}_2 = \mathbf{r}$ and $s_1 = s_2 = s$ holds this results in

$$\Psi_{MB}(\mathbf{r}, s; \mathbf{r}, s) = -\Psi_{MB}(\mathbf{r}, s; \mathbf{r}, s), \tag{2.7}$$

i.e. $\Psi_{MB}(\mathbf{r}, s; \mathbf{r}, s) = 0$. Therefore, no two electrons can simultaneously have the same position and spin coordinates. This holds for more than two particles, too. In practice this means that two electrons with the same spin will avoid being near to each other, giving a minimum in the spatial electron-electron correlation function known as the exchange hole. Because the electrons avoid coming close to each other, where the energy contributions from the Coulomb repulsion are high, the exchange hole leads to a lowering of the total energy.

2.1.1. Hartree- and Hartree-Fock approach

A first attempt to model the many-body wave function is a product ansatz (sometimes called Hartree ansatz) of one-particle wave functions

$$\Psi_H(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N) = \phi_{q_1}(\mathbf{r}_1)\, \phi_{q_2}(\mathbf{r}_2) \cdots \phi_{q_N}(\mathbf{r}_N), \tag{2.8}$$

where the q_i are different sets of quantum numbers. This wave function can easily be stored due to the separation of variables. A basis set of size d would then require the storage of Nd numbers. However, the Hartree ansatz leads to a description of the electrons without the explicit consideration of the Pauli principle. This was

fixed by the ansatz of a determinant of single particle wave functions proposed by Slater [61]

$$\Psi_{HF}(\mathbf{r}_1,\ldots,\mathbf{r}_n) = \begin{vmatrix} \phi_1(\mathbf{r}_1) & \phi_2(\mathbf{r}_1) & \cdots & \phi_n(\mathbf{r}_1) \\ \phi_1(\mathbf{r}_2) & \phi_2(\mathbf{r}_2) & & \phi_n(\mathbf{r}_2) \\ \vdots & & \ddots & \vdots \\ \phi_1(\mathbf{r}_n) & \phi_2(\mathbf{r}_n) & \cdots & \phi_n(\mathbf{r}_n) \end{vmatrix}. \tag{2.9}$$

The mathematical construct of determinants intrinsically satisfies the antisymmetry constraint imposed by the Pauli principle. Variation of the total energy

$$E_{tot} = \langle \Psi_{MB} | \hat{H}_{MB} | \Psi_{MB} \rangle \tag{2.10}$$

with respect to the single particle states $\phi_i(\mathbf{r})$ leads to the Hartree-Fock [62, 63] equations

$$\left(-\frac{1}{2}\Delta + \hat{V}_{ext}(\mathbf{r})\right)|\phi_i\rangle + \left(\sum_k \langle \phi_k | \frac{1}{|\mathbf{r}-\mathbf{r}'|} | \phi_k \rangle\right)|\phi_i\rangle - \sum_k \left(\langle \phi_k | \frac{1}{|\mathbf{r}-\mathbf{r}'|} | \phi_i \rangle\right)|\phi_k\rangle = \epsilon_i^{HF}|\phi_i\rangle \tag{2.11}$$

with the Hartree-Fock energy parameters ϵ_i^{HF}. This leads to a computational very expensive scheme that does not contain the full electron-electron interaction. Even though the exchange interaction is treated in an exact manner the effects of electron-electron correlations are not included.
Correlations in the statistical meaning are cross dependencies between the particles coordinates and spins. Assuming the many-body wave function as a product of single particle states as Equation (2.8) gives a totally uncorrelated description for the electrons, i.e. their probability distributions are independent of each other.

A rather different approach towards a solution of the MB problem was found by considering the degrees of freedom of the electron density rather than the full MB wave function.

2.2. Foundation of Density Functional Theory

The two theorems by Hohenberg and Kohn [1] build up the foundation of density functional theory. Their first theorem states that the electronic ground-state density $n_0(\mathbf{r})$ of a system determines the external potential $V(\mathbf{r})$ in a unique way such that also the ground-state many-body state $|\Psi_0\rangle$ can be found. As a consequence

all observable quantities are indirect functionals of the ground-state density $n_0(\mathbf{r})$. This is of particular interest for the expectation value of the Hamiltonian, i.e. the total energy. The total energy functional is defined as

$$E_{tot}[n] = \min_{\Psi \to n} \langle \Psi | \hat{T} + \hat{V}_{ee} | \Psi \rangle + \int d^3\mathbf{r}\, V(\mathbf{r})\, n(\mathbf{r}) \tag{2.12}$$

where the first expression is called universal functional. The minimization is considered over all N_e-electron states Ψ that lead to the density $n(\mathbf{r})$. $\hat{V}_{ee}$ is the electron-electron interaction $1/|\mathbf{r}_i - \mathbf{r}_j|$ for $i \neq j$. According to the second theorem the ground-state density $n_0(\mathbf{r})$ minimizes the total energy functional under the constraint of particle conservation $\int d^3\mathbf{r}\, n_0(\mathbf{r}) = N_e$. Nevertheless, the practical use of these theorems is small as long as we have no way to evaluate the universal functional.

2.2.1. Thomas-Fermi approach

The Thomas-Fermi approach offers a crude approximation for the universal functional $\langle \Psi | \hat{T} + \hat{V}_{ee} | \Psi \rangle$ in terms of the density alone. Here, the electron-electron interaction is replaced by the Hartree energy

$$U_H[n] = \frac{1}{2} \iint d^3\mathbf{r} d^3\mathbf{r}' \frac{n(\mathbf{r})\, n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \tag{2.13}$$

and the kinetic energy is formulated as a direct function of the density

$$E_{kin}^{TF}[n] = \frac{3}{10} (3\pi^2)^{\frac{2}{3}} \int d^3\mathbf{r}\, n^{\frac{5}{3}}(\mathbf{r}). \tag{2.14}$$

The variation of the total energy with respect to the density leads to the Thomas-Fermi equation

$$\frac{1}{2} (3\pi^2)^{\frac{2}{3}} n^{\frac{2}{3}}(\mathbf{r}) + V_H(\mathbf{r}) + V(\mathbf{r}) = \mu \tag{2.15}$$

where μ is a Lagrange multiplier that ensures the conservation of the particle number. $V_H(\mathbf{r})$ is the Hartree potential defined by

$$V_H[n](\mathbf{r}) = \frac{\delta U_H[n]}{\delta n(\mathbf{r})} = \int d^3\mathbf{r}' \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}. \tag{2.16}$$

Many electronic effects are neglected here as for example the exchange interaction that promotes magnetism is missing. As well, it is not accounted for correlation effects. Furthermore, the approximation for the kinetic energy is too inaccurate to even describe covalent bonding [64]. Nevertheless, the idea of a kinetic energy $E_{kin}[n]$ as a direct functional of the density is a promising candidate for a practical method [65].

2.2.2. Kohn-Sham system

Kohn and Sham (1965) [2] showed that an auxiliary system of non-interacting electrons can be constructed which reproduces the ground-state density $n_0(\mathbf{r})$ of the complicated many-electron system, henceforth called KS-system. Thus, $n_0(\mathbf{r})$ can be calculated by solving inexpensive single-particle Schrödinger equations

$$\hat{H}^{KS}|\psi_i^{KS}\rangle = \left(\hat{T} + \hat{V}_{eff}\{n\}\right)|\psi_i^{KS}\rangle = \epsilon_i^{KS}|\psi^{KS}\rangle \tag{2.17}$$

and occupy the N_e lowest states according to the Pauli principle. The density is then given by

$$n_0(\mathbf{r}) = \sum_i f^{FD}(\epsilon_i - \epsilon_F)|\psi_i^{KS}(\mathbf{r})|^2 \tag{2.18}$$

with the Fermi level ϵ_F. In Equation (2.17) the quantity ϵ^{KS} has the unit of an energy but does not necessarily represent a physically measurable quantity because it is the single-particle energy of an auxiliary, non-interacting system. From a mathematical point of view these energies rather need to be understood as Lagrange multipliers that arise due the conservation of the particle number, N_e. Nevertheless, the Kohn-Sham energies ϵ^{KS} are often treated as physically meaningful and so are derived quantities such as the density of states.

The total energy of the interacting system is expressed as

$$E_{tot}[n] = T_s[n] + U_H[n] + E_{xc}[n] + \int d^3\mathbf{r}\, V_{ext}(\mathbf{r})\, n(\mathbf{r}). \tag{2.19}$$

with the kinetic energy of the single particle states

$$T_s[n] = \sum_i \langle\psi_i^{KS}|\hat{T}|\psi_i^{KS}\rangle \tag{2.20}$$

Here, $E_{xc}[n]$ contains energy contributions due to exchange and correlation effects and the kinetic energy difference of non-interacting and interacting system. For the ground-state density, the variation of $E^{KS}[n]$ with respect to the density must become stationary, i.e.

$$\frac{\delta T_s}{\delta n(\mathbf{r})} + V_H(\mathbf{r}) + \frac{\delta E_{xc}}{\delta n(\mathbf{r})} + V_{ext}(\mathbf{r}) = \mu. \tag{2.21}$$

Thus, the effective potential operator $\hat{V}_{eff}$ includes the classical electrostatic interaction with the electron density given by the Hartree potential $V_H(\mathbf{r})$ defined in (2.16), the external potential $V_{ext}(\mathbf{r})$ and all complicated many-electron exchange and correlation effects in the exchange-correlation potential $V_{xc}[n](\mathbf{r})$ which results from the variation of E_{xc} with respect to $n(\mathbf{r})$. The effective potential then reads

$$V_{eff}[n](\mathbf{r}) = V_H[n](\mathbf{r}) + V_{ext}(\mathbf{r}) + V_{xc}[n](\mathbf{r}). \tag{2.22}$$

Since the effective potential $V_{\text{eff}}[n](\mathbf{r})$ depends on the density $n(\mathbf{r})$ and the density, in returns, depends on the KS states $|\psi_i^{KS}\rangle$, these equations need to be solved self-consistently.

2.2.3. Spin DFT

Barth and Hedin [66] introduced the electron spin into the DFT formalism. Under the assumption that the spin orientation is collinear, two separate eigenvalue problems arise from the Kohn-Sham equation

$$\left(\hat{T} + \hat{V}_{\text{eff}}^{\uparrow} - \epsilon_{\text{KS}}^{\uparrow}\right) |\psi_{\text{KS}}^{\uparrow}\rangle = 0, \tag{2.23}$$

$$\left(\hat{T} + \hat{V}_{\text{eff}}^{\downarrow} - \epsilon_{\text{KS}}^{\downarrow}\right) |\psi_{\text{KS}}^{\downarrow}\rangle = 0. \tag{2.24}$$

The full density $n(\mathbf{r})$ is then obtained by summation over both spin densities $n^{\sigma}(\mathbf{r})$ with $\sigma \in \{\uparrow, \downarrow\}$. Furthermore, a magnetization density $m(\mathbf{r})$ arises from the difference of $n^{\uparrow}(\mathbf{r})$ and $n^{\downarrow}(\mathbf{r})$. The magnetization density may also be interpreted as an expectation value of the 2×2 Pauli matrix $\hat{\sigma}_z$. The aforementioned conservation of the total number of particles holds for the sum of both partial particle numbers $N^{\uparrow} + N^{\downarrow}$ whereas there is no constraint to the total magnetic moment of the system which is proportional to the difference of the two partial particle numbers. The effective potential $\hat{V}_{\text{eff}}^{\sigma}$ differs with respect to the spin σ since the exchange-correlation potential $\hat{V}_{\text{xc}}$ is now a functional of the total electron density $n(\mathbf{r})$ and the magnetization density $m(\mathbf{r})$. Furthermore, the external potential may contain a magnetic field contribution.

`jüRS` **Input Syntax:** `spin 2`

2.2.4. Local density approximation

The simplest kind of approximation for the exchange-correlation energy E_{xc} gives a local potential $V_{\text{xc}}[n](\mathbf{r})$ that includes interactions with the total density $n(\mathbf{r}')$ only at the position $\mathbf{r}' = \mathbf{r}$. Therefore, it is called the *local density approximation* (LDA). It is based on the assumption that the exchange interaction can be approximated by the exchange of a single electron with a homogeneous density of electrons all over space that has the value $n(\mathbf{r})$. The exchange energy $E_x[n]$ then is given as

$$E_x[n] = \int_V d^3\mathbf{r}\, n(\mathbf{r})\, \epsilon_x(n(\mathbf{r})) \tag{2.25}$$

$$\epsilon_x(n) = -\frac{3}{4\pi}(3\pi^2\, n)^{\frac{1}{3}} \tag{2.26}$$

where $\epsilon_x[n]$ is the exact exchange energy density for the homogeneous electron gas. The corresponding potential, arising from the functional derivative with respect to the density, is given by

$$V_x\{n(\mathbf{r})\} = \frac{\delta E_x(n)}{\delta n(\mathbf{r})} = -\frac{1}{\pi}(3\pi^2)^{\frac{1}{3}}\, n(\mathbf{r})^{\frac{1}{3}} \tag{2.27}$$

The missing correlation part, $E_c[n]$, is evaluated by a numerically approximated function that has been fitted to exact many-body results achieved from Quantum Monte-Carlo simulations as first done by Ceperly and Alder [67]. Various parametrizations can be found in the literature [68, 69]. For magnetic systems the exchange-correlation potential depends on the total density $n(\mathbf{r})$ and the magnetization density $m(\mathbf{r})$ where $n(\mathbf{r}) = n^{\uparrow}(\mathbf{r}) + n^{\downarrow}(\mathbf{r})$ and $m(\mathbf{r}) = n^{\uparrow}(\mathbf{r}) - n^{\downarrow}(\mathbf{r})$ and is then called local *spin* density approximation (LSDA). Figure 2.1 shows the values of the energy density $\epsilon_{xc}(n)$ and the potential $V_{xc}(n)$ as function of the density n in the flavor of Perdew and Zunger [69].

jüRS **Input Syntax:** `xc PZ81`

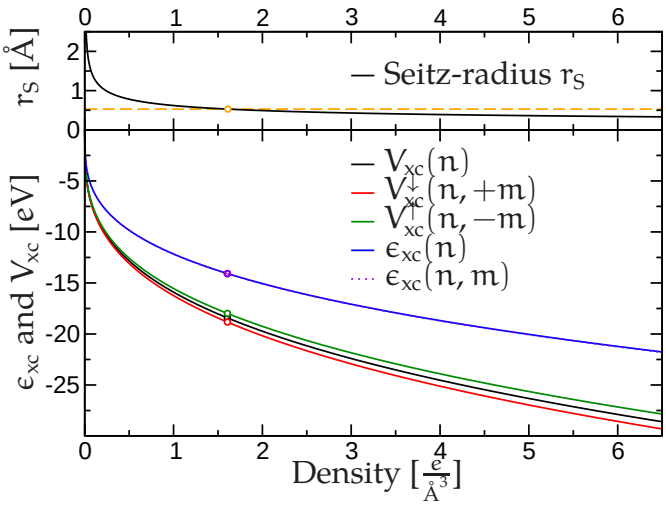


Figure 2.1.: Exchange-correlation potential $V_{xc}(n)$ and energy density $\epsilon_{xc}(n)$ as a function of the density n in the local density approximation (LDA) parametrized by Perdew and Zunger [69]. Including spin (LSDA), the potential also depends on the local magnetization density m. Here, $m = \pm 10\,\%$ of n is shown. Two different parametrizations for the correlation contribution are matched at a Seitz-radius $r_S = 1$ Bohr or 0.529 Å, see dots. The exchange-correlation energy density $\epsilon_{xc}(n)$ and energy density $\epsilon_{xc}(n, m)$ lie almost on top of each other.

CHAPTER 3

PROJECTOR AUGMENTED-WAVE METHOD

Considering all the electrons of a system explicitly in a DFT calculation can become very expensive and is, in most cases, not necessary for precise answers. Analyzing typical spectra of Kohn-Sham energies, we can classify two types of occupied Kohn-Sham states. The first type is core states that lie very deep in energy and are localized in spheres around the nuclear position. Secondly, we find the valence states with energies close to and at the Fermi level that form bands with a considerable dispersion. Comparing the core states from a calculation which includes all electrons to eigenstates of an isolated atom, we merely find tiny deviations in their spectrum and spatial distribution. We thus conclude that different chemical environments and coordinations of an atom hardly affect the properties of the core states. At most, their energetic positions might shift with respect to the potential average. Including the core states in the same way as the valence states in the calculation is problematic since the core states are only non-zero close to the nuclear site. There, the orthogonality requirement forces higher core states and valence states into rapid oscillations.

The projector augmente-wave method (PAW) proposed by Blöchl [22] is capable of an all-electron description of atoms in the framework of density functional calculations based on Kohn-Sham wave functions. However, a clear distinction between core states and valence states is usually introduced. Their treatment differs in the sense that the core states are restricted to the vicinity of the atom. The overlap of the core states on one site with core states of neighboring atoms is assumed to vanish. Core states are expected to mainly experience the spherically symmetric, deep, attractive atomic potential. We thus approximate the core states as discrete energy levels rather than energy bands with an energy dispersion, even for periodic arrangements of atoms. Valence states extend over several atomic sites and strongly experience the full structure of the system and, if applicable, its symmetry.

A criterion for identifying core and valence states is to compare their energies with the barrier height of the local effective potential between two neighboring atom sites. If the energy level is close to the barrier height or higher, the state needs to be treated as a valence state. For practical reasons, we should choose the core-valence separation such that the lowest valence state energy is separated by a relatively large gap from the highest core state energy. If the valence-core gap is too small, we have to deal with so-called semicore states, as for example in the case of 3d-states of zinc. Semicore states should be included in the set of valence bands because their spatial extent is usually larger than half the typical atom-atom distances. Therefore, we expect a non-negligible energy dispersion as depicted schematically in Figure 3.1. Figure 3.2 shows all atomic eigenenergies for $Z \in [1, 118]$ and where the separation between core and valence states is usually placed.

The introduction of core states together with the approximation of non-overlapping core state wave functions and those experiencing a spherical atomic potential reduces the number of bands to be calculated, since only valence bands will have to be treated to a spatial extent. Furthermore, the construction of the PAW projectors is simplified and the flexibility of core states is limited thus avoiding numerical difficulties.

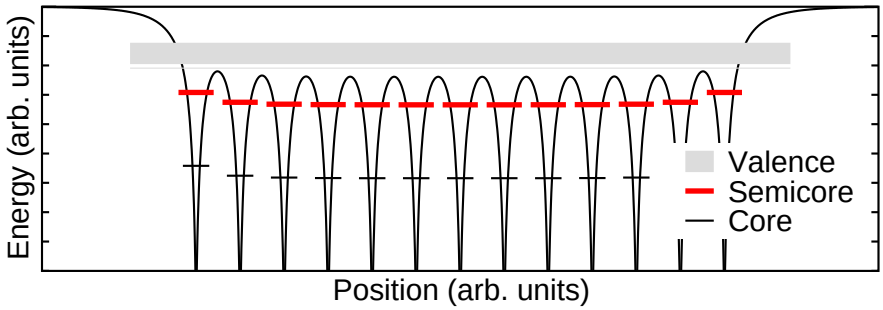


Figure 3.1.: Schematic classification of Kohn-Sham states in a solid by energy. Shown is the potential of a one-dimensional cluster of 13 atoms in real space. The bars stand for various KS states whereas their height indicates their position in energy, their width the energy dispersion and their length the extent in real space. Low-lying core states are strongly localized around the atomic nucleus and thus experience mainly the spherical potential. Valence state are delocalized over the entire system and are therefore sensitive to the environment. For semicore states, we need to decide whether they may be treated as core states or need to be included in the class of valence states.

Standard DFT pseudopotential calculations apply one occupied valence reference state per angular momentum ℓ as for example Troullier-Martins' norm-conserving pseudopotentials [28]. The PAW method is, in principle, able to use a larger num-

ber of valence reference states since there is no requirement for the pseudized (smooth) wave function to be nodeless [22]. This requires a very careful choice of projector functions, especially a treatment with more than one projector per angular momentum ℓ, to guarantee a proper description in the entire energy window containing two valence states of the same ℓ-symmetry.

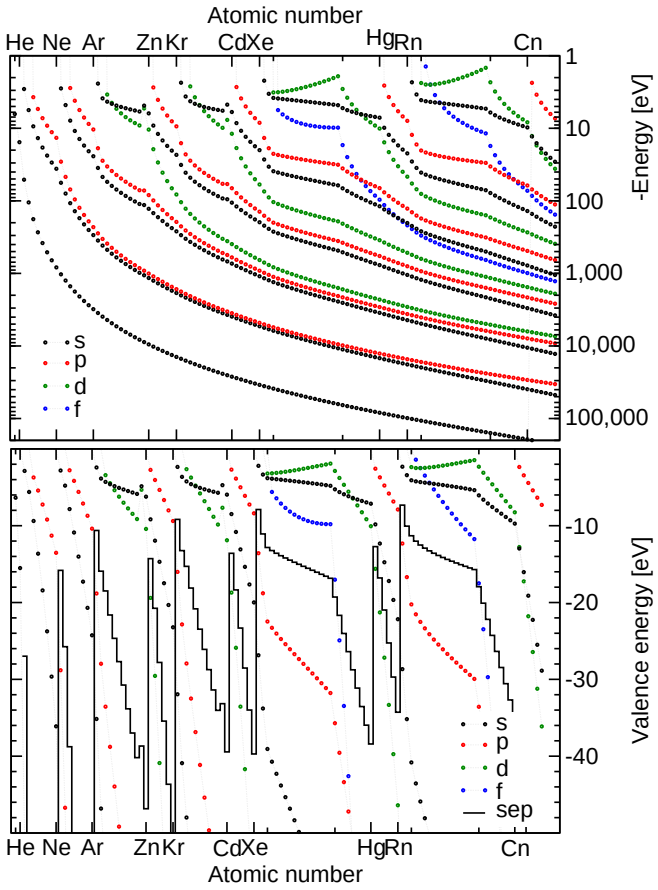


Figure 3.2.: Atomic KS energies for all atomic numbers up to Z=118. In the upper diagram, all levels are shown on a logarithmic scale. The lower graph shows only the valence energy range and indicates the separation of core and valence states (solid black line). The periodic shell structure becomes visible in the valence regime.

3.0.5. Spherical Approximation

The identification of core states and valence states introduced above allows further approximations. The strong degree of localization of core states around the atomic nuclei guarantees that the effect of non-spherical contributions $V_{\ell m}(r)$ ($r = |\mathbf{r}|$, $\ell > 0$) to the potential is negligibly small such that we can approximate core states

to experience the spherical part of the potential $V_{00}(|\mathbf{r}|)$ only. This leads to a degeneracy with respect to the magnetic quantum number m and a simpler treatment of the core states as $R^a_{nl}(|\mathbf{r}|)Y_{\ell m}(\hat{\mathbf{r}})$ on a radial grid.

3.0.6. Frozen-core Approximation

We can further assume that states which are low in energy are not very sensitive to the environment of the atom. A plausible explanation is the strong screening behavior of the valence states. Therefore, an enormous simplification of the method is to pre-compute the core states only once in an atomic all-electron calculation. We further integrate out the degrees of freedom of the core states and store only their density (and the sum of their kinetic energies, see Chapter A about the generation of PAW setups). Since the core density is not updated at a later moment, this approximation is referred to as frozen core.

3.1. Basics of PAW

The Projector Augmented Wave method [22] is a way to deal with the difficulties arising from a singular effective potential of the atomic nuclei that the electrons experience. There are two major aspects that lead to unfortunate convergence behavior with respect to the number of basis functions assuming that our basis functions are independent of the nuclear positions, as e.g. plane waves or equidistant grid points are. The first challenging aspect is the representation of the potential itself in terms of basis functions. An exact representation of the Coulomb potential that arises from the proton charges in the atomic nuclei would require an infinite number of plane waves or grid points. The second aspect is the representation of the (single-particle) wave functions in terms of the basis functions. The attractive core potential leads to a very high local kinetic energy density around the nuclei, even for the valence states, where the local kinetic energy density of a state Ψ is defined as

$$\epsilon_{\text{kin}}(\mathbf{r}) = \Psi^*(\mathbf{r}) \left(-\frac{1}{2}\Delta_{\mathbf{r}}\right) \Psi(\mathbf{r}). \tag{3.1}$$

Equivalently, the potential energy density of a state Ψ is defined as

$$\epsilon_{\text{pot}}(\mathbf{r}) = \Psi^*(\mathbf{r}) V_{\text{loc}}(\mathbf{r}) \Psi(\mathbf{r}). \tag{3.2}$$

The sum of kinetic energy density and potential energy density has to add up to the energy eigenvalue E times $|\Psi|^2$, assuming that Ψ is an eigenstate of the Hamiltonian $\hat{H} = \hat{T} + \hat{V}_{\text{loc}}(\mathbf{r})$. Hence,

$$\epsilon_{\text{pot}}(\mathbf{r}) + \epsilon_{\text{kin}}(\mathbf{r}) = E|\Psi(\mathbf{r})|^2. \tag{3.3}$$

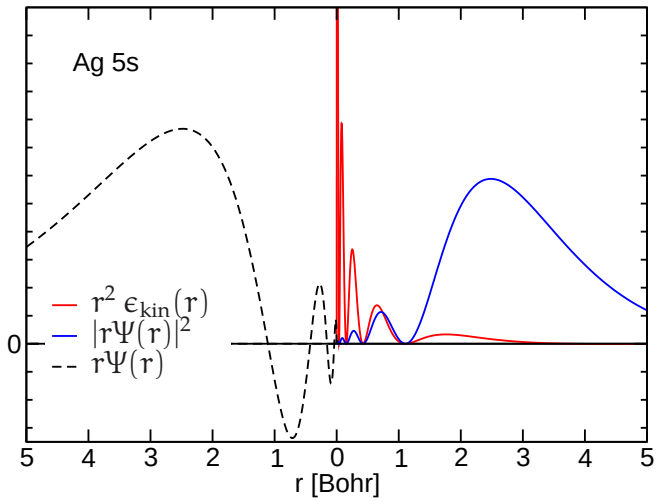


Figure 3.3.: Kinetic energy density of the Ag 5s valence state in an atomic calculation. The graph shows $r^2\,\epsilon_{kin}(r)$ for an isolated Silver atom. The kinetic energy density grows strongly (roughly as $\sim \frac{47}{r}$) approaching the singularity of the potential at $r = 0$. The left hand side shows $r\Psi(r)$ with its rapid oscillatory behavior inside $r_{max} = 1.32\,\text{Å}$.

Thus, a divergent local potential with the major contribution $-Z^a/|\mathbf{r}-\mathbf{R}^a|$ (Hartree atomic units) also leads to enormous values of the kinetic energy density of the eigenstates close to the nuclear site $\mathbf{R}^a$ of atom a. This corresponds to wave functions that show high frequency (short wavelength) oscillations near the nucleus for both, core and valence states. Additionally, *s*-like wave functions show a discontinuity of the first derivative (sometimes called *kink*) at the site of the singularity that cannot be represented by a finite set of smooth basis functions. A limited number of uniform basis functions, such as plane waves or an equivalent number of equidistant grid points, clearly cannot capture this behavior.

The PAW method is based on a linear transformation between the space of the true all-electron wave functions and smooth pseudo waves [22]. The characteristic property of the smooth waves is their slow variation with respect to the spatial coordinates which results in a fast convergence of their Fourier coefficients to zero for short wavelengths. An essential property of the function space of true wave functions is that it can capture both, fast and slow oscillations. Thus the smooth waves can be represented as well. The linear transformation $\hat{\mathcal{T}}$ acts onto the smooth wave function and returns the true wave function, i.e.

$$\hat{\mathcal{T}}|\tilde{\Psi}\rangle = |\Psi\rangle. \tag{3.4}$$

In the following, we use the tilde ($\tilde{}$) to denote all smooth quantities. In order to tackle the problem of high kinetic energy density of Kohn-Sham states in the vicinity of the atom cores we define the transformation to be a sum of single trans-

formation operations. Each single transformation is localized within a sphere $\mathcal{S}^a$ of radius r^a_{aug} around an atom site $\mathbf{R}^a$. Hence, the total transformation reads

$$\hat{\mathcal{T}} = \mathbf{1} + \sum_a \hat{\mathcal{T}}^a. \tag{3.5}$$

The restriction of the transformations $\hat{\mathcal{T}}^a$ into sphere shapes is convenient since the true potential in the crystal or molecule is dominated by the spherically symmetric contributions of the attractive nuclei. For the constructions of the transformation, the spherically symmetric potential of an isolated atom is assumed. In principle, other shapes could be used, as for example a regular dodecahedron for calculations in a regular face centered cubic (FCC) solid. However, the spherical shape is the most general and advantageous for the application within different geometries since it preserves spatial isotropy. Requesting the sphere radius r^a_{aug} to be small enough for the spheres not to overlap

$$S^a \cap S^{a'} = \emptyset \qquad \forall a, a', \tag{3.6}$$

simplifies the expressions for the expectation values of the transformed operators, as we will see later. Introducing non-overlapping spheres also defines two regions in space: The region inside any of the spheres $\bigcup_a \mathcal{S}^a$ and the region in between spheres $V \setminus (\bigcup_a \mathcal{S}^a)$, where V is the cell volume. The latter is called interstitial region (IR) in Augmented Plane Wave (APW) approaches [70]. Note that we use the same name here but the functionality of the IR in the PAW context is slightly different. The transformation $\hat{\mathcal{T}}$ is required to be unity inside the interstitial region since the atom-part of the transformation $\hat{\mathcal{T}}^a$ is only introducing changes inside the sphere volume $\mathcal{S}^a$, i.e.

$$\langle \mathbf{r}|\hat{\mathcal{T}}|\mathbf{r}'\rangle = \delta(\mathbf{r}-\mathbf{r}') \qquad \text{for } \mathbf{r}, \mathbf{r}' \in \text{IR}. \tag{3.7}$$

Furthermore, the requirement of non-overlapping spheres (Equation (3.6)) leads to the fact that the transformation operator does not relate spaces from different spheres, i.e.

$$\langle \mathbf{r}|\hat{\mathcal{T}}|\mathbf{r}'\rangle = 0 \qquad \text{for } \mathbf{r} \in \mathcal{S}^a \text{ and } \mathbf{r}' \in \mathcal{S}^{a'},\ \forall\{(a, a')|a \neq a'\}. \tag{3.8}$$

Since the transformation $\hat{\mathcal{T}}$ is linear, we can describe it by its action onto a basis. Using the same basis for both, the space of the smooth waves $\tilde{\Psi}$ and the space of the true waves Ψ, would not lead to a transformation useful for our particular problem. We want to keep the basis for the representation of the smooth waves as small as possible enhancing fast computations. We thus seek to find two different basis sets which are both strictly localized inside the sphere in order to fulfill Equation (3.8). To this end, we construct two sets of partial waves. First, the true partial waves

$\{|\phi_i^a\rangle\}$ that are suited to describe the behavior of the true wave functions inside the sphere. Further, we construct a set of smooth partial waves $\{|\tilde{\phi}_i^a\rangle\}$ that are suited to describe smoothly varying wave function $\tilde{\Psi}$ and need to match the true partial waves ϕ_i^a at the sphere boundary. The action of the PAW transformation consists of adding the wave function correction $\{|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\}$ in true space. The dual basis for the smooth space is spanned by a third set, the smooth projector functions $\{\langle\tilde{p}_i^a|\}$. Both sets, the correction functions $\{|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\}$ and the projector functions $\{\langle\tilde{p}_i^a|\}$, are localized within the atomic sphere $\mathcal{S}^a$

$$\langle \mathbf{r}| \left(|\phi^a\rangle - |\tilde{\phi}^a\rangle\right) = 0, \qquad \langle\tilde{p}^a|\mathbf{r}\rangle = 0 \qquad \forall\, \mathbf{r} \in \mathcal{V}\backslash\mathcal{S}^a. \tag{3.9}$$

Using

$$\hat{\mathcal{T}}^a = \sum_i \left(|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\right) \langle\tilde{p}_i^a|, \tag{3.10}$$

one can show that $\hat{\mathcal{T}}$ satisfies Equation (3.4) if the completeness

$$\sum_i |\tilde{\phi}_i^a\rangle\langle\tilde{p}_i^a| = \mathbf{1} \qquad \forall a \tag{3.11}$$

and the duality

$$\langle\tilde{p}_i^a|\tilde{\phi}_j^b\rangle = \delta_{ij}\,\delta_{ab} \tag{3.12}$$

are fullfilled. We assume that $\{|\phi_i^a\rangle\}$ and $\{|\tilde{\phi}_i^a\rangle\}$ are suitable to describe the true wave functions $|\Psi\rangle$ and the smooth wave functions $|\tilde{\Psi}\rangle$ inside the spheres, respectively. The full transformation then reads

$$\hat{\mathcal{T}} = \mathbf{1} + \sum_a \hat{\mathcal{T}^a} = \mathbf{1} + \sum_{ai} \left(|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\right) \langle\tilde{p}_i^a|. \tag{3.13}$$

As assumed, any smooth wave function $|\tilde{\Psi}\rangle$ can locally (inside a sphere $\mathcal{S}^a$) be expanded into smooth wave functions $|\tilde{\phi}_j^a\rangle$ as $|\tilde{\Psi}\rangle = \sum_j c_j^a|\tilde{\phi}_j^a\rangle$. Then, the corresponding true wave function $|\Psi\rangle$ has an expansion with the same coefficients c_j^a as $|\Psi\rangle = \sum_j c_j^a|\phi_j^a\rangle$ since this ensures an equivalence of $\tilde{\Psi}$ and Ψ in the interstitial region.

Applying the transformation $\hat{\mathcal{T}}$ onto any $|\tilde{\Psi}\rangle$ we see that

$$\begin{aligned}
\hat{\mathcal{T}}|\tilde{\Psi}\rangle &= |\tilde{\Psi}\rangle + \sum_{ai} \left(|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\right) \langle\tilde{p}_i^a|\tilde{\Psi}\rangle \\
&= |\tilde{\Psi}\rangle + \sum_{ai} \left(|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\right) \langle\tilde{p}_i^a| \sum_{jb} c_j^b|\tilde{\phi}_j^b\rangle \\
&= |\tilde{\Psi}\rangle + \sum_{ai} \left(|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\right) \sum_{jb} c_j^b\, \delta_{ij}\,\delta_{ab} \\
&= |\tilde{\Psi}\rangle + \sum_{ai} \left(|\phi_i^a\rangle - |\tilde{\phi}_i^a\rangle\right) c_i^a = |\Psi\rangle.
\end{aligned} \tag{3.14}$$

The smooth wave function $|\tilde{\Psi}\rangle$ is corrected inside each sphere to be the true wave function $|\Psi\rangle$. The inner products $\langle\tilde{p}_i^a|\tilde{\Psi}\rangle$ between smooth wave functions and smooth projector functions are called projection coefficients c_i^a and are essential quantities for any practical implementation of the PAW method.

The completeness relation (3.11) is only exact in the limit of an infinite number of partial waves. However, we seek to minimize the errors arising from a finite basis set, i.e. a set of limited number of partial waves $\{(|\phi_i^a\rangle, |\tilde{\phi}_i^a\rangle, \langle\tilde{p}_i^a|)\}$. An efficient basis for the true wave functions $\{\phi_i\}$ is found by an outwards integration of the all-electron Kohn-Sham equation under the assumption of a spherical reference potential $V_{\text{ref}}(r)$. The strength of the attractive nuclear Coulomb potential clearly dominates the ingredients to the local potential which justifies the spherical approximation. A computational benefit is the energetic m-degeneracy of the solutions. A set of chosen energies $\epsilon_{\ell n}$ will yield a set of radial all-electron wave functions $\phi_{\ell n}(r)$ (true partial waves) which are not required to be eigenstates of the underlying Hamiltonian. These functions are usually generated in a pre-computation where the applied potentials are effective spherical potentials of a chosen atomic configuration in an isolated environment. More about integration techniques and approximations can be found in in Section A.0.4, and further details about the generation process of the smooth partial waves $|\tilde{\phi}_i^a\rangle$ from $|\phi_i^a\rangle$ are given in Section A.0.2.

Using a finite number of partial waves $\{(|\phi_i\rangle, |\tilde{\phi}_i\rangle), \langle\tilde{p}_i|\}$ makes the PAW transformation not generally bijective. However, it works well even with a very small number of basis functions, because we can assume that the important valence states stay within a certain energy window of a few eV around the average band energies or atomic energy levels. Within the Linearized Augmented Plane Wave (LAPW) methods one can show that taking the energy derivative is equivalent to introducting a second projector. It has been reported in the literature that two projectors per ℓ-channel are sufficient for almost all cases [22]. Compared to pseudopotential approaches with only one projector function, the PAW method thus promises better transferability due to the increased flexibility in the spheres. DFT methods that rely on unoccupied states will probably need more than two partial waves per ℓ for sufficient flexibility at higher energies.

3.2. PAW Transformed Operators

Considering the all-electron operator $\hat{O}$ and its matrix element with respect to the all-electron states $|\Psi_n\rangle$ and $|\Psi_m\rangle$, we insert Equation (3.4) to find its matrix element evaluated in terms of smooth wave functions $|\tilde{\Psi}_n\rangle$ and $|\tilde{\Psi}_m\rangle$. Note that we use the $|\rangle$-notation here for all kinds of wave functions even though these are in general

not normalized to unity.

$$
\begin{aligned}
&\langle\Psi_m|\hat{O}|\Psi_n\rangle \qquad (3.15)\\
=&\langle\tilde{\Psi}_m|\hat{\mathcal{T}}^\dagger\hat{O}\hat{\mathcal{T}}|\tilde{\Psi}_n\rangle\\
=&\langle\tilde{\Psi}_m|\left(\mathbf{1}+\sum_b \mathcal{T}^{b\dagger}\right)\hat{O}\left(\mathbf{1}+\sum_a \mathcal{T}^a\right)|\tilde{\Psi}_n\rangle\\
=&\langle\tilde{\Psi}_m|\left(\mathbf{1}+\sum_{bj}|\tilde{p}_j^b\rangle\langle\phi_j^b-\tilde{\phi}_j^b|\right)\hat{O}\left(\mathbf{1}+\sum_{ai}|\phi_i^a-\tilde{\phi}_i^a\rangle\langle\tilde{p}_i^a|\right)|\tilde{\Psi}_n\rangle\\
=&\langle\tilde{\Psi}_m|\hat{O}|\tilde{\Psi}_n\rangle+\sum_{aibj}c_{mj}^{b*}\langle\phi_j^b-\tilde{\phi}_j^b|\hat{O}|\phi_i^a-\tilde{\phi}_i^a\rangle c_{ni}^a\\
&+\sum_{bj}c_{mj}^{b*}\langle\phi_j^b-\tilde{\phi}_j^b|\hat{O}|\tilde{\Psi}_n\rangle+\sum_{ai}\langle\tilde{\Psi}_m|\hat{O}|\phi_i^a-\tilde{\phi}_i^a\rangle c_{ni}^a
\end{aligned}
$$

with the projection coefficient $c_{ni}^a=\langle\tilde{p}_i^a|\tilde{\Psi}_n\rangle$.

3.2.1. Local and Semi-local Operators

The above expressions hold for any general operator $\hat{O}$. However, the density operator and thus also the overlap operator, the effective potential and the kinetic energy operator are local and semi-local operators, respectively. For these operators the expression can be simplified considerably. The correction functions $\langle\mathbf{r}|\phi^b-\tilde{\phi}^b\rangle$ are localized within the sphere $\mathcal{S}^b$. Thus all mixed-spheres expectation values $\langle\phi^a-\tilde{\phi}^a|\hat{O}_{loc}|\phi^b-\tilde{\phi}^b\rangle$ vanish for different spheres $a\neq b$ if the requirement of disjoint spheres (Equation 3.7) is fulfilled. This implies for the mixed expressions, second line in Equation (3.15), that we can use the expansion of $|\tilde{\Psi}_n\rangle$ and $\langle\tilde{\Psi}_m|$ inside the spheres. Inserting the completeness relation from Equation (3.11) we obtain

$$
\begin{aligned}
\langle\phi_j^b-\tilde{\phi}_j^b|\hat{O}_{loc}|\tilde{\Psi}_n\rangle&=\langle\phi_j^b-\tilde{\phi}_j^b|\hat{O}_{loc}\sum_i|\tilde{\phi}_i^b\rangle\langle\tilde{p}_i^b|\tilde{\Psi}_n\rangle,\\
\langle\tilde{\Psi}_m|\hat{O}_{loc}|\phi_i^a-\tilde{\phi}_i^a\rangle&=\langle\tilde{\Psi}_m|\sum_j|\tilde{p}_j^a\rangle\langle\tilde{\phi}_j^a|\hat{O}_{loc}|\phi_i^a-\tilde{\phi}_i^a\rangle,
\end{aligned}
$$

which yield

$$\begin{aligned}
&\langle\Psi_m|\hat{O}_{loc}|\Psi_n\rangle \qquad (3.16)\\
=&\langle\tilde{\Psi}_m|\hat{O}_{loc}|\tilde{\Psi}_n\rangle\\
+&\sum_{aibj}\delta_{ab}\,c^{b*}_{jm}c^{a}_{ni}\left(\langle\phi^b_j|\hat{O}_{loc}|\phi^a_i\rangle-\langle\tilde{\phi}^b_j|\hat{O}_{loc}|\phi^a_i\rangle-\langle\phi^b_j|\hat{O}_{loc}|\tilde{\phi}^a_i\rangle+\langle\tilde{\phi}^b_j|\hat{O}_{loc}|\tilde{\phi}^a_i\rangle\right)\\
+&\sum_{bj}c^{b*}_{mj}\sum_i\left(\langle\phi^b_j|\hat{O}_{loc}|\tilde{\phi}^b_i\rangle-\langle\phi^b_j|\hat{O}_{loc}|\tilde{\phi}^b_i\rangle\right)c^{b}_{ni}\\
+&\sum_{ai}\sum_j c^{a*}_{mj}\left(\langle\tilde{\phi}^a_j|\hat{O}_{loc}|\phi^a_i\rangle-\langle\tilde{\phi}^a_j|\hat{O}_{loc}|\tilde{\phi}^a_i\rangle\right)c^{a}_{ni}.
\end{aligned}$$

Renaming all b to a in the third line we find that all terms of the kind $\langle\phi|\hat{O}_{loc}|\tilde{\phi}\rangle$ (expectation value of a mixed pair of smooth and true partial waves) cancel out exactly. Also, the expressions $\langle\tilde{\phi}^a_i|\hat{O}_{loc}|\tilde{\phi}^b_j\rangle$ from the second and third line cancel. Thus a (semi-)local operator matrix elements in the PAW formalism consists of three contributions,

$$\langle\Psi_m|\hat{O}_{loc}|\Psi_n\rangle=\langle\tilde{\Psi}_m|\hat{O}_{loc}|\tilde{\Psi}_n\rangle+\sum_{aij}c^{a*}_{mj}\left(\langle\phi^a_j|\hat{O}_{loc}|\phi^a_i\rangle-\langle\tilde{\phi}^a_j|\hat{O}_{loc}|\tilde{\phi}^a_i\rangle\right)c^{a}_{ni}. \qquad (3.17)$$

Since we did not impose any restrictions onto the states Ψ_n and Ψ_m, Equation (3.17) allows us read off a representation of any (semi-)local operator with respect to smooth states $\tilde{\Psi}$

$$\hat{\mathcal{T}}^\dagger\hat{O}_{loc}\hat{\mathcal{T}}^\dagger=\hat{O}_{loc}+\sum_{aij}|\tilde{p}^a_j\rangle\left(\langle\phi^a_j|\hat{O}_{loc}|\phi^a_i\rangle-\langle\tilde{\phi}^a_j|\hat{O}_{loc}|\tilde{\phi}^a_i\rangle\right)\langle\tilde{p}^a_i|. \qquad (3.18)$$

3.2.2. Density Operator

The electron density is the expectation value of the local operator $\hat{\rho}(\mathbf{r})=|\mathbf{r}\rangle\langle\mathbf{r}|$. Therefore, we can apply Equation (3.17) to find the density of a Kohn-Sham state $|\Psi\rangle$

$$\langle\Psi|\mathbf{r}\rangle\langle\mathbf{r}|\Psi\rangle=\langle\tilde{\Psi}|\mathbf{r}\rangle\langle\mathbf{r}|\tilde{\Psi}\rangle+\sum_{aij}c^{a*}_i\left(\langle\phi^a_i|\mathbf{r}\rangle\langle\mathbf{r}|\phi^a_j\rangle-\langle\tilde{\phi}^a_i|\mathbf{r}\rangle\langle\mathbf{r}|\tilde{\phi}^a_j\rangle\right)c^a_j. \qquad (3.19)$$

The transformed valence density operator then reads

$$\hat{\rho}(\mathbf{r})=\hat{\tilde{\rho}}(\mathbf{r})+\sum_{aij}|\tilde{p}^a_i\rangle\left(\langle\phi^a_i|\mathbf{r}\rangle\langle\mathbf{r}|\phi^a_j\rangle-\langle\tilde{\phi}^a_i|\mathbf{r}\rangle\langle\mathbf{r}|\tilde{\phi}^a_j\rangle\right)\langle\tilde{p}^a_j|. \qquad (3.20)$$

In various situations, the smooth and true densities in the atomic sphere need to be treated. Therefore, an atomic density matrix D^a is defined by

$$D^a_{ij\sigma}=\sum_{nk}f_{n\sigma k}\,c^{a*}_{in\sigma\mathbf{k}}\,c^a_{jn\sigma\mathbf{k}} \qquad (3.21)$$

With this, the representation of the valence density reads

$$n_v^\sigma(\mathbf{r}) = \tilde{n}_v^\sigma(\mathbf{r}) + \sum_a D_{ij\sigma}^a \left(\phi_i^a(\mathbf{r})\phi_j^a(\mathbf{r}) - \tilde{\phi}_i^a(\mathbf{r})\tilde{\phi}_j^a(\mathbf{r})\right) \tag{3.22}$$

where the smooth valence density $\tilde{n}_v^\sigma(\mathbf{r})$ is constructed from the smooth KS wave functions

$$\tilde{n}_v^\sigma(\mathbf{r}) = \sum_{n\mathbf{k}} f_{n\sigma\mathbf{k}} \left|\tilde{\Psi}_{n\sigma\mathbf{k}}(\mathbf{r})\right|^2. \tag{3.23}$$

3.2.3. Overlap Matrix

The PAW transformation can be —but does not necessarily have to be —norm-conserving. However, the true wave functions must be normalizable in order to find the correct number of particles in the system. The requirement $\langle\Psi|\Psi\rangle = 1$ can be translated into the space of smooth wave functions by writing the norm as the expectation value of the unity operator which is clearly local. According to Equation (3.17), we find

$$1 = \langle\Psi|\mathbf{1}|\Psi\rangle = \langle\tilde{\Psi}|\tilde{\Psi}\rangle + \sum_{aij} c_i^{a*} \left(\langle\phi_i^a|\phi_j^a\rangle - \langle\tilde{\phi}_i^a|\tilde{\phi}_j^a\rangle\right) c_j^a. \tag{3.24}$$

We have therefore introduced an overlap operator $\hat{S}$ that is a metric to normalize the smooth wave functions $\langle\tilde{\Psi}|\hat{S}|\tilde{\Psi}\rangle = 1$, where

$$\hat{\mathcal{T}}^\dagger\hat{\mathcal{T}} = \hat{S} = \mathbf{1} + \sum_{aij} |\tilde{p}_i^a\rangle \Delta q_{ij}^a \langle\tilde{p}_j^a| \tag{3.25}$$

with the norm deficit matrix $\Delta q_{ij}^a = \langle\phi_i^a|\phi_j^a\rangle - \langle\tilde{\phi}_i^a|\tilde{\phi}_j^a\rangle$. For the special case of a norm-conserving construction of the smooth partial waves, the entire norm deficit matrix vanishes and an implementation of the overlap operator becomes redundant; see also Section A.0.2. The norm deficit matrix is often called charge deficit matrix because the elementary charge e=1 in (Hartree) atomic units and the same matrix is applied in the context of the electrostatic monopole deficit. This ensures that the normalization condition $\langle\tilde{\Psi}|\hat{S}|\tilde{\Psi}\rangle = 1$ also leads to a normalized generalized density.

3.2.4. Kinetic Energy Operator

The kinetic energy operator relies on spatial derivatives and is thus not a local but a semi-local operator since we can compute derivatives of a function at a given point holding information about this point's surrounding. This is done on real-space grids with the finite difference approximation that leads to a non-local but

localized operator. The PAW transformation for the kinetic energy operator reads

$$\langle\Psi|\hat{T}|\Psi\rangle = -\frac{1}{2}\langle\tilde{\Psi}|\Delta_{\mathbf{r}}|\tilde{\Psi}\rangle + \sum_{aij} c_i^{a*}\left(\langle\phi_i^a| - \frac{1}{2}\Delta_{\mathbf{r}}|\phi_j^a\rangle - \langle\tilde{\phi}_i^a| - \frac{1}{2}\Delta_{\mathbf{r}}|\tilde{\phi}_j^a\rangle\right) c_j^a, \quad (3.26)$$

and the transformed operator with respect to the smooth wave functions is given as

$$\hat{\mathcal{T}}^\dagger\hat{T}\hat{\mathcal{T}} = -\frac{1}{2}\Delta_{\mathbf{r}} + \sum_{aij}|\tilde{p}_i^a\rangle\left(E_{ij}^{a,kin} - \tilde{E}_{ij}^{a,kin}\right)\langle\tilde{p}_j^a| \quad (3.27)$$

The matrix elements $E_{ij}^{a,kin}$ of the true and smooth partial waves with kinetic energy operator are evaluated on the radial grid. Furthermore, we can exploit that the true partial waves have been found as solutions of a local potential to a given energy eigenvalue, such that $\hat{T}|\phi_{\ell n}\rangle = [\epsilon_{\ell n} - V_{ref}^a(r)]|\phi_{\ell n}\rangle$ which helps to evaluate $E_{ij}^{a,kin}$.

As shown in the following chapter of this thesis the derivatives $\Delta_{\mathbf{r}}$ are approximated by finite-differences on the real-space grid. This procedure involves a limited number if neighboring grid points such that the kinetic energy operator is non-local with a range finite $N_f h$. However, in the limit of small grid spacings, $h \to 0$, this range vanishes and the PAW transformation for local operators may still be applied.

3.2.5. Potential Operator

In the previous cases we used the PAW transformation in the sphere merely to replace the true wave functions by an expansion in smooth partial waves. The potential operator is treated differently in the sense that also the potential shape is replaced by a much smoother one. As mentioned in the beginning of the chapter, the singular Coulomb potential of the nucleus cannot be represented in a basis that captures smooth quantities only. We therefore augment the singularity and replace it with an —in principle arbitrary, but preferably —smooth potential $\tilde{V}$ inside the sphere $\mathcal{S}^a$

$$\langle\Psi|V|\Psi\rangle = \langle\tilde{\Psi}|\tilde{V}|\tilde{\Psi}\rangle - \sum_{aij} c_i^{a*}\left(\langle\phi_i^a|V|\phi_j^a\rangle - \langle\tilde{\phi}_i^a|\tilde{V}|\tilde{\phi}_j^a\rangle\right) c_j^a. \quad (3.28)$$

Thus, the potential operator with respect to the smooth wave functions reads

$$\hat{\mathcal{T}}^\dagger V\hat{\mathcal{T}} = \tilde{V} - \sum_{aij}|\tilde{p}_i^a\rangle\left(\langle\phi_i^a|V|\phi_j^a\rangle - \langle\tilde{\phi}_i^a|\tilde{V}|\tilde{\phi}_j^a\rangle\right)\langle\tilde{p}_j^a|. \quad (3.29)$$

The freedom in the shape of $\tilde{V}$ is usually restricted by the constraints that the local potential on the grid $\tilde{V}(\mathbf{r})$ is as smooth as possible. The construction of very smooth potentials can be achieved by a potential shape correction term, see Section A.0.2.

3.3. Comparison PAW and APW

In the Augmented Plane Wave method (APW, developed by C. Slater [61]) one divides space into two regions. The interstitial region (IR) and the so-called muffin-tin spheres (MT). Non-overlapping spheres are centered around each atomic nucleus. The interstitial region is the remaining space in between the sphere volumes. Choosing maximally large spheres (touching spheres), we can assume that valence states, densities and the effective potential are very smooth in the interstitial region and can therefore be well represented with a relatively modest number of plane waves. Inside the spheres a radial real-space grid is applied. A non-equidistant radial grid can easily capture the strong potential of the nucleus as well as rapid oscillations of wave functions and densities. The connection of the radial grid towards the interstitial basis function at the sphere boundary involves matching of values and derivatives. In the extension to Linearized APW (LAPW) the matching is energy-independent.

Here, one may see the mayor difference between PAW and APW methods: As pointed out in Equation (3.17), the expectation value for a local operator in PAW consists of three contributions (omitting all indices)

$$\text{PAW} \qquad \langle\Psi|\hat{O}|\Psi\rangle = \langle\tilde{\Psi}|\hat{O}|\tilde{\Psi}\rangle_{\text{space}} + \left(\langle\phi|\hat{O}|\phi\rangle - \langle\tilde{\phi}|\hat{O}|\tilde{\phi}\rangle\right)_{\text{sphere}} \tag{3.30}$$

while in APW one obtains

$$\text{APW} \qquad \langle\Psi|\hat{O}|\Psi\rangle = \left(\langle\tilde{\Psi}|\hat{O}|\tilde{\Psi}\rangle\right)_{\text{masked}} + \langle\phi|\hat{O}|\phi\rangle_{\text{sphere}}. \tag{3.31}$$

The interstitial contributions in APW are equivalent to $\langle\tilde{\Psi}|\hat{O}|\tilde{\Psi}\rangle - \langle\tilde{\phi}|\hat{O}|\tilde{\phi}\rangle$, i.e. an evaluation in the entire space except for the sphere volumes which are excluded by a mask function. The APW sphere contribution is the exact equivalent of the PAW contribution $\langle\phi|\hat{O}|\phi\rangle$ of the true quantities.

One can find more similarities between PAW and an implementation of LAPW as described by Soler and Williams [37] if the plane waves are allowed to penetrate the spheres for $\ell > \ell_{\text{cut}}$ which is reported to lead to converged results at similar values of ℓ_{cut} as usually applied in PAW. This approach can be regarded as PAW on a plane-wave ($\mathbf{G}$) basis set whereas a single smooth partial wave $\tilde{\phi}^a$ is chosen as the spherical Bessel function $j_\ell(Kr)$ for each $K = |\mathbf{k} + \mathbf{G}|$. A linear combination of ϕ^a and its energy derivative $\dot{\phi}^a$ is used as true partial wave, where the linear combination is chosen such that value and derivative at the sphere boundary match $j_\ell(Kr)$. The resulting transformation $\mathcal{T}^a$ depends on the plane wave energy $\frac{1}{2}K^2$ contrary to the transformation in the PAW method discussed here.

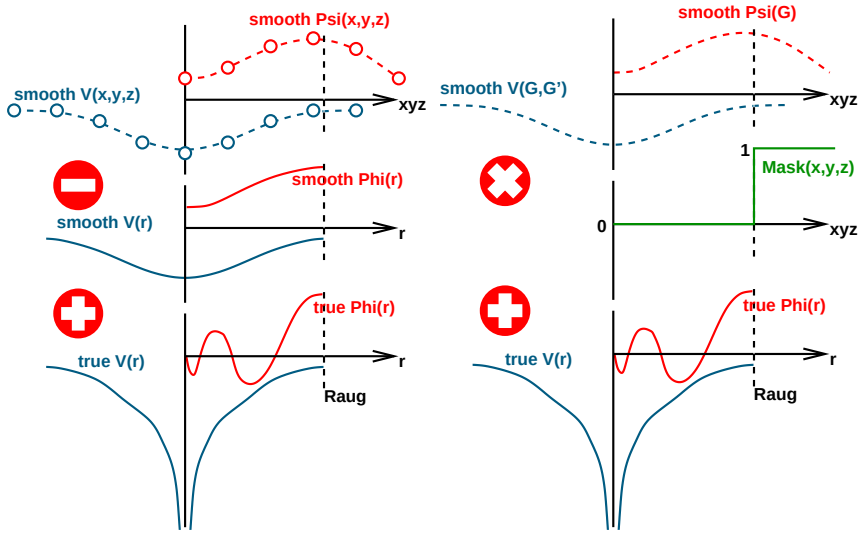


Figure 3.4.: Schematic comparison of the Projector Augmented Wave method (left) and the Augmented Plane Wave method (right). The interstitial contributions of APW are treated in a plane wave representation. Implementations of the PAW method are based on plane waves, localized basis functions such as wavelets or on real-space grid points as demonstrated in this work.

3.4. Total Energy

The total energy of a system is defined as the sum of the kinetic energy E_{kin}, the electrostatic energy E_{es}, the exchange-correlation contribution E_{xc}, and the external energy E_{ext}

$$E_{tot} = E_{kin} + E_{es} + E_{xc} + E_{ext}. \tag{3.32}$$

The external energy is a consequence pf an applied external field $V^{\sigma}_{ext}(\mathbf{r})$ and, hence, its contribution to the total energy is given as

$$E_{ext} = \sum_{\sigma} \int d^3\mathbf{r}\, n_{\sigma}(\mathbf{r}) V^{\sigma}_{ext}(\mathbf{r}). \tag{3.33}$$

The total kinetic energy E_{kin} can be evaluated directly from the auxiliary Kohn-Sham states

$$E_{kin} = \sum_{n\sigma\mathbf{k}} f_{n\sigma\mathbf{k}} \langle \Psi_{n\sigma\mathbf{k}} | \hat{T} | \Psi_{n\sigma\mathbf{k}} \rangle \tag{3.34}$$

with the occupation numbers $f_{n\sigma\mathbf{k}}$ or indirectly via the eigenvalue sum and a double counting correction as

$$E_{kin} = \sum_{n\sigma\mathbf{k}} f_{n\sigma\mathbf{k}} \epsilon_{n\sigma\mathbf{k}} - \sum_{\sigma} \int d^3\mathbf{r}\, n_\sigma(\mathbf{r})\, V^\sigma_{eff}(\mathbf{r}). \tag{3.35}$$

The exchange-correlation energy in the local-density approximation reads

$$E_{xc} = \int d^3\mathbf{r}\, n(\mathbf{r})\, \epsilon_{xc}[n](\mathbf{r}) \tag{3.36}$$

where $\epsilon_{xc}[n]$ is the exchange-correlation energy density of the homogeneous electron gas at density n. The electrostatic energy contribution E_{es} is discussed in detail in the following section.

3.4.1. The Electrostatic Energy

The electrostatic contribution to the total energy consists of three terms,

$$E_{es} = E_{ZZ} + E_{eZ} + E_{ee} \tag{3.37}$$

the core-core interaction, E_{ZZ}, the electron-core interaction, E_{eZ}, and the electron-electron term, E_{ee}, i.e. the electrostatic self-interaction of an electron density $n(\mathbf{r})$. The latter is given as

$$E_{ee} = \frac{1}{2}\iint d^3\mathbf{r}\, d^3\mathbf{r}'\, \frac{n(\mathbf{r})\, n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \tag{3.38}$$

$$= \frac{1}{2}\int d^3\mathbf{r}\, n(\mathbf{r})\, V_{es}[n](\mathbf{r}), \tag{3.39}$$

where the electrostatic potential of a charge distribution $\rho(\mathbf{r})$ is given by

$$V_{es}[\rho](\mathbf{r}) = \int d^3\mathbf{r}'\, \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}. \tag{3.40}$$

The evaluation of the electrostatic potential of a charge distribution that is not neutral in the spatial average is difficult. It involves either boundary values in the case of isolated systems or it requires the tuning for the convergence of an Ewald summation technique for periodic boundary conditions [71]. To overcome this difficulty, we consider the electrostatics of the atomic cores. The attractive Coulomb potentials of the atomic nuclei at positions $\mathbf{R}^a$ create an external potential which is experienced by the electrons. The contribution to the electrostatic energy is thus given by

$$E_{eZ} = \int d^3\mathbf{r}\, n(\mathbf{r}) \sum_a V_{es}[Z^a\delta](\mathbf{r}) = \int d^3\mathbf{r}\, n(\mathbf{r}) \sum_a \frac{Z^a}{|\mathbf{r}-\mathbf{R}^a|}, \tag{3.41}$$

where an additional Ewald summation over the atoms is required in the case of periodic boundary conditions. The charge distribution associated with the atomic nuclei is equivalent to $Z^a\delta(\mathbf{r}-\mathbf{R}^a)$. Z^a is the number of unit charges in the nucleus i.e. a negative integer. Finally, the core-core interaction is given by

$$E_{ZZ} = \frac{1}{2}\sum_{a}^{a'\neq a} \frac{Z^{a'}\, Z^a}{|\mathbf{R}^{a'}-\mathbf{R}^a|}. \tag{3.42}$$

Note that while each core does not interact electrostatically with itself it does interact with its periodic images, if applicable.

We can simplify the difficulties associated to Ewald summations of charged systems by introducing a generalized charge density of electrons and protons. If the

total number of charges (electrons minus protons) vanishes, then the three contributions to the electrostatic energy can be written as electrostatic energy of the neutral charge density $\rho(\mathbf{r})$ defined by

$$\rho(\mathbf{r}) = n(\mathbf{r}) + \sum_a Z^a \delta(\mathbf{r} - \mathbf{R}^a), \tag{3.43}$$

$$E_{es} = \frac{1}{2} \iint d^3\mathbf{r}\, d^3\mathbf{r}'\, \frac{\rho(\mathbf{r})\, \rho'(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}, \tag{3.44}$$

where the prime on ρ reminds us of the omission of the self-interaction of the cores.

3.4.2. Electrostatics and the Pseudocharge Construction

In analogy to the PAW transformed expectation values introduced above we split the electrostatic potential $V_{es}[\rho]$ defined in Equation (3.40) and the electrostatic energy yielding

$$E_{es} = \tilde{E}_{es} + \sum_a \left[E^a_{es} - \tilde{E}^a_{es} \right], \tag{3.45}$$

$$V_{es}[\rho](\mathbf{r}) = \tilde{V}_{es}(\mathbf{r}) + \sum_a \left[V^a_{es}(\mathbf{r}) - \tilde{V}^a_{es}(\mathbf{r}) \right]. \tag{3.46}$$

The smooth contribution $\tilde{V}_{es}(\mathbf{r})$ can be represented with a reasonable number of basis functions, e.g. plane waves or grid points, that do not depend on the atomic positions. The two contributions V^a_{es}, $\tilde{V}^a_{es}$ are treated on a radial grid inside the augmentation spheres S^a around each atom a. The radial grids are limited by the radius of the augmentation spheres r^a_{aug}.

The electrostatic potential of a charge distribution is found by solving the Poisson equation

$$\Delta_{\mathbf{r}} V_{es}[\rho](\mathbf{r}) = -4\pi\rho(\mathbf{r}) \tag{3.47}$$

with appropriate boundary conditions. In the case of finite systems, we align the electrostatic potential such that it assumes zero for large distances. Due to the treatment of neutral charge distributions the potential approaches zero as fast as $\sim \frac{\mathbf{D}\cdot\mathbf{r}}{|\mathbf{r}|^3}$, where $\mathbf{D}$ is the total dipole moment, or even faster if $\mathbf{D}$ vanishes. In the special case of three periodic boundaries, the value of $V_{es}(\mathbf{r})$ may be shifted by an arbitrary constant. To fix this, we demand that the spatial average of the electrostatic potential vanishes, $\int d^3\mathbf{r}\, V_{es}(\mathbf{r}) = 0$.

The solution of Poisson's Equation (3.47) can be found according to Jackson [72] via the Green function

$$G_{es}(\mathbf{r}, \mathbf{r}') = \frac{1}{|\mathbf{r} - \mathbf{r}'|} \tag{3.48}$$

since

$$\Delta_{\mathbf{r}'} G_{\mathrm{es}}(\mathbf{r},\mathbf{r}') = \Delta_{\mathbf{r}'} \frac{1}{|\mathbf{r}-\mathbf{r}'|} = -4\pi\delta(\mathbf{r}-\mathbf{r}') \tag{3.49}$$

in Hartree atomic units.

The integration over $\mathbf{r}'$ shows that the electrostatic potential depends on $\rho(\mathbf{r}')$ in all space. For this reason simply replacing $\rho(\mathbf{r}')$ by a smooth quantity inside the spheres would result in a different electrostatic potential in the interstitial region and is thus not suitable. However, we can expand the Coulomb kernel $|\mathbf{r}-\mathbf{r}'|^{-1}$ in terms of multipoles such that

$$\frac{1}{|\mathbf{r}-\mathbf{r}'|} = \sum_{L}^{\infty} \frac{r_<^\ell}{r_>^{\ell+1}} \frac{Y_L(\hat{\mathbf{r}}) Y_L(\hat{\mathbf{r}}')}{2\ell+1}, \tag{3.50}$$

where $r_>$ is defined as $\max\{|\mathbf{r}|,|\mathbf{r}'|\}$ and similarly $r_< = \min\{|\mathbf{r}|,|\mathbf{r}'|\}$. L is a combined index for ℓ and m running from 1 (ℓ=0, m=0) through $(\ell_{\max}+1)^2$.

We then split $\rho(\mathbf{r})$ into three parts, a smooth contribution $\tilde{\rho}(\mathbf{r})$ and two contributions $\rho^a(\mathbf{r})$, $\tilde{\rho}^a(\mathbf{r})$ treated on the radial grid, i.e.

$$\rho(\mathbf{r}) = \tilde{\rho}(\mathbf{r}) + \sum_a \left[\rho^a(\mathbf{r}) - \tilde{\rho}^a(\mathbf{r})\right]. \tag{3.51}$$

However, in this situation the non-locality of the electrostatic problem requires a special treatment. We demand that $\tilde{\rho}(\mathbf{r})$ reproduces the same electrostatic multipole moments as $\rho(\mathbf{r})$ inside each sphere. Hence, the atomic correction $\rho^a(\mathbf{r})$-$\tilde{\rho}^a(\mathbf{r})$ must have only vanishing multipole moments. Since $\tilde{\rho}(\mathbf{r})$ coincides with $\rho(\mathbf{r})$ in the interstitial region, we need to find the two identical charge distributions $\tilde{\rho}(\mathbf{r})$ and $\tilde{\rho}^a(\mathbf{r})$ that fulfill this requirement.

Again, $\tilde{\rho}(\mathbf{r})$ needs to be smooth enough to be accurately represented, so it can neither feature the rapid oscillations of the atomic core densities nor the nuclear charges $Z^a\delta(|\mathbf{r}-\mathbf{R}^a|)$. At this point we introduce the concept of the pseudocharge construction. The compensator function $\hat{g}_L^a(\mathbf{r}-\mathbf{R}^a)$ is strictly localized inside the sphere $\mathcal{S}^a$, centered at $\mathbf{R}^a$ and has unity multipole moment

$$\int d^3\mathbf{r}\, \hat{g}_L(\mathbf{r})\, |\mathbf{r}|^{\ell'} Y_{L'}(\hat{\mathbf{r}}) = \delta_{LL'}. \tag{3.52}$$

Using compensator functions, we can identify $\tilde{\rho}(\mathbf{r})$ as

$$\tilde{\rho}(\mathbf{r}) = \tilde{n}(\mathbf{r}) + \sum_{aL} \Delta q_L^a\ \hat{g}_L^a(\mathbf{r}-\mathbf{R}^a) \tag{3.53}$$

with

$$\Delta q_L^a = \int_{\mathcal{S}^a} d^3\mathbf{r}\, |\mathbf{r}|^\ell Y_L(\hat{\mathbf{r}})\, [n^a(\mathbf{r}) + Z^a\delta(\mathbf{r}) - \tilde{n}^a(\mathbf{r})] \tag{3.54}$$

such that $V_{es}[\tilde{\rho}](\mathbf{r})$ is the correct electrostatic potential outside of each sphere and shows the correct asymptotic behavior for $|\mathbf{r}| \to \infty$. The summation over ℓ is stopped at $2\ell_{max}$, where ℓ_{max} is the ℓ-cutoff of projectors and partial waves. The multipole moments Δq^a_L in the implementation are stored in `a%qlm(1:)`.

3.4.3. Electrostatic Potential in the Spheres

For notational convenience, we assume $\mathbf{R}^a$=0 in this section.

As we can see from Equation (3.50), the Coulomb kernel $|\mathbf{r} - \mathbf{r}'|^{-1}$ is diagonal with respect to L in an ℓm-representation. Given that all densities are stored on radial grid as $\rho(r, L)$, the electrostatic problem reduces to the radial integrals

$$\begin{aligned} V_{es}[\rho(r_2, L)](r_1, L) &= \int dr_2\, r_2^2 \frac{r_<^\ell}{r_>^{\ell+1}} \rho(r_2, L) \\ &= r_1^{-\ell-1} \int_0^{r_1} dr_2\, r_2^{2+\ell} \rho(r_2, L) + r_1^\ell \int_{r_1}^\infty dr_2\, r_2^{1-\ell} \rho(r_2, L), \end{aligned} \tag{3.55}$$

where the notation $V_{es}[\rho(r_2, L)]$ expresses that we only need to know $\rho(r_2, L)$ at all radii r_2 to find $V_{es}(r_1, L)$. In a real calculation, the second integral is only executed up to the boundary of the sphere $\mathcal{S}^a$.

To find the correct electrostatic potentials and electrostatic energy contributions in the spheres, we start by generating a preliminary potential, $\tilde{V}^{a,pre}_{es}[\tilde{\rho}^a(r', L)](r, L)$ and $V^{a,pre}_{es}[\rho^a(r', L)](r, L)$, in the sphere from the true density ρ^a and smooth density $\tilde{\rho}^a$, where we disregard the boundary conditions for both.

Also, due to the non-locality of the electrostatic problem, we cannot find $V^a_{es}(r, L)$ from $n^a(r, L)$ and $\tilde{V}^a_{es}(r, L)$ from $\tilde{n}^a(r, L)$ alone, respectively. Even if $\tilde{n}^a(r, L)$ matches with $\tilde{n}(\mathbf{r})$ on the grid, in the sense that

$$\sum_L \tilde{n}^a(|\mathbf{r}|, L) Y_L(\hat{\mathbf{r}}) = \tilde{n}(\mathbf{r}), \tag{3.56}$$

or equivalently,

$$\int d^3\mathbf{r}\, \tilde{n}(\mathbf{r}) Y_L(\hat{\mathbf{r}}) \delta(r - |\mathbf{r}|) = \tilde{n}^a(r, L), \tag{3.57}$$

the two potentials $\tilde{V}^{a,pre}_{es}(r, L)$ and $\tilde{V}_{es}(\mathbf{r})$ may still differ by a multipole source free expression.

$$\tilde{V}_{es}(\mathbf{r}) = \sum_L \tilde{V}^{a,pre}_{es}(|\mathbf{r}|, L) Y_L(\hat{\mathbf{r}}) + \sum_L \nu^a_L \underbrace{|\mathbf{r}|^\ell Y_L(\hat{\mathbf{r}})}_{\text{source-free}}. \tag{3.58}$$

The compensators $\hat{g}^a_L(\mathbf{r})$ are, by construction, orthonormal to $|\mathbf{r}|^\ell Y_L(\hat{\mathbf{r}})$ (see Equation (3.52)), such that we can multiply Equation (3.58) with the compensators $\hat{g}^a_L$ and

integrate over space. Then

$$\nu_L^a = \int d^3\mathbf{r}\, \tilde{V}_{es}(\mathbf{r})\, \hat{g}_L^a(\mathbf{r}) - \int dr\, r^2\, \tilde{V}_{es}^{a,pre}(r, L)\, \hat{g}_\ell^a(r), \tag{3.59}$$

where the first term is an integral of the compensators with the smooth electrostatic potential on the grid (c.f. `a%vlm(1:)`) that contains information about external potential shifts, electrical fields and higher multipoles. In contrast to Δq_L^a (c.f. `a%qlm(1:)`) which corrects the multipole moment for the far field, the shifts ν_L^a carry information about the electrostatics outside the spheres into the sphere and generate the correct near field

$$\tilde{V}_{es}^a[\tilde{\rho}^a(r', L)](r, L) = \tilde{V}_{es}^{a,pre}[\tilde{\rho}^a(r', L)](r, L) + \nu_L^a r^\ell, \tag{3.60}$$

$$V_{es}^a[\rho^a(r', L)](r, L) = V_{es}^{a,pre}[\rho^a(r', L)](r, L) + \nu_L^a r^\ell. \tag{3.61}$$

3.4.4. Electrostatic Energy in the Spheres

We can now compute the electrostatic energy in the spheres. At this point, the treatment of the smooth quantities and the true quantities differs slightly. The electrostatic energy of the smooth density in the sphere is a pure Hartree energy

$$\tilde{E}_{es}^a = \frac{1}{2} \sum_L \int dr\, r^2\, \tilde{V}_{es}^a(r, L) \tilde{\rho}^a(r, L). \tag{3.62}$$

In return, the true electrostatic energy E_{es}^a in the sphere consists of a Hartree energy and the Coulomb energy of the electron density in the singular potential of the atomic nucleus. It cannot be evaluated by $\int V_{es}^a[\rho^a]\rho^a$ with $\rho^a = n^a + Z^a\delta$ because that would involve a self-interaction of the nucleus. Instead

$$E_{es}^a = \frac{1}{2} \sum_L \int dr\, r^2\, V_{es}^a[n^a](r, L) n^a(r, L) + \int dr\, r^2\, n^a(r, 00) \frac{4\pi Y_{00} Z^a}{r} + \frac{1}{2} \nu_{00}^a Y_{00} Z^a. \tag{3.63}$$

This way, the electrostatic interaction of a nucleus with itself is excluded.

3.4.5. Compensator Functions

The pseudocharge compensators need to be rather smooth and strictly localized. We therefore adopt the polynomial construction as described by Weinert *et al.* [73] as depicted in Figure 3.5.

$$\hat{g}_\ell^{[N_W]}(\mathbf{r}) \sim |\mathbf{r}|^\ell \left(1 - \frac{|\mathbf{r}|^2}{R_{aug}^2}\right)^{(N_W - \ell)} Y_{\ell m}(\hat{\mathbf{r}}) \qquad \text{for} \qquad |\mathbf{r}| < R_{aug}. \tag{3.64}$$

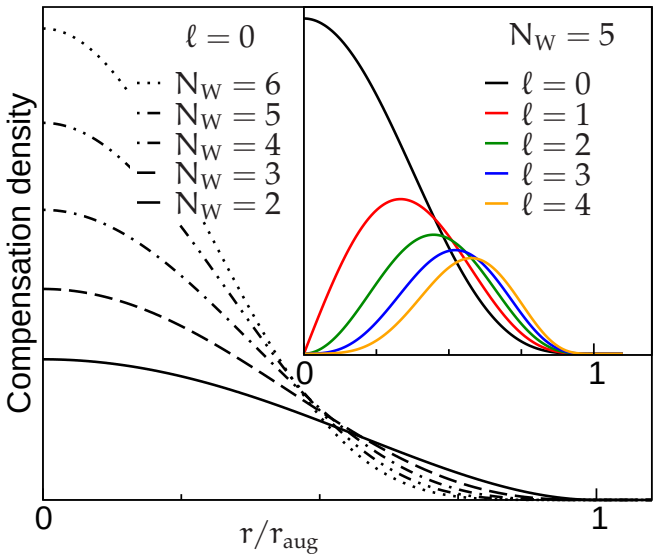


Figure 3.5.: Strictly localized compensator functions of the Weinert type. The exponent N_W controls the smoothness of the functions. The main graph shows the ℓ=0-compensator for different exponents, whereas the inlet shows the normalized radial functions for various values of ℓ.

3.4.6. Exchange-Correlation

A major advantage of the real-space representation is the ability to directly evaluate the exchange-correlation energy density $\epsilon_{xc}[n](\mathbf{r})$ and the potential $V_{xc}[n](\mathbf{r})$ in a local parametrization as LDA or GGA. However, the true density n differs inside the spheres from the smooth density $\tilde{n}$. The smooth density $\tilde{n}(\mathbf{r})$ is stored on the real-space grid, and the true densities inside the spheres are stored in (r, L)-representation. Therefore, the exchange-correlation energy has three contributions

$$E_{xc} = \int d^3\mathbf{r}\, \epsilon_{xc}[n](\mathbf{r})\, n(\mathbf{r}) = \tilde{E}_{xc} + \sum_a \left(E^a_{xc} - \tilde{E}^a_{xc}\right), \tag{3.65}$$

$$\tilde{E}_{xc} = \int d^3\mathbf{r}\, \epsilon_{xc}[\tilde{n}](\mathbf{r})\, \tilde{n}(\mathbf{r}), \tag{3.66}$$

$$E^a_{xc} = \sum_L \int^{r^a_{aug}} dr r^2\, \epsilon_{xc}[n^a](r, L)\, n^a(r, L), \tag{3.67}$$

$$\tilde{E}^a_{xc} = \sum_L \int^{r^a_{aug}} dr r^2\, \epsilon_{xc}[\tilde{n}^a](r, L)\, \tilde{n}^a(r, L). \tag{3.68}$$

Since those exchange-correlation parametrizations known to work are non-linear, the evaluation of ϵ_{xc} and the matrix elements of partial waves with V_{xc} has to be performed fully numerically at each step. The densities inside the spheres are eas-

ily represented in a radial representation and spherical harmonics for the angular part, i.e. $\tilde{n}^a(r, L)$, where r will be sampled from the origin to the sphere radius in a usually non-linear fashion. This allows us to capture the rapid oscillations in the all-electron densities with a reasonable number of radial support points. The advantages of an ℓm-representation (or L) are the simplicity of the cutoff parameter ℓ_{max} that controls the quality of this representation and the straightforward evaluation of electrostatics. However, the evaluation of the non-linear exchange-correlation potential ϵ_{xc} and V_{xc} cannot be executed in the ℓm-basis. Here, we need to transform into representation that samples the full solid angle Ω. This transformation is performed on a Legendre grid, where we sample φ and ϑ in an equidistant fashion.

The number of points for this sampling is usually $2M^2$ where M is chosen $\geqslant \ell_{max}$.

The transformation then reads

$$n^a(r, \vartheta_j, \varphi_i) = \sum_L n^a(r, L) Y_L(\vartheta_j, \varphi_i), \tag{3.69}$$

and the reverse transformation applied to the potential V_{xc} (and also ϵ_{xc}) reads

$$\begin{aligned} V_{xc}^a(r, L) &= \int_0^{2\pi} d\varphi \int_0^{\pi} d\vartheta \sin\vartheta \; V_{xc}^a(r, \vartheta, \varphi) \, Y_L(\vartheta, \varphi) \\ &\approx \sum_j^M w_j \sum_i^{2M} V_{xc}^a(r, \vartheta_j, \varphi_i) \, Y_L(\vartheta_j, \varphi_i) \end{aligned} \tag{3.70}$$

where the indices i and j sample φ and ϑ, respectively. A Legendre grid for $M = 12$ is depicted in Figure 3.6. The closer the points are to one of the poles (ϑ=0 or ϑ=π), the denser is this grid. Therefore, the weights w_j with

$$w_j = \sin(\vartheta_j) \tag{3.71}$$

need to be included in the reverse transformation.

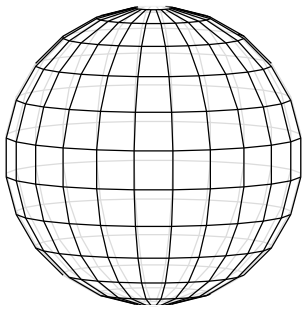


Figure 3.6.: Example Legendre grid for sampling the solid angle with M=12, i.e. 288 points. Close to the poles, the grid points are denser which is compensated by the integration weights $w_j = \sin(\vartheta_j)$ for an isotropic treatment of the full solid angle.

3.5. Hamiltonian

We can derive the full transformed PAW Hamiltonian by the variation of the total energy with respect to the smooth wave functions $\tilde{\Psi}$, where we suppress the indices at $\tilde{\Psi}_{n\sigma\mathbf{k}}$

$$\begin{aligned}\frac{\delta E_{tot}}{\delta\tilde{\Psi}^*(\mathbf{r})} &= f\int d^3\mathbf{r}'\hat{\tilde{H}}(\mathbf{r},\mathbf{r}')\tilde{\Psi}(\mathbf{r}') \\ &= \frac{\delta}{\delta\tilde{\Psi}^*(\mathbf{r})}\left(E_{kin}[\{\tilde{\Psi}\}] + E_{xc}[\tilde{n}] + E_{es}[\tilde{\rho}] + \sum_a \Delta E^a\right).\end{aligned} \tag{3.72}$$

ΔE^a stands for the difference of true minus smooth atomic contributions, i.e. $E^a - \tilde{E}^a$. Let us consider each part separately starting from the kinetic energy

$$\frac{\delta E_{kin}[\{\tilde{\Psi}\}]}{\delta\tilde{\Psi}^*(\mathbf{r})} = \frac{\delta\left(f\int d^3\mathbf{r}'\,\tilde{\Psi}^*(\mathbf{r}')\left(-\frac{1}{2}\Delta_{\mathbf{r}'}\right)\tilde{\Psi}(\mathbf{r}')\right)}{\delta\tilde{\Psi}^*(\mathbf{r})} = f\left(-\frac{1}{2}\Delta_{\mathbf{r}'}\right)\tilde{\Psi}(\mathbf{r}')\delta(\mathbf{r}-\mathbf{r}').$$

The exchange-correlation energy yields

$$\frac{\delta E_{xc}[\tilde{n}]}{\delta\tilde{\Psi}^*(\mathbf{r})} = \int d^3\mathbf{r}'\frac{\delta E_{xc}[\tilde{n}]}{\delta\tilde{n}(\mathbf{r}')}\frac{\delta\tilde{n}(\mathbf{r}')}{\delta\tilde{\Psi}^*(\mathbf{r})} = f\int d^3\mathbf{r}'\,V_{xc}[\tilde{n}](\mathbf{r}')\tilde{\Psi}(\mathbf{r}')\delta(\mathbf{r}-\mathbf{r}'). \tag{3.73}$$

The contribution from the electrostatic energy reads

$$\begin{aligned}\frac{\delta E_{es}[\tilde{\rho}]}{\delta\tilde{\Psi}^*(\mathbf{r})} &= \int d^3\mathbf{r}'\frac{\delta E_{es}[\tilde{\rho}]}{\delta\tilde{\rho}(\mathbf{r}')}\frac{\delta\tilde{\rho}(\mathbf{r}')}{\delta\tilde{\Psi}^*(\mathbf{r})} = \int d^3\mathbf{r}'\,V_{es}[\tilde{\rho}](\mathbf{r}')\frac{\delta\left[\tilde{n}(\mathbf{r}') + \sum_{aL}\Delta q^a_L\hat{g}^a_L(\mathbf{r}')\right]}{\delta\tilde{\Psi}^*(\mathbf{r})}. \\ &= \int d^3\mathbf{r}'\,V_{es}[\tilde{\rho}](\mathbf{r}')\left(f\tilde{\Psi}(\mathbf{r}')\delta(\mathbf{r}-\mathbf{r}') + \sum_{aL}\frac{\delta\Delta q^a_L}{\delta\tilde{\Psi}^*(\mathbf{r})}\hat{g}^a_L(\mathbf{r}')\right),\end{aligned} \tag{3.74}$$

where the Δq_L^a depend on the atomic density matrices D_{ij}^a. Hence,

$$\frac{\delta E_{xc}[\tilde{n}]}{\delta\tilde{\Psi}^*(\mathbf{r})} = f\int d^3\mathbf{r}' V_{es}[\tilde{\rho}](\mathbf{r}')\tilde{\Psi}(\mathbf{r}')\delta(\mathbf{r}-\mathbf{r}') + \sum_{aLij}\frac{\delta\Delta q_L^a}{\delta D_{ij}^a}\frac{\delta D_{ij}^a}{\delta\tilde{\Psi}^*(\mathbf{r})}\underbrace{\int d^3\mathbf{r}' V_{es}[\tilde{\rho}](\mathbf{r}')\hat{g}_L^a(\mathbf{r}')}_{\nu_L^a}. \tag{3.75}$$

The variation of the atomic density matrices D_{ij}^a yields

$$\frac{\delta D_{ij}^a}{\delta\tilde{\Psi}^*(\mathbf{r})} = f\tilde{p}_i^a(\mathbf{r})\int d^3\mathbf{r}'\tilde{p}_j^a(\mathbf{r}')\tilde{\Psi}(\mathbf{r}')$$

which also appears in

$$\frac{\delta\Delta E^a}{\delta\tilde{\Psi}^*(\mathbf{r})} = \sum_{ij}\frac{\delta\left(E^a-\tilde{E}^a\right)}{\delta D_{ij}^a}\frac{\delta D_{ij}^a}{\delta\tilde{\Psi}^*(\mathbf{r})}. \tag{3.76}$$

The true atomic sphere contributions are

$$\begin{aligned}\frac{\delta E^a}{\delta D_{ij}^a} &= \langle\phi_i^a|\hat{T}|\phi_j^a\rangle + \frac{\delta E_{es}^a}{\delta D_{ij}^a} + \frac{\delta E_{xc}^a}{\delta D_{ij}^a}\\ &= \langle\phi_i^a|\hat{T}|\phi_j^a\rangle + \sum_L\int dr\, r^2\left[V_{es}^a[\rho^a](r,L)\frac{\delta\rho^a(r,L)}{\delta D_{ij}^a} + V_{xc}^a[n^a](r,L)\frac{\delta n^a(r,L)}{\delta D_{ij}^a}\right]\\ &= \langle\phi_i^a|\hat{T}|\phi_j^a\rangle + \sum_L\int dr\, r^2\left[V_{es}^a[\rho^a](r,L) + V_{xc}^a[n^a](r,L)\right]\phi_i^a(r)\phi_j^a(r)G\left(L_i,L_j;L\right),\end{aligned} \tag{3.77}$$

whereas the smooth sphere contributions are

$$\begin{aligned}\frac{\delta\tilde{E}^a}{\delta D_{ij}^a} &= \langle\tilde{\phi}_i^a|\hat{T}|\tilde{\phi}_j^a\rangle + \frac{\delta\tilde{E}_{es}^a}{\delta D_{ij}^a} + \frac{\delta\tilde{E}_{xc}^a}{\delta D_{ij}^a}\\ &= \langle\tilde{\phi}_i^a|\hat{T}|\tilde{\phi}_j^a\rangle + \sum_L\int dr\, r^2\left[V_{es}^a[\tilde{\rho}^a](r,L)\frac{\delta\tilde{\rho}^a(r,L)}{\delta D_{ij}^a} + V_{xc}^a[\tilde{n}^a](r,L)\frac{\delta\tilde{n}^a(r,L)}{\delta D_{ij}^a}\right]\\ &= \langle\tilde{\phi}_i^a|\hat{T}|\tilde{\phi}_j^a\rangle + \sum_L\int dr\, r^2\left[V_{es}^a[\tilde{\rho}^a](r,L) + V_{xc}^a[\tilde{n}^a](r,L)\right]\tilde{\phi}_i^a(r)\tilde{\phi}_j^a(r)G\left(L_i,L_j;L\right)\\ &\quad + \sum_L\frac{\delta\Delta q_L^a}{\delta D_{ij}^a}\int dr\, r^2\, V_{es}^a[\tilde{\rho}^a](r,L)\hat{g}_\ell^a(r)\end{aligned} \tag{3.78}$$

with

$$\frac{\delta\Delta q_L^a}{\delta D_{ij}^a} = \int dr\, r^2\left[\phi_i^a(r)\phi_j^a(r) - \tilde{\phi}_i^a(r)\tilde{\phi}_j^a(r)\right]r^\ell\, G\left(L_i,L_j;L\right). \tag{3.79}$$

Collecting all terms we obtain

$$\int d^3\mathbf{r}'\hat{\tilde{H}}(\mathbf{r},\mathbf{r}')\tilde{\Psi}(\mathbf{r}') = \left(-\frac{1}{2}\Delta_{\mathbf{r}} + \tilde{V}_{es}[\tilde{\rho}](\mathbf{r}) + \tilde{V}_{xc}[\tilde{n}](\mathbf{r})\right)\tilde{\Psi}(\mathbf{r}) \tag{3.80}$$
$$+ \sum_{aij} \underbrace{\int d^3\mathbf{r}'\,\tilde{\Psi}(\mathbf{r}')\tilde{p}_j^a(\mathbf{r}')}_{c_j^a} \left[\frac{\delta\Delta E^a}{\delta D_{ij}^a} + \sum_L \frac{\delta\Delta q_L^a}{\delta D_{ij}^a}\nu_L^a\right] \tilde{p}_i^a(\mathbf{r}).$$

If an external potential is present, $V_{ext}(\mathbf{r})$ needs to be incorporated into both the local parts, expression in parenthesis in Equation (3.80), and the atomic Hamiltonian matrix elements H_{ij}^a, expression in square brackets in Equation (3.80).

3.6. Forces

This section follows closely the notes by C. Rostgaard [74].

Once a self-consistent electron density according to the ion positions is found, the Hellman-Feynman forces acting onto the ions can be calculated from

$$\mathbf{F}^a = -\frac{dE_{tot}}{d\mathbf{R}^a} = -\frac{\partial E_{tot}}{\partial \mathbf{R}^a} - \sum_{n\sigma\mathbf{k}} \left(\frac{\partial E_{tot}}{\partial|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle}\frac{d|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle}{d\mathbf{R}^a} + \text{h.c.}\right). \tag{3.81}$$

We can insert the Hamiltonian for $\frac{\partial E_{tot}}{\partial\langle\tilde{\Psi}|} = f\hat{\tilde{H}}|\tilde{\Psi}\rangle$ and the eigenvalue equation $\hat{\tilde{H}}|\tilde{\Psi}\rangle = \epsilon\hat{\tilde{O}}|\tilde{\Psi}\rangle$ to find

$$\mathbf{F}^a = -\frac{\partial E_{tot}}{\partial \mathbf{R}^a} - \sum_{n\sigma\mathbf{k}} f_{n\sigma\mathbf{k}}\epsilon_{n\sigma\mathbf{k}}\left(\langle\tilde{\Psi}_{n\sigma\mathbf{k}}|\hat{\tilde{O}}|\frac{d\tilde{\Psi}_{n\sigma\mathbf{k}}}{d\mathbf{R}^a}\rangle + \text{h.c.}\right) \tag{3.82}$$
$$= -\frac{\partial E_{tot}}{\partial \mathbf{R}^a} + \sum_{n\sigma\mathbf{k}} f_{n\sigma\mathbf{k}}\epsilon_{n\sigma\mathbf{k}}\langle\tilde{\Psi}_{n\sigma\mathbf{k}}|\frac{d\hat{\tilde{O}}}{d\mathbf{R}^a}|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle, \tag{3.83}$$

where we exploited that the Kohn-Sham state are normalized to the overlap operator

$$\langle\tilde{\Psi}_i|\hat{\tilde{O}}|\tilde{\Psi}_j\rangle = \delta_{ij}. \tag{3.84}$$

Taking the derivative with respect to $\mathbf{R}^a$ on both sides of this equation yields

$$\frac{d}{d\mathbf{R}^a}\langle\tilde{\Psi}_i|\hat{\tilde{O}}|\tilde{\Psi}_j\rangle = 0 \tag{3.85}$$

and after reformulating

$$\langle \frac{d\tilde{\Psi}_i}{d\mathbf{R}^a} | \hat{\tilde{O}} | \tilde{\Psi}_j \rangle + \text{h.c.} = -\langle \tilde{\Psi}_i | \frac{d\hat{\tilde{O}}}{d\mathbf{R}^a} | \tilde{\Psi}_j \rangle, \tag{3.86}$$

where

$$\frac{d\hat{\tilde{O}}}{d\mathbf{R}^a} = \sum_{ij} \Delta q^a_{ij} \left(|\tilde{p}^a_i\rangle \langle \frac{d\tilde{p}^a_j}{d\mathbf{R}^a}| + \text{h.c.} \right). \tag{3.87}$$

We further need to analyze the explicit dependence of the total energy on the atomic positions

$$\frac{\partial E_{tot}}{\partial \mathbf{R}^a} = \sum_\sigma \int d^3\mathbf{r} \, \frac{\delta E_{tot}}{\delta \tilde{n}^\sigma(\mathbf{r})} \frac{\partial \tilde{n}^\sigma(\mathbf{r})}{\partial \mathbf{R}^a} + \sum_{ij\sigma} \frac{\partial E_{tot}}{\partial D^a_{ij\sigma}} \frac{\partial D^a_{ij\sigma}}{\partial \mathbf{R}^a} + \sum_L \int d^3\mathbf{r} \, \frac{\delta E_{tot}}{\delta \hat{g}^a_L(\mathbf{r})} \frac{\partial \hat{g}^a_L(\mathbf{r})}{\partial \mathbf{R}^a} \tag{3.88}$$

$$= \sum_\sigma \int d^3\mathbf{r} \, \tilde{V}^\sigma_{eff}(\mathbf{r}) \frac{\partial \tilde{n}^a_c(\mathbf{r})}{\partial \mathbf{R}^a} + \sum_{ij\sigma} H^a_{ij\sigma} \frac{\partial D^a_{ij\sigma}}{\partial \mathbf{R}^a} - \sum_L \int d^3\mathbf{r} \, \frac{\partial \tilde{V}_{es}(\mathbf{r})}{\partial \mathbf{r}} \Delta q^a_L \hat{g}^a_L(\mathbf{r}). \tag{3.89}$$

In the first term we used that the smooth density $\tilde{n}^\sigma(\mathbf{r})$ only depends on the atomic positions explicitly via the smooth core densities which are independent of the spin σ, i.e.

$$\tilde{n}^\sigma(\mathbf{r}) = \tilde{n}^\sigma_v(\mathbf{r}) + \sum_a \tilde{n}^a_c(|\mathbf{r} - \mathbf{R}^a|). \tag{3.90}$$

In the third term of Equation (3.89) changes its sign due to a partial integration which effects that we have to consider the gradients of the electrostatic potential $\tilde{V}_{es}(\mathbf{r})$ rather than the derivative of each localized atomic compensator function $\hat{g}^a_L(\mathbf{r})$.

The derivatives of the atomic density matrix $D^a_{ij\sigma}$ consists of derivatives of projection coefficients. These can be calculated by taking the derivative of either the wave functions or the projector functions which leads to different scaling behavior with respect the system size. Anyway, the derivative results in

$$\frac{\partial D^a_{ij\sigma}}{\partial \mathbf{R}^a} = \sum_{n\mathbf{k}} f_{n\sigma\mathbf{k}} \left(\langle \tilde{\Psi}_{n\sigma\mathbf{k}} | \tilde{p}^a_i \rangle \frac{\partial}{\partial \mathbf{R}^a} \langle \tilde{p}^a_j | \tilde{\Psi}_{n\sigma\mathbf{k}} \rangle + \text{h.c.} \right). \tag{3.91}$$

Taking the derivative of the atomic projectors with respect to their origins yields expressions that can be unified using the derivative of the overlap operator from Equation (3.83)

$$\frac{\partial D^a_{ij\sigma}}{\partial \mathbf{R}^a} = \sum_{nk} f_{n\sigma k} \left(\langle \tilde{\Psi}_{n\sigma k} | \tilde{p}^a_i \rangle \langle \frac{d\tilde{p}^a_j}{d\mathbf{R}^a} | \tilde{\Psi}_{n\sigma k} \rangle + \text{h.c.} \right) \tag{3.92}$$

$$= \sum_{nk} f_{n\sigma k} \left(c^{a*}_{in\sigma k} \, \partial c^a_{jn\sigma k} + \text{h.c.} \right). \tag{3.93}$$

Hence, the total expression for the force consists of three contributions

$$\mathbf{F}^a = -\sum_L \int d^3\mathbf{r} \, \frac{\partial \tilde{V}_{es}(\mathbf{r})}{\partial \mathbf{r}} \Delta q^a_L \hat{g}^a_L(\mathbf{r}) \tag{3.94}$$

$$+ \sum_{n\sigma k} f_{n\sigma k} \sum_{ij} \left(H^a_{ij\sigma} - \epsilon_{n\sigma k} \Delta q^a_{ij} \right) \left(c^{a*}_{in\sigma k} \, \partial c^a_{jn\sigma k} + \text{h.c.} \right) \tag{3.95}$$

$$+ \sum_\sigma \int d^3\mathbf{r} \, \tilde{V}^\sigma_{eff}(\mathbf{r}) \frac{\partial \tilde{n}^a_c(\mathbf{r})}{\partial \mathbf{R}^a} \tag{3.96}$$

In practice, the total effective potential usually contains a contributions $\bar{V}(\mathbf{r})$ that corrects the potential to be smoother and less attractive inside the augmentation spheres. This leads to an additional contribution to the force which then reads

$$\mathbf{F}^a = -\sum_L \int d^3\mathbf{r} \, \frac{\partial \tilde{V}_{es}(\mathbf{r})}{\partial \mathbf{r}} \Delta q^a_L \hat{g}^a_L(\mathbf{r}) \tag{3.97}$$

$$+ \sum_{n\sigma k} f_{n\sigma k} \sum_{ij} \left(H^a_{ij\sigma} - \epsilon_{n\sigma k} \Delta q^a_{ij} \right) \left(c^{a*}_{in\sigma k} \, \partial c^a_{jn\sigma k} + \text{h.c.} \right) \tag{3.98}$$

$$+ \sum_\sigma \int d^3\mathbf{r} \, \left(\tilde{V}^\sigma_{eff}(\mathbf{r}) - \bar{V}(\mathbf{r}) \right) \frac{\partial \tilde{n}^a_c(\mathbf{r})}{\partial \mathbf{R}^a} \tag{3.99}$$

$$+ \sum_\sigma \int d^3\mathbf{r} \, \frac{\partial \bar{V}(\mathbf{r})}{\partial \mathbf{R}^a} \tilde{n}^\sigma_v(\mathbf{r}). \tag{3.100}$$

jüRS **Input Syntax:** `forces 1`

3.7. Preparations for a PAW Calculation

- filter projector functions $\tilde{p}_{\ell n}$ and restore bi-orthogonality $\langle \tilde{p}_{\ell n} | \tilde{\phi}_{\ell n'} \rangle = \delta_{nn'}$ in the radial basis
- compute tensors for the components of the atomic Hamiltonian $H^a_{ij\sigma}$, i.e. kinetic energy deficit matrix, multipole deficit tensor

- create localized projector functions $\tilde{p}_i^a(\mathbf{r}-\mathbf{R}^a)$ on the cartesian grid using the DoubleGrid method
- create localized compensator functions $\hat{g}_i^a(\mathbf{r}-\mathbf{R}^a)$ on the cartesian grid
- create frozen smooth core density $\tilde{n}_c(\mathbf{r}) = \sum_a \tilde{n}_c^a(|\mathbf{r}-\mathbf{R}^a|)$ on the cartesian grid. Here, frequently more than 27 periodic images are required.
- create localized potential correction $\bar{V}(\mathbf{r}) = \sum_a \bar{V}^a(|\mathbf{r}-\mathbf{R}^a|)$ on the cartesian grid, max. 27 periodic images.
- load $\tilde{\Psi}_{n\sigma\mathbf{k}}(\mathbf{r})$ from a file or generate an initial guess from atomic valence orbitals $\tilde{\phi}_{\ell n}(|\mathbf{r}^a|)Y_{\ell m}(\hat{\mathbf{r}^a})$, where $\mathbf{r}^a = \mathbf{r}-\mathbf{R}^a$ and n=1 is the atomic bound state (for details see Section 3.8).
- load $\tilde{n}_v^\sigma(\mathbf{r})$ from a file or generate an initial guess from the wave functions $\sum_{n\mathbf{k}} f_{n\sigma\mathbf{k}}|\tilde{\Psi}_{n\sigma\mathbf{k}}(\mathbf{r})|^2$.

3.8. Initial Wave Functions

An excellent choice of a set of initial wave functions are complete shells of atomic orbitals. We therefore set $\tilde{\Psi}_{n\sigma\mathbf{k}}(\mathbf{r})$ $\tilde{\phi}_{\ell\, n=1}(|\mathbf{r}^a|)Y_{\ell m}(\hat{\mathbf{r}}^a)M(|\mathbf{r}^a|/R_{max})$, where $\mathbf{r}^a = \mathbf{r}-\mathbf{R}^a$ and n=1 is the atomic bound state. In order to limit the costs, a maximum number of only 27 periodic images of the atomic position are taken into account. This requires a limitation of the range of the initial orbital to R_{max} that is given by the smallest extend of the cell in a direction with periodic boundary conditions. The mask function $M(x)$ takes care of this. $M(x)$ is unity around the origin and tends to 0 at x=1, beyond which it vanishes exactly. A typical choice of the mask function shape is $(1-x^8)^4$ for $|x| \leqslant 1$ and 0 elsewhere. This limitation ensures that symmetries of the system are not broken by the initial guess functions despite the finite number of periodic images. However, in some cases we want to break a symmetry as, e.g., in a spin polarized calculation. To test if the system tends to be magnetic or not we break the symmetry between spin up and spin down by scaling the length of $\mathbf{r}^a$ by a few percent up and down, respectively. Thus, one kind of orbitals becomes more localized than the others and their energetic degeneracy is lifted.

If no density file is found at startup of a calculation, the density is generated from the initial guess wave functions, i.e. usually atomic orbitals. Therefore, it is necessary to create the right set of atomic orbitals to create an atomic density. The choice is made by always adding full shells of orbitals to the set of initial states. In this procedure, $|\tilde{\phi}_{\ell\, n=1}\rangle$ states that have been occupied during the PAW data generation are selected with priority.

`jüRS` **Input Syntax:** `./paw -l -L`

3.9. The Self-consistency Cycle

- start from smooth wave functions $|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle$ and occupation numbers $f_{n\sigma\mathbf{k}}$
- find the coefficients $c^a_{in\sigma\mathbf{k}} = \langle \tilde{p}^a_i | \tilde{\Psi}_{n\sigma\mathbf{k}} \rangle$
- create the atomic density matrix $D^a_{ij\sigma}$ from $f_{n\sigma\mathbf{k}}$ and $c^a_{in\sigma\mathbf{k}}$
- determine the compensation charge multipole moments $\Delta q^a_{\ell m}$ from $D^a_{ij\sigma}$
- compose the smooth valence density $\tilde{n}^\sigma_v(\mathbf{r})$ from $|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle$ and $f_{n\sigma\mathbf{k}}$
- evaluate $V_{xc}[\tilde{n}^\sigma]$ and $\epsilon_{xc}[\tilde{n}]$, where $\tilde{n}^\sigma = \tilde{n}^\sigma_v + \tilde{n}^\sigma_c$
- evaluate $V_{es}[\tilde{\rho}]$, where $\tilde{\rho} = \sum_\sigma \tilde{n}^\sigma_v + \tilde{n}^\sigma_c + \Delta q^a_{\ell m} \hat{g}_{\ell m}$
- find potential shifts $\nu^a_{\ell m}$ from $V_{es}[\tilde{\rho}]$
- set up atomic Hamiltonian matrix $H^a_{ij\sigma}$ from $D^a_{ij\sigma}$ and $\nu^a_{\ell m}$
- compose the new Hamiltonian $\hat{\tilde{H}}^\sigma = \hat{T} + V_{es}[\tilde{\rho}] + V^\sigma_{xc}[\tilde{n}] + \bar{V} + |\tilde{p}^a_i\rangle H^a_{ij\sigma} \langle \tilde{p}^a_j|$
- diagonalize $\hat{\tilde{H}}^\sigma$ to find the new $|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle$ and $\epsilon_{n\sigma\mathbf{k}}$
- compute new Fermi level and occupation numbers $f_{n\sigma\mathbf{k}}$ from $\epsilon_{n\sigma\mathbf{k}}$

3.10. Summary

The PAW method introduces a transformation mapping from smooth KS wave functions to the cooresponding all-electron wave function and, thus, allows for an all-electron treatment of all elements in the framework of DFT. However, the frozen core approximation is implemented in jüRS for simplicity. Inside the atomic augmentation spheres, the core density $n_c(r^a)$ is replaced by $\tilde{n}_c(r^a)$ and all valence states are represented as linear combinations of the smooth partial waves $\tilde{\phi}^a(\mathbf{r}^a)$. The all-electron wave functions are retrieved from the same linear combination of the true partial waves $\phi^a(\mathbf{r}^a)$ which match the smooth partial waves at the sphere boundaries in value and derivative(s). Localized projector functions $\tilde{p}^a(\mathbf{r}^a)$ are required to find the corresponding linear combination.

The proper electrostatics outside the sphere are ensured by smooth localized multipole compensator functions $\hat{g}^a(\mathbf{r}^a)$ in the spheres. In return, multipole-free corrections of the local electrostatic potential are required for the alignment of the smooth and true potential inside the sphere. The potentials in the sphere provide an atomic non-local correction to the Hamiltonian H^a. Furthermore, the PAW transformation is not necessarily norm-conserving which leads to the charge deficit Δq^a which is generally non-zero.

The quantities $\tilde{n}_c(r^a)$, $\hat{g}^a(\mathbf{r}^a)$, $\tilde{\phi}^a(\mathbf{r}^a)$ and $\tilde{p}^a(\mathbf{r}^a)$ exhibit, compared to the corresponding all-electron quantities, smoothness, i.e. a rapidly converging represen-

tation in reciprocal space, and can thus be treated accurately with a reasonable number of plane-waves or on equidistant real-space grids, as this thesis shows.

The variation of the total energy E_{tot} with its contribution from the smooth quantities on the complete simulation volume and corrections inside the sphere with respect to the degrees of freedom of a smooth KS wave function $\tilde{\Psi}(\mathbf{r})$ leads to the Hamiltonian $\hat{\tilde{H}}$ in the generalized eigenvalue equation

$$\left(\hat{\tilde{H}} - \epsilon\hat{\tilde{O}}\right)|\tilde{\Psi}\rangle = 0 \tag{3.101}$$

with the PAW overlap operator defined by

$$\hat{\tilde{O}} = \mathbf{1} + \sum_a |\tilde{p}^a\rangle \Delta q^a \langle\tilde{p}^a| \tag{3.102}$$

and the PAW transformed Hamiltonian

$$\hat{\tilde{H}} = -\frac{1}{2}\Delta + \tilde{V}_{eff}(\mathbf{r}) + \sum_a |\tilde{p}^a\rangle H^a \langle\tilde{p}^a| \tag{3.103}$$

which resembles closely the effective Hamiltonian of a pseudopotential method.

Hellman-Feynman forces can be computed deriving either the extended wave functions $\tilde{\Psi}(\mathbf{r})$ or the localized projector functions $\tilde{p}^a(\mathbf{r}^a)$ with respect to the atomic positions $\mathbf{R}^a$.

CHAPTER 4

CARTESIAN REAL-SPACE GRID

The most time consuming part of a DFT calculation is the solution of the Kohn-Sham eigenvalue equation

$$\left(\frac{1}{2}\hat{\mathbf{p}}^2 + V_{\text{eff}}^{\sigma}(\hat{\mathbf{r}}) - E_{n\sigma\mathbf{k}}\right)|\Psi_{n\sigma\mathbf{k}}\rangle = 0. \tag{4.1}$$

In this equation we can find the momentum operator $\hat{\mathbf{p}}$ and the real-space position operator $\hat{\mathbf{r}}$. The non-relativistic kinetic energy operator $\hat{T} = \frac{1}{2}\hat{\mathbf{p}}^2$ is merely a quadratic function of $\hat{\mathbf{p}}$ and the potential operator V_{eff}^{σ} is a function of $\hat{\mathbf{r}}$. These two operators are incommensurate as their commutator

$$[\hat{\mathbf{p}}, \hat{\mathbf{r}}] = [i\frac{\partial}{\partial \mathbf{r}}, \mathbf{r}] = i$$

does not vanish. We thus cannot find a common eigenstate of both operators. The eigenstates of the momentum operator $|\mathbf{p}\rangle$ are plane waves. Choosing these as a basis the Kohn-Sham equation reads

$$\int d^3\mathbf{p}' \left(\left[\frac{1}{2}\mathbf{p}^2 - E_{n\sigma\mathbf{k}}\right]\delta(\mathbf{p}-\mathbf{p}') + \langle\mathbf{p}|\hat{V}_{\text{eff}}^{\sigma}|\mathbf{p}'\rangle\right)\langle\mathbf{p}'|\Psi_{n\sigma\mathbf{k}}\rangle = 0 \qquad \forall\mathbf{p}. \tag{4.2}$$

Casting this equation into a finite basis set of N plane waves $|\mathbf{p}\rangle$, usually chosen out of a sphere with $\frac{1}{2}\mathbf{p}^2 \leqslant E_{\text{cut}}$, we need to evaluate (in principle N^2 but due to symmetry merely) 2^3N matrix elements of the potential operator $\langle\mathbf{p}|\hat{V}_{\text{eff}}^{\sigma}|\mathbf{p}'\rangle$ every time the effective potential changes due to a density update. On the other hand, choosing the eigenstates of the position operator $|\mathbf{r}\rangle$ as a basis, the Kohn-Sham equation in real-space reads

$$\int d^3\mathbf{r}' \left(\langle\mathbf{r}|\frac{1}{2}\hat{\mathbf{p}}^2|\mathbf{r}'\rangle + [V_{\text{eff}}^{\sigma}(\mathbf{r}) - E_{n\sigma\mathbf{k}}]\,\delta(\mathbf{r}-\mathbf{r}')\right)\langle\mathbf{r}'|\Psi_{n\sigma\mathbf{k}}\rangle = 0 \qquad \forall\mathbf{r}. \tag{4.3}$$

In this work, we employ three-dimensional equidistant real-space grids, i.e. the real-space coordinates $\mathbf{r}$ are sampled with a constant grid spacing h_i, $i \in \{x, y, z\}$. Then, the matrix elements of the kinetic energy operator $\hat{T}$ in real-space representation are constants. Their evaluation is discussed in Section 4.1. Further, we apply the PAW transformation in order to model the scattering properties and energy contributions of the atoms properly, see Chapter 3.

4.0.1. Kinetic Energy Operator in Real-Space

The maximum frequency to be represented on an equidistant grid with a constant grid spacing of h is $k_{max} = \frac{\pi}{h}$. This is called sampling theorem and k_{max} denotes the Nykvist-frequency. The theoretical maximum of kinetic energy $\frac{\pi^2}{2h^2}$ will further on be called *nominal cutoff energy* E_{cut}. In order to represent highly oscillating quantities the grid spacing h must be decreased and thus E_{cut} increased. The computational costs grow at least cubically with the number of gridpoint and thus $\sim h^{-3}$. The usage of the PAW method (see Chapter 3) with its augmentation procedure allows to represent only smooth quantities in full space and thus enables us to use relatively coarse real-space grids. Furthermore, the application of the potential operator in real-space is a simple product. However, the kinetic energy operator $\hat{T}$ is non-local since its eigenstates are plane waves that extend over the full space. An exact treatment of the kinetic energy on real-space grids would thus require Fourier transforms. Even though we can make use of the Fast Fourier Transform (FFT) algorithms that scale only $N \log(N)$ with the number of grid points N, we will avoid this method and approximate the kinetic energy by a localized Finite Difference (FD) derivative scheme. The reason for this are two aspects of scaling behavior. FD allows a linear scaling in the sense that we can compute the application of $\hat{T}$ to a vector in $\mathcal{O}(N)$ operations which will always be faster than FFT for larger N. Further, the evaluation of the kinetic energy operator in FD approximation can be parallelized with a restricted communication pattern, whereas FFT requires unrestricted all-to-all communication.

4.1. Finite Differences

The kinetic energy operator $\hat{T}$ appears in the Schrödinger and in the Kohn-Sham equation (see Equation 2.17) in the form of the Laplacian, i.e. the second derivative with respect to all three spatial directions. The central Finite Difference (FD) method truncates this derivative to a difference quotient. Consider an analytically given function $f(x)$ that varies slowly in space and its representation on a uniform grid with a constant grid spacing h, i.e. $x_i = ih$. The values $f_i = f(x_i)$ correspond to the function evaluated at the grid points x_i. By Taylor expansion up to second

order we can express the value of the analytical function at neighboring points to both sides.

$$\begin{aligned} f_{i-1} = f(x_{i-1}) = f(x_i - h) &\overset{\text{Taylor}}{=} f(x_i) - hf'(x_i) + \frac{1}{2}h^2 f''(x_i) + \mathcal{O}(h^3) \\ f_i \quad = f(x_i) & \\ f_{i+1} = f(x_{i+1}) = f(x_i + h) &\overset{\text{Taylor}}{=} f(x_i) + hf'(x_i) + \frac{1}{2}h^2 f''(x_i) + \mathcal{O}(h^3) \end{aligned} \tag{4.4}$$

This can be rewritten in matrix form as

$$\begin{pmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{pmatrix} = \begin{pmatrix} 1 & -1 & \frac{1}{2} \\ 1 & 0 & 0 \\ 1 & 1 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} f(x_i) \\ hf'(x_i) \\ h^2 f''(x_i) \end{pmatrix} \tag{4.5}$$

This is a matrix relation between the vector of function values (f_{i-1}, f_i, f_{i+1}) and the zeroth, (approximate) first and (approximate) second derivative $(f(x_i), hf'(x_i), h^2 f''(x_i))$ of the function $f(x)$ at the central grid point x_i. The corresponding powers of h have been kept for simplicity of the notation. Inversion of this matrix shows that the zeroth, first and second derivative of f can be expressed as a linear combination of the values f_{i-1}, f_i and f_{i+1}

$$\begin{pmatrix} 0 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} \\ 1 & -2 & 1 \end{pmatrix} \begin{pmatrix} f_{i-1} \\ f_i \\ f_{i+1} \end{pmatrix} = \begin{pmatrix} f(x_i) \\ hf'(x_i) \\ h^2 f''(x_i) \end{pmatrix} \tag{4.6}$$

$$(N_f = 1) \;\Rightarrow\; f''(x_i) = \frac{1}{h^2}\left(f_{i-1} - 2f_i + f_{i+1}\right) \tag{4.7}$$

Thus, the coefficients for the second derivative are $c''_{-1} = 1$, $c''_0 = -2$ and $c''_1 = 1$ considering only nearest neighbor points. This scheme can be expanded to higher derivatives and more neighboring point as shown below. With this method a more accurate approximation for the second derivative can be achieved. Table 4.1 shows the coefficients for various numbers of considered neighbors, N_f. It is easy to show that even derivatives (zeroth, second, fourth, ...) have a symmetric coefficient scheme while the coefficients are antisymmetric for odd derivatives (first, third, ...) on an equidistant grid.

The case of arbitrary order $2N_f$ reads

$$f(x_{i+j}) = f(x_i + jh) = f_{i+j} = \sum_{k=0}^{2N_f} \frac{(jh)^k}{k!} \left.\frac{\partial^k f(x)}{\partial x^k}\right|_{x=x_i} + \mathcal{O}(h^{2N_f+1}) \,, j \in [-N_f, N_f]. \tag{4.8}$$

The matrix of Taylor coefficients then is $t_{jk} = j^k/k!$. So the coefficient scheme for the first $2N_f$ derivatives are given by the rows of the inverse matrix of t_{jk}. For the m-th derivative, one finds

$$f_{FD}^{(m)}(x_i) = \frac{1}{h^m} \sum_{l=-N_f}^{N_f} \left\{ [t_{jk}]^{-1} \right\}_{ml} f_{i+l} \tag{4.9}$$

An explicit formula for high order coefficients of the first and second derivatives that does not require the inversion of a $(2N_f+1) \times (2N_f+1)$ matrix can be found later in this thesis.

$\lvert i-j \rvert$	0	1	2	3	4	5	6
N_f=1	-2.00000	1.00000					
N_f=2	-2.50000	1.33333	-0.08333				
N_f=3	-2.72222	1.50000	-0.15000	0.01111			
N_f=4	-2.84722	1.60000	-0.20000	0.02540	-0.00179		
N_f=5	-2.92722	1.66667	-0.23810	0.03968	-0.00496	0.00032	
N_f=6	-2.98278	1.71429	-0.26786	0.05291	-0.00893	0.00104	-0.00006

Table 4.1.: Finite-difference coefficients of the second derivative. The number of finite difference neighbors $2N_f$ determines the order of approximation. The error from this approach is of the order $\mathcal{O}(h^{2N_f})$.

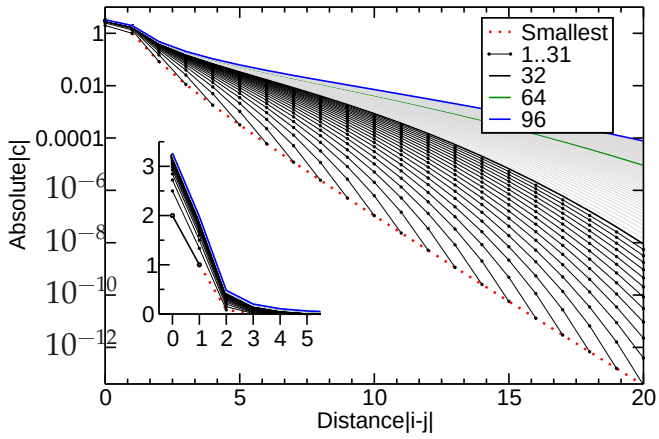


Figure 4.1.: Absolute value of the finite-difference coefficients, $c_{FD}(i-j)$, of the second derivative for $N_f \in [1, 96]$. The smallest element converges to zero roughly as $0.05\,e^{-\sqrt{2}N_f}$. The scale is logarithmic except for the inlet.

The second derivative relates the grid points x_i and x_j for all $|i-j| < N_f$ so the Laplacian operator is not local (diagonal in spatial representation) and is represented by a symmetric banded matrix. In the case of one spatial dimension these

bands enclose the diagonal from the upper and lower side as depicted in Figure 4.2 for N_f=1.

The sparsity of these matrices enables a banded storage form. Periodic boundary conditions cause non-zero matrix elements relating the first and last grid points. The second picture in Figure 4.2 differs from the first only by an entry in the upper left and the adjoint entry in the lower right corner.

The absolute value of these matrix element is as before determined from the finite difference coefficients. However, there is one degree of freedom in its complex phase. The physical meaning behind this is a phase modulation of a complex wave function which extends over an infinite number of unit cells. Therefore the Bloch-theorem can be applied and the complex phase factor of the upper right element is set to $c_1\, e^{-ikL}$ where L denotes the extend of the one-dimensional unit cell. The kinetic energy operator is an observable and so its matrix representation is hermitian. This fully determines the lower left element to be $c_1\, e^{ikL}$, where c_1 is real.

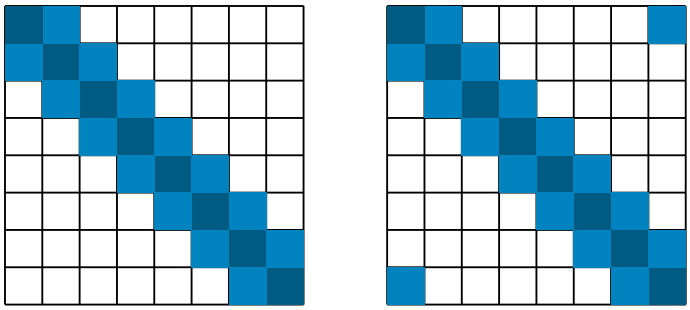


Figure 4.2.: Matrix scheme for finite difference representation of the second derivative for $N_f = 1$ on a one-dimensional equidistant grid with N_g=8 grid points. On the left side, finite boundary conditions are depicted, assuming that any function $f(x)$ vanishes outside the super cell. On the right hand side, a matrix with periodic boundary conditions is depicted. The matrix representation in general is very sparse, i.e. the ratio of nonzero entries is $\frac{(2N_f+1)}{N_g}$ in the one-dimensional case.

4.1.1. Error Estimate

To analyze the error of the truncation of the calculation of derivatives to a finite number of neighboring grid points, an arbitrary function $f(x)$ represented on an equidistant grid is expanded into Taylor polynomials. Different from before, two orders more then requested to set up the matrix are taken into account. We achieve an estimate for the error of the finite-difference approach for the second derivative:

$$f(x_i \pm h) \overset{\text{Taylor}}{=} f(x_i) \pm hf'(x_i) + \frac{h^2}{2}f''(x_i) \pm \frac{h^3}{6}f^{(3)}(x_i) + \frac{h^4}{24}f^{(4)}(x_i) \pm \mathcal{O}(h^5). \quad (4.10)$$

Solving for the second derivative yields (in vector notation)

$$f''(x_i) \approx \frac{1}{h^2} \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}^T \cdot \begin{bmatrix} f(x_i - h) \\ f(x_i) \\ f(x_i + h) \end{bmatrix} - \frac{h^2}{12} f^{(4)}(x_i). \tag{4.11}$$

Obviously the error in the second derivative is proportional to h^2. It can be shown that for $N_f > 0$ the error enters with h^{2N_f} multiplied by the $(2\,N_f{+}2)$-th derivative and an exponentially decreasing coefficient. Essentially, there are two ways to improve the quality of these derivatives. On one hand, the grid spacing h can be decreased which leads to an enormous increase of the computational costs. The other possibility is employing a higher order finite difference formula which leads to an increase in communicational costs as we will discuss later. A tradeoff needs to be found here.

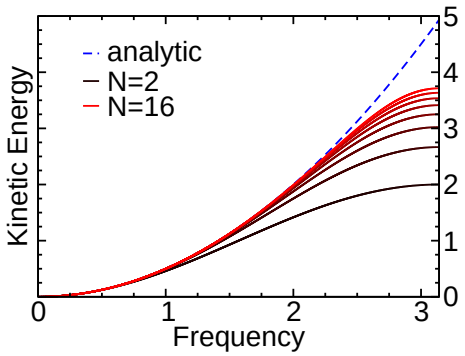


Figure 4.3.: Error estimate of the finite-difference approximation for the kinetic energy applied to one-dimensional plane waves. A plane wave e^{ikx} is an eigenstates of the kinetic energy operator and its eigenvalue is given analytically as $\frac{1}{2}k^2$. For the allowed k-values in $[0, \frac{\pi}{h}]$, the FD-approximated second derivative has been evaluated at the even orders 2 through 16. The grid spacing h is set to unity here. The dispersion relation matches the exact derivative quite well as long as the k values stay below a certain fraction of the nominal cutoff wave vector length $\frac{\pi}{h}$. The lowest (black) curve corresponds to the lowest order FD approximation (second order). A higher order finite difference approximation exploits the cutoff energy more efficiently.

Figure 4.3 shows the dispersion relation of a plane wave where the kinetic energy is computed with finite difference approximations of various orders and analytically for comparison. It can be seen that the second derivative deviates more from the analytical solution $\frac{1}{2}k^2$ for low order finite difference sets (lower curves). This means that the FD approximation is only accurate up to a certain fraction of the nominal cutoff energy $\frac{1}{2}\left(\frac{\pi}{h}\right)^2$. A very efficient way to reduce this error is by

increasing the number of finite difference neighbors such that a larger fraction of the nominal cutoff energy can be exploited. However, this leads to a less sparse and less localized kinetic energy operator and thus an increase in computation and communication.

4.2. Lagrange Interpolation

The Taylor expansion technique and so the central point finite difference formula is equivalent with Lagrange interpolation: a polynomial $Q(x)$ of degree $2\,N_f$ is fitted such that it matches the values f_i at the grid points x_i perfectly i.e. $Q(x_i) = f_i$ for all $i \in [-N_f, N_f]$. Since a polynomial of degree m has exactly m+1 degrees of freedom, we have to fit it to exactly $2\,N_f$+1 pairs of point and value (x_i, f_i). To keep it symmetric we choose one central point x_0 and N_f neighboring points to each side i.e. $x_{-N_f}, x_{-N_f+1}, ..., x_0, ..., x_{N_f-1}, x_{N_f}$. The generic solution for the Lagrange interpolated polynomial centered around x_0 reads

$$Q(x) = \sum_i f_i \prod_{k\neq i} \frac{x - x_k}{x_i - x_k} = \sum_i f_i\, c_i^{(0)}(x) \tag{4.12}$$

(all indices in $\sum$ or $\prod$ run from $-N_f$ to N_f). The expansion coefficients for the interpolated function are thus given by

$$c_i^{(0)}(x) = \prod_{k\neq i} \frac{x - x_k}{x_i - x_k} \tag{4.13}$$

So the first derivative of the polynomial reads

$$Q'(x) = \sum_i f_i \frac{\partial}{\partial x} c_i^{(0)}(x) = \sum_i f_i\, c_i^{(1)}(x) \tag{4.14}$$

$$c_i^{(1)}(x) = \sum_{j\neq i} \frac{1}{x_i - x_j} \prod_{k\neq i,j} \frac{x - x_k}{x_i - x_k} \tag{4.15}$$

and the second derivative

$$Q''(x) = \sum_i f_i \frac{\partial^2}{\partial x^2} c_i^{(0)}(x) = \sum_i f_i\, c_i^{(2)}(x) \tag{4.16}$$

$$c_i^{(2)}(x) = \sum_{j\neq i} \frac{1}{x_i - x_j} \sum_{l\neq i,j} \frac{1}{x_i - x_l} \prod_{k\neq i,j,l} \frac{x - x_k}{x_i - x_k} \tag{4.17}$$

This yields a direct formula to compute high order central FD coefficients for the second derivative $c_i^{(2)}(x = 0)$ without inverting the matrix in Equation (4.6). On an

equidistant grid, the difference in the denominator $x_j - x_k$ is an integer multiple of the grid spacing h and never zero because the case $k = j$ is explicitly excluded such that equation (4.17) can be simplified to

$$c_i^{(2)}(0) = \frac{1}{h^2} \sum_{j \neq i} \frac{1}{i-j} \sum_{l \neq i,j} \frac{1}{i-l} \prod_{k \neq i,j,l} \frac{k}{i-k} \tag{4.18}$$

The coefficients $h^2 c_i^{(2)}(0)$ are displayed in Table 4.1 for N_f in $[1, 6]$. Figure 4.4 shows the (derived) Lagrange polynomials $c_i^{(k)}(x)$ for k in $[0, 3]$ and N_f=2.
The consistency of the FD approach with the Lagrange interpolation technique is important because in later sections of this work, Lagrange interpolation will be applied to find values of a wave function in between two grid points. The effect of interpolation with a finite number of grid points onto sharp-featured functions can be seen as spurious residuals. The interpolation operator $\hat{\mathcal{I}}$ applies Lagrangian interpolation to transfer values from a coarse grid to a finer grid. This operator shows a sparse matrix representation if the order of interpolation is small compared the total number of grid points. The operators matrix representation does not even require an explicit storage if the coarse grid spacing, h_{coarse}, is an integer multiple of the fine grid spacing, h_{fine}. The reverse operation $\hat{\mathcal{I}}^\dagger$ translates from a finer grid to the coarse grid. However, this operator may not be called the inverse of $\hat{\mathcal{I}}$ because both matrices are not square. The consequent execution of both operations, $\hat{\mathcal{I}}^\dagger\hat{\mathcal{I}}$, will be the unity operation and lets the function on the coarse grid unchanged as long as the order of interpolation is equal to the number of grid points (and the interpolation operator is a full matrix). Any truncated formula with $2\,N_{\text{itp}} < N_g$ will introduce an error to sharp features of the function.

4.2.1. Non-uniform Grids

Non-uniformity of the grid implies that the distances between grid points differs and therefore are functions of the grid point position or of its index. In principle one can determine the coefficients for the finite difference representation by the same technique as outlined above but the coefficients as well become functions of the grid index. This is fairly simple considering one-dimensional grids. Three spacial dimensions, however, require mathematical constructs like non-diagonal Riemann tensors. This in general includes so called *adaptive coordinates* [39] and *finite element* methods [75].

The advantage of a non-uniform grid is the possibility of increasing the grid point density in the vicinity of atomic cores. This enables cheaper calculations and in principle also all-electron calculations where fast oscillations close to the core and slow variation in the interstitial regions are represented on the same grid. Also in the case of a widely extended vacuum region inside the super cell, this could help to save resources of both, computational time and memory.

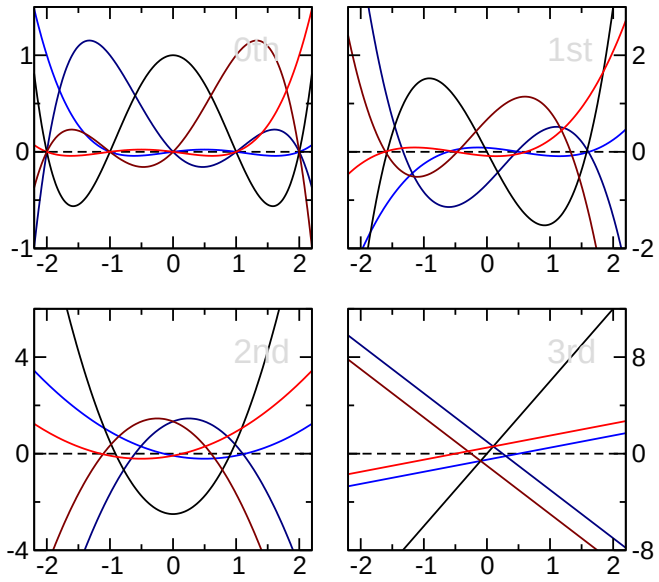


Figure 4.4.: Lagrange polynomials for N_f=2 un-derived (upper left panel), first, second and third derivatives.

4.2.2. Grid Anisotropy

Choosing the same grid spacings h_i in all three spatial directions $i \in \{x, y, z\}$ allows a cartesian grid to be commensurate with all 48 symmetries of a cube, i.e. the octahedral group. Nevertheless, the cartesian grid cannot be commensurate with 5-fold or 6-fold rotations. Also 3-fold rotations are only possible around the space diagonals, i.e. the (111) direction.

The simulation volume is bound by a rectangular box that introduces anisotropy by its shape. The real-space implementation of Chelikowsky makes use boundary conditions for isolated systems that sample gridpoints up to a certain radius from the center of mass of all atoms [76] rather than a rectangular box. This reduces the anisotropy by the effect of the box-shape boundaries. In the case of periodic boundary conditions, the periodic images of the simulation cell give rise to similar effects as a bounding box.

4.3. Filtering Methods

The translational symmetry of a system with periodic boundary conditions is cast into a discrete translational symmetry, i.e. space will only look exactly the same once we translate by an integer multiple of the grid spacing. However, translation by an arbitrary distance might affect those terms that relate the absolute position and the positions of the grid points. The most illuminating example for these terms

are the localized atomic projectors. Shifting the atomic position by a fraction of the grid spacing will make the projector representation change. This effect is stronger the coarser one chooses the grid and the more structured the projector function is. A measure for structure of a localized function can be found from its Fourier transform (on a 3-dimensional grid) or its Bessel transform (on a radial grid). This section describes how the irrepresentable components of projector functions in real-space can be reduced in order to decrease the spurious effect due to their strong localization that appears during continuous translations of the atomic positions.

Localized functions represented on the radial grids have much higher Fourier components than representable on the three dimensional cartesian grid with finite sampling h due to their strict localization. Hence, the interface between the two grids leads to the so called *egg box*-effect. To avoid these unphysical oscillations of the inner product between a radially given projector function and functions represented on the cartesian grid, the radial projector function (multiplied by spherical harmonics) may not have strong Fourier-components larger than the nominal cutoff frequency of the grid, given by $k_{max} = \frac{\pi}{h}$ where h is the grid spacing. To ensure this, different filter methods in reciprocal space can be applied. The only difficulty is to find the trade-off between frequency cutoff and strength of projector localization. Since the unfiltered projector functions tend to an exact zero beyond a radial cutoff r_{cut}, their representation in Fourier space (Bessel-transformed) is delocalized i.e. all frequency components are non-zero. In return, simply eliminating all components higher g_{max} destroys the localization of the backtransform in the radial representation.

For PAW calculations it is of major importance to keep the bi-orthogonality of the filtered projectors with the smooth partial waves (compare Equation 3.12). The reorthogonalization is usually performed only on the radial grids after the Fourier filtering steps (Section 4.3.1 and Section 4.3.2) and, hence, does not take the effects of the reversed interpolation method (see Section 4.3.3) into account.

The following three methods show how to reduce the high frequency components of a strictly localized projector function preserving as much localization as possible. The first two methods act onto the projector function on a radial grid. The third method forms an interface between the radial function and its representation on a cartesian grid.

4.3.1. Fourier Filtering Method

Given a projector function $p(\mathbf{r}) = p_\ell(|\mathbf{r}|)Y_{\ell m}(\hat{\mathbf{r}})$. The filtering can be applied on the radial grid, where the Fourier transform becomes a Bessel transform. For simplicity of the notation we will use the same symbol p for the projector function in real-

space and in reciprocal space representation. The transformation is given by

$$p(g) = \sqrt{\frac{2}{\pi}} \int_0^R dr\, r^2\, p(r)\, j_\ell(g\, r) \tag{4.19}$$

and the back-transformation by

$$p(r) = \sqrt{\frac{2}{\pi}} \int_0^G dg\, g^2\, p(g)\, j_\ell(g\, r). \tag{4.20}$$

Once found $p(g)$ we choose a maximum frequency G_{max}, usually twice the cutoff frequency of the grid, G_{cut}. The smoother projector function $\tilde{p}$ is constructed as

$$\tilde{p}(g) = \begin{cases} p(g) & 0 \leqslant g \leqslant G_{cut} \\ f(g) & G_{cut} < g \leqslant G_{max} \\ 0 & G_{max} < g \end{cases} \tag{4.21}$$

where $f(g)$ is given by the following operations:

$$f(g) = \int_{G_{cut}}^{G_{max}} dg_3 \left[\frac{\pi}{2}\delta(g - g_3)\, g^2 - \mathcal{J}_\ell^{[R]}(g, g_3)\right]^{-1} \int_0^{G_{cut}} dg_2\, \mathcal{J}_\ell^{[R]}(g_3, g_2)\, p(g_2) \tag{4.22}$$

with the overlap integral of two Bessel function $\mathcal{J}$ defined in real-space

$$\mathcal{J}_\ell^{[R]}(q_1, q_2) = q_1^2\, q_2^2 \int_0^R dr\, r^2\, j_\ell(q_1\, r)\, j_\ell(q_2\, r). \tag{4.23}$$

The symmetric kernel $\mathcal{J}_\ell^{[R]}(q_1, q_2)$ can be understood as the scattering matrix of a sharp mask function (step function on the radial grid) which is 1 for $|\mathbf{r}| < R$ and zero elsewhere. Here, R denotes an enlarged cutoff radius. The realization of this procedure uses a equidistant g-mesh to discretize the reciprocal space and a simple matrix inversion for the inverse of the symmetric operator.

The overlap function $\mathcal{J}$ can be evaluated analytically. We scale all wave numbers $q \to q\, R$ and radii $r \to r/R$. For $\ell \in \{0, 1, 2\}$, the kernel $\mathcal{J}_\ell(q_1, q_2)$ is then given by

$$R\, \mathcal{J}_0(q_1, q_2) = \frac{q_1\, q_2}{q_2^2 - q_1^2} \left(q_1 \cos(q_1) \sin(q_2) - q_2 \cos(q_2) \sin(q_1)\right) \tag{4.24}$$

$$R\, \mathcal{J}_1(q_1, q_2) = R\, \mathcal{J}_0(q_1, q_2) - \sin(q_1) \sin(q_2) \tag{4.25}$$

$$R\, \mathcal{J}_2(q_1, q_2) = R\, \mathcal{J}_0(q_1, q_2) - 3\left(\cos(q_1) - \sin(q_1)/q_1\right)\left(\cos(q_2) - \sin(q_2)/q_2\right). \tag{4.26}$$

These expressions appear to be irregular in the limit $q_1 \to q_2$. Using addition theorems we can find simplified regular expressions for the diagonal elements $\mathcal{J}_\ell(q, q)$. These are

$$R\,\mathcal{J}_0(q,q) = \frac{1}{2}q\,(q + \cos(q)\sin(q)) \tag{4.27}$$

$$R\,\mathcal{J}_1(q,q) = \frac{1}{2}q\,(q + \cos(q)\sin(q)) - \sin^2(q) \tag{4.28}$$

$$R\,\mathcal{J}_2(q,q) = \frac{1}{2}q\,(q + \cos(q)\sin(q)) - 3. \tag{4.29}$$

4.3.2. Masked Fourier Filter Method

Another flavor of pre-filtering the radial projector function and preserving the localization in real-space is outlined by Tafipolski and Schmid [77], and earlier by Lin-Wang [78]. In these methods a real-space mask function $\mathcal{M}(r)$ is introduced. High frequency components of the auxiliary function $p(r)/\mathcal{M}(r)$ are strongly suppressed with a second mask function $\mathcal{M}(q)$ in Bessel-space which is usually exponentially decaying and thus suppressing high frequency components. The real-space mask is then multiplied to the backtransform again and thus restores most of the localization. With the Bessel-transform (BT) defined in Equation (4.20) and its back-transform (BT^{-1}) the procedure reads

$$\tilde{p}(r) = \mathcal{M}(r) \cdot BT^{-1}(r) \left\{ \mathcal{M}(q) \cdot BT(q) \left\{ \frac{p(r')}{\mathcal{M}(r')} \right\} \right\} \tag{4.30}$$

and is also depicted in Figure 4.5. The shape of the two mask functions (green) yields a degree of freedom in the realization of this very efficient procedure. In order to gain a smoother, less localized, but still strictly localized, projector function, a real-space mask function the assumes zero exactly beyond a larger cutoff radius $R_{cut} > r_{cut}$ is advantageous. In reciprocal space a Gaussian type mask function performs well to reduce high, irrepresentable Fourier components. Figure 4.6 shows the action of the masked Fourier filtering procedure in real-space representation. The x-y-contourplots show the smoother output function $\tilde{p}(y)$ when the input is a sharp delta function $\delta(x)$ for $\ell \in [0, 3]$.

Nevertheless, the smoother projector function $\tilde{p}(r)$ will still contain contributions to relatively high Fourier components that have been introduced by the multiplication of the mask function $M(r)$. Therefore, the next section describes a careful treatment in the interface between radial grid $\tilde{p}(r)$ and cartesian real-space grid $\tilde{p}(|\mathbf{r}|) \cdot Y_{\ell m}(\hat{\mathbf{r}})$. Figure 4.6

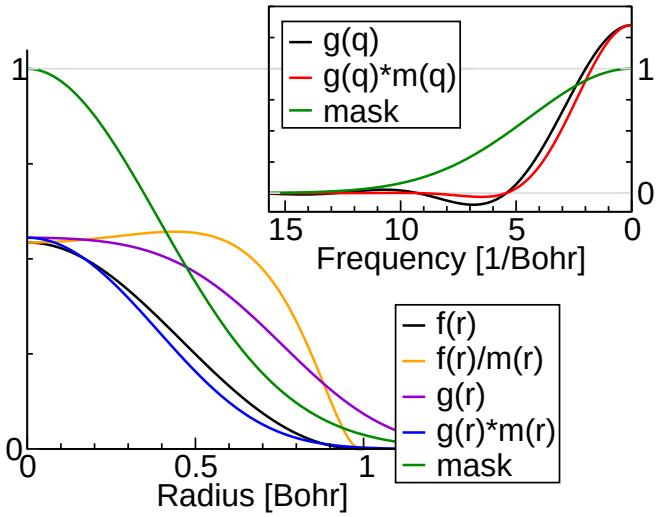


Figure 4.5.: Example of the Masked Filtering technique. The input function $f(r)$ (black line) is strictly localized within $r \in [0,1]$, whereas the output function $g(r) \cdot m(r)$ (blue line) may extend beyond $r = 1$.

4.3.3. Reversed Interpolation Method

The need of accurate inner products between spatially extended wave functions $\tilde{\Psi}(\mathbf{r})$ and localized projector functions $\tilde{p}(\mathbf{r})$ is eminent in the PAW method. This section outlines a method to reduce the unphysical dependence of the projection coefficient $c = \langle \tilde{p} | \tilde{\Psi} \rangle$ on the position of the projector's center, called *egg-box effect*. Assume a smooth wave function $\tilde{\Psi}(\mathbf{r})$ represented on a coarse uniform real-space grid with grid spacing h_c. Then, the *egg-box effect* can be suppressed strongly by interpolating the wave function to an M-times denser grid with grid spacing $h_d = h_c/M$ and performing the inner product there. Then, the projection coefficient is given by

$$c = \sum_j \tilde{p}(\mathbf{r}_j)\, \tilde{\Psi}(\mathbf{r}_j)\, \frac{\Delta V}{M^3} \tag{4.31}$$

$$= \sum_j \tilde{p}(\mathbf{r}_j) \left(\sum_i \mathcal{I}_{ji}\, \tilde{\Psi}(\mathbf{r}_i) \right) \frac{\Delta V}{M^3} \tag{4.32}$$

where the index i runs over the degrees of freedom of the wave function on the coarse grid and the index j is used for the dense grid in M^3-times more operations. In usual implementations of PAW (frozen core, fixed energy parameters) the projectors do not change as long as the atomic positions are kept constant. The wave functions are updated during the self-consistency iterations but the localized projector functions are unchanged. As shown by Ono [79, 50, 53] it is, therefore,

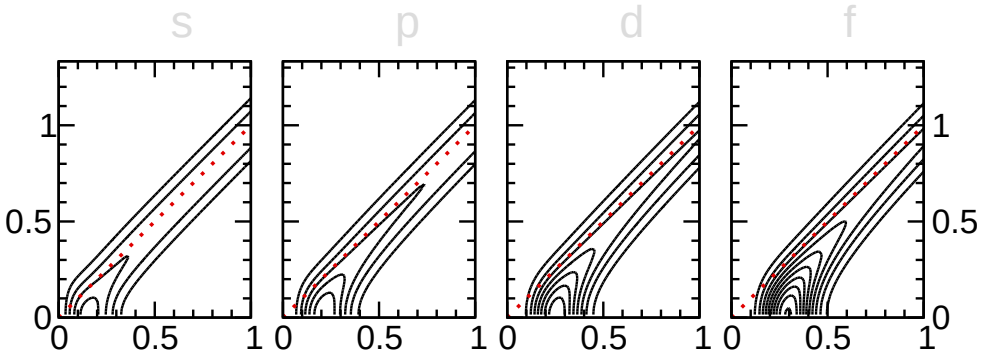


Figure 4.6.: x-y-Contourplot of the action of the masked Fourier filtering for $\ell \in [0,3]$. The input function is a delta peak at x with $x \in [0,1]$. The output $\tilde{p}(y)$ shows a softening. Observe how the maximum of $\tilde{p}(y)$ is repelled stronger from the origin the higher ℓ is. For $x \to 1$ and higher the action converges to a simple broadening function.

advantageous to interchange the summation order in the computation

$$c = \sum_i \left[\frac{1}{M^3} \sum_j \tilde{p}(\mathbf{r}_j)\, \mathcal{I}_{ji} \right] \tilde{\Psi}(\mathbf{r}_i)\, \Delta V \tag{4.33}$$

such that we can prepare the expression in square brackets, $\tilde{p}_i$, in a setup calculation preceding the self-consistency loop. Then, during the self-consistency iterations the projection coefficients are computed as

$$c = \sum_i \tilde{p}_i\, \tilde{\Psi}(\mathbf{r}_i)\, \Delta V. \tag{4.34}$$

The new effective projector elements $\tilde{p}_i$ is now represented on the same grid as the wave functions $\tilde{\Psi}$. Therefore, we can interpret $\tilde{p}_i$ also as $\tilde{p}(\mathbf{r}_i)$ and compare the real-space representation to the original projector function $\tilde{p}(\mathbf{r})$. Figure 4.7 shows the three levels, the projector function on the radial grid $\tilde{p}(r)$, on the dense grid $\tilde{p}(\mathbf{r}_j)$ and after reversed interpolation on the coarse grid $\tilde{p}(\mathbf{r}_i)$.

Assuming that we start from a projector function that is given as a radial function $\tilde{p}(r)$ times an angular function (e.g. a single spherical harmonic $Y_{\ell m}(\hat{\mathbf{r}})$), our localization region is a sphere of radius R_{cut}. Applying the reversed interpolation scheme as outlined above to our projector on the M-times denser real-space grid will increase the radius within which non-zero elements of $\tilde{p}(\mathbf{r}_i)$ can be found by the range R_{itp} of the interpolation. Therefore, the freedom of this method is given by the mesh refinement number M, usually integer, and the kind of interpolation used to generate $\mathcal{I}_{ji}$. Figure 4.8 shows the one-dimensional interpolation function $\mathcal{L}_N^{[1D]}(x)$ of the Lagrange interpolation (compare Section 4.2) for $N \in [1,8]$. The

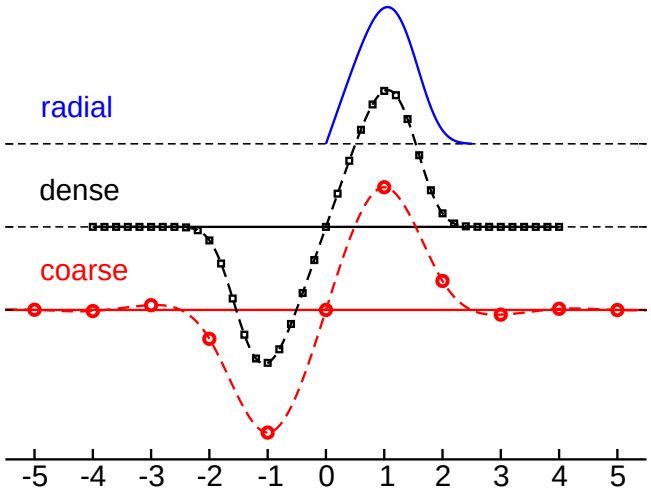


Figure 4.7.: Schematic graph for the reversed interpolation filtering in one dimension. A radial prefiltered function $\tilde{p}_\ell(r)$ (blue) here with ℓ=1 is represented on a 3-dimensional dense cartesian real-space grid (black). Reversed interpolation yields the representation of the projector function on the coarse grid (red). One can observe how the localization is gradually lost.

interpolation coefficients $\mathcal{I}_{ji}$ are given by

$$\mathcal{I}_{ji} = F_N^{[3D]}(\mathbf{r}_j - \mathbf{r}_i). \tag{4.35}$$

with the grid points at $\mathbf{r}_i$ on the coarse grid and $\mathbf{r}_j$ on the dense grid. We can exploit that Lagrange interpolation factorizes, i.e. the three dimensional interpolation function is a product of the one-dimensional functions,

$$\mathcal{L}^{[3D]}(\overrightarrow{xyz}) = \mathcal{L}^{[1D]}(x)\mathcal{L}^{[1D]}(y)\mathcal{L}^{[1D]}(z). \tag{4.36}$$

The application of the localized interpolation scheme of order N with 2N-1 grid points, $N \in \mathbb{N}$, affects the localization of $\tilde{p}(\mathbf{r}_i)$. The radius to which non-zero projector values may be found is now $R_{cut} + R_{itp}$ where $R_{itp} = N\,h_c$, i.e. the radius grows proportional to the order of the Lagrange interpolation.

The limit case $N \to \infty$ lets the interpolation produce exact results. However, it leads to the interpolating function

$$\mathcal{L}_\infty(x) = \frac{\sin(\pi x)}{\pi x} \tag{4.37}$$

which has an infinite range R_{itp}, i.e. $\tilde{p}(r_i)$ is not a localized function any more but has non-vanishing values on each grid point. This leads to a disadvantageous scaling behavior, similar to the situation of a plane wave basis $\{\mathbf{G}_i\}$, where the projector representation in Fourier space $\tilde{p}(\mathbf{G}_i)$ is non-zero for all plane waves. Each projection operation thus scales proportional to the number of basis functions which

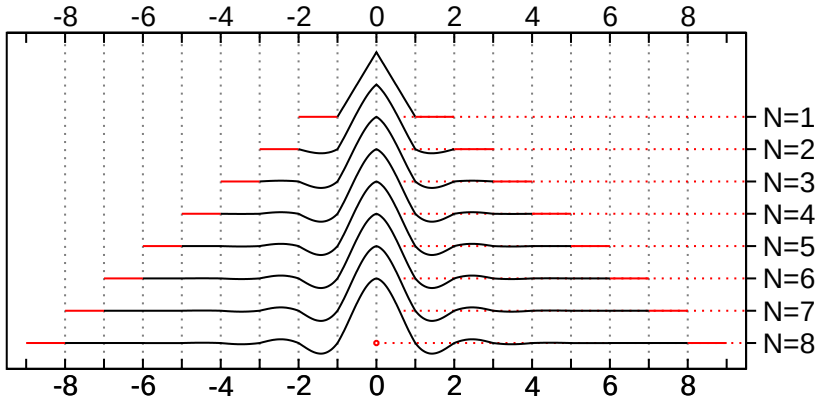


Figure 4.8.: Lagrange interpolation function $\mathcal{L}_N(x)$ in one-dimension for various N. The interpolation coefficients between coarse and dense grid can be read of given the grid refinement M and the relative alignment s of the two grids. $\mathcal{L}_N(x)$ vanishes exactly for $|x| \geqslant N$. Note that a Lagrange interpolation function of any finite order N is continuous but its first derivative is discontinuous as clearly visible for N=1. The red solid lines indicate from where the interpolating functions vanish exactly.

makes the overall workload for projection scale quadratically in the system size. The real-space representation with a finite interpolation order N leads the linear scaling with a prefactor that depends on N.

A systematic way to understand the effect of reversed interpolation filtering is to analyze the transfer function, i.e. the effect of reversed interpolation in reciprocal space. The filter is defined as $\mathcal{L}_N(q) = \langle q|\hat{\mathcal{I}}_N^{[1D]}|q\rangle$ where $|q\rangle$ are simple one-dimensional plane waves with wave number q and $\hat{\mathcal{I}}^{[1D]}$ is the one-dimensional interpolation operator. Reformulating this, we see that $F(q)$ is the Fourier transform of the interpolation coefficients of $\hat{\mathcal{I}}^{[1D]}$.

Figure 4.9 shows the one-dimensional transfer function $\mathcal{L}_N^{[1D]}(q)$ for several configurations (N, M). Analyzing this we have to keep in mind that these filters have been constructed from interpolation weights of the one-dimensional interpolation scheme. The three-dimensional equivalent shows even sharper transitions from transmission ($\mathcal{L}^{[3D]}(q) \approx 1$) to suppression ($\mathcal{L}^{[3D]}(q) \ll 1$) as we can see in Figure 4.10. As residual of the finite refinement factor ($M = 5$) we can observe the transmission bump up to 3.9 % for $q = 2.8\,\frac{\pi}{h}$ in linear interpolation in the (100) direction. The height of this maximum is only slowly reduced with the interpolation order N and its position seems to shift towards $3\,q_{max}$. A second set of residulas can be seen in the (100)-direction close to the maximum frequency on the dense grid. However, these are smaller by a factor of ten. In the (110)-direction, some contribution around 0.2 % becomes visible around $q = 4q_{max}$. In general, we find that the suppression of erroneous frequencies above 2.0 is best for (111), good (110)

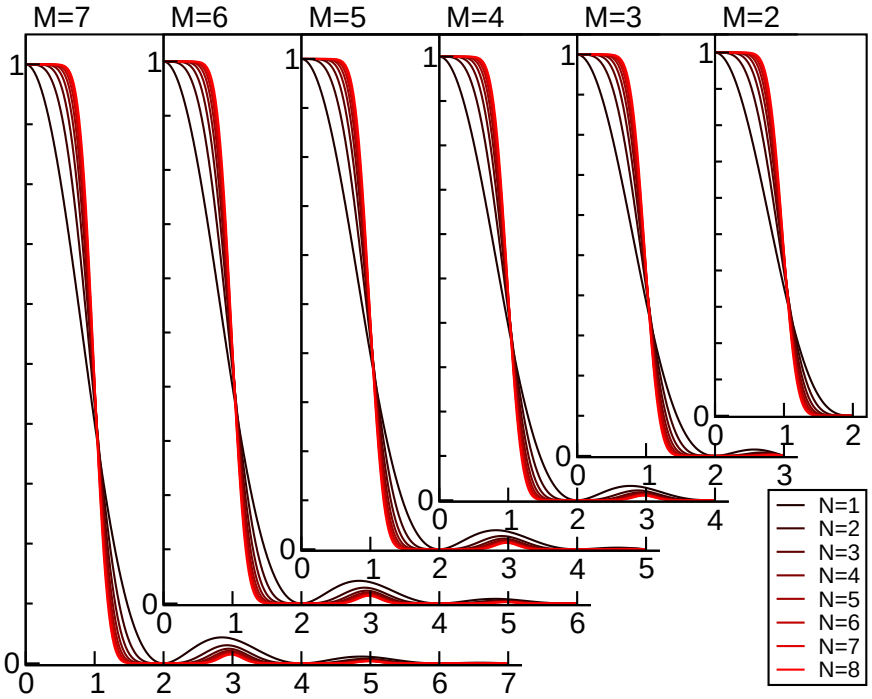


Figure 4.9.: Frequency dependent transfer function of the reversed interpolation method in one dimension. Shown are the transfer ratios $\mathcal{L}_N(q)$ as a function of the reduced frequency q/q_{max} for the interpolation order N =1 (linear, black) through 8 (red) and for different ratios of dense grid to coarse grid M =7 (largest) through 2 (rightmost graph). The frequencies q of functions represented on on the dense grid can only assume values up to $M \cdot q_{max}$ where $q_{max} = \frac{\pi}{h_c}$ with the grid spacing h_c of the coarse grid. We can observe a stronger suppression of *error frequencies* , i.e. for $q/q_{max} > 1$, at larger values of N and larger values of M. Especially for larger interpolation orders N, the frequencies $< q_{max}$ are transmitted better i.e. the transfer function is closer to the perfect mask function which would be unity for $q < q_{max}$ and zero elsewhere.

and relatively good exept the one bump for (100). However, in the range 1.0 to 2.0, the complete opposite is the case. In the limit $N \to \infty$, the transfer function should approach a Heavyside function with the edge at 1.0. This seems to be the case for (100), however, the position of the edge for (110) and (111) is renormalized by $\sqrt{2}$ and $\sqrt{3}$, respectively.

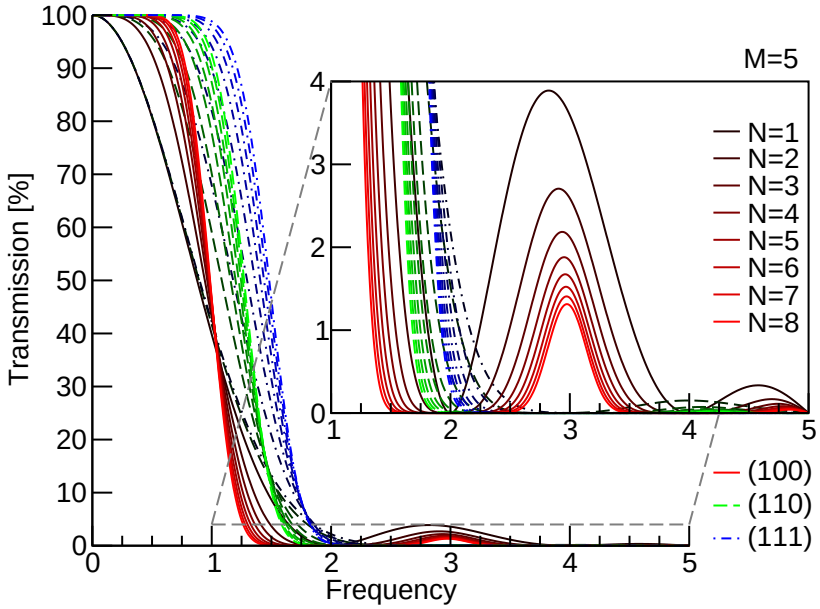


Figure 4.10.: Frequency and orientation dependent transfer function of the reversed interpolation method in 3D. Shown are the transmission ratios $\mathcal{L}_N(q)$ as a function of the relative frequency q/q_{max} for the interpolation order N from 1 (dark) through 8 (colored) and with the ratio of dense grid to coarse grid M=5. The inlet shows a closeup onto the suppression range ($q/q_{max} > 1$) on a larger scale. We can clearly see different transfer functions for different spatial orientations. Only the high symmetry directions (100), (110) and (111) are depicted here in solid red, dashed green and dotted-dashed blue lines, respectively.

`jüRS` **Input Syntax:** `dg N M`

There are several possibilities to align the dense grid to the coarse grid if the grid ratio is M, allowing only integer values for M. One option is to align the position of the coarse grid to the exact position of every Mth dense grid point (*matched*, shift s=0). Another way is to center the coarse grid points in between two dense grid points (*between*, shift $s=\frac{1}{2Mh_c}$). h is the grid spacing of the coarse grid. A third possibility is an arbitrary shift (*moving*, shift $s \in \mathbb{R}$ and $-\frac{1}{2Mh} < s \leqslant \frac{1}{2Mh_c}$) that depends on the position of the radial center of the localized function. Figure 4.11 shows all three cases graphically. In the case *moving*, the dense grid is attached to the atomic position. Then, only the reversed interpolation operator is position dependent. One might consider this approach for Car-Parinello MD implemen-

tation on real-space grids since CPMD allows an update of the nuclear positions every iteration. The costs of recomputing the representation of the localized functions on the dense grid after a position update could be saved. However, the application of the reversed interpolation operator is usually more time consuming than the setup of the projector functions on the dense grid. The actual implementation forsees the same M and N for each spatial direction and uses the method *between* for even values of M and the *matched* method if M is odd. The odd case is connected to an advantageous workload since 2N-2 of the 2NM-1 coefficients of the one-dimensional interpolation operator $\mathcal{I}^{[1D]}$ vanish exactly. This reduces the number of operations by $\frac{2N-2}{2NM-1}$ saving roughly 28 % for N=2 and M=5 during the execution of the three-dimensional application of the reversed interpolation.

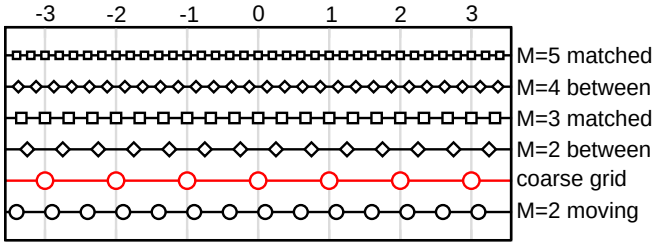


Figure 4.11.: Different alignments of the dense grid with respect to the coarse grid for the grid refinement ratio $M \in [2, 5]$. The alignment option *moving* is shown for M=2 and an atomic position that is 0.1 h_c off a coarse grid point.

4.4. Algorithms

The choice of a real-space grid instead of a basis set leads to the necessity of various methods and algorithms that are introduced in the following sections. For the generation of the effective potential we need to solve the electrostatic problem. The procedure is outlined below. Further, the generalized eigenvectors of the Hamiltonian $\hat{\tilde{H}}$ with the overlap operator $\hat{\tilde{O}}$ according to the lowest eigenvalues is requested. Therefore, we apply a combination of DIIS band update (see Section 4.4.2) and a subsequent explicit diagonalization in the subspace of the updated bands (see Section 4.4.3).

4.4.1. Poisson solver

The classical electrostatic potential $V_{es}(\mathbf{r})$ of the charge distribution $\rho(\mathbf{r})$ needs to be found in every self-consistency iteration. A straightforward implementation of the Green function defined in Equation (3.48) leads to a disadvantageous scaling

with respect to the number of grid points. Each grid elements requires the information of each other grid elements. If periodic boundary conditions are involved, the solution also needs to be converged with respect to a finite number of periodic images. Consider the differential form of the electrostatic problem,

$$\Delta_{\mathbf{r}} V_{\mathrm{es}}(\mathbf{r}) = -4\pi\rho(\mathbf{r}). \tag{4.38}$$

From the Green function we know that the solution is intrinsically non-local. We can find the exact solution by a Fourier transform $\rho(\mathbf{G}) = \mathrm{FT}\{\rho(\mathbf{r})\}$ and the algebraic solution of

$$-\mathbf{G}^2\, V_{\mathrm{es}}(\mathbf{G}) = -4\pi\rho(\mathbf{G}) \tag{4.39}$$

in Fourier space. Finally, the back transformation $\mathrm{FT}^{-1}\{V_{\mathrm{es}}(\mathbf{G})\}$ into real-space gives yields $V_{\mathrm{es}}(\mathbf{r})$. We can already see that the component $V_{\mathrm{es}}(\mathbf{G} = 0)$ is not determined by this equation. This corresponds to an arbitrary global shift of the local potential. In return, the spatial average of the density $\rho(\mathbf{G} = 0)$ must vanish for a charge neutral system. We see that the treatment of the Poisson equation in Fourier space is very simple but only applies to neutral charge distributions in periodic systems. The demand of a neutral charge density is essential for three periodic boundary conditions. However, finding the FT with the Fast Fourier transform (FFT) algorithm

$$\rho(\mathbf{r}) \longrightarrow \rho(\mathbf{G}) \qquad \text{and} \qquad V_{\mathrm{es}}(\mathbf{G}) \longrightarrow V_{\mathrm{es}}(\mathbf{r})$$

scales as N log N for the serial calculation where N is the number of points in either space. For a parallelized version of the FFT, the total amount of communication is expected to grow quadratically with N such that the efficiency will rapidly drop at a high degree of parallelism.

A solution of the electrostatic problem in real-space can be found through the differential form of the Poisson equation as in Equation (4.38). Let us focus on the derivatives $\Delta_{\mathbf{r}}$. We can approximate the second derivative in a finite-difference (FD) scheme (compare Section 4.1), then the evaluation of $\Delta_{\mathbf{r}} V_{\mathrm{es}}(\mathbf{r})$ at a certain grid point involves the value of $V_{\mathrm{es}}(\mathbf{r})$ and the values of V_{es} at neighboring grid points. This operation is localized in a range which depends on the order of the finite-difference expansion, i.e. its evaluation scales linear with the number of elements in the FD stencil and linear with total number of grid points. The iterative solution requires a start guess for the potential $V^{\mathrm{start}}(\mathbf{r})$. This guess does not need to have a special shape, even a completely vanishing guess function may be used. However, the closer the start guess is to the final solution, the less iterations will be needed to achieve convergence. From $V_{\mathrm{es}}(\mathbf{r}) = V_{\mathrm{es}}^{\mathrm{start}}(\mathbf{r})$ we can evaluate the approximate charge density $\rho^{\mathrm{app}}(\mathbf{r})$ by applying the Laplace operator.

$$\rho^{\mathrm{app}}(\mathbf{r}) = -\frac{1}{4\pi}\Delta_{\mathrm{FD}}[V_{\mathrm{es}}(\mathbf{r})] \tag{4.40}$$

the residual vector R is then defined as

$$R(\mathbf{r}) = \rho(\mathbf{r}) - \rho^{app}(\mathbf{r}). \tag{4.41}$$

We apply iterative residuum minimization techniques such as steepest descent or conjugate gradients (CG) to minimize the residual norm r with the definition

$$r^2 = \langle R|R\rangle = \int_V d^3\mathbf{r}\, R^2(\mathbf{r}). \tag{4.42}$$

Since the self-consistency cycle is meant to lead to a converging charge density $\rho(\mathbf{r})$ and thus the changes of in the charge density become smaller over the number of cycles, the solution of the Poisson equation of the previous iteration is a relatively good start vectors.

An alternative to the FD approximation are the Mehrstellen algorithm [80] where a short-ranged non-local stencil is applied to both sides of Equation (4.38). Different from FD which only requires information of grid points in the six cartesian (100)-directions, these stencils access the complete environment of each grid point.

4.4.2. DIIS method

For an accurate description of the atoms with the PAW method a certain minimum density of the grid is necessary. This number differs according to the type of atom, in particular the augmentation radius r^a_{aug} and ℓ^a_{max}. However, the number of degrees of freedom of a smooth KS wave function $\tilde{\Psi}(\mathbf{r})$ easily exceeds the number of electrons N_e in the system by a factor of hundred or more. We thus only need to find the lowest 1 % of the eigenvalues and eigenvectors of $\hat{\tilde{H}}$, or less. Nevertheless, a direct method which scales cubically with the matrix dimension which makes it practically impossible to reach to systems with more than a hundred atoms.

As outlined above, the KS-Hamiltonian $\hat{\tilde{H}}$ in real-space representation consists of three contributions. The constant FD stencil which approximates the kinetic energy operator with $6\,N_f$ elements, the smooth local effective potential and a non-local potential part that is a sum of dyads of localized atomic projector functions. The local potential is diagonal in real-space representation the FD approximation leads to a banded but sparse representation. If now the system extends L_i are much larger than the projection regions of the non-local potential, the Hamiltonian is very sparse. We therefore employ iterative diagonalization methods to find the lowest N_e eigenvalues and eigenvectors of the Hamiltonian. The sparsity and the structure of the Hamiltonian make an explicit storage of the matrix elements $\langle \mathbf{r}'|\hat{\tilde{H}}|\mathbf{r}\rangle$ redundant. Instead, only the FD stencil t_{FD}, the local effective potential $\tilde{V}_{eff}(\mathbf{r})$, localized projector functions $\tilde{p}^a_i(\mathbf{r})$ and atomic Hamiltonian matrix

elements H^a_{ij} are stored. We can thus compute the action of the Hamiltonian onto a trial vector within a number of operations that scales linearly with the number of grid points. The implicit action of $\hat{\tilde{H}}$ and $\hat{\tilde{O}}$ is all that is needed. The required eigenvectors $|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle$ and, consequently, their eigenenergies $E_{n\sigma\mathbf{k}}$ are found with the Direct Inversion of the Iterative Subspace (DIIS) method proposed by Pulay [81]. An iterative sequence of vectors is found by adding a trial vector each iteration. The norm of the residual vector

$$|R\rangle = \left(\hat{\tilde{H}} - \epsilon\hat{\tilde{O}}\right)|\tilde{\Psi}\rangle = |\hat{\tilde{H}}\tilde{\Psi}\rangle - \epsilon|\hat{\tilde{O}}\tilde{\Psi}\rangle \tag{4.43}$$

is minimized in the iterative subspace. The energy eigenvalue ϵ is approximated by the Ritz value of $|\tilde{\Psi}\rangle$, i.e.

$$\epsilon = \frac{\langle\tilde{\Psi}|\hat{\tilde{H}}|\tilde{\Psi}\rangle}{\langle\tilde{\Psi}|\hat{\tilde{O}}|\tilde{\Psi}\rangle}. \tag{4.44}$$

In the first iteration, the subspace consists of the initially given vector $|\tilde{\Psi}\rangle$ only. The subspace is extended by the gradient direction $|R\rangle$ every iteration and a linear combination of the sequence of residual vectors is found that minimizes the new residual. This method is implemented in `jüRS` without explicit orthogonalization. Therefore, a careful treatment of the eigenvectors is required. The maximum size of the subspace is usually limited to four steps. After the DIIS update, reorthogonalization of all eigenstates is necessary. This can either be done be a Gram-Schmidt-scheme or by subspace rotation as outlined in the next section.

4.4.3. Subspace Rotation

The subspace rotation method is necessary when the DIIS scheme, as described in the previous section (Section 4.4.2), is applied to the bands without explicit reorthogonalization. Its duty is to restore the orthogonality of all bands of a $\sigma\mathbf{k}$-point orthogonal after they have been updated in the DIIS scheme. The Hamiltonian is block diagonal with respect to different σ and $\mathbf{k}$, therefore, bands of different $\sigma\mathbf{k}$-points are orthogonal by symmetry. During subspace rotation and for each $\sigma\mathbf{k}$-point, the Hamiltonian and the overlap matrix are represented in the basis of the newly updated bands $\tilde{\Psi}$, i.e. $H_{nm} = \langle\tilde{\Psi}_n|\hat{\tilde{H}}|\tilde{\Psi}_m\rangle$ and $S_{nm} = \langle\tilde{\Psi}_n|\hat{\tilde{O}}|\tilde{\Psi}_m\rangle$ for $n, m \in [1, N_{\text{bands}}]$. The subscript $\sigma\mathbf{k}$ is suppressed here showing only the band index. Since solving for eigenstates with an iterative band update method the KS-wave states $|\tilde{\Psi}_n\rangle$ are never completely converged to exact eigenstates. Furthermore it also may occur that the potential changes sufficiently such that two bands are tending to interchange their energy order. For both reasons the two matrices are not diagonal matrices, even though often very close to a diagonal matrix, i.e. the strength of off-diagonal elements is small compared to diagonal entries. An exceptional case of many non-vanishing off-diagonal elements is the first iteration,

when the potential is generated from a superposition of atomic densities and the start guess states are localized atomic orbitals. In this special case, the operators resemble the Hamiltonian and overlap matrix of a tight-binding or LCAO (Linear Combination of Atomic Orbitals) approach. Figure 4.12 and Figure 4.13 show examples of the subspace Hamiltonian matrices.

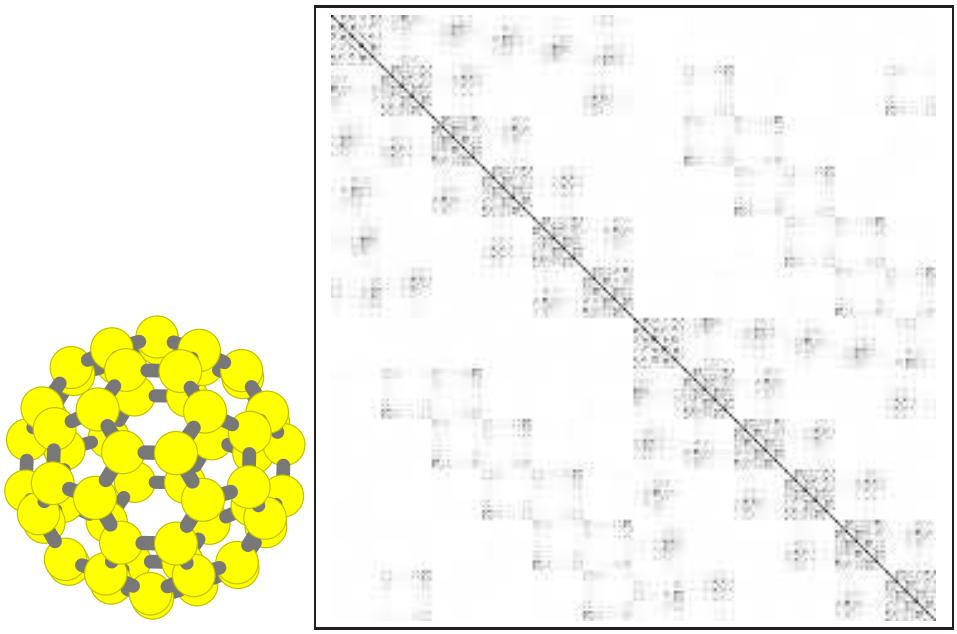


Figure 4.12.: Structure and subspace Hamiltonian matrix for a C_{60} Buckminster fullerene (Bucky ball) in the first iteration: The start guess KS wave functions are atomic orbitals. Centered at each of the 60 Carbon atoms, a set of $\{s, p_x, p_z, p_y\}$ atomic orbitals at each atomic position has been generated. Thus the matrix dimension is 60×4=240. The effective potential is created from a superposition of the atomic densities where s-orbitals are occupied with $2\,e$ and each p-orbital hosts $\frac{2}{3}\,e$. One can see various features in this plot: Twelve five-fold rings of Carbon atoms couple strongly among themselves visible as dark blocks of 20x20 occupying the matrix diagonal. The couplings of 5-rings to other 5-rings appear as 5 lighter blocks in each row or column. Further, the non-coupled atom pairs are indicated by blank entries as e.g. atoms on the other side of the soccer ball-like structure.

The subspace rotation method consists of three steps of which the first is the setup of the subspace matrices

$$H_{nm} = \langle \tilde{\Psi}_n | \hat{\tilde{H}} | \tilde{\Psi}_m \rangle, \tag{4.45}$$

$$S_{nm} = \langle \tilde{\Psi}_n | \hat{\tilde{O}} | \tilde{\Psi}_m \rangle. \tag{4.46}$$

For this step, the Hamiltonian and overlap operator have to be applied onto all

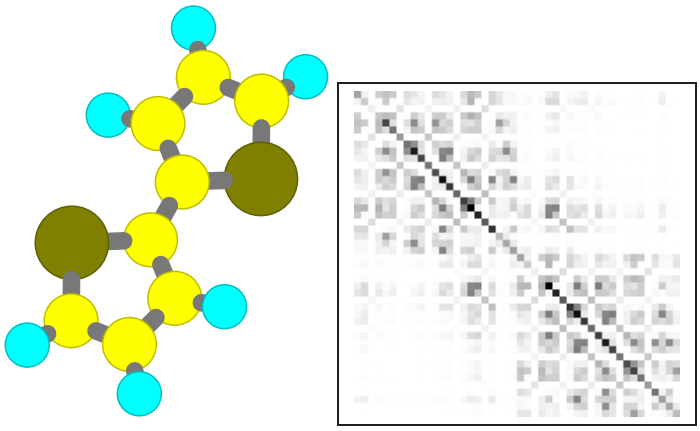


Figure 4.13.: Structure and subspace Hamiltonian for oligothiophene $(SC_4H_3)_2$ in the basis of atomic start orbitals. Strong inner-ring coupling becomes visible in two blocks of 23x23 on the diagonal. In addition to that, the C-C bond between the two rings appears as eminent off-diagonal matrix elements.

preliminary states $\tilde{\Psi}_n$, $n \in [1, N_{\text{bands}}]$, and $2N_{\text{bands}}^2$ scalar products need to be evaluated. Fortunately, we can exploit that $\hat{H}$ and $\hat{O}$ are hermitian operators. Hence, the subspace matrices are hermitian (symmetric) for complex (real) wavefunctions $\tilde{\Psi}$, $H_{nm} = H_{mn}^*$ and $S_{nm} = S_{mn}^*$. Hence, only their lower (or upper) triangular matrix needs to be evaluated. Then, the total number of scalar products is $N_{\text{bands}}^2 + N_{\text{bands}}$.

The second step is the solution of the generalized eigenvalue problem

$$(H_{nm} - E_n S_{nm}) X_{mk} = 0 \tag{4.47}$$

where a conventional eigensolver is applied. Finally, the third step is the rotation of the KS-states that gives rise to the name of this algorithm. The eigenstates of $\hat{H}$ are given by

$$\tilde{\Psi}'_n(\mathbf{r}) = \sum_k X_{nk} \tilde{\Psi}_k(\mathbf{r}). \tag{4.48}$$

The combination of both methods, the DIIS band update and the subspace rotation, take most of the computation time for an increasing system size. Therefore, we show the strategies of parallelization applied to both methods in the next chapter. The solution of the Poisson equation is parallelized as well. However, the Laplacian operator is implemented in a finite-differences and its action onto a trial vector is parallelized in complete analogy to the kinetic energy in the Kohn-Sham equation. Therefore, it is not discussed in particular detail.

CHAPTER 5

PARALLELIZATION

Self-consistent density functional calculations consist of three major tasks: the generation of the effective potential given an electron density, the solution of the Kohn-Sham equation for the lowest eigenstates, and the generation of a new electron density. Analyzing the scaling behavior of the different tasks we find clearly that the solution of the generalized eigenvalue problem that arises from the Kohn-Sham equation in the PAW formalism

$$\left(\hat{\tilde{H}}_{\sigma\mathbf{k}} - E_{n\sigma\mathbf{k}}\hat{\tilde{O}}_{\mathbf{k}}\right)|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle = 0 \tag{5.1}$$

becomes computationally most demanding when the system size is increased.

For this reason, `jüRS` parallelizes the solution of Equation (5.1) on three levels that differ strongly by their degree of communication. In priority order:

- $\sigma\mathbf{k}$-point parallelization → Section 5.2
- three-dimensional real-space domain decomposition → Section 5.3
- band parallelization → Section 5.4

Depending on the communication network of the computer system in use, one can achieve higher parallel efficiency by interchanging the latter two levels in priority. However, in the case of massively parallel supercomputer architectures connected by a torus network that provide a high bandwidths and small latencies in communication with the nearest cartesian neighbor nodes this ordering is advantageous.

5.1. Introduction to Parallel Efficiency

Amdahl's law [82] for parallelization of independent tasks without communication states that the total execution time of a workload parallelized with N processes,

$T_{exe}(N)$, consists of the time each process spends in serial tasks, T_{serial}, and the time spent in parallel tasks. The cumulative time that all processes spend on the execution of the parallel tasks, $T_{parallel}$, is a constant if we neglect possible overhead times of the task distribution that might depend on N. Assuming that the work can be distributed equally among all processes. Then the total execution time reads

$$T_{exe}(N) = T_{serial} + \frac{T_{parallel}}{N}. \tag{5.2}$$

We define the serial fraction α by

$$\alpha = \frac{T_{serial}}{T_{exe}(1)} = \frac{T_{serial}}{T_{serial} + T_{parallel}}. \tag{5.3}$$

Then, the speedup of the parallelization is defined by

$$\text{Speedup}(N) = \frac{T_{exe}(1)}{T_{exe}(N)} \tag{5.4}$$

and the parallel efficiency by

$$\text{Efficiency}(N) = \frac{\text{Speedup}(N)}{N} = \frac{T_{exe}(1)}{N \cdot T_{exe}(N)}. \tag{5.5}$$

Figure 5.1 shows the monotonously growing speedup for an infinite number of independent tasks. The asymptotic behavior for large N is the maximum speedup defined by

$$\text{Speedup}^{max} = \frac{T_{exe}(1)}{T_{exe}(\infty)} = \frac{1}{\alpha}. \tag{5.6}$$

The asymptotic behavior for small numbers of processes ($N \ll \frac{1}{\alpha}$) is the ideal speedup $\text{Speedup}^{ideal}(N) = N$ which corresponds to 100 % efficiency.

5.1.1. Load imbalance effects

We assumed earlier that the number of tasks to be distributed is unlimited. However, considering a finite number of tasks M which all require the same compute time, $T_{parallel}/M$, we need to include imbalances due to the distribution of M tasks onto N processes into the formula. The final synchronization makes all processes wait until the process with the largest number of tasks assigned has finished. The largest number of tasks is gives as $\lceil M/N \rceil$. $\lceil x \rceil$ is the nearest integer number larger than x. Equivalently, $\lfloor x \rfloor$ is the nearest integer number smaller than x. With this the total execution time can be rewritten as

$$T_{exe}(N) = T_{serial} + \frac{T_{parallel}}{M} \left\lceil \frac{M}{N} \right\rceil. \tag{5.7}$$

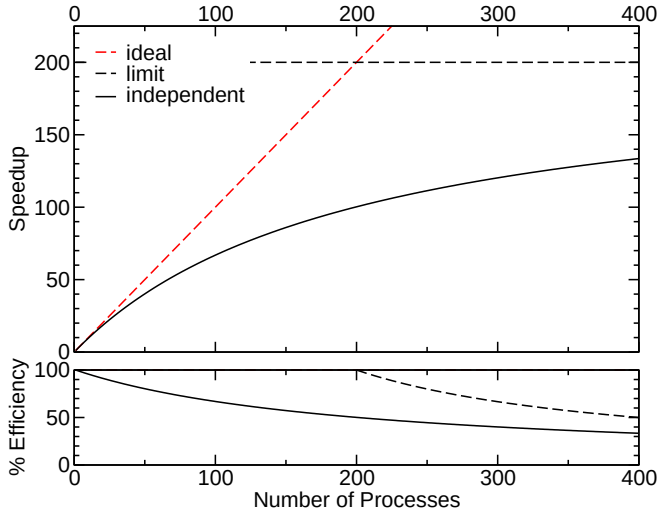


Figure 5.1.: Theoretical limitation of the parallel speedup (black solid line) described by Amdahl's law about non-parallel code. Here a serial fraction of α=0.5% limits the max. speedup to 200. In the absence of communication and load imbalance (infinite number of independent tasks) the speedup grows asymptotically towards the maximum $S_{max}=[\alpha]^{-1}$ (black dashed line).

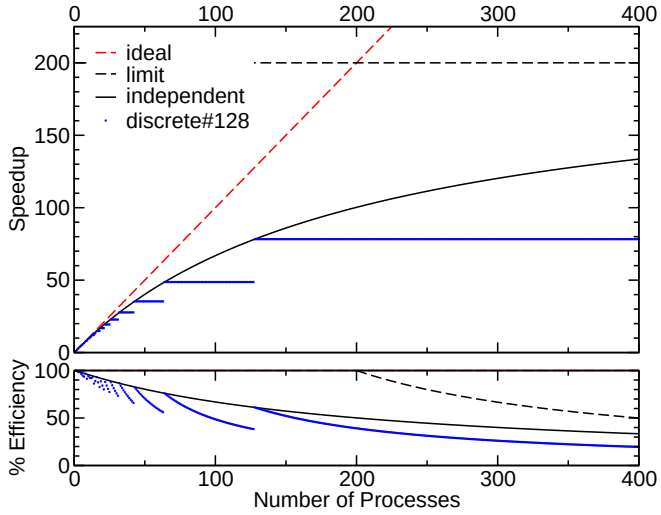


Figure 5.2.: Parallel speedup including load imbalance due to a discrete number of M=128 tasks and a serial fraction of α=0.5%. The maximum speedup cannot grow beyond $S(M)=[\alpha+\frac{1}{M}(1-\alpha)]^{-1}$, 78.3 in this plot.

The effect of load imbalances can be seen in Figure 5.2. Besides the limitation by the serial fraction α the speedup is now limited to the upper bound $\text{Speedup}(M) = [\alpha + (1-\alpha)/M]^{-1}$, the case where each processes is assigned exactly one task. For a parallelization with more processes than tasks, $N > M$, idle processes let the parallel efficiency drop strongly.

5.1.2. Intensive communication

In the previous sections we looked at Amdahl's law for independent tasks. However, the need for strong degrees of parallelism is introduced by large problem sizes that can only be divided into a combination of separate tasks plus communications. Assuming again that all tasks need the same compute time we can neglect losses due to load imbalances and assume a perfect synchronization. Nevertheless, the overhead of communication also introduces a time contribution T_{comm}. We thus rewrite Equation (5.2) into

$$T_{\text{exe}}(N) = T_{\text{serial}} + \frac{T_{\text{parallel}}}{N} + T_{\text{comm}}(N). \tag{5.8}$$

Here, the dependence of the communication time on the number of processes N and the type of communications plays a crucial role. The type of communication determines the communication pattern, i.e. which processes need to exchange data. We discuss two different types of communication here: the first is communication with a constant number of processes which leads to a constant communication time, independent of N. This can be combined with T_{serial} such that we find a renormalized serial fraction as shown in Figure 5.3. The second type involves all-to-all communication. Here, we need to distinguish two limit cases and one intermediate case. If the communication is volume intensive and the data packages become smaller with increasing degree of parallelization N, the communication time T_{comm} consists of an overhead part of the communication, e.g. `MPI`-handshake etc. that is independent of N and the actual sending process that scales with the transferred data volume. Assuming a data package size that behaves $\sim 1/N$, the latter contribution is a constant in the limit $N \longrightarrow \infty$ and renormalizes the serial code fraction. Also, if the data packages are of constant size but negligibly small the communication time can be approximated as being proportional to N. Then, only the overhead is seen and $T_{\text{comm}} = \gamma^{\text{lin}} N$. This leads to a growing contribution to the total execution time for growing N and thus a reduction of the speed in the limit of large N. A clear maximum of the speedup can be seen for all-to-all communication (orange line) in Figure 5.3.

An intermediate case between constant and linear scaling communication times can be found in global reduction operations. Usually, the data network is based on a (e.g. binary) tree structure such that the communication times arising due to

global reductions scales with the number of tree levels, i.e. $T_{comm} \sim \ln(N)$. This contribution grows when increasing N but its growths is weaker than a linear dependence.

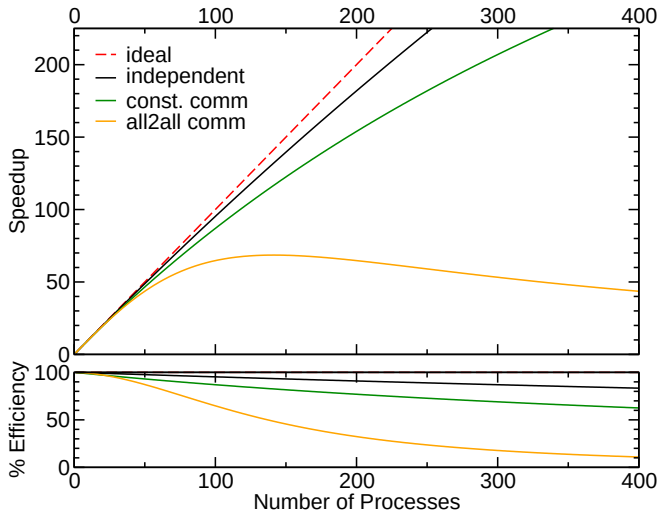


Figure 5.3.: Theoretical speedup including a serial fraction of α=0.05% and communication. A constant communication time (green line) can be regarded similar to a larger serial fraction. The new limitation then is $S_{max}=[\alpha+\gamma^{cst}]^{-1}$. Here, $\gamma^{cst}=10^{-3}$. All-to-all communication time scales linear with the number of processes and thus its speedup (orange line) shows a maximum at $\sqrt{(1-\alpha)/\gamma^{lin}}$. With $\gamma^{lin}=5\cdot10^{-5}$ the maximum can be found at 141.4 in this plot.

An example of a discrete number of independent tasks are spins and **k**-points discussed in the following section. Communicating tasks with a restricted communication pattern in `jüRS` are introduced in the framework of real-space domain decomposition (Section 5.3) and data volume intensive all-to-all communication is necessary in band parallelization (Section 5.4).

5.2. Spin- and k-point Parallelization

Periodic systems require a sampling of the first Brillouin zone by **k**, so called **k**-points. The Bloch theorem tells us that the KS Hamiltonian is block diagonal in **k** due to the locality of the effective potential. Furthermore, magnetic systems show a difference in the effective potentials $V_\uparrow$ and $V_\downarrow$. Restricting magnetic systems to those with a collinear spin, all coupling elements $H_{\uparrow\downarrow}$ vanish, such that the generalized eigenvalue problem

$$\left(\hat{\tilde{H}}_{\sigma\mathbf{k}} - E_{n\sigma\mathbf{k}}\hat{\tilde{O}}_{\mathbf{k}}\right)|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle = 0 \tag{5.9}$$

for both, **k**-points and collinear spins σ, can be treated as completely independent tasks. The results of these tasks need to be collected at two points.

- For the determination of the occupation numbers $f_{n\sigma\mathbf{k}}$ of the next iteration the eigenenergies $E_{n\sigma\mathbf{k}}$ of all states are required (see subroutine `Fermidistribution`).
- For the generation of the new valence density $\tilde{n}_v^\sigma(\mathbf{r})$ the eigenvectors $\tilde{\Psi}_{n\sigma\mathbf{k}}(\mathbf{r})$ of all states are required (see subroutine `construct_smooth_density`).
  The atomic density matrices $D^\alpha_{ij\sigma}$ are computed from the eigenvector's projection coefficients $\langle\tilde{p}^\alpha_i|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle$ (see subroutine `find_density_matrix`).

Both communication operations may be performed after the local summations have been executed so the only significant quantities that require global reduction operations are $\tilde{n}_v^\sigma(\mathbf{r})$ and $D^\alpha_{ij\sigma}$. The associated MPI communicator is called `equi_comm` since it connects regions of equivalent space **r** and images of the same atoms α belonging to different σ**k**-points.

The σ**k**-point parallelization is very efficient as long as the total workload for the diagonalization T_{diag} of the KS Hamiltonian is much larger than the time T_{pot} for the generation of the effective potentials. According to Section 5.1 we find an efficiency for the σ**k**-point parallelization

$$\text{Efficiency}_{\sigma\mathbf{k}}(N_p) = \frac{T_{diag} + T_{pot}}{N_p T_{pot} + T_{diag}} \tag{5.10}$$

$$\approx 1 - (N_p - 1)\frac{T_{pot}}{T_{diag}} \qquad \text{for} \qquad T_{pot} \ll T_{diag} \tag{5.11}$$

This estimate for parallel efficiency does not include losses due to load imbalance. The global communication operations along `equi_comm` introduce a synchronization that makes all ranks wait for the slowest one to finish its tasks. We therefore have to use the following formula including Load Balance (LB) effects. $M_{\sigma\mathbf{k}}$ is the number of discrete σ**k**-sampling points.

$$\text{Efficiency}^{LB}_{\sigma\mathbf{k}}(N_p) = \frac{T_{pot} + T_{diag}}{N_p T_{pot} + T_{diag}\frac{N_p}{M_{\sigma\mathbf{k}}}\left\lceil\frac{M_{\sigma\mathbf{k}}}{N_p}\right\rceil} \tag{5.12}$$

which is less or equal the above defined efficiency since

$$\frac{N_p}{M_{\sigma\mathbf{k}}}\left\lceil\frac{M_{\sigma\mathbf{k}}}{N_p}\right\rceil \geqslant 1 \qquad \text{for} \qquad M_{\sigma\mathbf{k}}, N_p \in \mathbb{N} \tag{5.13}$$

Obviously, $M_{\sigma\mathbf{k}}$ sets an upper limit for N_p. In the special case $M_{\sigma\mathbf{k}} = N_p$, the speedup is largest and each process elements performs calculations of one task. For higher N_p, we will encounter idle processes.

Regarding T_{pot}, a large fraction of the potential setup time goes to the spin-independent electrostatic potential. Therefore, the parallelization over collinear spins has not been extended to the generation of the spin-dependent XC-potentials.

The solution of the Kohn-Sham eigenvalue problem and the generation of the effective potentials are subject to the grid parallelization described in the following section.

5.3. Real-Space Domain Decomposition

The special strength of the real-space approach is the possibility to distribute the grid points of the uniform grid to parallel processes in a very simple and efficient way such that local operations can be performed without and localized operations with little inter-process communication, respectively. Equal-size domain decomposition divides each of the three spatial extends of the full grid into small cuboids, called domain. A good load balance for local operations is thus guaranteed when domains is assigned to one parallel process and the size of each domain is the same. Assuming that the real-space grid consists of $N_x \times N_y \times N_z$ grid points the distribution into domains reads

$$N_x \times N_y \times N_z = (p_x n_x) \times (p_y n_y) \times (p_z n_z). \tag{5.14}$$

It is necessary that the total number of processes p_{tot} can be factorized to $p_x \times p_y \times p_z$. For an equal numbers of grid points per domain, n_i, the grid sizes N_i with $i \in \{x, y, z\}$ must be a multiple of the domain numbers p_i. A distribution with n_i that differ among the processes is discussed in Section 5.3.3.

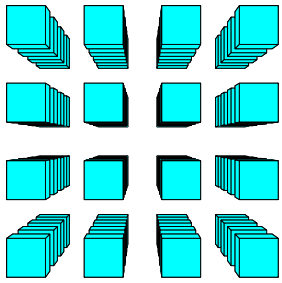


Figure 5.4.: Domain Decomposition with $4 \times 4 \times 4$ processes. Depending on the system's network the `MPI` task placement is important for the scaling behavior of the communication times of the nearest-neighbor communication.

5.3.1. Communication with cartesian neighbors

In real-space representation we can distinguish two major classes of operations, local and non-local. In the class of non-local operations we can identify the subset of global operations that involve all points in space and localized operations

with a finite range. Finite-difference derivatives (compare Section 4.1), for example, are localized operations whose range is given by the order of the finite-difference approximation. To perform localized operations we need to transfer data from the neighboring domains into a temporary storage which is realized by an halo-enlarged array, i.e. the array bounds are extended by a halo at the lower and at the upper boundary. The halo thickness is n_i^h grid points for each spatial direction $i \in \{x, y, z\}$. The range of the localized operation determines the minimum thickness of the halo in terms of number of grid points n_i^h. An upper limit for n_i^h in `jüRS` is the number of grid points per domain n_i. Any operation with a larger range than n_i would require that the data exchange procedure is called multiple times or direct communication with a second shell of neighboring processes needs to be established.

Standard finite-difference operators involve only non-local access to grid points along the cartesian directions. Thus we need to exchange $n_i^h n_x n_y n_z / n_i$ data points with the 6 cartesian face-neighbor processes, i.e. $n_x^h n_y n_z$, $n_x n_y^h n_z$ and $n_x n_y n_z^h$ in x, y and z direction, respectively. Figure 5.5 shows a classification of the 26 neighboring processes of the first shell into face-neighbors, edge-neighbors and vertex-neighbors. The ordering of the directions is arbitrary and best performed simultaneously. Some other localized operations, e.g. the Mehrstellen algorithm [80] for the approximation of the Laplacian operator involve the complete environment of a grid point and thus also $n_i n_x^h n_y^h n_z^h / n_i^h$ data points from the 12 edge-neighboring process along the diagonals and $n_x^h n_y^h n_z^h$ data points from 8 vertex-neighboring processes along the space diagonals. In order to avoid extra MPI communications with 12+8 processes for considerably smaller data packages (here we assumed that n_i^h is smaller than n_i, the only restriction however is $n_i^h \leqslant n_i$), we can circumvent this by introducing a non-arbitrary ordering of the data exchange operations: x, y, z, where the x-direction is associated with innermost array dimension `array(ix,:,:)`. After $n_x^h n_y n_z$ data points have been exchanged in x-direction, the copying for the y-direction will also include array from the x-halo region, i.e. $(n_x + 2n_x^h) n_y^h n_z$ data points are exchanged in y-direction. The last step then exchanges $(n_x + 2n_x^h)(n_y + 2n_y^h) n_z^h$ data points in z-direction. The information exchange with the edge-neighbors and vertex-neighbors is thus cast into a 2-step forwarding procedure. See Figure 5.6 for an example of a 1-step forwarding procedure in two dimensions.

5.3.2. Boundary Condition

A specialty of the real-space grid approach is the flexibility of boundary conditions (BCs). Three boundary conditions have to be chosen, one for each spatial direction. Standard BCs can either be periodic(1) or isolated(0).

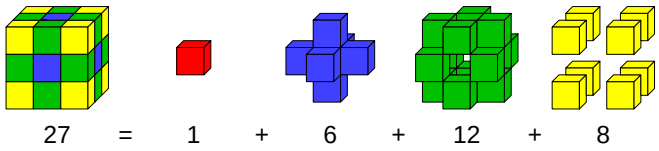


Figure 5.5.: Cartesian neighbor domains in a three-dimensional domain decomposition. The cube in the center (red) of 3×3×3 cubes has 26 neighbors, six cartesian direct neighbors (blue) sharing faces, so called face-neighbors, twelve edge-neighbors (green) and eight vertex-neighbors (yellow).

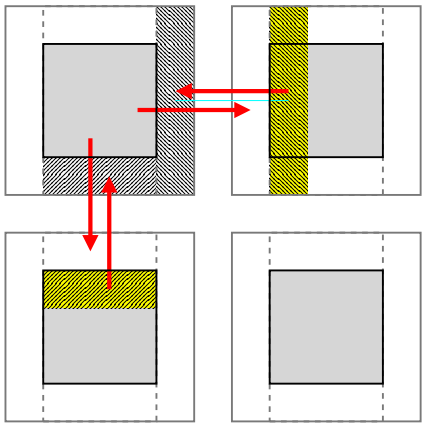


Figure 5.6.: Two-dimensional showgraph of x-y-ordered data exchange. The information flow mediates data from the vertex-neighbors by forwarding it along the cartesian neighbors such that communication is necessary only with direct cartesian edge-neighbors in 2D (face-neighbors in 3D). The data network of JUGENE supports direct neighbor communication on the hardware level in a three-dimensional torus network.

- Isolated BCs means that the simulation volume ends. Wave functions and potentials vanish outside the volume and thus the data exchange will return zeros. Array halos are set to zero in the routine `dataexchange`.
- Periodic boundary conditions connect the lower end of the volume at $-\frac{1}{2}L_i$ to the upper end at $\frac{1}{2}L_i$ and vice versa. L_i is the extend of the simulation volume in direction i where $i \in \{x, y, z\}$. During data exchange across a periodic boundary, a $\mathbf{k}$-dependent Bloch-factor $\mathbf{e}^{i\mathbf{kL}}$ is multiplied the the wave functions because the convention in `jüRS` is that the full Bloch wave functions $\tilde{\Psi}_{\mathbf{k}}(\mathbf{r})$ are treated rather than their lattice periodic parts $\tilde{u}_{\mathbf{k}}(\mathbf{r})$ where

$$\tilde{\Psi}_{\mathbf{k}}(\mathbf{r}) = \mathbf{e}^{i\mathbf{k}\cdot\mathbf{r}}\tilde{u}_{\mathbf{k}}(\mathbf{r}) \tag{5.15}$$

  and

$$\tilde{\Psi}_{\mathbf{k}}(\mathbf{r}+\mathbf{T}) = \mathbf{e}^{i\mathbf{k}\cdot(\mathbf{r}+\mathbf{T})}\tilde{u}_{\mathbf{k}}(\mathbf{r}+\mathbf{T}) = \mathbf{e}^{i\mathbf{k}\cdot\mathbf{T}}\mathbf{e}^{i\mathbf{k}\cdot\mathbf{r}}\tilde{u}_{\mathbf{k}}(\mathbf{r}) = \mathbf{e}^{i\mathbf{k}\cdot\mathbf{T}}\tilde{\Psi}_{\mathbf{k}}(\mathbf{r}) \tag{5.16}$$

  must hold for any allowed periodic translation vector $\mathbf{T}$.
- One may think of more elaborate BCs [50].

The total communication volume V_{comm} of the cartesian data exchange of each domain is a linear function of the artificial interfaces A_i created by the decompositions into domains. These surfaces and interfaces are given as

$$A_x = n_y n_z, \tag{5.17}$$

$$A_y = (n_x + 2n_x^h)n_z, A_z \qquad = (n_x + 2n_x^h)(n_y + 2n_y^h) \tag{5.18}$$

and the total communication volume

$$V_{comm} = 2n_x^h A_x + 2n_y^h A_y + 2n_z^h A_z. \tag{5.19}$$

Usually one chooses all halo thicknesses $n_i^h = n^h$ independent of the spatial direction such that

$$V_{comm} = 2n^h (A_x + A_y + A_z) \tag{5.20}$$

$$= 2n^h n_y n_z + 2n^h (n_x + 2n^h) n_z + 2n^h (n_x + 2n^h)(n_y + 2n^h) \tag{5.21}$$

$$= 2n^h (n_y n_z + n_x n_z + n_x n_y) + \left(2n^h\right)^2 (n_z + n_y + n_x) + \left(2n^h\right)^3 \tag{5.22}$$

We have to determine n_i such that V_{comm} is minimal under the constraint that the total number of grid points per domain $n_x n_y n_z$ is a constant. The solution is as close to a cube ($n_x = n_y = n_z$) as the constraint of integer numbers allows.

The inner product of two wave functions stored on a real-space grid with domain decomposition is performed in each domain locally first. Finally, a global reduction

over all domain ranks is necessary:

$$\int_{\text{Volume}} d^3\mathbf{r}\,\tilde{\Psi}(\mathbf{r})\tilde{\Psi}'(\mathbf{r}) = \sum_{r}^{p^x p^y p^z} \int_{\text{Domain}(r)} d^3\mathbf{r}\,\tilde{\Psi}(\mathbf{r})\tilde{\Psi}'(\mathbf{r}). \tag{5.23}$$

The cartesian `MPI` subcommunicator for the parallelization in three-dimensional domain decomposition is called `grid_comm` or `g%comm`, where `g` is of `type(grid)`, the cartesian grid descriptor.

The real-space domain decomposition in `jüRS` is applied to the coarse grid of the smooth wave function $\tilde{\Psi}_{n\sigma\mathbf{k}}(\mathbf{r})$. The resulting smooth density $\tilde{n}^{\sigma}(\mathbf{r})$ is treated on a twice as dense grid with grid spacing $h_d = h_c/2$. The solution of the Poisson equation (see Section 4.4.1) is either calculated on the dense grid or on an extra fine grid with grid spacing $h_e = h_d/2$. For simplicity of the implementation the denser grids are congruent with the coarse wave function grid and make use of the same grid communicator.

5.3.3. Non-uniform domain decomposition

The simplest approach towards a good load balance is a uniform domain decomposition of a grid, as pointed out in the previous section. However, this requires that the total number of grid points N can be factorized by the number of processes p, i.e. $N = n \cdot p$. Let us assume that N cannot be factorized but needs to be parallelized with p processes. Then we can find a distribution $N = p_> \cdot n_> + p_< \cdot n_<$ where $p_> + p_< = p$ and $n_> = n_< + 1$ i.e. the local numbers of grid points only differ by one. The load balance of this setup is $\frac{n_<}{n_>}$ (unless N can be factorized). On a three dimensional grid the effective load balance cannot be smaller than in one dimension.

The overall performance of the three-dimensional domain decomposition in `jüRS` is not affected by the placement of the domains that host $n_<$ or $n_>$ grid points if the cell volume is homogeneously filled with atoms. However, isolated boundary conditions require some vacuum regions where the wave functions may decay without being squeezed into the confinement given by the cell volume. Processes in the domain decomposition that are located in these vacuum regions will have to perform the application of the total kinetic energy and local potential operator only and most likely not involved into atomic projection operations. The workload of the latter depends strongly on the local geometry. Then it is advantageous to partition the processes in three groups, i.e. $N = \left\lfloor \frac{p_>}{2} \right\rfloor \cdot n_> + p_< \cdot n_< + \left\lceil \frac{p_>}{2} \right\rceil \cdot n_>$. This procedure guarantees that central processes of the domain decomposition store and process less grid elements which is advantageous for isolated BCs with vacuum

regions towards the boundary.

$$\underbrace{n_>, \ldots, n_>}_{\left\lfloor \frac{p_>}{2} \right\rfloor \text{ times}}, \underbrace{n_<, n_<, \ldots, n_<, n_<}_{p_< \text{ times}}, \underbrace{n_>, \ldots, n_>}_{\left\lceil \frac{p_>}{2} \right\rceil \text{ times}}.$$

An example of a three dimensional case is shown in Figure 5.7.

Deactivate `EQUAL_DOMAINSIZES` in module `configuration` for this feature.

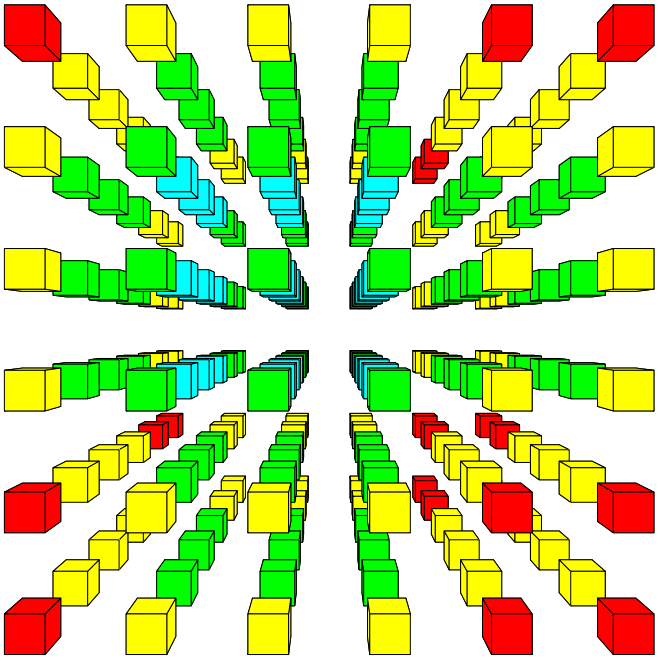


Figure 5.7.: Load balancing of the grid tasks in a $6 \times 6 \times 6$ domain decomposition of $6n + 3$ grid points, $n \in \mathbb{N}$. Four colors are present according to the total number of grid points per domain which is $\in \{n_<^3, n_<^1 n_>^2, n_<^2 n_>^1, n_>^3\}$.

The atomic projection operation and the addition operation are implemented in a very similar code structure. It can be observed that their execution time differs on most platforms by less than 5 %. Hence, their work balance with respect to the domain decomposition is theoretically equal. However, minor differences can be seen in Figure 5.8 at the example of a C_{60} molecule distributed on $8 \times 8 \times 8$ domains.

Considering a heterogeneous distribution of the atoms in the simulation volume we may optimize the workload balance of the domains using a non-uniform domain decomposition. The domain size is adjusted according to the local concentration of atoms (weighted with the number of projector functions of each atom). The domain size of domains with a high (low) concentration of atoms are decreased (increased). The workload of the kinetic energy in finite-difference approximation (red blocks in Figure 5.9) is proportional to the domain size and the workload of the

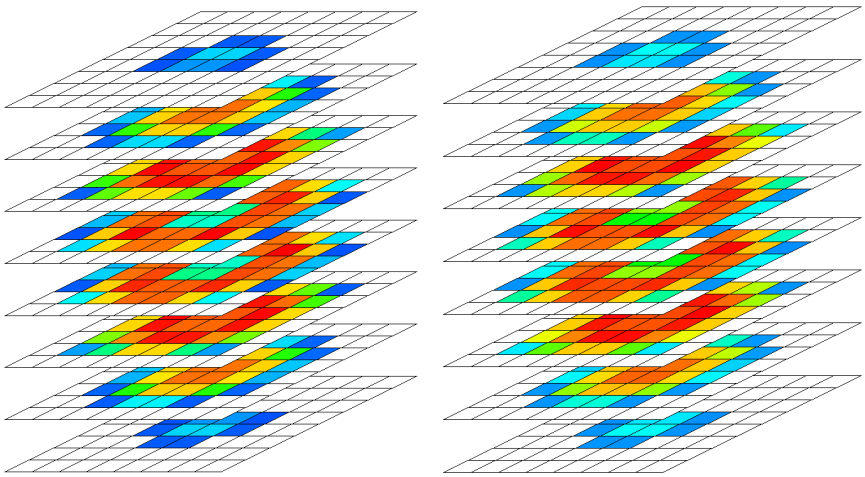


Figure 5.8.: `Scalasca` [83, 84] visualization of the load balance of the atomic projections $c_i = \langle \tilde{p}_i | \tilde{\Psi} \rangle$ (left) and addition operations $|\tilde{\Psi}'\rangle := |\tilde{\Psi}\rangle + \sum_i c_i |\tilde{p}_i\rangle$ (right). The measured system is a C_{60} Buckminster fullerene in a $8 \times 8 \times 8$ domain decomposition. White colored processes are not involved in the operations (vacuum).

atomic projection/addition operations (blue/green block in Figure 5.9) grows with increasing domain size as well. This growths cannot be described as proportional since we assumed variations in the local concentration of atomic tasks. Figure 5.9 shows that adjusted domain sizes help to reduce the total execution time by 26 % assuming a load balance of 74 % before. The communication between the atomic tasks is indicated as a blocking operation in setup 1, i.e. the communication introduces synchronization. In setup 2 the communication is indicated at different times. The concept of non-blocking communication can be applied here but the domains involved in more than one projections/additions need to reorder the processing of the atomic tasks according to the gradient of the centration of atoms. Otherwise, also non-blocking communication may lead to idle processes waiting from `MPI` messages. This constraint increases the complexity of finding an optimized atom order, see 5.3.6.

5.3.4. Atom owner processes

As an intrinsic property of the PAW method, some calculations have to be performed on the radial grids inside the spheres. In order to balance the workload and memory usage these atomic tasks are assigned to one grid process, i.e. one rank of the domain decomposition that *owns* the atom.

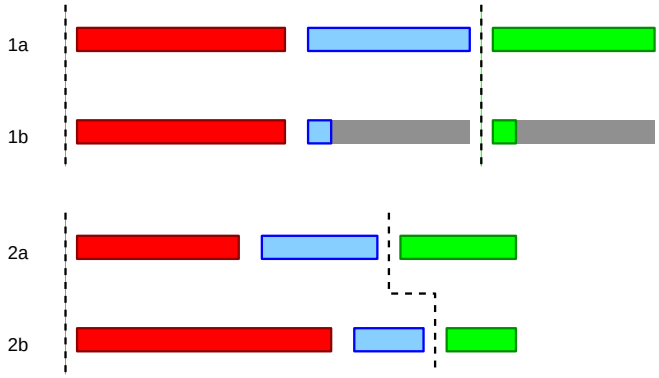


Figure 5.9.: The execution time of the application of $\hat{\tilde{H}}$ to a vector $|\tilde{\Psi}\rangle$ consists of three contributions: FD-kinetic energy (red), atomic projection operation (blue) and atomic addition operation (green). Equal domain sizes and blocking communication lead to setup 1 where domains with less atoms (1b) idle (gray) while waiting for the domain with the most workload in projection and addition (1a). Setup 2 is the ideal case since the domain sizes are adjusted to the local concentration of atoms and the communication does not introduce synchronization.

5.3.5. Parallelization of the projection

The artificial interfaces from the domain decomposition will intersect with the cloud of grid points $\mathbf{r}$ where localized atomic functions are nonzero. These may be either the PAW projectors $\tilde{p}^a(\mathbf{r})$ (see Section 4.3.3) or in a similar fashion the atomic compensators $\hat{g}^a(\mathbf{r})$ (see Section 3.4.5). We discuss this issue with the atomic projectors here. The domain decomposition forces us to perform a local application of the projector part of the localized function that intersects with the domain. For the projection operation $\langle\tilde{p}|\tilde{\Psi}\rangle$ this requires a global reduction over the domain ranks r:

$$\int_{\text{Volume}} d^3\mathbf{r}\, \tilde{p}(\mathbf{r})\tilde{\Psi}(\mathbf{r}) = \sum_{r}^{p^x p^y p^z} \int_{\text{Domain}(r)} d^3\mathbf{r}\, \tilde{p}(\mathbf{r})\tilde{\Psi}(\mathbf{r}). \tag{5.24}$$

Within this reduction operation the contribution from most domains will vanish due to the localized character of the projector. We can therefore identify for each atom a group of processes that are involved in the projection operations with a non-vanishing contribution. This group is a subset of the grid communicator. In the current implementation, this is realized by forming a separate `MPI` communicator `a%comm` for each atom[1]. Introducing atomic communicators leads to the advantage of reduction operations that only need to be performed in contributing processes. The result of the reduction is stored in the atom owner process only. For conve-

[1]Note that the number of `MPI` communicators is limitated to a maximum that depends in the implementation of the communication library.

nience, the atom owner has MPI-rank 0 (master) within the atomic MPI communicator. Choosing the atom owner rank such that the atomic nucleus is located inside its domain will ensure that the atom owner is simultaneously a contributing process. However, a distribution of atom owner processes that minimizes the number of atomic ownerships per domain could balance the execution of atomic task better.

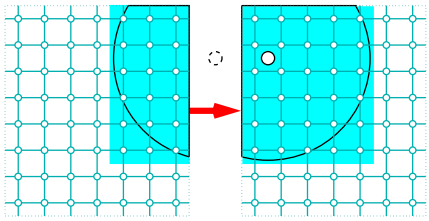


Figure 5.10.: Parallelization in domain decomposition of the non-local projection operations. The inner product of a wave function with the localized projector function is executed in each domain locally. The results need to be reduced in the atom owner process, usually the domain that hosts the atomic nucleus. The atomic tasks on the radial grids are assigned to the right domain. A circle indicates the position of the atomic nucleus. The dashed circle shows the nuclear position for the left domain.

The atomic communicator serves for both directions, i.e. to gather the domain contributions to the projection coefficients and to broadcast information that are stored in the atom owner process to the domains contributing to the localized functions of this atom. This is needed for the addition of the projector functions to a wave function as appearing in the action of the non-local parts of the PAW transformed Hamiltonian (Equation (3.80)) and, similarly, for the multipole moments $\Delta q^a_{\ell m}$ of the compensators in the electrostatic construction (Equation (3.53)).

5.3.6. Atomic communication ordering

In order to avoid serialization of the atomic communications, we have to find an ordering, such that these steps are performed in a minimum number of cycles. Within each cycle only such groups of processes are allowed to communicate (`MPI_Broadcast` or `MPI_Allreduce`) that are disjoint, i.e. no process is member of more than one of the groups in this cycle. The following algorithm describes a way to set up the table of disjoint atomic communicator groups.

Each process stores an array of boolean values $b_r(a)$ which is `true` if and only if the process r is involved in the atomic communicator of atom a. The spatial shape of the group of processes that belongs to an atomic communicator reminds

of a rather edgy cloud. The logic table $B(a_1, a_2)$ is `true` if any of the processes in `a1%comm` is also involved in the communications along `a2%comm`. Here, B stands for blocking. B is found via a global reduction operation with `OR` over all N_p processes ranks, i.e.

$$B(a_1, a_2) = \bigvee_{r=0}^{N_p-1} b_r(a_1) \wedge b_r(a_2) \tag{5.25}$$

Obviously, $B(a_1, a_2)$ is symmetric under exchange $a_1 \leftrightarrow a_2$ and the diagonal entries $B(a, a)$ are always `true`, such that the minimum number of blockings is the number of atoms. These two aspect can be exploited for saving communicational volume and temporary memory.

Now, the table $A(a_1, a_2)$ of groups which may be treated in the same cycle and thus at the same time is the exact negation of B:

$$A(a_1, a_2) = \neg B(a_1, a_2)$$

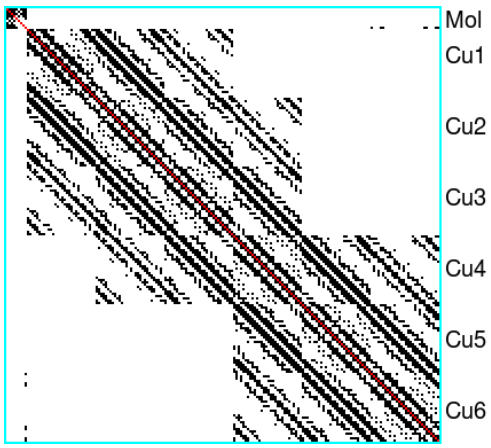


Figure 5.11.: Boolean matrix of blockings for thiophene on a Cu(111) surface. The order of atoms in this setup is 4×C, 4×H, S followed by the Cu layers 1 through 6 with 30 Cu atoms in each layer. The $8 \times 8 \times 16$ domain decomposition is not aligned to the layers of the Copper slab in z-direction. The upper left 9×9 submatrix stems from the molecule. It is almost block-diagonal except for some matrix elements that disable the atomic communication operation of S and the Cu surface atoms of layer 6 to be treated simultaneously.

If we consider A to be the adjacency matrix of the graph $G(\{a\}, E)$, where the set of all atomic communicators $\{a\}$ represents the vertices of the graph and the edges E are determined via the matrix A, we have reduced the problem to the well known mathematical multi-coloring problem, i.e. a problem of finding the minimum number of complete subgraphs. Each of those subgraphs then stands for one color, i.e. one subset of atomic communicators that can be treated at the same time without

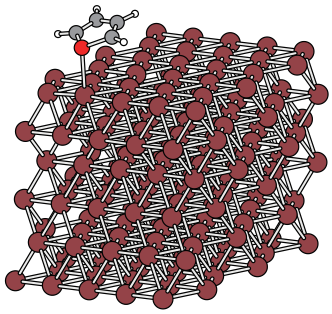


Figure 5.12.: Cu(111) surface with an adsorbed thiophene molecule on top of layer 6. Layers 1 through 3 have been kept at the distance according to the bulk FCC lattice constant $a_{bulk} = 3.636\,\text{Å}$. Courtesy of M. Callsen and Sh. Tsukamoto.

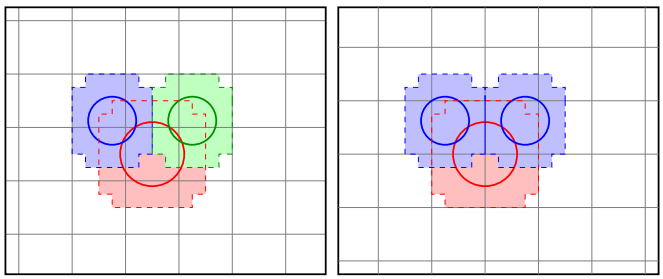


Figure 5.13.: Blocking of atomic communicators depends on the alignment of the domain boundaries. The same H_2O molecule may lead to complete serialization (left) where 2 domains contribute to all three projection operations or the simultaneous treatment of the two Hydrogen atoms in one color (right).

serialization of the operations. The requirement of completeness of each subgraph corresponds to the requirement that no pair of two atomic communicators in a color is blocking.

Given the logic table $B(a_1, a_2)$ the following pseudo code will find a coloring:

```
for a1 in atoms
    for c in colors
        for a2 in c
            if B(a1,a2)
                go to next c in colors
        no atom has been blocking
            add a1 to c
    end of colors reached but no open slot found
        create new color c
        add c to colors
        add a1 to c
```

The multi-coloring problem has many solutions. Which one of all possible solutions is found by this algorithm depends on the initial order of atoms. The solution of this problem is only unique in two trivial limit cases. The first trivial case occurs when all atoms only block with themselves. Then the average number of blockings per atom is unity, $N_b = N_a$ and only one color is needed, $N_c = 1$. If the average number of blockings per atom matches N_a, i.e. $N_b = N_a^2$, all atoms block with all other atoms an the communication will serialize completely. Then the number of colors N_c equals N_a as visible in Figure 5.14. These data have been generated by applying the above algorithm to random blocking tables while measuring the resulting number of colors. We can observe that N_c converges from above towards a constant for growing N_a and fixed ratio N_b/N_a. If we now interpret the ratio N_b/N_a as a coordination number, we can see that the convergence of the the number of colors, i.e. the number of communication cycles, converges to a function of N_b/N_a for large system sizes.

Figure 5.15 shows an analysis of the number of colors as a function of a constant ratio N_b/N_a. Assuming that all atoms with k shells of an face-centered cubic (FCC) crystal structure are in a blocking relation, the ratio equals the number of atoms in all shells up to this radius. For $k \in [0, 9]$ the number of atoms in the FCC shell at radius $a_0\sqrt{\frac{k}{2}}$ is 1, 12, 6, 24, 12, 24, 8, 48, 6 and 36, respectively. We can find the sums of all shells up to this radius in the legend of Figure 5.15. We can observe how all numbers of colors converge to a constants. The convergence occurs slower, the larger the blocking radius k is. Furthermore, the converged numbers of colors depend on the number of blockings in a non-linear fashion.

The serial multi-coloring algorithm scales linear with N_a in its memory consumption and the computational time is proportional to N_a^2 where the prefactor is smaller than 1.5 ns on a 2.5 GHz compute core. For example $N_a = 1024$ takes

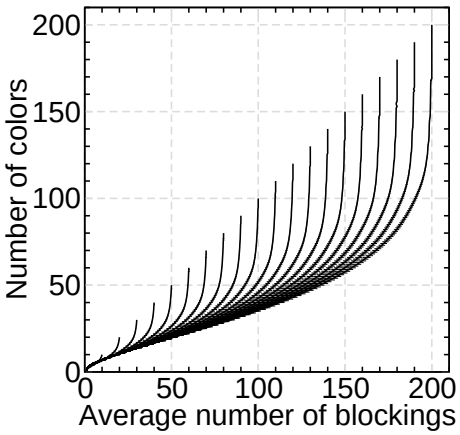


Figure 5.14.: Number of colors N_c as a function of the average number of blockings per atom N_b/N_a. Each branch of the graph corresponds to a different number of atoms N_a with $N_a \in \{10, 20, ..., 190, 200\}$. At a constant ratio N_b/N_a, the number of colors converges to a constant for increasing system size N_a.

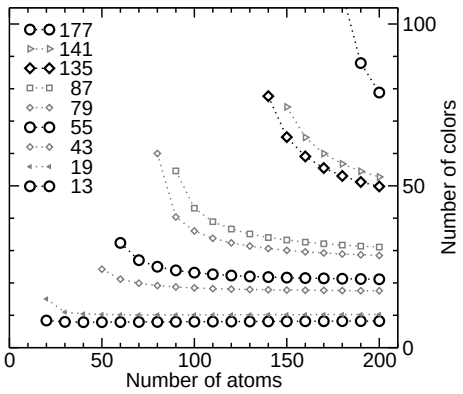


Figure 5.15.: Number of colors N_c as a function of the number of atoms N_a for various blockings per atom N_b/N_a. Assuming an FCC crystal structure for example, each atom blocks with its neighbors. Different shells of neighbors have been considered up to the radii $a_0\sqrt{\frac{k}{2}}$ with $k \in [1, 9]$ and the FCC lattice constant a_0.

1.3 msec. These execution times are much smaller than the time consumed by e.g. IO performed at start-up of the calculation. For this reason, a parallelization of this algorithm is obsolete.

5.3.7. Performance

The parallelization of the real-space grid in domain decomposition is expected to be extremely efficient on JUGENE due to the machines three-dimensional torus network that provides small latencies for the communication with the nearest cartesian neighbor nodes.

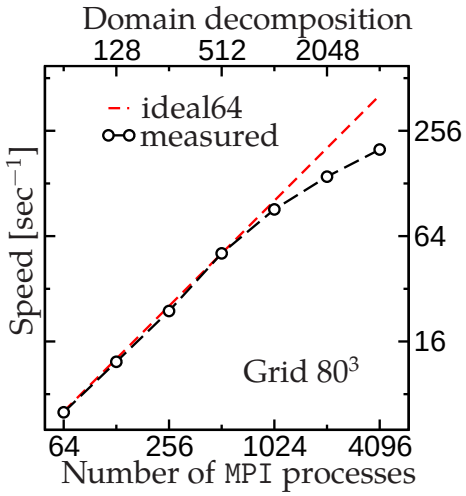


Figure 5.16.: Strong scaling of the grid parallelization. A real-space grid with $80 \times 80 \times 80$ points is parallelized in 4^3 through 16^3 domains where the number of domains per direction are chosen from $\{4, 8, 16\}$. The ideal performance at 8^3 domains can be explained by caching effects.

Figure 5.16 shows the strong scaling for the parallelization of the real-space grid in domain decomposition. $80 \times 80 \times 80$ real-space grid points are parallelized in 4^3 through 16^3 domains where the number of domains per direction are chosen from $\{4, 8, 16\}$. One may observe the ideal performance at 8^3 domains where caching effects compensate for the speed losses due to communication as discussed by Gustafson [85]. The local problem size at the rightmost data point is as small as 5^3 grid points per domain, n_i=5 for $i \in \{x, y, z\}$. This is very close the limit when the communication volume is the complete local array size, i.e. $n_i = n_i^h$. jüRS is limited by $n_i \geqslant n_i^h$ since communication operations with next-nearest neighbors or recursive calling of the data exchange routine would be required otherwise. We can observe that the efficiency is close to 100 % up to 1024 domains and drops to roughly 50 % for 16^3 domains. The drop is expected at higher parallelization numbers for larger grid sizes. As discussed in Section 5.1, the speedup will grow asymp-

totically towards a constant limit and not exhibit a maximum as in communcation intensive parallelizations. However, the restriction $n_i \geqslant n_i^h$ ensures a reasonable parallel efficiency at any time. Once could think of an optmization that improves the speedup in this exact case by omitting the reshaping of the array to their send buffers, if the application is frequently running in this limit.

5.3.8. Subspace Rotation in Domain Decomposition

A time critical point during the calculation of a large systems is the subspace rotation as introduced in Section 4.4.3. This section outlines the optimization that is necessary to use this algorithm efficiently with the domain decomposition of the real-space grid.

Due to its enormous size the KS-Hamiltonian $\hat{\hat{H}}$ in the representation of the real-space grid points $\langle \mathbf{r}|\hat{\hat{H}}|\mathbf{r}'\rangle$ is never stored explicitly. The same holds for the overlap operator $\hat{\hat{O}}$. However, the structure and especially the sparsity of $\hat{\hat{H}}$ and $\hat{\hat{O}}$ allow that we can provide the action of the operators onto any wave function $\tilde{\Psi}(\mathbf{r})$. The evaluation of this action scales linearly with the grid size. For the setup of the subspace matrices the matrix elements

$$H_{nm} = \langle \tilde{\Psi}_n|\hat{\hat{H}}|\tilde{\Psi}_m\rangle, \tag{5.26}$$

$$S_{nm} = \langle \tilde{\Psi}_n|\hat{\hat{O}}|\tilde{\Psi}_m\rangle \tag{5.27}$$

need to be computed for all $n, m \in [1, N_{\text{bands}}]$. This is done by applying the operators to $|\tilde{\Psi}_m\rangle$ and evaluating the inner products $\langle \tilde{\Psi}_n|\hat{\hat{H}}\tilde{\Psi}_m\rangle$ and $\langle \tilde{\Psi}_n|\hat{\hat{O}}\tilde{\Psi}_m\rangle$ for $n \in [1, N_{\text{bands}}]$. Because of the domain decomposition, each inner product is associated to a global reduction operation over all the domain ranks, see Equation (5.23). A straightforward implementation would thus call `MPI_allreduce` $2N_{\text{bands}}^2$ times for the reduction of a single number, $N_{\text{bands}}^2 + N_{\text{bands}}$ times if the hermiticity is exploited. The data volume of a single number is negligible such that the communication time is roughly the number of calls times the overhead time needed to establish the communication. As mentioned earlier, global reduction operations in a tree network scale with the number of tree levels, thus a domain decomposition with a larger number of `MPI` processes is expected to require more communication time.

An optimization to prevent this unfortunate scaling of the communication time with respect to N_{bands} is introduced by rearranging the inner products $\langle \tilde{\Psi}_n|\hat{\hat{H}}\tilde{\Psi}_m\rangle$ and $\langle \tilde{\Psi}_n|\hat{\hat{O}}\tilde{\Psi}_m\rangle$. The domains first perform the inner products locally and store the temporary matrices h_{nm} and s_{nm}. Then, the global reduction operation is called for the two matrices. In this situation, the communication overhead time becomes negligible and the communication time is determined by the data volume and the

Method	# of Ops	Vol.	Example # of Ops	N_{bands}=8×437$\frac{4}{8}$ Vol.	Comm. time[sec]
Simple	$2N_{bands}^2$	1	24,500,000	8 Byte	245
Double	N_{bands}^2	2	12,250,000	16 Byte	-
COLLECT	$N_{bands}N_p$	$2\,n_{bnd}$	28,000	7 kByte	51
COLLECT2	N_p^2	$2\,n_{bnd}^2$	64	3 MByte	40

Table 5.1.: Acceleration of the setup of the matrix elements H_{nm} and S_{nm} for subspace rotation. The communication needed for the global reduction of each scalar product over the 16^3 domain decomposition can be collected in order to reduce the number of communication calls and thus the time spent in operational overhead. The example system has been measured on 32768 JUGENE processes ($16^3 \times 8$). A total of 3500 bands are distributed over 8 sets as $4 \times 438 + 4 \times 437$ for a good load balance, see Section 5.4. In general $N_{bands} = N_p \times n_{bnd}$. From the two extreme cases we can read off an overhead time of about 10 µsec and an effective bandwidth of 4.8 MByte/sec.

bandwidth of the network. The results of this collection of communication operations are shown in Table 5.1 in combination with band parallelization. An effective bandwidth of 4.8 MByte/sec on a 16^3 domain decomposition (JUGENE,-mode VN) and overhead times of 10 µsec have been found.

The storage of the subspace matrix elements H_{nm} and S_{nm} and the temporary quantities h_{nm} and s_{nm} of domain-local scalar products on each node can easily exceed the available memory. The next section shows how we can solve this problem.

5.3.9. Parallel Eigensolver in the Subspace

As the central task of the subspace rotation method we need to solve the generalized eigenvalue problem

$$\sum_m \left(H_{nm} - E_n^{new} S_{nm}\right) X_{mk} = 0. \tag{5.28}$$

All indices n, m and k are in the range $[1, N]$. In the subspace rotation method outlined in Section 4.4.3 the matrix dimension N is the number of bands, N_{bands}. There are two major technical issues about this step:

- Workload of a direct matrix diagonalization $\sim N^3$
- Memory footprint of a full matrix $\sim N^2$

Trying to perform the diagonalization step of the subspace rotation in serial (e.g. using DSYGV or ZHEGV from LAPACK) will lead to a hard limit given by memory

and a soft limit given by an enormous drop of the overall parallel efficiency due to Amdahl's law about non-parallel code sections (see Section 5.1). Therefore, we aim for solving both, memory and workload issues, at a time by applying a parallel eigensolver.

We have chosen the ScaLAPACK library [86] as parallel eigensolver, in particular the routines for the real symmetric (`PDSYGVX`) and complex hermitian (`PZHEGVX`) generalized eigenvalue problem. ScaLAPACK is based on a distributed matrix storage scheme that can tackle the memory-per-node issue. This parallelized version of LAPACK enables to decrease the execution time of the matrix diagonalization. ScaLAPACK is based on the Basic Linear Algebra Communication Subprograms (BLACS). All `MPI` processes available for the eigensolver are incorporated into a two-dimensional BLACS grid. The two dimensions represent the two indices of a matrix. The real-space domain decomposition is running on p_{tot} `MPI` processes where p_{tot} is factorized as $p_x p_y p_z$. Since we treat only square matrices and, according to the ScaLAPACK documentation [86], the performance for square matrices is best at a square BLACS grid, we restrict the number of `MPI` processes involved to the smallest square number $N_p^2 \leqslant p_{tot}$.

ScaLAPACK's distribution scheme is a block cyclic distribution, i.e. the full matrix is divided into N_B blocks. Each block has a constant block size of BS elements along both, row and column index [86], i.e. $N_B = \left\lceil \frac{N}{BS} \right\rceil$. These blocks are distributed among the BLACS grid ranks such that the maximum number of blocks per rank is $M_B = \left\lceil \frac{N_B}{N_p} \right\rceil$. Given a global index $j \in [1, N]$ (row or column index) of a matrix element we can compute the block index $j_B \in [0, N_B\text{-}1]$ by $j_B = \left\lfloor \frac{j-1}{BS} \right\rfloor$. A one-dimensional example is shown in Table 5.2. The process rank $r \in [0, N_p\text{-}1]$ on the BLACS grid determines whether a matrix block is supposed to be stored in the local memory of this process. The block cyclic distribution requires that the rank r must match $j_B \% N_p$. Only the storing processes further computes the corresponding index i of the local storage array. It determines the local block index $i_B = \left\lfloor \frac{j_B}{N_p} \right\rfloor$ with $i_B \in [0, M_B\text{-}1]$. Then, the local array index is given by $i = j - BS\,(j_B - i_B)$, where $i \in [1, M]$. The local memory is limited by the upper maximum of $M = M_B \cdot BS$ elements in both directions, rows and columns. Advantageous values for the block size BS are found by performance analysis and depend on the implementation and cache optimization of the underlying Basic Linear Algebra Subprograms (BLAS) library.

The recently developed parallel eigensolver Elemental [87] makes use of a cyclic distribution, a simpler distribution scheme than ScaLAPACK. One can retrieve it from the block cyclic distribution with a block size of unity. Then $j_B = j\text{-}1$ and $i_B = i\text{-}1$ simplify the distribution prescription. Table 5.3 shows an example of this distribution scheme.

rank 0			rank 1			rank 2		
[01 02]	[07 08]	[13 14]	[03 04]	[09 10]	[15]	[05 06]	[11 12]	[ ]

Table 5.2.: Example for ScaLAPACK's block cyclic distribution in one dimension. Distribution of 15 vector entries (01-15) onto 3 ranks with 8 blocks of block size 2, indicated by square brackets. The blocks are introduced to increase the performance of local BLAS operations on memory-coherent parts of the vectors or matrices. Usual block sizes are 32, 64 or 128, depending on the cash line length of the system.

rank 0	rank 1	rank 2
01 04 07 10 13 16	02 05 08 11 14 17	03 06 09 12 15

Table 5.3.: Example for a one-dimensional cyclic distribution. Distribution of 17 vector entries (01-17) onto 3 ranks. The memory usage is more balanced than in the case of a block cyclic distribution.

Performance

The solution of the generalized eigenvalue problem in parallel yields a substantial speedup compared to single processor performance which easily take several hours for this task. However, incorporating too many `MPI` processes into the BLACS grid may hinder a speed gain in the solution of the generalized eigenvalue problem since this is a strongly communicating task. The last row in Table 5.4 shows that mapping all 4096 processes of a 16^3 domain decomposition onto a 64^2 BLACS grid only accelerates by 9.6 % compared to the performance of the 32^2 grid. In the latter situation 75 % of the `MPI` processes were running idle during the execution of `PDSYGVX`. This relatively low speed gain tells us that the 64^2 grid for a 14k-matrix is already close to the turning point when the execution time starts to grow for a parallelization with more processes. For a smaller matrix size N the turning point is expected to occur earlier. We can observe this behaviour at $N = 3500$ in Table 5.4. It is therefore useful to restrict the BLACS grid the the number of processes at which the highest speedup can be achieved.

5.4. Band Parallelization

Density functional calculations in the framework of a wave function based method require the evaluation of (at least) the number of occupied KS-states. When the number of atoms in a system grows, both, its volume and its number of (valence) electrons grow linearly. To be able to tackle the more than quadratically growing workload we apply the real-space parallelization to the simulation volume as pointed out in Section 5.3. We still have to face the challenge of a linearly growing

Matrix Size N	BLACS grid	Time[sec]	System
3500	16^2	10.1	GST_{555}
3500	64^2	11.0	GST_{555}
14336	16^2	141.0	GST_{888}
14336	32^2	94.1	GST_{888}
14336	64^2	85.8	GST_{888}

Table 5.4.: Performance data of the ScaLAPACK routine `PDSYGVX` on `JUGENE` for various node numbers. A block size of 64 has been used in all runs. Using all available 16^3 `MPI` processes of the domain decomposition for solving the generalized eigenvalue problem of dimension 14336 can be as fast as 85.8 sec where a workstation would need more than ten hours.

number of KS-states. Different $\sigma\mathbf{k}$-points are parallelized on the outermost level as pointed out in Section 5.2. However, large systems often treat only the Γ-point such that the maximum is 2-fold parallelization of spins if applicable. Therefore, we focus in this section onto the number of bands N_{bands}. We employ a simple distribution of the total number of bands onto p_{bnd} parallel sets as $N_{bands} = p_{bnd} \times n_{bnd}$. Each set stores n_{bnd} bands. We use the term *sets* here, since each set is internally parallelized in a domain decomposition such that we cannot speak of a single `MPI`-process. n_{bnd} differs by 1 band between the sets if the total number of bands is not an integer multiple of p_{bnd}.

The DIIS band update method, as introduced in Section 4.4.2, acts onto each band separately. Therefore, each band set perform the DIIS update onto the locally stored bands. Workload and memory are distributed simultaneously. However, band parallelization is more complex for the case of reorthogonalization by means of the subspace rotation, as pointed out in the following Section.

5.4.1. Subspace matrix setup

The parallelization of bands introduces substantial communication into the matrix setup step. Consider the subspace rotation algorithm as described in Section 4.4.3 (serial) and Section 5.3.8 (parallel).

The total number of bands is parallelized by $p_{bnd} > 1$ band sets, i.e. $N_{bands} = p_{bnd} \times n_{bnd}$. The distribution subdivides the matrices H_{nm} and S_{nm} into blocks of $n_{bnd} \times n_{bnd}$ elements. The diagonal blocks consist of scalar products of $|\hat{\tilde{H}}\tilde{\Psi}_m\rangle$ ($|\hat{\tilde{O}}\tilde{\Psi}_m\rangle$) and bands $|\tilde{\Psi}_n\rangle$, both in local storage as in Section 5.3.8. In addition, we need to evaluate $p_{bnd}(p_{bnd}-1)$ off-diagonal matrix blocks. These consist of scalar products between the locally stored $|\hat{\tilde{H}}\tilde{\Psi}_m\rangle$ and bands $|\tilde{\Psi}_n\rangle$ stored in the other sets.

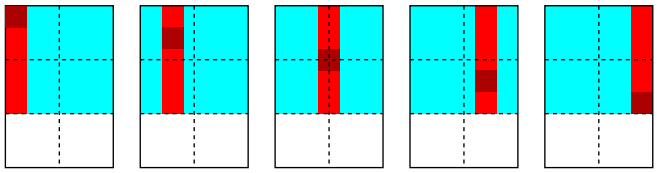


Figure 5.17.: Setup of matrix elements in band parallelization: In this example the inner real-space parallelization level contains 6 processes in a $3 \times 2 \times 1$ domain decomposition. A 2×2 subset of the 6 grid processes contributes to the BLACS grid since square grids achieve the highest performance on square matrix operations in ScaLAPACK. White-colored processes idle during the ScaLAPACK call. Bands are distributed over 5 parallel sets. Each band set evaluates the columns of the matrix H_{nm} (and equivalently S_{nm}) that match the locally stored bands (indicated in red). Those matrix elements that result from scalar products of two locally stored bands are marked in a darker tone and can be computed without exchanging wave functions along the band communicator. For the other matrix element, volume intensive communication between the band sets in necessary.

Figure 5.17 shows the setup procedure schematically. For these scalar products, the wave functions need to be exchanged in explicit send-receive operations via the band communicator `band_comm`. A straightforward implementation without the exploitation of hermiticity would require p_{bnd}-1 communication cycles. Then, each band set achieves access to the $N_{bands} \times n_{bnd}$ matrix elements H_{nm} (S_{nm}) that are evaluated involving the locally stored $|\hat{\tilde{H}}\tilde{\Psi}_m\rangle$ ($|\hat{\tilde{O}}\tilde{\Psi}_m\rangle$). Since the generalized eigenvalue problem is solved in each band set separately access to the complete matrices is needed. Therefore, a reduction operation along the band communicator is performed as final step of the matrix setup procedure. The matrix elements are stored on the distributed BLACS grid which has a similar process distribution in each bands set. Therefore, the reduction operation only works on two $M \times M$ matrices where M is defined in the ScaLAPACK context as

$$M = \left\lceil \frac{\left\lceil \frac{N_{bands}}{BS} \right\rceil}{N_p} \right\rceil \cdot BS \tag{5.29}$$

with the BLACS grid of $N_p \times N_p$ processes. These two communication operations are introduced additionally in band parallelization of which the exchange of the wave functions takes considerably more time than the reduction operation.

Exploiting that the Hamiltonian and the overlap operator are hermitian (symmetric for real numbers) quantities, i.e. $H_{nm} = H^*_{mn}$ and $S_{nm} = S^*_{mn}$. This leads to a simple setup of $N_{bands}(N_{bands}+1)/2$ elements of the lower triangular matrix when all bands are stored locally. Savings of 44 % compared to the full setup of N^2_{bands} elements can be achieved for N_{bands}=3500 as shown in Table 5.5. This is a

good result considering that roughly 12 % of of the time for the full matrix setup is spent in the application of the operators.

	T_{full} [sec]	T_{sym} [sec]	Speedup
no band parallelization	250.88	141.40	1.774
16× band parallelization	23.86	13.39	1.782

Table 5.5.: Times for the subspace matrix setup exploiting the symmetry of symmetric/hermitian operators (sym) compared to the setup of all matrix elements (full). The system was $Ge_{125}Sb_{250}Te_{500}$ on a 16^3 domain decomposition, with and without band parallelization. In both cases roughly 44 % of the matrix setup time are saved.

In band parallelization $N_{bands} = n_{bnd} \times p_{bnd}$, we also have to evaluate matrix blocks of locally stored bands with remotely stored bands. The symmetry allows to compute all matrix blocks in $\lfloor \frac{p_{bnd}}{2} \rfloor$ cycles instead of p_{bnd}-1 cycles. Figure 5.18 shows this scenario for p_{bnd}=3 (odd) and p_{bnd}=4 (even). The odd case is advantageous because the number of off-diagonal matrix blocks that need to be evaluated in each process is $\frac{p_{bnd}(p_{bnd}-1)}{2}$ which is integer. We thus only need to perform $\lfloor \frac{p_{bnd}}{2} \rfloor = \frac{p_{bnd}-1}{2}$ cycles where each has process evaluates $\frac{p_{bnd}-1}{2}$ off-diagonal blocks. In the case of even p_{bnd}, $\frac{p_{bnd}-1}{2}$ is not integer such that only half of the processes need to perform a block evaluation in the last cycle. Nevertheless, also for a 16-fold band parallelization we observe 44 % savings in the matrix setup times, compare Table 5.5.

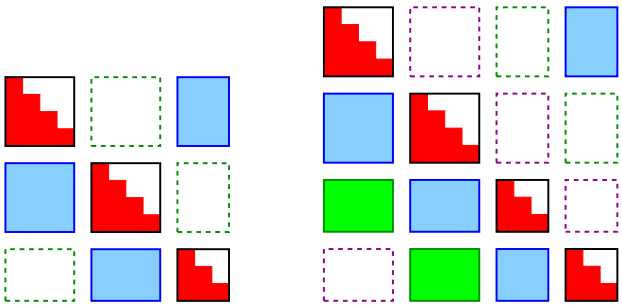


Figure 5.18.: Setup of subspace matrix elements of hermitian operators. Only the lower triangular matrix is evaluated (red). In band parallelization (3-fold on the left, 4-fold on the right) matrix elements between locally stored bands and remotely stored bands can be evaluated in $\lfloor p_{bnd}/2 \rfloor$ cycles. In the second cycle (green) of the 4-fold band parallelization only half of the processes need to evaluate their matrix blocks.

A workflow description of the ScaLAPACK interface can be found in the Appendix C.1.9.

5.4.2. Performance

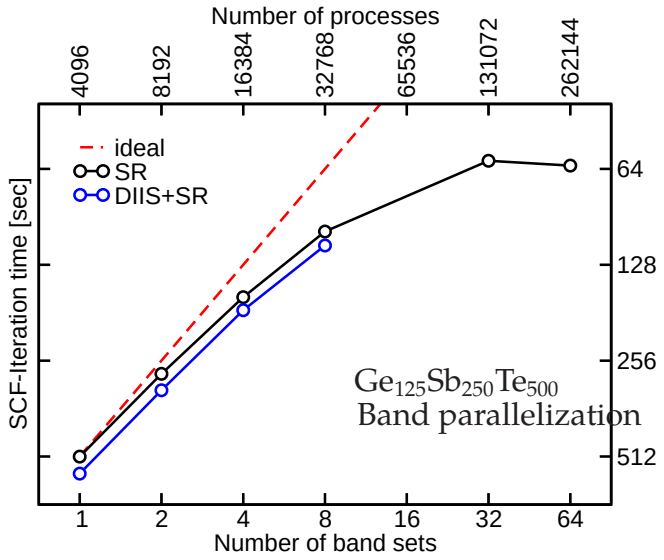


Figure 5.19.: Strong scaling of the SCF-iteration time in band parallelization. Increasing the number of processes in the band parallelization, we find that the execution time decreases for up to a 32-fold parallelization. The underlying real-space grid parallelization is a $16 \times 16 \times 16$ domain decomposition, i.e. 4096 `JUGENE` cores. The last data point (64) has been running on 2^{18} cores, 88.9% of all available nodes.

As outlined in Section 5.1.2 intensive all-to-all communication causes a communication time which grows proportionally to the number of communication partners, i.e. the number of parallel band sets in this case. Figure 5.19 shows the strong scaling behavior of `JUGENE` in band parallelization. We can easily see a maximum of the speedup at a band parallelization with 32 parallel sets, each with an internal domain decomposition on 16^3 `MPI` processes.

5.5. Size Scaling

For the calculation of large-scale problems, the it is important to know the scaling behavior of the code components with respect to the system size, i.e. with the number of atoms. In the previous sections, we have seen the strong scaling (hard scaling) behavior of the two parallelization levels which give information about the parallel efficiency in dependence on the degree of parallelization and the problem size. In order to estimate the total workload of a calculation, we want to find the dependence of the total workload on the system size. We therefore analyze the weak scaling, i.e. we increase the system size and the number of processors at the same time.

The subsequent section increases the real-space cell and decreases the **k**-space sampling simultaneously which leads to an effective scaling that is linear in the number of atoms. However, Section 5.5.2 does samples the Brillouin zone only with the Γ-point. Real-valued KS wave functions can be used but the full quadratic scaling of the method becomes visible.

5.5.1. Real-Space vs. k-Space

The total workload of a real-space DFT calculation scales as

$$N_{\text{bands}} N_{\text{grid}} N_{\text{kpoints}} \left(1 + \gamma N_{\text{bands}}\right) \tag{5.30}$$

where the number of bands N_{bands} depends in a linear fashion on the number of valence electrons in the system and, thus, on the number of atoms N_{atoms}. The total number of grid points N_{grid} is proportional to the systems volume which, as well, depends on the number of atoms in the system, assuming no vacancies. The number of **k**-points can be considered to scale roughly with the inverse of the number of atoms. Large systems in real-space require a less dense sampling of the Brillouin zone for a description providing the same accuracy. However, the number of **k**-points cannot be less than one. In this section, we assume $N_{\text{kpoints}} > 1$. Then, the overall scaling of the workload has two contributions, one which scales linear with the system size and one which is quadratic in the system size. The ratio of the two components is labelled γ here. The subspace rotation method, in principle, also has a contribution proportional to N_{bands}^2 from the solution of the generalized eigenvalue problem in the subspace. However, we neglect this contribution in this analysis since its prefactor is comparably small.

In order to extract the scaling behavior with respect to the number of bands we performed a set of calculations of carbon in the diamond crystal structure. The number of unit cells treated in real-space is doubled each time whereas the number of **k**-point sampling is reduced by the same factor. Thus, in all calculations the resulting density in each unit cell is the same and, making use of the real-space parallelization and the parallel treatment of **k**-points, the total number of `MPI` processes is the same, merely the factorization differs. We found the deviations in total energy per atom to be as small as 1.2 meV comparing the converged results for 64 atoms and 512 atoms. We measured the time to solution (see Table 5.6) and the time spent in the major ingredients of the calculation (see Figure 5.20). The scaling of the solution of the Poisson equation exhibits no size dependence as expected for a system of high symmetry and a constant number of grid points per domain. The execution times of the DIIS method, T_{DIIS}, are expected to grow $\sim N_{\text{bands}} N_{\text{grid}} N_{\text{kpoints}}$ neglecting communication. With 4 bands per atom and 8^3 grid points per atom and $N_{\text{kpoints}} = 2^{11}/N_{\text{atoms}}$ we expect the DIIS time to scale $\sim N_{\text{atoms}}$. In Figure 5.20 an

average power of 1.06 can be seen. The application of the subspace rotation consists of four parts: First, the action of the Hamiltonian and overlap operator need to be calculated for each band, i.e. $N_{bands}N_{grid}N_{kpoints} \sim N_{atoms}$. Second, N_{bands}^2 matrix elements have to be computed for each **k**-point which scales $\sim N_{atoms}^2$. The third contribution is the direct diagonalization of the subspace matrices that scales $N_{band}^3 N_{kpoints} \sim N_{atoms}^2$. The fourth contribution is the new linear combination of bands $\sim N_{bands}^2 N_{grid} N_{kpoints}$ thus also $\sim N_{atoms}^2$. However, in Figure 5.20 an average power of 1.76 for the subspace rotation can be extracted, indicating a large fraction of linear scaling components. A fit to a polynomial with linear and quadratic powers results in $T_{SR} \approx 19\,\text{msec}\,N_{atoms} + 0.72\,\text{msec}\,N_{atoms}^2$.

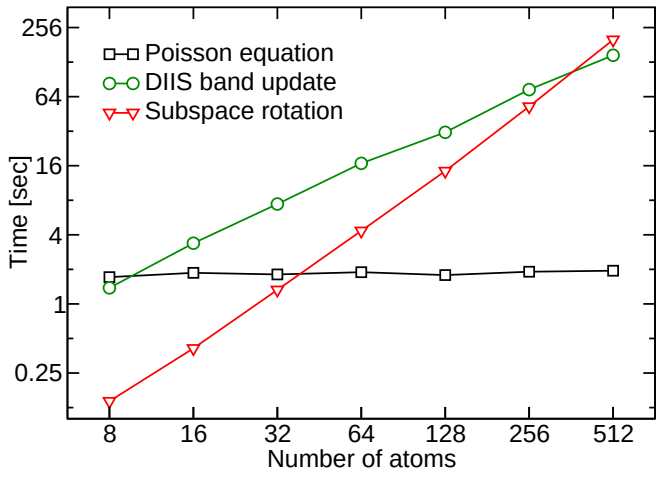


Figure 5.20.: Times for the major ingredients of a self consistency step: Solution of the Poisson equation, DIIS band update and subspace rotation method. This test system is bulk carbon in diamond crystal structure with 8 atoms in the cubic unit cell. 2048 `MPI` processes in virtual node mode have been running on `JUGENE` in all seven calculations. The domain size (grid points per `MPI` process) has been kept constant increasing the simulation volume in real-space from $1 \times 1 \times 1$ to $4 \times 4 \times 4$ cubic unit cells with 8 atoms each (see upper labels). Simultaneously the Brillouin zone sampling is decreased from $8 \times 8 \times 8$ to $2 \times 2 \times 2$. The difference in total energy per atom is 1.2 m*e*V comparing 64 atoms and 512 atoms.

The scaling in terms of memory $N_{bands}N_{grid}N_{kpoints}$ is linear in N_{atoms} in these calculations. A set of wave functions occupies 67.1 MByte per atom. However, typical large scale DFT calculations use merely Γ-point sampling of the Brillouin zone. Then, the total memory requirement of a set of wave functions is $\sim N_{bands}N_{grid}$ and thus proportional to N_{atoms}^2. Γ-point sampling has the advantage that real-valued KS wave functions can be used. Considering the scaling of the methods for the solution of the eigenvalues problem we need to face the quadratic behavior of T_{DIIS}

and both, quadratic and cubic contributions to T_{SR}. The at least quadratical scaling of the method with respect to system size strongly motivates the two levels of parallelization, i.e. real-space grid parallelization and band parallelization.

N_{atoms}	$m_x\ m_y\ m_z$	$k_x\ k_y\ k_z$	E_{tot} [eV]	E_{tot}/atom [eV]	Time [sec]
8	$1 \times 1 \times 1$	$8 \times 8 \times 8$	-131.035443	-16.37943	36.5
16	$2 \times 1 \times 1$	$4 \times 8 \times 8$			38.4
32	$2 \times 2 \times 1$	$4 \times 4 \times 8$			58.7
64	$2 \times 2 \times 2$	$4 \times 4 \times 4$	-1047.116956	-16.36120	85.3
128	$4 \times 2 \times 2$	$2 \times 4 \times 4$			153.0
256	$4 \times 4 \times 2$	$2 \times 2 \times 4$			364.8
512	$4 \times 4 \times 4$	$2 \times 2 \times 2$	-8377.590679	-16.36248	995.3

Table 5.6.: Bulk carbon in diamond structure for $m_x \times m_y \times m_z$ unit cells and a Brillouin zone sampling with $k_x \times k_y \times k_z$ **k**-points. Total energies are given only for the cubic case of 1^3, 2^3 and 4^3 unit cells. The other calculations have not been completed self-consistently. The SCF-iteration times have been measured on JUGENE running 2048 MPI processes.

In this section the weak scaling for a system with **k**-point sampling has been studied. As mentioned earlier, calculations in the large-scale regime usually sample only the Γ-point of the Brillouin zone, as shown in the following section.

5.5.2. Size Scaling of GeSbTe

As an example of large-scale calculations we have performed a scaling analysis on the Xeon cluster JUROPA [88] and the massively parallel supercomputer JUGENE [54] using different supercell sizes containing the phase-change material $Ge_1Sb_2Te_4$ (see Section 6). All simulated supercells are cubic and have an edge length of $m \cdot 6.04$ Å with an integer multiplier m. To enable a direct comparison of the systems of different edge lengths $m \cdot 6.04$ Å a constant grid spacing of 0.19 Å has been chosen such that the wave function grid contains $32m$ grid points in each direction. For a simple analysis, band parallelization has not been applied in any of these calculations. However, the parallelization of the real-space grid in domain decomposition has been scaled proportional to the system size. Two different series have been running on each machine: JUROPA with m^3 processes, JUROPA with $(2m)^3$ processes, JUGENE also with $(2m)^3$ processes and finally JUGENE also with $(4m)^3$ processes.

For simplicity we assume that the time for the diagonalization of the subspace eigenvalue problem in parallel is negligible compared to all other tasks. Without parallelization, this assumption is impossible due to the cubic scaling behavior of the serial diagonalization workload of a full matrix. With these settings, however,

enough parallel processes are available for diagonalizing the $(28\mathfrak{m}^3) \times (28\mathfrak{m}^3)$ matrix in a reasonably short time by using the ScaLAPACK library (see Section 5.3.9). The total workload of the DIIS method scales linear in the grid size and linear in the number of bands. Both factors are proportional to the system size and thus the total workload of the DIIS methods $\sim \mathfrak{m}^6$. With a parallelization $\sim \mathfrak{m}^3$, we find that the time behaves roughly proportional to $\sim \mathfrak{m}^3$. This can be seen in the upper right panel of Figure 5.21. The DIIS band update on JUGENE-4 (red solid line) takes 1 sec for $2 \times 2 \times 2$ unit cells of $Ge_1Sb_2Te_4$. The DIIS time for 8^3 unit cells is about 64 sec.

With the above assumption, the subspace rotation method scales linear in the grid size. However, the number of subspace matrix elements that need to be evaluated and the number of eigenvector coefficients scale quadratically with respect to the system size. The scaling behavior of the total workload of the subspace rotation is thus a cubic function of the system size. As above outlined for the DIIS method, the computing time is divided by as real-space parallelization $\sim \mathfrak{m}^3$. Hence, the time for the subspace rotation scales quadratically with the system size, i.e. with the sixth power of the edge length $\mathfrak{m}$. We can observe this power law in the lower left panel of Figure 5.21. The subspace rotation time on JUGENE 4 grows from 1.4 sec for $\mathfrak{m}$=2 to 4867 sec for the largest system. This corresponds to an effective power of 5.9 which indicates that there are also minor components in the subspace rotation which exhibit linear and quadratic scaling with the system size. However, if the compute time of each of the $64\mathfrak{m}^3$ processes scales $\sim \mathfrak{m}^{5.9}$ we can clearly see the cubic trend in the total workload and an overall workload of the problem that grows with the ninth power of the edge length.

All 4 panels of Figure 5.21 exhibit a deterioration of the parallel performance on JUROPA [88]. Although only a faint indication can be read of the JUROPA-1 data (blue dashed line) where the number of grid points per process was 32^3, the JUROPA-2 times (green dashed line) are clearly higher than the linear (quadratic) scaling behavior for the DIIS method (subspace rotation). Since only the grid parallelization is applied in this set of calculations, we can conclude that this an intrinsic signature of long communication times compared to computation. It can be directly assigned to the network speed. We see that only the short latencies and large bandwidth of the cartesian three-dimensional torus network are required to achieve a favorable parallel efficiency. However, the JUGENE torus network [54] enables this fast communication only if the task placement is matched to the torus shape. Therefore, JUGENE-4 data are shown only for the edge length multipliers $\mathfrak{m} \in \{2, 4, 8\}$.

5.5.3. Grid- vs. State- Parallelization

The additional communication that is introduced in band parallelization for the subspace matrix setup has been discussed in Section 5.4.1. Similar communica-

SCF-it.	JUROPA-1	JUROPA-2	JUGENE-2	JUGENE-4
$m = 2$	49.1	4.6	37.5	9.8
$m = 3$	191.1	25.8		
$m = 4$	801.4	230.1	531.1	104.4
$m = 5$		3305.6		
$m = 6$				
$m = 7$				
$m = 8$				5000.4
[sec]	blue dashed	green dashed	green solid	red solid

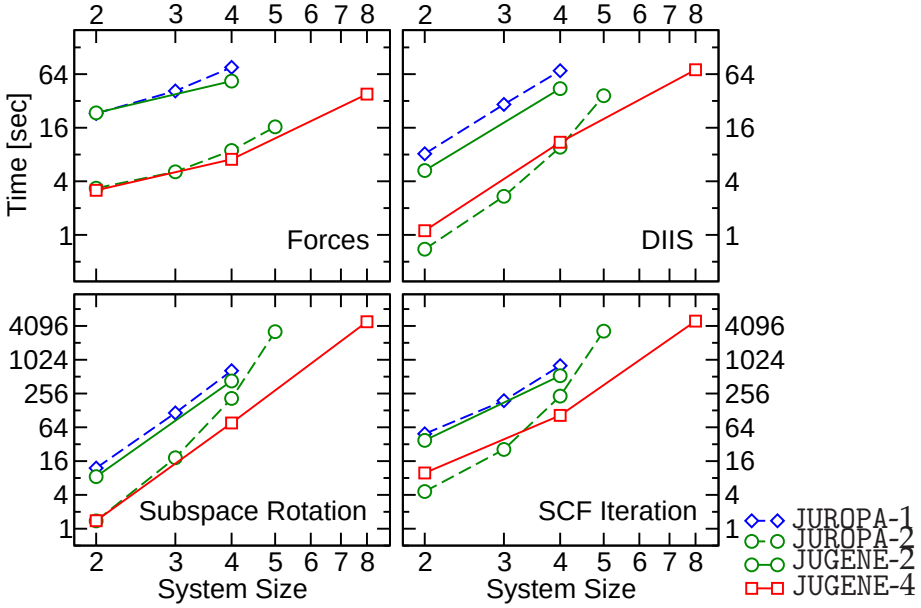


Figure 5.21.: Scaling of the computing time with the system size for the major components of a structural relaxation step. A system size of 2 stands for a distordered geometry of 2^3 unit cells of disordered $Ge_1Sb_2Te_4$. Consequently, system size 8 is a calculation with 3584 atoms (see Chapter 6.3). A relatively dense grid spacing of 0.19 Å was used in these calculations. The supercomputer systems JUROPA [88] has been used with 1^3 and 2^3 and JUGENE [54] with 2^3 and 4^3 MPI processes per unit cell in domain decomposition, respectively. The times for a self-consistency iteration (lower right panel) are listed in the above table.

tion operations are needed in the process of applying the rotations to the previous states to find the orthogonalized eigenstates of the Hamiltonian. These are not discussed in detail here but we can observe that this step takes often up to twice as long as the matrix setup since hermiticity cannot be exploited. Considering the amount of communication in each level we can maximize the parallel speedup by parallelizing the calculation by the most independent tasks, i.e. $\sigma\mathbf{k}$-points, if applicable. Then, continue with the real-space domain decomposition and finally apply the band parallelization if this last level is needed to fit the wave functions into memory or to get the result faster. However, we need to ensure that the grid parallelization stays in the range of reasonable parallel efficiencies. A limitation in `jüRS` is given by the requirement that the number of grid points per domains $n_i \geqslant n_i^h$ where n_i^h is the halo thickness. In this limit, $n_i = n_i^h$, one can imagine that the code will spend a large fraction of the compute time in communication since the entire local part of the wave functions needs to be transferred. Furthermore, the process of copying arrays into temporary buffers for `MPI_Sendrecv`-call is an overhead operation of the parallelization.

Given a total number of available `MPI` processes, N_p^{all}, we need to find the optimal distribution $(N_p^{\text{skpoints}}, N_p^{\text{bands}}, N_p^{\text{grid}})$ such that

$$N_p^{\text{all}} = N_p^{\text{skpoints}} \times N_p^{\text{bands}} \times N_p^{\text{grid}} \tag{5.31}$$

where we need to respect that N_p^{grid} needs to be factorizable according to Equation 5.14. As discussed in Section 5.1 and Section 5.2, the parallel efficiency of the $\sigma\mathbf{k}$-point parallelization is extremely efficient due to a low requirement for communication and is only reduced by a serial fraction that is cause by the generation of the effective potential. Therefore, the parallelization of $\sigma\mathbf{k}$-points always has the highest priority in the large-scale limit. However, large systems often do not sample the Brillouin zone with more than the Γ-point. Therefore, we can restrict our discussion to the distribution of N_p^{bands} and N_p^{grid}.

In order to find the most efficient distribution of the parallel processes we analyze the scaling of the major ingredients of a typical large-scale calculation. The wave function grid contains 192^3 grid points and we need to treat 14336 bands, Γ-point sampling, no spin polarization. The maximal number of available processes on `JUGENE` is $3^3 2^{15}$. The first setting factorizes N_p^{all} as 9×32^3 (Space), i.e. a $9\times$ band parallelization and the 192 grid points are distributed to 32 domains in each spatial direction. The number of finite difference neighbors n_f controls the halo-thickness n_i^h which is the default value of 4. We can already see that each domains contains 1593 wave functions with only 6^3 grid points each such that we come close to the limit case outlined above, $n_i \approx n_i^h$. The second setting assigns more parallel processes to the band parallelization (Bands), i.e. N_p^{all} is factorized as 72×16^3. In this scenario each band set owns 200 bands and the local grid size is eight times as large.

Table 5.7 summarizes these details for the first setting (Space) and the second setting (Bands) and shows resulting compute times of the most important steps of a self-consistency iteration. The grid parallelization numbers for the domain decomposition, 16 and 32 for (Space) and (Bands), respectively, are chosen to match the three-dimensional torus network of JUGENE. The hard wired cartesian nearest-neighbor connectors can only be linked at the surfaces of $8 \times 8 \times 8$-blocks, so called midplanes.

Name		Space	Bands
Band parallelization		$9\times$	$72\times$
Domain decomposition		$32 \times 32 \times 32$	$16 \times 16 \times 16$
Grid points/domain		$6 \times 6 \times 6$	$12 \times 12 \times 12$
$E_{tot} - E_0$	[eV]	- 0.08378510893	- 0.08378510616
DIIS	[sec]	29.9	4.5
Subspace setup	[sec]	437.6	151.8
Subspace solve	[sec]	168.7	170.5
Subspace lin.	[sec]	2433.1	854.6

Table 5.7.: Competition of the two parallelization levels. The total number of MPI processes, 294912 JUGENE cores in virtual node mode [54], is distributed as 9×32^3 (Space) and 72×16^3 (Bands) for the calculation of the electronic structure of $Ge_{512}Sb_{1024}Te_{2048}$. Numerical inaccuracies due to the different summation orders lead to deviations in the total energy as small as $3 \cdot 10^{-9}$ eV. The total energy offset E_0 is -35100 eV. A 64-bit representation for real numbers is used by default in the entire implementation.

The DIIS band update method works on each band independently. Therefore, we can interpret the ratio of 6.64 of the DIIS times as a clear indication that the grid parallelization with 32^3 domains brings only 15 % of the efficiency that is achieved with 16 domains per direction. The remaining times listed in Table 5.7 (subspace setup and subspace lin.) show a total efficiency drop to 35 % comparing (Space) to (Bands).

The solution of the generalized eigenvalue problem in the subspace shows no difference. In order to keep ScaLAPACK at its highest speedup, the block size has been customized to 224 and the BLACS grid extends over 64×64 processes. In the case of (Bands), the grid parallelization of 16^3 is exactly reshaped into the 64^2 BLACS grid. In the case of (Space), a 4096-subset of 32^3 grid processes takes care of the diagonalization of the subspace Hamiltonian.

The results have shown that the grid parallelization has an efficiency limit. When the local grid size is as small as 216 grid points, the communication operations with the nearest cartesian neighbors spend a large fraction of their time in operational

overhead. In order to as run at the parallelization limit $n_i = n_i^h$ the parallel efficiency can only be increased by avoiding the copying process to the communication buffer arrays and collecting the communication for a group of wave functions which would result in substantial changes in the code structure.

5.6. Summary

Two levels of distributed memory parallelization with MPI are introduced here. The first is a domain decomposition of the real-space grid. The exploitation of data locality and the truncation of the limited range of the finite-difference operator for the kinetic energy and Laplacian in the Poisson equation leads to a restricted communication pattern where only very little explicit communication is needed with the direct Cartesian neighbors. This leads to the excellent parallel efficiency of the domain decomposition, which is an intrinsic property of the real-space grid approach.

The second level of MPI parallelization introduced in this chapter is parallelization over KS states, which is trivial except for the parallelization of bands, a strong communication task. We have showed that even here we can obtain a substantial parallel speedup and, simultaneously avoid the memory-per-node boundary, a prerequisite for large-scale calculations due to the quadratic memory scaling of this real-space DFT algorithm.

5.6.1. Outlook

For a further improvement in the performance of the domain decomposition, a collection of nearest-neighbor communication could introduce an even higher efficiency when the action of the Hamiltonian is required for a set of KS wave functions. This is the case for the setup of the subspace matrices. However, the DIIS method treats the bands sequentially – up to now. Similar to the collection of communications in Chapter 5.3.8, the communication overhead times compared to transmission times would then fade and the large maximal bandwidth is exploited better. Furthermore, a mapping of the atomic projection onto a `BLAS3`-optimized matrix-matrix multiplication then becomes possible which will certainly increase the overall performance and the load balance for heterogeneous geometries.

An emulation of a virtual global memory has been realized, for example by one-sided communication [89]. Loading from memory is formulated into a request and the explicit sending of the remote compute node is hidden from the user. Although this apparently brings about some programming convenience, well-balanced synchronized two-sided communication should outperform this approach by far.

Another trend is hardware accelerators such as GPUs that perform outstandingly with large numbers of independent tasks yielding an enormous speed gain. It has been shown that finite-difference operations can be performed efficiently on GPUs [90, 91].

For the results of this work, `JUGENE` was mostly used in the virtual node mode (`-mode VN` [54]) with an `MPI` process on each of the four CPUs in each physical com-

pute node. Upcoming supercomputer architectures tend towards a larger number of compute cores per node. This motivates a hybrid parallelization, e.g. with OpenMP [55] or pthreads. In particular, the overlapping of computation and communication is promising in order to increase the parallel performance of a real-space grid code such as jüRS. A computation of the grid tasks, i.e. the finite-difference approximated kinetic energy, can easily be accelerated with threaded parallelism. However, an OpenMP parallelization of the atomic projection operation routine introduces more overhead than acceleration for the application of a single set of projectors onto a single KS wave function. Therefore, OpenMP should be applied in order to distribute the projections of different atomic spheres onto the threads. In terms of thread-load balancing, this results in a well-studied packing problem since the overlaps of the projection spheres with the rectangular domains leads to a variety of total operation counts of the projection routine.

CHAPTER 6

APPLICATION TO GESBTE

6.1. Introduction

Compounds of germanium, antimony and tellurium have been found to fulfill the prerequisites for a phase-change material (PCM) used for data storage [60]. PCMs consisting of ternary alloys of Ge:Sb:Te can now be found in media applications such as rewritable DVDs, Blueray^{TM}discs and non-volatile random access memory. The ability to store data is due to two (meta-) stable phases of these materials corresponding to binary bit states. One is the amorphous state and the other is the crystalline phase. Due to different optical properties, i.e. transmission and reflectivity, these states can be identified by a laser of low intensity. Information can be written to a thin film of PCM by melting the structure locally with a short and intense laser pulse. The material solidifies in the amorphous phase. The process can be reversed by applying a long laser pulse of lower intensity. The system is kept below the melting temperature and crystallizes again. The speed of this crystallization process is remarkable and allows data reading and writing at speeds of several Mbyte/sec.

Although alloys of Ge:Sb:Te have been used in industrial applications for many years, recent investigations aim to understand the functionality of these materials on a microscopic level. In this context Hegedüs *et al.* revealed that the fast recrystallization in $Ge_2Sb_2Te_5$ and $Ge_1Sb_2Te_4$ is assisted by a high vacancy concentration [92]. Further understanding of the fast formation dynamics was provided by Matsunaga *et al.*, who found stable rings of bonds which persist the melting and, hence, accelerate recrystallization during the annealing process [93]. Recent investigations have even indentified these PCMs as possible candidates for resistive data storage. Especially the combination of both resistive reading or writing

and optical access in the same material moved into the fields of interest [57].

In this work, we focus on the effects of structural relaxations onto the electronic structure of $Ge_1Sb_2Te_4$ in the rocksalt phase [60, 94, 95, 58].

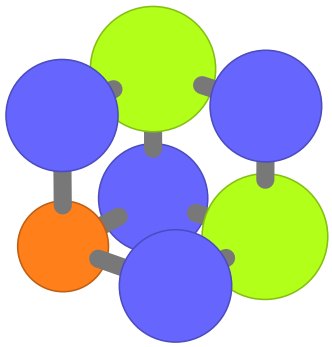


Figure 6.1.: Simplest unit cell of $Ge_1Sb_2Te_4$. In the rocksalt phase, the first FCC sublattice is completely occupied with Te atoms. The second sublattice is half occupied by Sb, a quarter Ge and a quarter vacancies.

Radius $[\frac{1}{2}a]$	1	$\sqrt{2}$	$\sqrt{3}$	2	$\sqrt{5}$	$\sqrt{6}$	$\sqrt{8}$	3	$\sqrt{10}$
Te—Te		24		12		48	24		48
Sb—Te	12		16		48			60	
Ge—Te	6		8		24			30	
Sb—Sb		6		3		12	6		12
Ge—Sb		6		3		12	6		12
Ge—Ge		$\frac{3}{2}$		$\frac{3}{4}$		3	$\frac{3}{2}$		3
v —Ge		3		$\frac{3}{2}$		6	3		6
v —Te	6		8		24			30	
v —Sb		6		3		12	6		12
v —v		$\frac{3}{2}$		$\frac{3}{4}$		3	$\frac{3}{2}$		3
$\sum$	24	48	32	24	96	96	48	120	96

Table 6.1.: Theoretical occurrence of interatomic distances in disordered $Ge_1Sb_2Te_4$ normalized to one unit cell. The lattice symmetry does not allow distances at $\frac{a}{2}\sqrt{7}$. The lowest line ($\sum$) indicates the number of atoms in an FCC shell of the corresponding radius. The occurrences are proportional to the product of the concentrations, c. On the second sublattice c_{Ge}, c_{Sb} and c_v are $\frac{1}{4}$, $\frac{1}{2}$ and $\frac{1}{4}$, respectively.

6.1.1. Electronic structure

Crystalline $Ge_1Sb_2Te_4$ (GST) in the rocksalt structure consists of two FCC sublattices which are shifted by half a lattice constant a in the (100)-direction. The first

sublattice is completely filled with Tellurium atoms (Te). The second sublattice is randomly occupied with Antimony (Sb), Germanium (Ge) and vacancies (v). The stoichiometry of $Ge_1Sb_2Te_4$ defines that the ratio of Sb and Ge is 2:1 and the number of Ge atoms equals the number of vacancies. Figure 6.1 shows a typical unit cell of $Ge_1Sb_2Te_4$, however, we deal with a disordered alloy so the randomness of occupations of the second sublattice may cause that two Ge atoms or two vacancies can be found as close as $a/\sqrt{2}$ as shown in Table 6.1. The first column describes the occurrence of nearest-neighbor bonds of bond length $\frac{a}{2}$, $k = 1$, $k \in \mathbb{N}$. These bonds can only occur between the different FCC sublattices. This holds for all bonds with odd k, except for some special cases as $k = 7$ which is impossible because 7 cannot be expressed as $m_x^2 + m_y^2 + m_z^2$ with all m_i integer. Even values of k represent atom pairs on the same sublattice. Of particular interest in the following analysis of the data, we consider the near-field coordination, i.e. nearest neighbors (nn) and next-nearest neighbors (nnn), corresponding to $k = 1$ and $k = 2$, respectively. The values in Table 6.1 are normalized to one unit cell of $Ge_1Sb_2Te_4$ such that each of the two Sb atoms feels the surrounding six Te atoms as nn which leads to an occurrence of 12 for Sb-Te at the bond distance, $\frac{a}{2}$. The bottom line in the table shows the sum of all occurrences which corresponds to the number of sites in an SC crystal structure with lattice constant $\frac{a}{2}$ at the given distances.

In the disordered alloy, there is a non-vanishing probability of large regions without any Ge atoms or without vacancies. The distance occurrence is proportional to the product of two concentrations. The concentration of Te on the first sublattice is unity, $c_{Ge}^{(1.)} = 1$, and vanishes on the second sublattice, $c_{Ge}^{(2.)} = 0$. For the other three components, Ge, Sb and vacancies (v), all concentrations on the first sublattice vanish. The stoichiometry of $Ge_1Sb_2Te_4$ defines that $c_{Ge}^{(2.)} = \frac{1}{4}$, $c_{Sb}^{(2.)} = \frac{1}{2}$ and $c_v^{(2.)} = \frac{1}{4}$. Now, the occurrence of a combination of two components, e.g. Sb-v, is proportional to the product of their concentrations, $c_{Sb}^{(2.)} \cdot c_v^{(2.)} = \frac{1}{8}$ and the coordination number of the FCC shell.

All calculations of the electronic structure of $Ge_1Sb_2Te_4$ have been performed using `jüRS`, the implementation of real-space DFT discussed in this work. The local density approximation [96] has been applied and a lattice constant of 6.04 Å has been chosen [97]. The PAW configuration strings

```
Ge  4s* 2 4p* 2 4d | 1.8
Sb  5s* 2 5p* 3 5d | 2.5
Te  5s* 2 5p* 4 5d | 2.2
```

indicate that Ge is treated with a frozen core density [Ar]3d, the cores of Sb and Te are configured [Kr]4d. *s* and p states of all three species are treated in the valence band with two projectors, each. In the atomic reference configuration, the *s* valence states have been doubly occupied whereas the p valence states hosted 2,3

and 4 electrons for Ge, Sb and Te, respectively. Table 6.2 shows the energies of the spherical all-electron calculation that precedes the augmentation procedure, compare Section A.0.1 and A.0.2. Core states are indicated by (c) and valence states with their occupation number. An additional d projector is added for each of the three species. The corresponding d state is unbound and hence unoccupied. The augmentation radii (given in atomic units above) are 0.953, 1.323 and 1.164 Å for Ge, Sb and Te, respectively, and are chosen independent of the angular momentum character, ℓ.

[eV]	Ge		Sb		Te	
1s	-10940.723	c	-30269.596	c	-31592.174	c
2s	-1369.390	c	-4607.071	c	-4845.614	c
2p	-1198.005	c	-4138.436	c	-4352.371	c
3s	-167.721	c	-902.692	c	-963.773	c
3p	-115.370	c	-749.917	c	-803.425	c
3d	-29.548	c	-516.090	c	-560.267	c
4s	-11.948	2	-148.479	c	-163.979	c
4p	-4.052	2	-101.591	c	-114.183	c
4d			-33.546	c	-41.689	c
5s			-13.087	2	-15.359	2
5p			-4.998	3	-6.116	4

Table 6.2.: Atomic eigenstates for the reference configuration of Ge, Sb and Te. c indicates a fully occupied frozen core shell. Numbers stand for the occupation of a valence state.

A grid spacing of 0.25 Å has been used in all calculations shown in this Chapter and the projectors where represented on the coarse grid using the reversed interpolation method with N=2 and M=5 (compare Section 4.3.3).

Atomic density of states (DOS) plots discussed in this chapter are created from the sum over s, p and d of the atomic ℓ-resolved DOS, where the d contributions are negligible in all calculations. The atomic ℓ-resolved DOS results from a weighting of each KS state with the weight $w^{a\ell}_{n\sigma\mathbf{k}}$ of its PAW projection coefficient, i.e. $w^{a\ell}_{n\sigma\mathbf{k}} = \sum_i |\langle\tilde{\Psi}_{n\sigma\mathbf{k}}|\tilde{p}^a_i\rangle|^2$ where the sum over projectors i has been restricted to the lowest partial waves with the proper ℓ-character. All DOS-plots show the Fermi level at the origin.

6.2. Classification of Te in $Ge_{125}Sb_{250}Te_{500}$

For a meaningful analysis of the electronic structure of a disordered alloy, we need to ensure that finite size effects do not play a role. In particular, a too small unit cell

in real-space leads to the spurious interactions of the atoms with their periodic images. In nature, the quasi-crystal structure does not possess this symmetry. Therefore, various numbers of $Ge_1Sb_2Te_4$ unit cells have been investigated by Thieß[97]. We consider a $Ge_{125}Sb_{250}Te_{500}$ consisting of $5 \times 5 \times 5$ unit cells as large enough. The configuration for the occupation of the second sublattice has been adopted from the samples calculated by Thießfor a better comparison. Figure 6.2 shows the calculated structure schematically. The atomic radii have been chosen according to the PAW augmentation radii of each element.

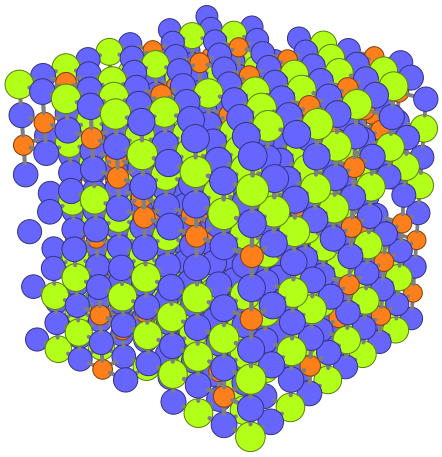


Figure 6.2.: Visualization of the random alloy $Ge_{125}Sb_{250}Te_{500}$. The first FCC sublattice consists completely of Te atoms (blue). The second sublattice is randomly occupied with Ge:Sb:vacancies in the stoichiometry 1:2:1. Ge and Sb are colored orange and green, respectively.

A determination of the electronic structure with `jüRS` yields the systems DOS and atomically resolved DOS, if requested. Figure 6.3 shows the atomic ℓ-resolved DOS averaged over the three different species for $Ge_{125}Sb_{250}Te_{500}$ in the geometry of the ideal FCC lattice positions, i.e. all atomic coordinates are integer multiples of the bond distance, 3.02 Å. A clear cut between s states and p states can be found at -6 *e*V. For the s states we can observe a pronounced peak for each of the three species. The Te-5s peak has a shoulder towards the positions of Ge-4s and Sb-5s and, vice versa, Ge and Sb also exhibit a shoulder at the position of Te-5s. All shoulders below -6 *e*V have as minor contribution of p-character which is indicated by the dashed lines in the corresponding colors. This indicates the hybridization of the s valence states. Comparing the s peak positions to the energy eigenvalues of the atomic reference configuration given in Table 6.2 we find that the Sb-5s peak has its maximum 0.4 *e*V lower than expected, neglecting the global shift of the energy origin.

GST exhibits a relatively clear separation between occupied and unoccupied states. Nevertheless, a real band gap is not seen. The DOS shows a dip around the Fermi

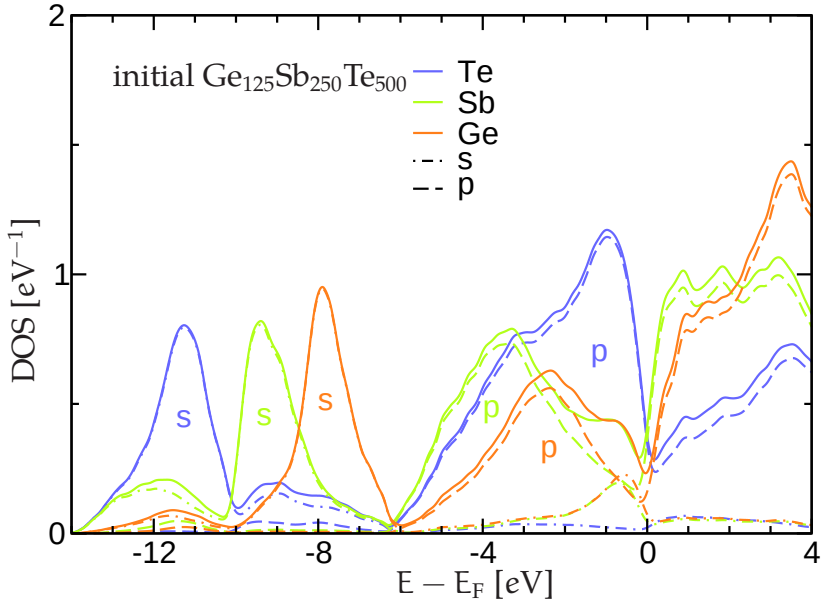


Figure 6.3.: Average Density of States projected onto the atomic Ge-4s, Ge-4p, Sb-5s, Sb-5p, Te-5s and Te-5p states. The color coding is orange, green and blue for Ge, Sb and Te, respectively. Solid lines show the sum of both, s- and p-like contributions which are illustrated with dash-dotted and dashed lines, respectively.

level on all three species at slightly different positions; Sb 0.15 eV below, Te 0.15 eV above and Ge at the Fermi energy. The directional bonds in GST can be attributed to occupied p states which are found between -6 eV and the Fermi level in Figure 6.3. On the rising side of the p-peaks (-6 to -3 eV) the DOS shows a similar behavior for the three species. However, the range between -2 eV and the Fermi level carries some extraordinary features. The Te-5p states have an extra sharp peak centered at -1 eV where Sb and Ge exhibit an s-like contribution (green and orange dash-dotted lines, almost on top of each other) in the range $[-1.5, 0.0]$ eV. GST is known to exhibit resonant p-bonding, i.e. covalent bonds which are occupied with less than two electrons [60]. We investigate the KS states closely below the Fermi level in more detail.

Figure 6.4 shows the DOS resolved for all atoms of $Ge_{125}Sb_{250}Te_{500}$ around the Fermi level. The pseudo-gap becomes visible formed by the (mostly) unoccupied Sb states starting at the Fermi level and and the Te states reaching to E_F. The plots also makes clear that the sharp contributions of Te at the Fermi level can be assigned to a few atoms only. Thieß[97] has shown that these strongly localized p states of Tellurium occur in dependence of the nn vacancy coordination, n_{Vac}, introducing seven classes which are identified by n_{Vac} from 0 to 6.

We have refined the classification of Te atoms according to the nn vacancy co-

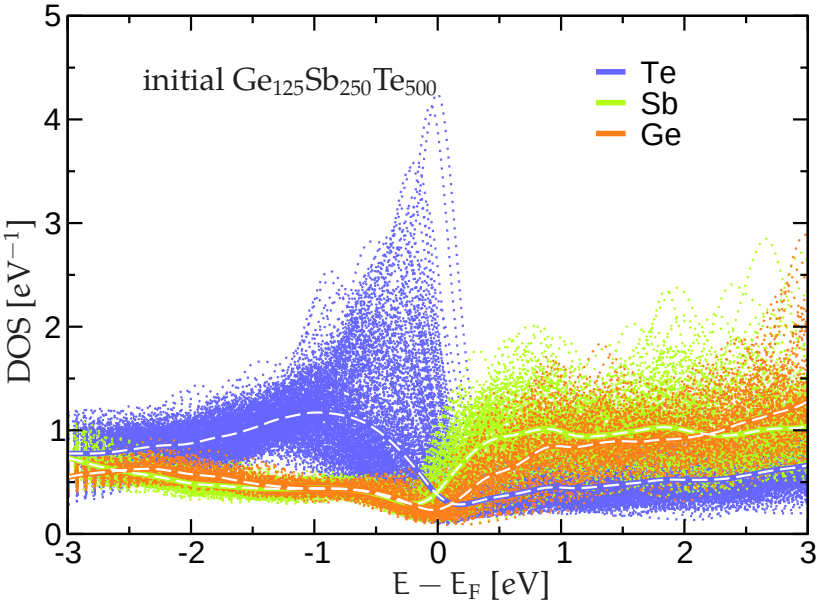


Figure 6.4.: Density of States around the Fermi energy of the initial structure of $Ge_{125}Sb_{250}Te_{500}$. The dotted lines depict the DOS projected onto each atom where Ge, Sb and Te are colored in orange, green and blue, respectively. The average DOS of each species is indicated by dashed white lines. We can see clearly that some very localized Te states lie at the Fermi level forming a pseudo-gap with the unoccupied Sb and Ge states.

ordination into ten classes {0,1,2,3,2′,3′,4,4′,5,6} of which the classes 5 and 6 do not occur in the random configuration of $Ge_{125}Sb_{250}Te_{500}$ which has been investigated here. The class names stand for the number of vacancies, n_{Vac}, that are located at $\frac{a}{2}$ from a Te atom. The further distinction is done by introducing the classes 2′,3′ and 4′. For example, Te atoms in the classes 2 and 2′ posses 4 bondings towards Ge or Sb atoms but the local coordination differs by the bond angles. Table 6.3 defines the ten classes in detail and shows the predicted occurrence of classes in an infinitely large quasi-crystal. For a random distribution of one quarter vacancies on the second sublattice in an infinitely large sample of GST, we can predict the probability of occurrence of Te atoms in the ten classes. All six nn of each Te atom have a probability of being a vacancy of p_{Vac} and a probability p_X of being either Sb or Ge. We thus have a binomial distribution for the probability p the number of vacancies, n_{Vac}, defined by

$$1 = (p_{Vac} + p_X)^6 = \sum_{n_{Vac}=0}^{6} p(n_{Vac}). \tag{6.1}$$

With $p_{Vac} = \frac{1}{4}$ and $p_X = \frac{3}{4}$ we find that

$$p(n_{Vac}) = \frac{1}{4^6}\binom{6}{n_{Vac}} 3^{6-n_{Vac}} \tag{6.2}$$

finding the probabilities 17.8, 35.6, 29.7, 13.2, 3.3, 0.4 and 0.024 % for $n_{Vac} = 0$ through 6, respectively. The distinction by bond angles leading to the classes 2′, 3′ and 4′ introduced before has not been made yet. Hence, e.g. $p(3) = p'(3) + p'(3')$ with $p'(\text{class})$ being the occurrence probability as a function of the class. The classes 3 and 3′ share their total probability of 13.2 % in a ratio of 8:12 because of the 20 possible configurations with 3 bond in 6 directions we can find eight configurations whith three bond angles of 90° (class 3). In the other 12 configurations all bonds lie in a plane. Then, two bond angles are 90° and one is 180° (class 3′). Hence, the probability $p'(3) = 5.3\,\%$ and $p'(3') = 7.9\,\%$. For $n_{Vac} = 2$ and 4, we can exploit the symmetry $v \leftrightarrow X$, exchanging bonding and vacancy nn sites. Let us consider $n_{Vac} = 4$. In three of 15 configurations, the bonds are at a bond angle of 180°, therefore, the probability $p(4)$ is divided 12:3 giving $p'(4) = 2.6\,\%$ and $p'(4') = 0.66\,\%$. Using the aforementioned symmetry, we split up $p(2)$ in the same ratio leading to $p'(2) = 23.7\,\%$ and $p'(2') = 5.9\,\%$ with 2′ being the class where the two vacancies are on opposite sidesof the Te atom. All probabilities p' can be found in Table 6.3. The listed order of the classes has been chosen such that the DOS on the Te atoms is a monotoneously growing function of the class in Figure 6.5.

Figure 6.5 illustrates the atomic DOS on the Te atoms in $Ge_{125}Sb_{250}Te_{500}$ averaged within the ten classes defined above. The highest and sharpest DOS values shows

Class	n_{Vac}	Bonds	90°	180°	555	888	p′ [%]
0	0	6	12	3	84	364	17.798
1	1	5	8	2	180	723	35.596
2	2	4	5	1	120	478	23.730
3	3	3	3		21	117	5.273
2′	2	4	4	2	37	134	5.933
3′	3	3	2	1	39	166	7.910
4	4	2	1		15	45	2.637
4′	4	2		1	4	10	0.659
5	5	1				10	0.439
6	6	0				1	0.024

Table 6.3.: Classification of Te atoms in distordered $Ge_1Sb_2Te_4$ according to the number of vacancies on nearest-neighbor sites, n_{Vac}, and the bond angles. The last columns show the occurrence in $Ge_{125}Sb_{250}Te_{500}$ (555), $Ge_{512}Sb_{1024}Te_{2048}$ (888) and the theoretical probability of the occurrence for a random distributions of one quarter vacancies in an infinite quasi-crystal. The expectation of class 6 occurring in (888) is 49 %. Fortunately, the setup investigated here contains exactly one.

class 4′. Here, four vacancies surround the Te atom in a plane, i.e. the two Te-X bonds form an angle of 180° where X may be Ge or Sb. In class 4 the Te atom has two covalent bonds at an angle of 90° and is, therefore, close to an sp^3-hybridization which occurs in tetrahedral coordinations (109.5 %). In contrast thereto, the bonds in class 4′ are formed out of linear sp-hybridized orbitals such that the two non-bonding p orbitals are close to degenerate in the plane perpendicular to the bonds.

It can be observed that the classified DOS shows several trends. Firstly, the width of the peak is smaller the higher the class. Secondly, the height of each maximum grows with the class rank. A third observation is a shifting of the position of the maxima and, simultaneously, of the dip at $E \geqslant E_F$ of about +0.1 *e*V per class to higher energies. This shift is even more pronounced in the position of the Te-5s states (not shown) where we find about +0.2 *e*V per class. The pronounced shifts indicate that the local potential in the surrounding of those Te atoms is elevated compared to the potential average. This effect can be assigned to a depletion of charge on these Te atoms which is a direct consequence of their p states being lifted above the Fermi level. The lower shift in those Te-5p states compared to the 5s states on the same atom can be regarded as screening behavior which only occurs in metallic systems. A high accumulation of vacancies on the second sublattice obviously causes a local metallicity. The small broadening of the Te-5p states of class 4 and 4′ visible in the DOS (Figure 6.5) indicates a strong localization in real-space. This agrees with the model of localized states in a pseudo-gap that have been found for quasi-crystals [98].

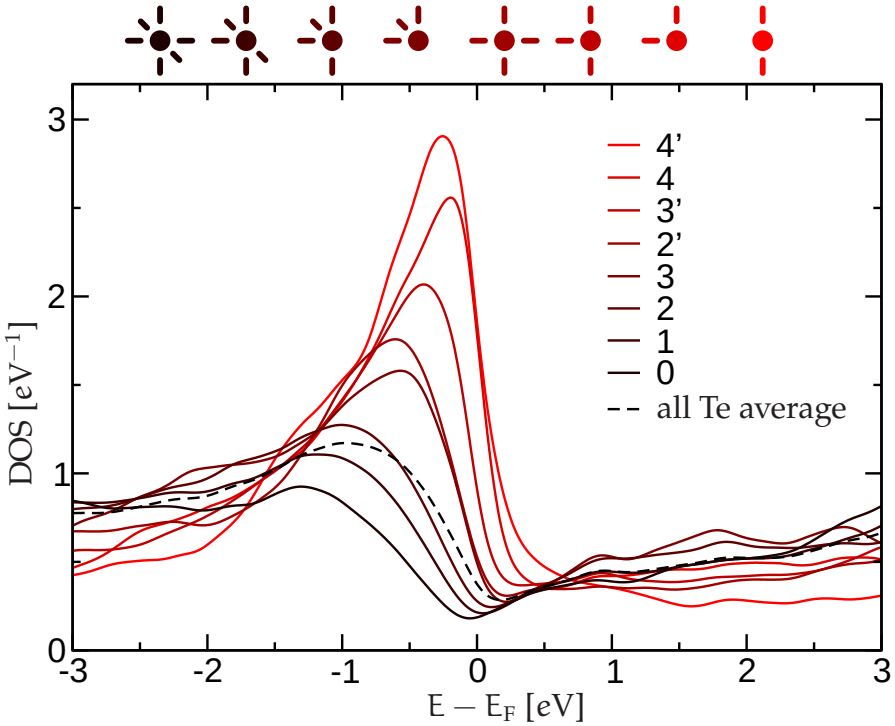


Figure 6.5.: Density of States on Te atoms averaged within eight classes. The class of the Te atom depends on the near field vacancy coordination and bond angles. Only four of 500 Te atoms are found in class number 4′ where four vacancies surround the Te atom and the bond angle is 180°.

Although the positions of the vacancies are randomly distributed, we need to take into consideration that a localization center cannot consist of a single Te atom in class 4 (or higher) alone. The presence of four vacancies around one Te atom necessarily causes that these four vacancies are at a distance of $\frac{a}{2}\sqrt{2}$ from each other. This is accompanied with at least four neighboring Te atoms which are at least in class 2 (two bonds missing at an angle of 90°). In other words, the probability of finding a Te atom in class c_1 on the nnn site from a Te atoms in class c_0 is not simply given by the values for $p'(c_1)$ in Table 6.3 but there is a positive correlation between $p(n_{Vac})$ for nnn sites. The correlation coefficient for n_{Vac} on two nnn Te sites is $\frac{1}{3}$ since these two Te atoms have two of six random variables in common. We hereby see that vacancy clusters exert an potential elevating effect onto a small surrounding region of Te atoms leading to localized states close to the Fermi level.

6.2.1. Relaxed Structure $Ge_{125}Sb_{250}Te_{500}$

Structural relaxation of $Ge_{125}Sb_{250}Te_{500}$ leads to a slightly distorted rocksalt structure and, of course, larger relaxation of the local structure in the vicinity of large concentrations of vacancies. A simple, but interesting number is the total number of atomic bonds. Two atoms are considered bonding if their interatomic distance does not exceed the bond length of 3.02 Å plus a tolerance of 25 %. Also, the bond length should not undergo −20 % from the equilibrium bond length since atoms would come too close. This, however, is not the case in the relaxed coordinates. Due to the relaxations, the total number of bonds was reduced by 2.5 %. This is counter-intuitive considering that the high concentration of vacancies in the system produces as large number of dangling bonds and one would expect the relaxation to finish at a geometry where more bands are formed, i.e. an increased bond count. Figure 6.6 shows the structure

The changes in geometry are accompanied with changes in the electronic structure. Figure 6.7 shows the atom-species averaged DOS of $Ge_{125}Sb_{250}Te_{500}$ before and after structural relaxation. Neglecting the larger smoothness of the DOS curves of the initial geometry (dashed lines) we can hardly see strong changes in the lower range of the valence energies except for a relaxation of the Te-5s states.

In the electronic structure calculated from the initial coordinates, the *s*-states of Tellurium were located at -11.48 ± 0.76 *e*V. After relaxation, we can find the peak position at 0.13 *e*V lower, at -11.61 ± 0.57 *e*V, and a little sharper. These data have been found by integrating the Te-DOS only up to −10 *e*V, i.e. the shoulder from hybridization with the *s* valence states of the other two species has not been included evaluating mean and rms values. A clear development due to the relaxation is the clear separation of *s*-states and p-states. In fact, the DOS of the relaxed structure does not show any states between −6.3 and 6.0 *e*V. Similarly, the pseudo-gap at the

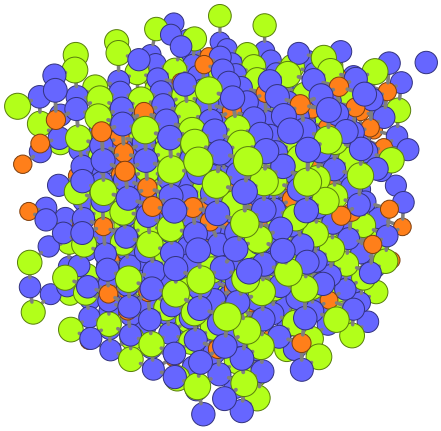


Figure 6.6.: Visualization of the relaxed geometry of $Ge_{125}Sb_{250}Te_{500}$. Ge, Sb and Te are colored orange, green and blue, respectively. Compared to the initial geometry (Figure 6.2), this structure looks almost amorphous. However, the upper right corner clearly exhibits (111)-planes of Te atoms.

Fermi level has lowered its value and the dips of the three species are aligned about 60 meV below E_F. The special features discussed for the electronic structure of the initial geometry are still present, i.e. a high density of Te-5p state between $-1.5\,e$V and the Fermi level and s-like contributions from Ge and Sb to these states.

However, after relaxation of the structure $Ge_{125}Sb_{250}Te_{500}$, a classification of Te atoms according to the number of nn vacancies and bond angles is not as simple as before. In analogy to Figure 6.4 for the initial structure, Figure 6.8 shows an atom-resolved DOS around the Fermi level. We can observe that the very localized Te states at have been reduced in height. In general, Te states are hardly see directly above E_F. In addition, the Ge states are completely unoccupied here leading to a reduced total DOS value at the Fermi level which is smaller by a factor of 2.3 than that of the initial geometry. The number of states in the vicinity of E_F is even reduced by a factor of 4. Here, the vicinity is defined by the derivative of the Fermi-Dirac function with a temperature T_{FD} of 1000 K. However, we can still see some behavior of local metallicity. Therefore, we evaluate the local near field coordination of the Te atoms. As mentioned earlier, the number of nn vacancies n_{Vac} needs to be redefined. We therefore approach via the bonds and their lengths. For the relaxed structure of $Ge_{125}Sb_{250}Te_{500}$, we define n_{Vac} for the Te atoms by

$$n_{Vac} = 6 - \sum_i \frac{3.02\,\text{Å}}{b_i} \tag{6.3}$$

where b_i are the bond length of nn bonds, i.e. Te-Ge or Te-Sb. With this definition, n_{Vac} is obviously not integer any more. Figure 6.9 shows the pDOS at the Fermi level on the Te atoms as a function of n_{Vac} comparing the initial and the relaxed

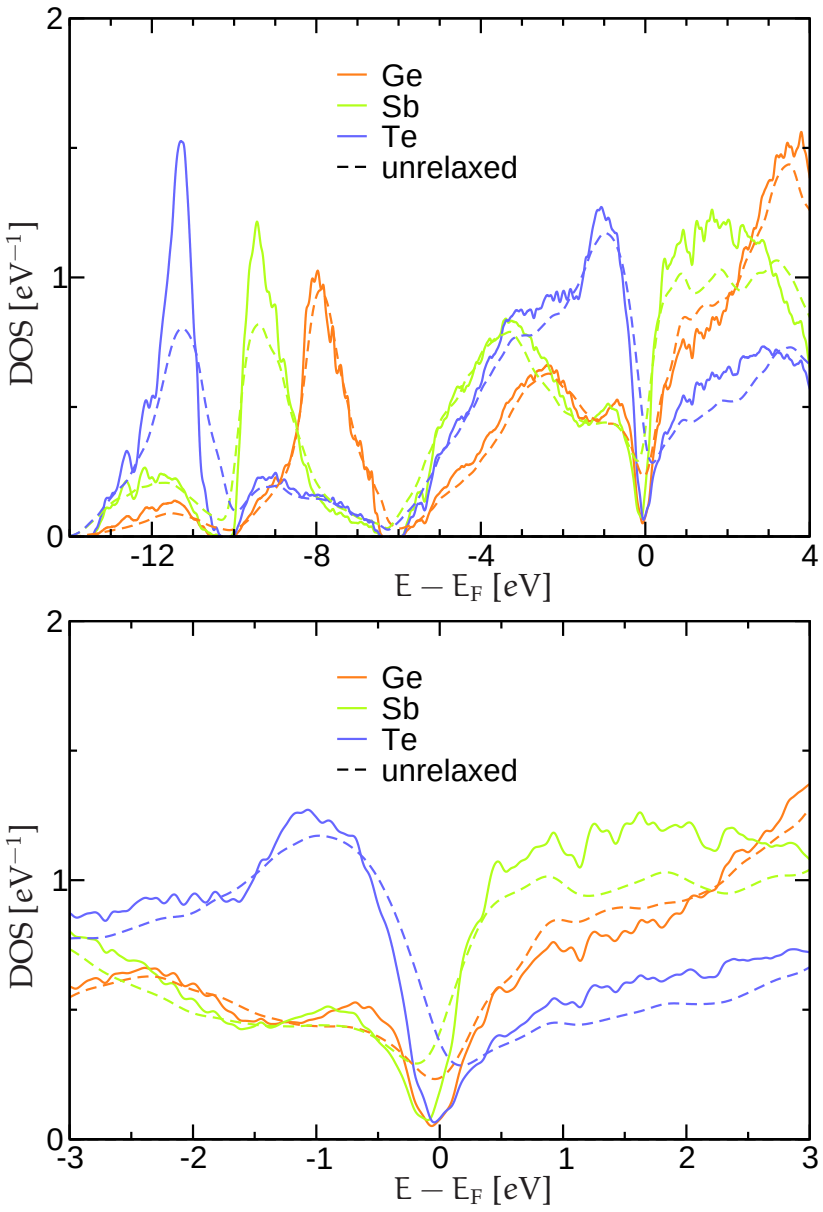


Figure 6.7.: Average Density of States of $Ge_{125}Sb_{250}Te_{500}$ comparing the initial (dashed lines) and the relaxed geometry (solid lines) for Ge, Sb and Te in orange, green and blue, respectively. The lower panel illustrates the energy range around the Fermi level exhibiting the pseudogap.

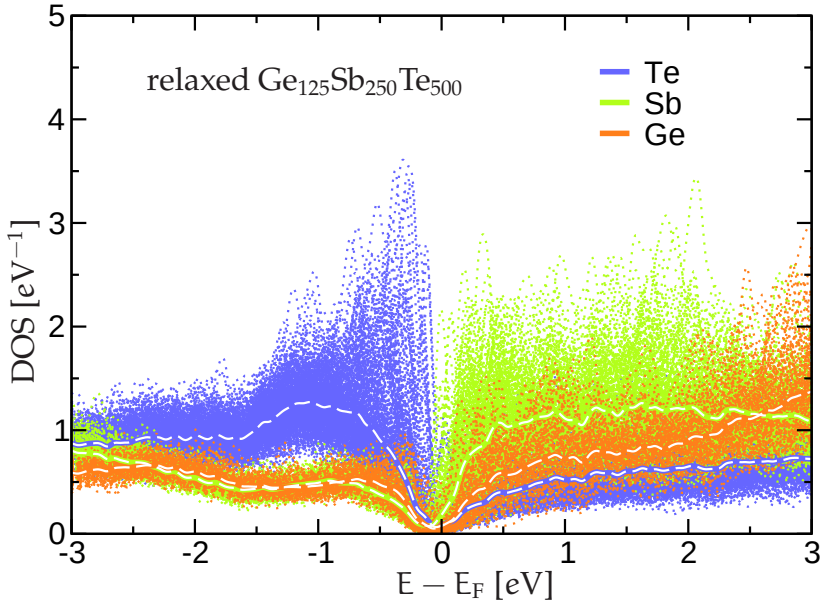


Figure 6.8.: Atom-resolved Density of States around the Fermi energy of the relaxed structure of $Ge_{125}Sb_{250}Te_{500}$. The dashed white lines indicate the DOS average within each species. Ge, Sb and Te are colored orange, green and blue, respectively. Compared the the initial structure (Figure 6.4) the number of states at E_F has been reduced considerably by the structural relaxation.

geometry. We can see from the position of the data points that, although continuous, n_{Vac} is usually found close to an integer value for the relaxed coordinates. The error bars indicate the variation in the bond lengths b_i.

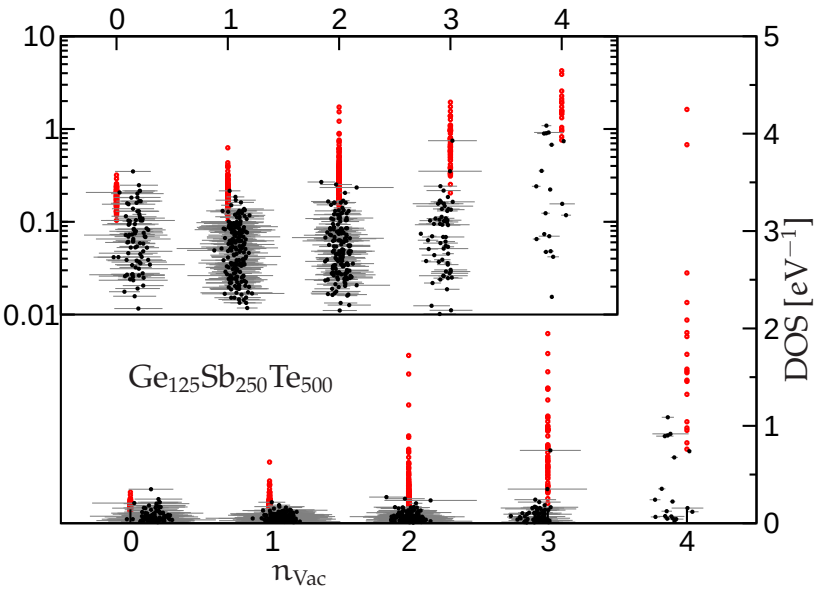


Figure 6.9.: Density of States on Te atoms at the Fermi level as a function of n_{Vac}. In the initial geometry, n_{Vac} is an integer given by the number of vacancies surrounding a Te atom (red dots). After relaxation, n_{Vac} is estimated from the number of bonds to the nearest neighbor atoms and their bond lengths (black dots), see text. The error bars indicate the variation of the bond lengths. The inlet shows the same numbers on a logarithmic scale.

As the most important trend, we observe that the DOS(E_F) has reduced for all Te atoms. Furthermore, we can see that Te atoms in class 0 (six bonds to Ge or Sb) are tending towards slightly longer bonds than 3.02 Å whereas classes 4 and 4′ with a lower bond coordination have short bonds in average. Class 2 and 2′ seem to be hardly affected. The annealing reduced the strength of the large density of states especially in the higher classes.

6.3. Large System: $Ge_{512}Sb_{1024}Te_{2048}$

The calculation of 5^3 unit cells of $Ge_1Sb_2Te_4$, i.e. $Ge_{125}Sb_{250}Te_{500}$ was, as expected, large enough to show no finit size effects. However, not all of the ten classes of Te atoms have been found in this systems. In fact, the chance of having a Te atoms surrounded by six vacancies in $5 \times 5 \times 5$ unit cells is below 12.2 %. Therefore, a similar analysis as above has been made for 8^3 unit cells of $Ge_1Sb_2Te_4$ again with

a random occupation of the second sublattice. Figure 6.10 illustrates the initial geometry of 3584 atoms with Te, Sb and Ge atoms colored in blue, green and orange, respectively.

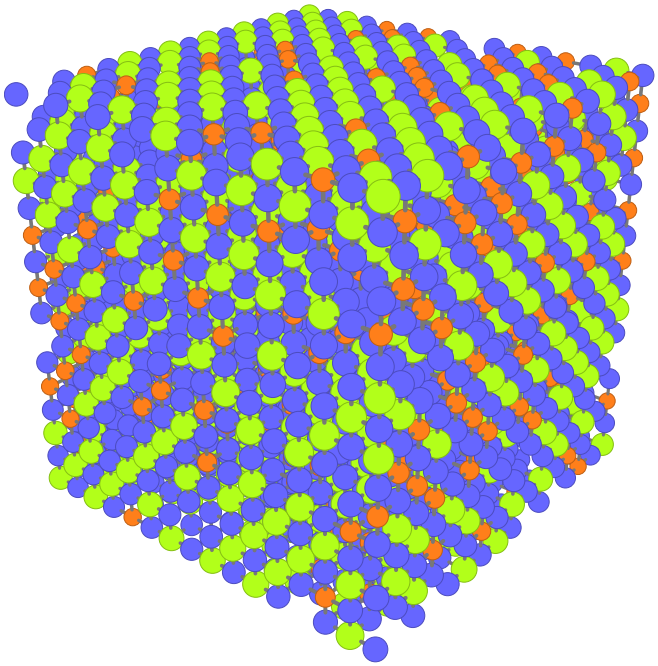


Figure 6.10.: Visualization of the random alloy $Ge_{512}Sb_{1024}Te_{2048}$ in the initial geometry of the ideal FCC positions.

Fortunately, this occupation configuration contains all of the ten classes specified in Section 6.2, although the probability of a Te atom in class 6 was 49 % within 2048 Te atoms, compare Table 6.3. Class 6 means that the all six nn sites of this Te atom unoccupied (vacancies). Table 6.4 show the occurrence of atomic distances. We can observe that in the large setup of 8^3 unit cells the occurrences of distances on the second FCC sublattice deviated less than 2.6 % compared to the ideal values given in Table 6.1.

The atom number 2711 in the calculation was found in class 6. Therefore, it feels only the nnn sites which are eight Te at a distance of 4.27 Å. As found for $Ge_{125}Sb_{250}Te_{500}$, we expect the local potential to be strongly elevated at the site of the atoms and a sharp density of states since there is no real bonding and thus no hybridization. Figure 6.11 shows the projected DOS of $Ge_{512}Sb_{1024}Te_{2048}$ for atom number 2711 in class 6 atom. As predicted, the Te-5s and 5p peaks are very sharp indicating a tiny degree of hybridization with the surrounding. Furthermore, the position of the 5s peaks at -9.8 eV is very high compared to the usual band center of the Te-5s states, e.g. in the initial geometry of $Ge_{125}Sb_{250}Te_{500}$ where the band was centered at -11.48 ± 0.76 eV, compare Figure 6.7.

After relaxation, the projected DOS on atom number 2711 (Te) shows a regular

[Å]	3.02	4.27	5.23	6.04	6.75	7.40	8.54	9.06	9.55
$[\frac{1}{2}a]$	1	$\sqrt{2}$	$\sqrt{3}$	2	$\sqrt{5}$	$\sqrt{6}$	$\sqrt{8}$	3	$\sqrt{10}$
Te—Te		24		12		48	24		48
Sb—Te	12		16		48			60	
Ge—Te	6		8		24			30	
Sb—Sb		6.035		3.000		11.994	5.936		11.924
Ge—Sb		5.946		3.006		12.014	6.059		12.000
Ge—Ge		1.503		0.752		2.953	1.469		3.021
Te— v	6		8		24			30	
Sb— v		5.984		2.994		11.998	6.070		12.152
Ge— v		3.052		1.490		6.080	3.004		5.957
v— v		1.483		0.758		2.961	1.463		2.945
$\sum$	24	48	32	24	96	96	48	120	96

Table 6.4.: Occurrence of interatomic distances in the initial coordinates of $Ge_{512}Sb_{1024}Te_{2048}$ normalized to one unit cell.

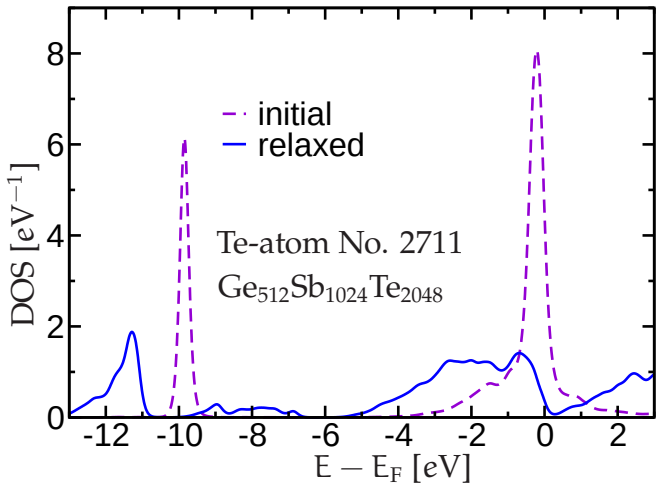


Figure 6.11.: Projected density of states on the atom No. 2711 before and after relaxation. Only this Te atom in $Ge_{512}Sb_{1024}Te_{2048}$ was in class 6 in the initial coordinates, i.e. no bonds to nearest neighbors. After relaxation the atom can be classified roughly in class 3, i.e. 3 bonds have formed at angles around 90°.

behavior compared to the average DOS. Three bonds have formed at angles close to 90° which allows a rough re-classification in class 3, compare Table 6.3. In order to gain a more statistical understanding of the effects of the relaxations, we analyze the changes of the interatomic distances. In the initial geometries, the atomic positions where integer multiples of $\frac{a}{2}$ and thus, all interatomic distance values were given by $\frac{a}{2}\sqrt{k}$ with k integer.

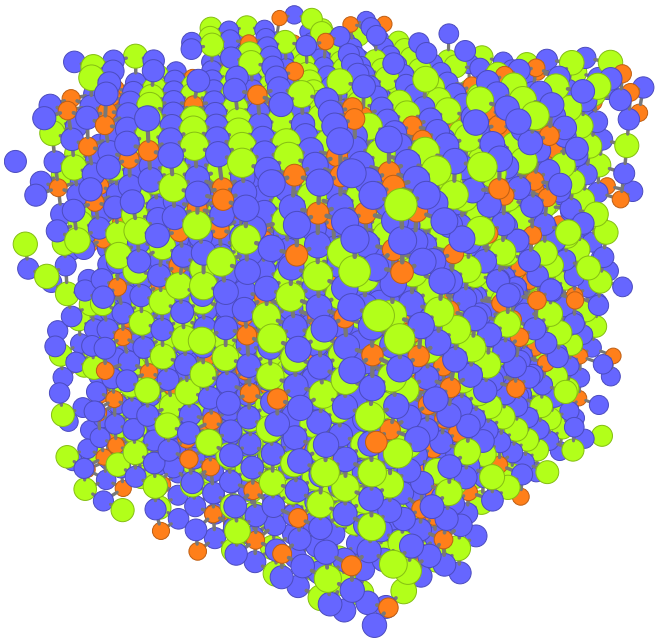


Figure 6.12.: Visualization of the geometry of $Ge_{512}Sb_{1024}Te_{2048}$ after relaxation.

Structural relaxation lets the distances become a continuous variable. However, the occurrences of interatomic distances of the relaxed geometry $Ge_{512}Sb_{1024}Te_{2048}$ exhibits a similar feature as the Tables 6.1 and 6.4 visible in Figure 6.13. Distances between sites of the same FCC sublattice (even k) are zero for Te-Ge and Te-Sb. In the same sense, distances connecting the different sublattices (odd k) do not occur for Ge-Ge, Sb-Sb or Te-Te. Unfortunately, the position of the vacancies is undetermined after structural relaxation.

Focussing onto the nn bonds, we can extract that there is a small difference between Sb and Ge in their bonding behavior towards Tellurium. The relaxation shows that the in Ge-Te bonds tends to be shorter by 1.9 pm in average which is a bond relaxation of $-0.6\,\%$ (Ge-Te 3.001 ± 0.15 Å). In contrast to that, the Ge-Sb bonds are elongated by $+2.2\,\%$ to 3.086 ± 0.12 Å. This tendency leads to a distortion from the regular rocksalt structure also without vacancies. Their respective width of the distance distribution indicated that the Te-Ge bond (dashed orange-blue line in Figure 6.13) occurres in local configurations with a higher degree of variation compared to Te-Sb. The nn peak of Te-Sb at 3.08 Å is relatively sharp

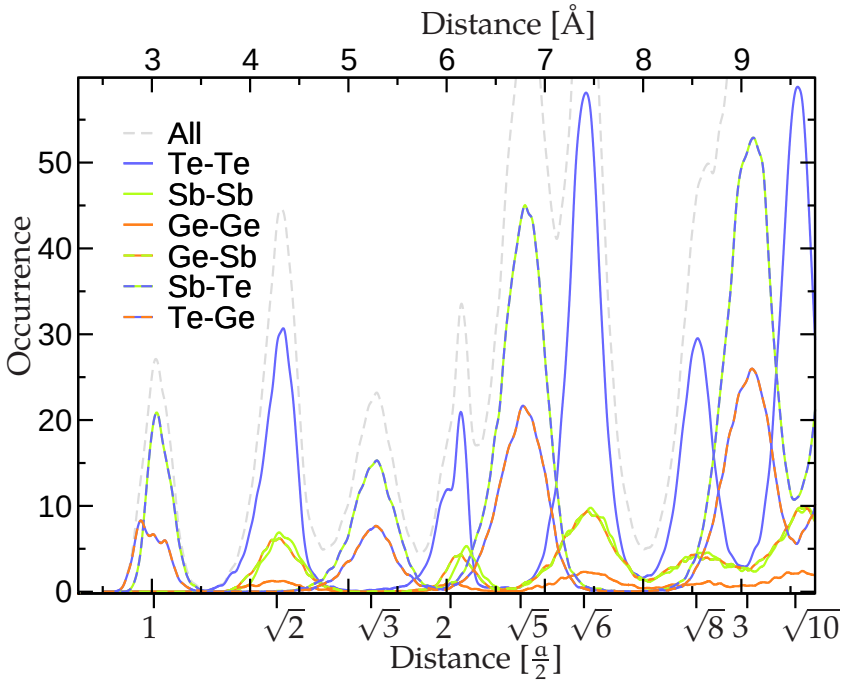


Figure 6.13.: Occurrence of interatomic distances in the relaxed structure of $Ge_{512}Sb_{1024}Te_{2048}$. The Sb-Te bonds centered at 3.086 ± 0.12 Å and are thus 2.2 % longer than the initial bonds with 3.02 Å. The Ge-Te bonds contracted by 0.6 % to 3.001 ± 0.15 Å.

and exhibits only a faint indication of a shoulder towards larger distances (dashed green-blue line). In contrast to that, the nn peak of Te-Ge shows a triple peak with different heights of the local maxima at 2.888, 3.015 and 3.131 Å, respectively.

Furthermore, it can be observed that both, the Ge-Sb and the Sb-Sb curve have their k=4 peak at +1.5 and +2.6 %, respectively. This indicates a repelling tendency beyond the Antimony atoms and a not as strong repulsion between Sb and Ge. The effect can also be seen at the nnn sites (k=2) but is much weaker there. In a similar fashion, the Te-Te nnn peak is a bit asymmetric and has its maximum at +1.5 %. However, its center of mass can be located at 4.278 ± 0.187 Å which corresponds only to an increase by 0.16 %. The k=4 Te-Te peak reveals a clear shoulder towards lower distances.

The electronic structure of the relaxed geometry of $Ge_{512}Sb_{1024}Te_{2048}$(Figure 6.14) shows the same signature as the DOS in Figure 6.8. Again, the blue lines illustrate the atomic DOS on Te atoms, green for Sb and orange for Ge. The sharp atomic DOS peaks of Te atoms which have been observed in the initial geometries indicating localized states of Te atoms surrounded by vacancies can hardly be found here.

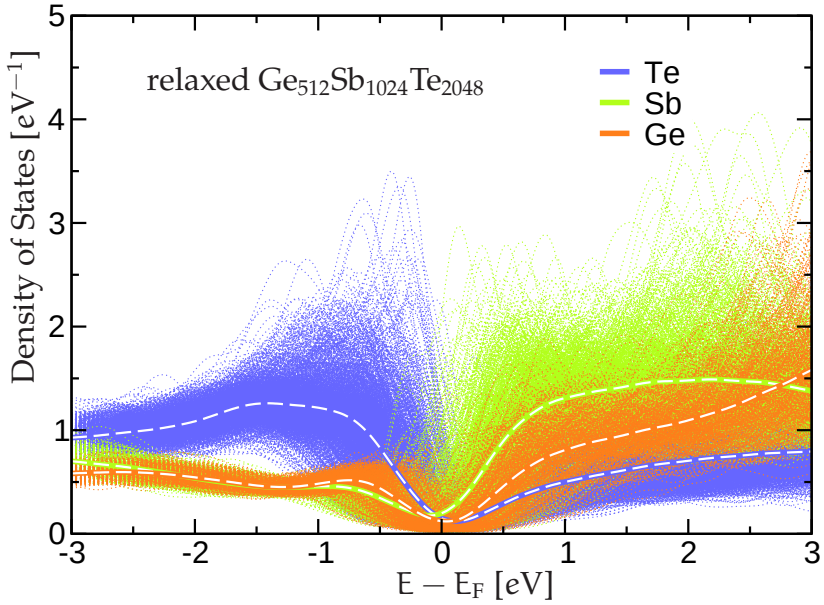


Figure 6.14.: Atomic density of states of $Ge_{512}Sb_{1024}Te_{2048}$. Snapshot after two SCF iterations with the relaxed geometry. A similar signature tp that in $Ge_{125}Sb_{250}Te_{500}$ becomes visible. The density of Te states below the Fermi level is strongly reduced compared to the DOS before relaxation. In particular, there are fewer Te atoms with sharply localized peaks.

6.4. Summary

The investigations on the electronic structure of disordered structures of $Ge_1Sb_2Te_4$, in particular $Ge_{125}Sb_{250}Te_{500}$ and $Ge_{512}Sb_{1024}Te_{2048}$, have shown that the initial geometries in the rocksalt structure exhibit large scattering centers with an elevated local potential and localized Te states in the region of high vacancy concentrations, i.e. vacancy clusters. The electronic structure was calculated with `jüRS`, the Jülich Real-Space DFT code for both system sizes. This implementation enables the parallel treatment of $Ge_{512}Sb_{1024}Te_{2048}$ on all 294,912 compute cores of the massively parallel supercomputer `JUGENE` [54].

We showed that structural relaxation decreases the strength of these scattering regions leading to a reduced resistivity. A similar trend was observed in the conductivity measurements on thin films of $Ge_1Sb_2Te_4$ by Zhang *et al.* [58]. The resistivity decreases with increasing temperature in the annealing process and drops even more for longer annealing times. This coupling highlights $Ge_1Sb_2Te_4$ and some other stoichiometries of Ge:Sb:Te as promising candidates for electrical switching between their crystalline and amorphous state.

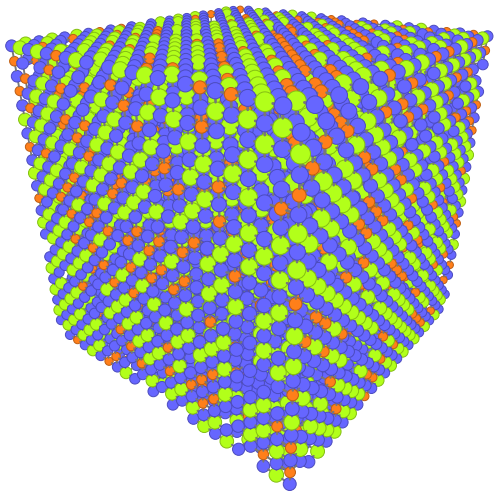


Figure 6.15.: Challenging setup of 12^3 unit cells of disordered $Ge_1Sb_2Te_4$ (12096 atoms). With a grid spacing of 0.252 Å, this calculation requires a storage volume of 9.25 Tbyte for all 48384 wave functions and about three times as much main memory during the calculation.

Even larger systems than those calculated are still challenging. However, the route for further progress has already been mapped out. For example, a setup with twelve thousand atoms, as illustrated in Figure 6.15, has moved closer to becoming feasible than it has been the case.

CHAPTER 7

CONCLUSIONS

The central goal of this thesis was the development of the parallel application jüRS [99] for large-scale calculations in the framework of density functional theory. The implementation is specially customized to the electronic structure and force calculations of large systems containing thousands of atoms employing massively parallel supercomputers as JUGENE, the IBM Blue Gene/P installed at Forschungszentrum Jülich. This machine has a nominal peak performance of one petaflops distributed onto 294,012 compute cores.

In contrast to a variety of complex basis sets for the representation of the Kohn-Sham wave functions and electron densities, as applied in other DFT calculations, we chose a uniform real-space grid which is parallelized in terms of memory and workload per grid point. A real-space grid can represent both periodic and isolated boundary conditions. Hence, the three-dimensional grid can combine different boundary conditions for different spatial directions permitting a proper description of not only molecules and bulk systems but also wire structures and surfaces.

The projector augmented-wave method enables an accurate description of the energy contributions and scattering properties of the atomic cores. For the sake of simplicity and an improved convergence behavior, the core electrons only enter the calculation as frozen core densities. Despite the great smoothness of the augmented wave functions and sophisticated filtering methods for the projector functions, the accurate representation of an atom on a real-space grid requires a number of grid points per atom that is up to two orders of magnitude larger than the number of valence electrons which come with this atom. Hence, the dimensionality of the Kohn-Sham Hamiltonian of a large system in the real-space representation is huge. However, the locality of the effective potential, the finite-difference approximation for the kinetic energy operator and the localization of the atomic PAW projectors cause the extreme sparsity of the Hamiltonian operator in real-space

representation. In fact, the number of non-vanishing elements in the Hamiltonian scales linearly with the system size. The application of the Hamiltonian operator onto a trial vector is thus of order-N. Furthermore, we only need to find the lowest 1 % of all eigenvalues and eigenvectors. These aspects, together with the fact that the Hamiltonian changes only slightly in the tail of convergence of the local potential, favor iterative diagonalization methods for the solution of the Kohn-Sham eigenvalue problem.

We found that a very efficient method for the iterative diagonalization of the real-space Hamiltonian is the combination of the DIIS band update with a rotation in the subspace of the occupied bands, which replaces an explicit Gram-Schmidt-like reorthogonalization. The generation of the effective potential and both components of the iterative diagonalization, the DIIS band update and the subspace rotation, are parallelized in a three-dimensional domain decomposition where each domain is assigned to one compute node. Each domain represents a rectangular portion of the Cartesian real-space grid. The finite-difference approximation introduces a limited range of non-locality for the evaluation of derivatives on the grid, which leads to a restriction of the communication pattern to the nearest-neighbor domains. Supercomputers with a three-dimensional torus network provide high bandwidths and short latencies for exactly these communication patterns. Hence, the mapping of the real-space grid domains to the three-dimensional grid of compute nodes allows excellent parallel efficiency of up to roughly 8 compute nodes per atom.

In the large-scale limit, the total workload of computations grows at best quadratically with system size. Therefore, a second parallelization level distributes the Kohn-Sham states over equivalent sets of compute nodes in domain decomposition. This state parallelization on top of the real-space parallelization simultaneously solves the memory bottleneck of 500 Mbyte/core on `JUGENE` and introduces a significant speedup. In particular, the parallelization of bands is a prerequisite for handling the large number of valence electrons in calculations of large-scale structures. Despite the need for data-intensive MPI communication to parallelize the subspace rotation method, an acceleration of the computation speed by a factor of eight has been achieved.

The implemented Jülich Real-Space DFT code project, `jüRS`, was applied to large disordered systems of the phase-change material $Ge_1Sb_2Te_4$. Short intense laser pulses can switch this ternary alloy from its crystalline phase to an amorphous state. During longer and less intense laser pulses, the material crystallizes again. This reversible phase change is accompanied by an alteration of the optical properties which promotes the class of Ge:Sb:Te-based materials for storage media as realized in rewritable DVDs. In the crystalline phase of $Ge_1Sb_2Te_4$, the clustering of vacuum impurity sites leads to strong relaxations of the local structure. The electronic structure was analyzed for the initial and the relaxed geometry $Ge_{125}Sb_{250}Te_{500}$ in

order to gain insight into the dependence of electrical resistance on annealing. The largest setup studied in this work contained 3584 atoms parallelized onto all resources of JUGENE. A self-consistency iteration with these settings thus takes 30 minutes whereas a work station would take about three years and would consume twice as much power.

Upcoming supercomputer architectures are based on multi- and many-core chipsets in their compute nodes, which promotes the usage of a node-internal shared memory parallelization and, to exploit the multi-core environment optimally, the overlapping of computation and communication. Furthermore, the integration of accelerator units into the compute nodes becomes increasingly significant such as graphical processing units in the combination CPU+GPU. Also, the improvements of the inter-node connections is substantial. The network of the next generation of IBM Blue Gene supercomputers is designed to connect the compute nodes to a five-dimensional torus. This matches the needs of jüRS perfectly in the sense that we map the real-space grid domains to three dimensions as shown in this work. The remaining two dimensions promise to accelerate the communication in the second parallelization level, in particular the compute nodes in band parallelization are then direct neighbors.

In this thesis, we showed that the real-space grid method for density functional calculations in combination with the accuracy of the PAW method can be parallelized efficiently and, applied on massively parallel supercomputers, can permit the determination of large-scale ground-state structures up to several thousands of atoms. The strength of our implementation was proven for the bulk system $Ge_{512}Sb_{1024}Te_{2048}$ with an edge length of 48.3 Å including disorder and relaxations. A larger field of application has thus become accessible. The combination of periodic and isolated boundary conditions invites us to investigate the relaxations of large functionalized molecules on substrates and complicated surface relaxations. Further conceivable topics include accurate predictions of the long-range interaction of impurity atoms in host crystals, doping in amorphous materials and a realistic description of entire nanodevices.

APPENDIX A

PAW DATA GENERATION

This chapter describes how the necessary precomputations for the Projector Augmented Wave (PAW) [22] transformation described in Chapter 3 are performed. PAW data are generated with some degrees of freedom. Starting from a spherically symmetric atomic all-electron calculation with a selected ionic configuration of occupation numbers f_ℓ, we need to identify the valence states and choose a set of energy parameters for each ℓ-channel.

A.0.1. Spherical Atomic All-Electron Calculation

Given the atomic number Z and occupation numbers for the designated states

$$f_{n\ell} = \begin{cases} 2 & \text{core states} \\ f_\ell & \text{valence states} \end{cases} \tag{A.1}$$

we start the atomic all-electron calculation by guessing a start density

$$n^{\text{guess}}(r) = (Z-2)\alpha\sqrt{\frac{\alpha r\, 108}{\pi}}e^{-3\alpha r}/r^2 + \gamma^3 e^{-\gamma r} \tag{A.2}$$

where $\alpha = 0.3058\sqrt[3]{\frac{Z}{2}}$ and $\gamma = \sqrt{4 + 3.2\Delta q}$ with the additional electron charge Δq.

The all-electron potential $V^{\text{ref}}[n](r)$ is found as a functional of the density $n(r)$ by simple radial integrations for the electrostatic potential. Furthermore, the exchange correlation potential can also be evaluated on the radial grid directly since a spherically symmetric representation $n_{L=00}(r)$ equals the real-space representation $n(\mathbf{r})$ for $|\mathbf{r}| = r$. Also for gradient corrected XC-functionals, only the radial

derivatives are required since all gradients w.r.t. to the angular coordinates vanish.

Useful for a good start guess are also approximate energy eigenvalues $\epsilon_{n\ell}$. A simple formula for qualitatively good start energies is

$$\epsilon_{n\ell}^{\text{guess}} = \underbrace{-\frac{1}{2}\left(\frac{Z}{n}\right)^2}_{\text{Hydrogen-like}} \underbrace{\left[0.783517 + 25.791 \cdot 10^{-6}\left(\frac{Z}{n}\right)^2\right]}_{\text{fit to 1s energies}} \underbrace{\mathbf{e}^{-(n-1)Z/100}}_{\text{screening}} \tag{A.3}$$

These start guess energies match best for the 1s-energies, since the middle part has been fit to the the 1s-energies [100] and the screening factor is unity for n=1. The energies $\epsilon_{n\ell}^{\text{guess}}$ are ℓ-degenerate although the true energies will depend strongly on ℓ. However, the ℓ-degeneracy is lifted already after the first self-consistency cycle.

The exact eigenvalues are found with the shooting method described in Section A.0.1. The results are depicted in Figure 3.2. Once all occupied eigenstates are found, a new density can be created

$$n^{\text{new}}(r) = \frac{1}{r^2}\sum_{n\ell} f_{n\ell}\,[r\Psi_{n\ell}(r)]^2 \tag{A.4}$$

and self-consistency is achieved by mixing of either the densities or the effective potentials. The latter has shown to achieve self-consistency in a reasonable number of SCF iterations, even with a simple mixing scheme $V^{\text{next}}(r) = V^{\text{last}}(r)(1-\alpha) + \alpha V[n^{\text{new}}](r)$ and α=$\frac{1}{3}$.

The all-electron atom solver will also provide the spherical true core density $n_c(r)$.

Shooting method

The shooting method is an iterative algorithm to solve a one-dimensional second order differential equation with two boundary conditions, as in our case finding an eigenvalue and an eigenstate of the radial Kohn-Sham equation

$$\left(-\frac{1}{2}\frac{d^2}{dr^2} + \frac{\ell(\ell+1)}{2r^2} + V_{\text{eff}}(r) - \epsilon\right) r\Psi(r) = 0 \tag{A.5}$$

The second order differential equation is cast into two coupled first order differential equations, such that a direct integration is possible. The method chooses a point r_m in the interval $(0, R_{\text{max}})$ and performs a Runge-Kutta type outwards integration for $[0, r_m]$ finding $r\Psi^{\text{out}}(r)$. Similarly, an inwards integration for $[r_m, R_{\text{max}}]$ starting from tiny values and derivatives at r=R_{max} finds $r\Psi^{\text{in}}(r)$. One of the two

partial solutions is scaled to match the other partial solution in value at the matching radius r_m. For a numerically stable method one can use the classical return radius as r_m, i.e. $\frac{\ell(\ell+1)}{2r_m^2} + V(r_m) = \epsilon$. However, the matched solutions do not coincide in their first derivatives at r_m as long as ϵ is not an exact eigenvalue. We can compute the so-called kink and use its strength to determine the energy correction for the next iteration. In order to find the correct eigenvalue, it is important to hit the right branch of solutions with the initially guess energy ϵ^{guess}. The branch can be determined by counting the number of nodes in both, $r\Psi^{in}(r)$ and $r\Psi^{out}(r)$. The wanted solution will show (n-ℓ-1) nodes where n is the principal quantum number and ℓ the angular quantum number. We therefore need to lower the guess energy as long as the number of nodes is higher than requested. Once the branch is hit, the kink is a monotonous function of energy so that we can easily find its zero with a bisection algorithm.

The physical interpretation of the shooting method is a matching of the logarithmic derivative of the core region with the logarithmic derivative of the potential tail. The energy must correspond to a bound state, i.e. should be lower than the $V(R_{max})$.

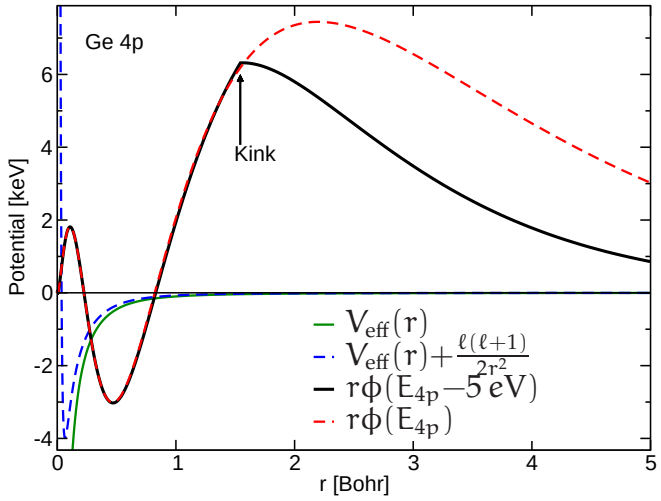


Figure A.1.: Schematic of the shooting method. For a given local effective potential, an inwards integrated solution of the radial Kohn-Sham equation (Equation A.5) is matched to an outwards integrated solution at a trial energy. The strength of the resulting kink at the matching radius r_m determines the energy correction until an eigenstate is found. The radial wave functions $r\phi(r)$ in the plot are scaled arbitrarily.

jüRS **Input Syntax:** `./paw inp -g`
File **inp** should contain the line `element Ag 5s* 1 5p* 4d 10 | 2.6`

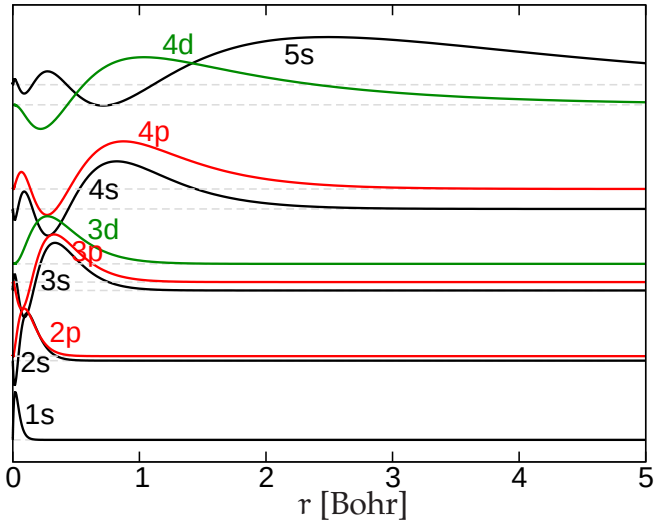


Figure A.2.: Radial atomic KS states for a Silver atom. The axes for each atomic wave function have been shifted according to their energy eigenvalue (logarithmically). The radial wave functions $r\phi_{n\ell}(r)$ are normalized to a maximal amplitude for better visibility.

A.0.2. Augmentation Procedure

Generation of true partial waves

Given a self-consistent spherically symmetric potential $V^{\text{ref}}(r)$ that has been generated with the ionic configuration f_ℓ of the valence states, we have to chose reference energies $\epsilon_{\ell n}$ for the true partial waves. Now n counts the number of partial waves in an ℓ-channel (c.f. `enn=1...s%nn(ell)`). Here, the lowest n=1 energy usually matches the eigenvalue of selected valence state. In this case we deal with an eigenstate of the reference Hamiltonian $\hat{H}^{\text{ref}} = \hat{T} + \hat{V}^{\text{ref}}$ that is normalizable and normalized. The advantage of this procedure is –besides numerical reasons –that the diagonal values of the atomic PAW density matrix $D^a_{ii\sigma}$ will get the physical significance of atomic occupation numbers. We will henceforth refer to higher n-values as excited partial waves since these may even be unbound scattering solutions. The true partial waves $\phi_{\ell n}$ are outwards integrated solutions of the radial Kohn-Sham equation A.5 the self-consistent all-electron reference potential $V^{\text{ref}}(r)$ at the chosen energies $\epsilon_{\ell n}$.

During the application of the PAW setup only a limited number $n_\ell^{\max}$ of partial waves according to the reference energies $\epsilon_{\ell n}$ are used. Two partial waves per ℓ for $\ell \in [0, \ell_{\text{phys.}}]$ plus one at $\ell_{\text{cut}} = \ell_{\text{phys.}}+1$ are typically a good choice. Figure A.3 defines $\ell_{\text{phys.}}$ for all elements of the periodic table.

There are different ways to generate the higher solutions $\phi_{\ell n}$ for $n > 1$ if the n=1-energy has been set to the valence state eigenvalue. One ways is to choose an

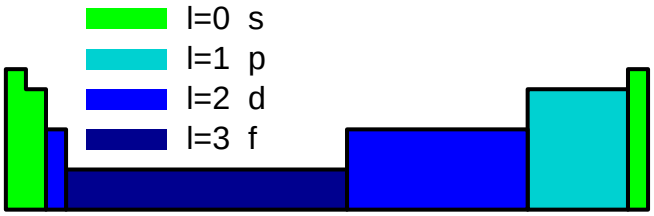


Figure A.3.: Physical $\ell_{\text{phys.}}$ for all elements of the periodic table. A good choice for ℓ_{cut} is $\ell_{\text{phys.}}$+1. The grid spacing of the coarse grid should be fine enough such that a minimum of $2(\ell_{\text{cut}}+1)^2$ grid points lie within the sphere. These are 8, 18, 32 and 50 for s-, p-, d- and f-elements, respectively.

energy $\epsilon_{\ell 2} = \epsilon_{\ell 1} + 1\,\text{Ha}$ which is usually an unbound state. As an alternative, one may generate an approximate energy derivative (as appearing in LAPW methods) by $\phi_{\ell 2} \approx \frac{1}{2\delta}\left(\phi_\ell[\epsilon_{\ell 1}+\delta] - \phi_\ell[\epsilon_{\ell 1}-\delta]\right)$. We use the latter method with δ=25 m*e*V in `subroutine true_partial_waves` which also allows for a second energy derivative $\phi_{\ell 3} \approx \frac{1}{\delta^2}\left(\phi_\ell[\epsilon_{\ell 1}+\delta] - 2\phi_\ell[\epsilon_{\ell 1}] + \phi_\ell[\epsilon_{\ell 1}-\delta]\right)$.

We can even find the exact energy derivative by solving the inhomogeneous radial Kohn-Sham equation

$$\left(\hat{H}-\epsilon\right)|\dot{\Psi}\rangle = |\Psi\rangle$$

where an inhomogeneous integration technique for the full SRA-Hamiltonian is required, c.f. Section A.0.4.

Generation of smooth partial waves

We first need to choose a augmentation radius r_{cut}. It is possible to use different radii for the different ℓ-channels where the smallest one will be used for the augmentation of the local potential. Simply add more values (in Bohr) for the radii after the separator `|` in the species configuration string as e.g. `element Ag 5s* 1 5p* 4d 10 | 2.4 2.6 2.5`. This will set r_s=2.4, r_p=2.6, r_d=2.5 and r_f=2.5. We pseudize the partial waves $\phi_{\ell n}(r)$ to find $\tilde{\phi}_{\ell n}(r)$ by replacing the heavy oscillating behavior in the core region ($r < r_{\text{cut}}$) with a smooth function, that matches the true wave function at r_{cut} in value, 1st, 2nd and 3rd derivative and has the correct r^ℓ-behavior at the origin. The norm does not need to be conserved.

Here I use the 4 polynomials r^ℓ, $r^{\ell+2}$, $r^{\ell+4}$ and $r^{\ell+6}$ to match the true wave function in 0th through 3rd derivative at r_{cut}. The 0th derivative is the value of the wave function itself. The treatment with polynomials simplifies the computation of the matrix elements of the kinetic energy operator $\langle\tilde{\phi}_{\ell n}|\hat{T}|\tilde{\phi}_{\ell n'}\rangle$ w.r.t. the smooth partial waves in the sense, that we can compute the derivative w.r.t. r analytically.

Assume that a smooth partial wave inside the cutoff radius is given by

$$r\tilde{\phi}(r) = r^{\ell+1}\left(\sum_{i=0}^{3} c_i\, r^{2i}\right)$$

then the kinetic energy operator for these $r\tilde{\phi}$-states in radial representation

$$\hat{T} = \frac{1}{2m}\left(\frac{\ell(\ell+1)}{r^2} - \frac{d^2}{dr^2}\right)$$

applied to the polynomials yields

$$\hat{T}\, r\tilde{\phi}(r) = \frac{1}{2m} r^{\ell+1} \sum_{i=0}^{3} c_i\, [\ell(\ell+1) - (\ell+2i)(\ell+1+2i)]\; r^{2i-2}.$$

We see that the term for $i=0$ cancels completely such that there are no negative powers of r left. This is the reason why we required that the smooth wave function behaves like r^ℓ at the origin. Then

$$\begin{aligned}
\hat{T}\, r\tilde{\phi}(r) &= \frac{1}{2m} r^{\ell+1} \sum_{i=1}^{3} c_i\, [\ell(\ell+1) - (\ell+2i)(\ell+1+2i)]\; r^{2i-2} \\
&= \frac{1}{2m} r^{\ell+1} \sum_{i=0}^{2} c_{i+1}\, [\ell(\ell+1) - (\ell+2i+2)(\ell+1+2i+2)]\; r^{2i} \\
&= -\frac{1}{2m} r^{\ell+1} \sum_{i=0}^{2} c_{i+1}(2i+2)(2i+2+2\ell+1)\, r^{2i}
\end{aligned}$$

So in a matrix representation w.r.t. the polynomial coefficients, we can write

$$-2m\hat{T} = \begin{pmatrix}
0 & 2(2\ell+3) & 0 & 0 \\
 & 0 & 4(2\ell+5) & 0 \\
 & & 0 & 6(2\ell+7) \\
 & & & 0
\end{pmatrix}.$$

The evaluation of matrix elements of the kinetic energy operator w.r.t. the true partial waves is performed via the energy parameter ϵ in order to capture the relativistic effects that have been accounted for in the radial SRA solver, if SRA is applicable:

$$\hat{T}\, r\phi(r) = [\epsilon - V_{\text{loc}}(r)]\; r\phi(r).$$

Augmentation of the core electron density

The true core electron density $n_c(r)$ has been found during the self-consistent atomic calculation, c.f. Section A.0.1. We augment the core density $n_c(r)$ in the same way as the smooth partial waves to find $\tilde{n}_c(r)$. There is a certain degree of freedom here. However, it is in general useful to have an only positive function $\tilde{n}_c(r) > 0\,\forall r$, because $\tilde{n}_c(r)$ will enter the exchange correlation potential that should not be evaluated with negative density values. Here again, the norm does not need to be conserved. When the smooth core density enters the electrostatic expressions, a compensation charge density will account for the missing charge. Since the core density is spherically symmetric, the corresponding compensation charge moment is only a monopole q_{00} (`s%q00`) that also includes the charge $-Z$ of the protons in the atomic nucleus. For the electrostatics,

$$\int_S r^2 dr\, n_c(r) - Z = \int_S r^2 dr\, \tilde{n}_c(r) + q_{00} Y_{00} = -N_v \tag{A.6}$$

must hold, where N_v is the number of valence electrons in the start configuration (if neutral).

Compute the norm deficit

$$\Delta q_{\ell n} = \langle \phi_{\ell n} | \phi_{\ell n} \rangle - \langle \tilde{\phi}_{\ell n} | \tilde{\phi}_{\ell n} \rangle = \int dr\, r^2 \left(\phi^2_{\ell n}(r) - \tilde{\phi}^2_{\ell n}(r) \right) \tag{A.7}$$

and

$$\Delta q_{\text{core}} = \int dr\, r^2 \left(n_c(r) - \tilde{n}_c(r) \right) = N_c - \int dr\, r^2\, \tilde{n}_c(r) \tag{A.8}$$

Usually, these integrals are performed on a radial grid in real space, so the integration only needs to run up to r_{cut} since both partial waves are identical beyond this radius.

The local potential

We construct the smooth charge densities

$$\tilde{n}_v(r) = \sum_\ell f_\ell \tilde{\phi}^2_{\ell 1}(r) \tag{A.9}$$

$$\tilde{\rho}(r) = \tilde{n}_v(r) + \tilde{n}_c(r) + \left(\sum_\ell f_\ell\, \Delta q_{\ell 1} - N_v + q_{\text{core}} \right) \hat{g}_0(r) \tag{A.10}$$

where N_v is the number of valence electrons. Further, we evaluate the augmented potentials $V_{es}[\tilde{\rho}]$ and $V_{xc}[\tilde{n}_v + \tilde{n}_c]$. The new PAW Hamiltonian

$$\hat{\tilde{H}} = \hat{T} + V_{es}[\tilde{\rho}](\mathbf{r}) + V_{xc}[\tilde{n}_v + \tilde{n}_c](\mathbf{r}) + \bar{V}(\mathbf{r}) + \hat{\tilde{H}}_{\text{nonlocal}} \tag{A.11}$$

now has some freedom: A potential $\bar{V}^a(\mathbf{r})$ localized inside the spheres $\mathcal{S}^a$ may be added to the local part and subtracted in the nonlocal Hamiltonian matrix elements H^a without affecting the KS-eigenvalues. This can be exploited to construct the local potential such that the scattering properties in the lowest ℓ-channel which does not have any projectors ($\ell_{\max}+1$) coincide with the all-electron potential at a chosen energy $\epsilon_{\ell_{\max}+1}$. The procedure for finding the shape of $\bar{V}(r)$ could follow the construction rules for local pseudopotentials [28]:

$$V_{\text{loc}}(r) = \epsilon_\ell - \frac{\hat{T}\tilde{\phi}_\ell(r)}{\tilde{\phi}_\ell(r)} = V_{\text{es}}[\tilde{\rho}](r) + V_{\text{xc}}[\tilde{n}_v + \tilde{n}_c](r) + \bar{V}(r) \tag{A.12}$$

Alternatively, the local potential can be constructed as smooth as possible.

Generation of projector functions

Finally, we construct the projector functions $\tilde{p}_{n\ell}(r)$ according to

$$\tilde{p}_{\ell n}(r) = \left(\hat{\tilde{H}}_{\text{loc}} - \epsilon_\ell\right)\tilde{\phi}_{\ell n}(r) = \left(\hat{T} + V_{\text{es}}[\tilde{\rho}](r) + V_{\text{xc}}[\tilde{n}_v + \tilde{n}_c](r) + \bar{V}(r) - \epsilon_\ell\right)\tilde{\phi}_{\ell n}(r) \tag{A.13}$$

These functions are nonzero since the smooth partial waves have been chosen maximally smooth under the constraints of matching the true partial waves at the sphere boundary and showing the correct behavior at the origin. Figure A.5 illustrates the projector functions of Copper. In PAW, the smooth partial waves are no eigenstates of the smooth local potential. These projector functions however are good for the lowest projector of each ℓ-channel. For higher projectors found that way, the orthogonality with the smooth partial waves $\langle\tilde{p}_{\ell n}|\tilde{\phi}_{\ell n'}\rangle = \delta_{nn'}$ has to be enforced. This method agrees with the way of Vanderbilt [30].

The augmented local potential is smooth and behaves quadratically at $r = 0$. Since the smooth partial waves $\tilde{\phi}_{\ell n}$ are constructed to behave as r^ℓ at the origin, the application of the smooth local Hamiltonian will transfer this behavior to the preliminary projector functions. The term in the kinetic energy that goes as $r^{\ell-2}$ arising from $-\frac{1}{2}\frac{d}{dr^2}$ will exactly cancel with the centrifugal potential $\frac{\ell(\ell+1)}{2r^2}$ arising from the angular parts of the Laplacian operator. Usually, the lowest s-projector will be a strictly localized nodeless function, the next higher projector will have a single node after the orthogonalization has been performed. See Figure A.5 for an example.

A.0.3. Scattering of the PAW Potential

The scattering of a spherically symmetric local potential is completely determined by the scattering phase shift δ_ℓ [101]. If our potential at hand vanishes for large

distances, i.e. it has no long-range part, the scattering phase is a constant. In the case of a spherically symmetric local potential, we can find the scattering wave by an outwards integration of the Schrödinger equation

$$\left(-\frac{1}{2m}\frac{d^2}{dr^2}+\frac{\ell(\ell+1)}{2mr^2}+V_{loc}(r)-E\right)\Psi(r)=0 \tag{A.14}$$

or its equivalent of the Scalar Relativistic Approximation (SRA) as described in Section A.0.4. The logarithmic derivative is then found by taking the ratio of the wave functions first derivative and its value at a certain radius.

$$G_\ell^{tru}(R,E)=\left.\frac{\partial_r\psi_{out}^{[E]}(r)}{\psi_{out}^{[E]}(r)}\right|_{r=R} \tag{A.15}$$

Since we want to compare the scattering of the true local potential and the augmented non-local potential, we choose this radius R to be larger than the augmentation radius R_{cut} of the potential. For the non-local potential, we are not left with a second order differential equation any more. Thus we cannot solve it by reduction of order as exploited in the solution of the purely local potential. We are rather left with an integro-differential equation of the kind

$$\left(\hat{T}+\hat{\tilde{V}}_{loc}-E+\sum_{ij}|\tilde{p}_i\rangle\left(H_{ij}-ES_{ij}\right)\langle\tilde{p}_j|\right)|\tilde{\Psi}\rangle=0 \tag{A.16}$$

where the kinetic energy operator $\hat{T}$ contains spatial derivatives and the inner product $\langle p_j|\tilde{\Psi}\rangle$ a spatial integral. In order to solve this, we first solve for the homogeneous solution of the local potential

$$\left(\hat{T}+\hat{\tilde{V}}_{loc}-E\right)|\tilde{\psi}_0^{[E]}\rangle=0 \tag{A.17}$$

and then solve the inhomogeneous equations for each projector state $|\tilde{p}_i\rangle$

$$\left(\hat{T}+\hat{\tilde{V}}_{loc}-E\right)|\tilde{\psi}_i^{[E]}\rangle=|\tilde{p}_i\rangle\qquad,i>0 \tag{A.18}$$

The integro-differential equation is then cast into a set of linear equations by the *ansatz* (the superscript $^{[E]}$ is dropped here for simplicity of the notation)

$$|\tilde{\Psi}\rangle=\sum_{k=0}b_k|\tilde{\psi}_k\rangle \tag{A.19}$$

which results in

$$\sum_{k=0}b_k\left(\hat{T}+\hat{\tilde{V}}_{loc}-E\right)|\tilde{\psi}_k\rangle+\sum_{ij}|\tilde{p}_i\rangle\left(H_{ij}-ES_{ij}\right)\langle\tilde{p}_j|\sum_{k=0}b_k|\tilde{\psi}_k\rangle=0 \tag{A.20}$$

$$\sum_{k=1}b_k|\tilde{p}_k\rangle+\sum_{ij}|\tilde{p}_i\rangle\left(H_{ij}-ES_{ij}\right)\langle\tilde{p}_j|\sum_{k=0}b_k|\tilde{\psi}_k\rangle=0 \tag{A.21}$$

where the first summation over k starts from 1 because $|\tilde{p}_0\rangle$ vanishes in all space. We can now exploit that the projectors $|\tilde{p}_i\rangle$ are by construction orthogonal to the smooth partial waves $\langle\tilde{\phi}_n|$. Multiplying $\langle\tilde{\phi}_n|$ from the left will lead to a set of linear equations

$$\sum_{k=1} b_k \delta_{nk} + \sum_{k=0} \sum_{ij} \delta_{ni} \left(H_{ij} - E S_{ij}\right) \langle\tilde{p}_j|\tilde{\psi}_k\rangle b_k = 0 \tag{A.22}$$

$$\text{with} \quad \sum_{j} \left(H_{ij} - E S_{ij}\right) \langle\tilde{p}_j|\tilde{\psi}_k\rangle := A_{ik} \tag{A.23}$$

$$b_n + \sum_{k=0} b_k A_{nk} = 0 \tag{A.24}$$

that determine the coefficients b_n relative to b_0. Nevertheless, the choice of b_0 is irrelevant for the logarithmic derivative. As an example, for two projectors in the ℓ-channel at hand the system will look like this

$$\begin{bmatrix} 1 & 0 & 0 \\ A_{10} & A_{11}+1 & A_{12} \\ A_{20} & A_{21} & A_{22}+1 \end{bmatrix} \cdot \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{A.25}$$

The logarithmic derivative is thus found by the superposition of the particular solutions.

$$G_\ell^{\mathrm{PAW}}(R, E) = \left.\frac{\sum_{k=0} b_k \partial_r \tilde{\psi}_k^{[E]}(r)}{\sum_{k=0} b_k \tilde{\psi}_k^{[E]}(r)}\right|_{r=R} \tag{A.26}$$

Comparing the logarithmic derivatives over the energy range of the occupied valence states is essential to check the quality of the non-local PAW potential. However, we also have to check the augmented potential to show no resonances at energies below the lowest valence band, because these would lead to so-called *ghost states* [102], i.e. states that tend to fall into a pseudo-core state. This is an effect of the approximation that core states are not treated explicitly and so valence states cannot be kept orthogonal to the core states.

`jüRS` **Input Syntax:** `./paw -g -cm 2`

A.0.4. Solving the Scalar-Relativistic Equation

The scalar-relativistic (SR) equation

$$\left(-\frac{\hbar^2}{2m(r)}\frac{d^2}{dr^2}+\frac{\hbar^2\ell(\ell+1)}{2m(r)\,r^2}+V_{Hxc}(r)-\frac{Ze^2}{r}-E\right)r\phi(r)=0 \tag{A.27}$$

is solved by reduction of order: Besides $g(r)=r\phi(r)$, a so-called small component auxiliary wave function, the large component wave function $f(r)$ is introduced. The second order differential equation can then be reduced to a set of two first order differential equations:

$$\frac{d}{dr}\begin{pmatrix} g \\ f \end{pmatrix}=\begin{bmatrix} \frac{1}{r} & m(r) \\ 2W(r) & -\frac{1}{r} \end{bmatrix}\begin{pmatrix} g \\ f \end{pmatrix}:=\hat{\partial}\begin{pmatrix} g \\ f \end{pmatrix} \tag{A.28}$$

$$W(r)=\frac{\ell(\ell+1)}{2m(r)r^2}+V_{Hxc}(r)-\frac{Ze^2}{r}-E \tag{A.29}$$

in Hartree atomic units, i.e. $\hbar=m_e=e^2=1$ and $c=137.036$, the speed of light. For the scalar-relativistic version, the relativistic increase of the electron mass at high kinetic energies has to be taken into account:

$$m(r)=m_e\sqrt{1+\frac{E_{kin}}{m_ec^2}}\approx m_e+\frac{E_{kin}(r)}{2c^2}\qquad E_{kin}(r)=E-\left(V_{Hxc}(r)-\frac{Ze^2}{r}\right)$$

The Schrödinger equation is found by replacing the relativistic mass $m(r)$ by the rest mass m_e of the electron. The approximation of the squareroot is valid for all elements of the lower half of the periodic table, even though the effects to the core levels and total energy of atomic configurations may be of several Hartree it does not alter the valence states. For larger Z, it will affect the valence states. For example Gold shows differences of 0.25 eV in the valence levels (5d and 6s).

Now the first order equations are integrated using the Adam's-Moulton implicit 3 step 4th order multistep method:

$$y_{i-1}+\frac{h}{24}\left(9\hat{\partial}y_i+19\hat{\partial}y_{i-1}-5\hat{\partial}y_{i-2}+\hat{\partial}y_{i-3}\right)=y_i+\mathcal{O}(h^5) \tag{A.30}$$

Since both, the value y_i and the derivative $\hat{\partial}y_i$ are unknown, they can be brought to the left hand side

$$\frac{8}{3h}y_i-\hat{\partial}y_i=\frac{1}{9}\left(24/h\,y_{i-1}+19\hat{\partial}y_{i-1}-5\hat{\partial}y_{i-2}+\hat{\partial}y_{i-3}\right):=b_i \tag{A.31}$$

Now the derivative $\hat{\partial}y$ is a linear function of y, given by the 2×2 matrix $\hat{\partial}$ in Equation (A.28) from above. (The implementation of $\hat{\partial}$ includes the factor dr.)

Finally, y_i is found by applying the inverse operator to the right hand side (which was defined as b_i in Equation (A.31)).

$$y_i = \left[\frac{8}{3h} - \hat{\partial}\right]^{-1} b_i \tag{A.32}$$

For the inversion of the 2×2 operator $[\frac{8}{3h} - \hat{\partial}]$ we can use the simple formula adjoint divided by determinant.

Behavior at the origin

To analyze the wave function's behavior at $r = 0$, we expand the two components $g(r)$ and $f(r)$ in power series.

$$m(0) = m_e + \frac{E - V_{Hxc}(0)}{2c^2} + \frac{Ze^2}{2c^2}\left(\frac{1}{r}\right) \tag{A.33}$$

$$:= m_1 + \frac{m_Z}{r} \tag{A.34}$$

$$2W(0) = \frac{\ell(\ell+1)}{m(0)r^2} + 2\left(V_{Hxc}(0) - E - \frac{Ze^2}{r}\right) \tag{A.35}$$

$$:= \frac{\ell(\ell+1)}{m_1 r^2 + m_Z r} + w_1 + \frac{w_Z}{r} \tag{A.36}$$

Two special cases simplify these expressions drastically:
ℓ=0:

$$2W(0) := w_1 + \frac{w_Z}{r} \tag{A.37}$$

$$\frac{d}{dr}\begin{pmatrix} g \\ f \end{pmatrix} = \frac{1}{r}\begin{bmatrix} 1 & m_Z \\ w_Z & -1 \end{bmatrix}\begin{pmatrix} g \\ f \end{pmatrix} + \begin{pmatrix} m_1 f \\ w_1 g \end{pmatrix} \tag{A.38}$$

leads to the double recursive relation

$$f_i = \frac{w_Z g_i + w_1 g_{i-1}}{i+1} \tag{A.39}$$

$$g_{i+1} = \frac{m_Z w_1/(i+2) g_i + m_1 f_i}{i - m_Z w_Z/(i+2)} \tag{A.40}$$

and Z=0:

$$m(0) := m_1 \tag{A.41}$$

$$2W(0) := \frac{\ell(\ell+1)}{m_1 r^2} + w_1 \tag{A.42}$$

$$\Rightarrow \quad g_{k+1} = \frac{m_1 w_1}{k(k+1) - \ell(\ell+1)} g_{k-1} \tag{A.43}$$

$$\Rightarrow \quad f_k = \frac{1}{k+1}\left[\frac{\ell(\ell+1)}{m_1} g_{k+1} + w_1 g_{k-1}\right] \tag{A.44}$$

It is simple to show that both special cases coincide, if ℓ=0 and Z=0.

For the general expression ($\ell > 0, Z > 0$) it is advantageous to expand the expression

$$\frac{\ell(\ell+1)}{m_1 r^2 + m_Z r} = \frac{\ell(\ell+1)}{m_Z r(1+\frac{m_1}{m_Z} r)} \approx \frac{\ell(\ell+1)}{m_Z r}\left(1 - \frac{m_1}{m_Z} r\right)$$

because then, $2W(0)$ can be reformulated as

$$2W(0) := \frac{\ell(\ell+1)}{m_Z r} - \frac{m_1 \ell(\ell+1)}{m_Z^2} + w_1 + \frac{w_Z}{r}$$

Solving the inhomogeneous equation

The inhomogeneous equation

$$\left(-\frac{\hbar^2}{2m(r)}\frac{d^2}{dr^2} + \frac{\hbar^2 \ell(\ell+1)}{2m(r)\, r^2} + V_{Hxc}(r) - \frac{Ze^2}{r} - E\right) r\,\phi(r) = p(r) \tag{A.45}$$

leads to a derivative operator

$$\frac{d}{dr}\begin{pmatrix} g \\ f \end{pmatrix} = \begin{bmatrix} \frac{1}{r} & m(r) \\ 2W(r) & -\frac{1}{r} \end{bmatrix}\begin{pmatrix} g \\ f \end{pmatrix} - \begin{pmatrix} 0 \\ 2p(r) \end{pmatrix} := \hat{\partial}_p \begin{pmatrix} g \\ f \end{pmatrix} \tag{A.46}$$

$$\tag{A.47}$$

as one can show easily verify

$$g' = \frac{g}{r} + mf \qquad \text{and} \tag{A.48}$$

$$f' = 2Wg - \frac{f}{r} - 2p \tag{A.49}$$

$$\Leftrightarrow \frac{mf}{r} = \frac{g'}{r} - \frac{g}{r^2} \tag{A.50}$$

$$\text{then } g'' = \frac{g'}{r} - \frac{g}{r^2} + mf' \tag{A.51}$$

$$= 2mWg + 2mp \tag{A.52}$$

$$\Leftrightarrow -\frac{g''}{2m} + Wg = p \tag{A.53}$$

Then the multistep method differs slightly:

$$y_{i-1} + \frac{h}{24}\left(9\hat{\partial}_p y_i + 19\hat{\partial}_p y_{i-1} - 5\hat{\partial}_p y_{i-2} + \hat{\partial}_p y_{i-3}\right) = y_i + \mathcal{O}(h^5) \tag{A.54}$$

Again, the value y_i and the derivative $\hat{\partial}_p y_i$ are unknown. However this time, we split $\hat{\partial}_p$ into the 2x2 matrix $\hat{\partial}$ and the inhomogeneity 2-component vector $(0, -2p)$. Only $\hat{\partial} y_i$ is brought to the left hand side which leads to

$$\frac{8}{3h}y_i - \hat{\partial}y_i = \frac{1}{9}\left(24/h\, y_{i-1} + 19\hat{\partial}_p y_{i-1} - 5\hat{\partial}_p y_{i-2} + \hat{\partial}_p y_{i-3}\right) - 2(0,p)\frac{8}{3} := b_i \tag{A.55}$$

Mind the different operators $\hat{\partial}_p$ and $\hat{\partial}$ with and without inhomogeneity, respectively. Also here, the implementation requires a multiplication with the factor dr.

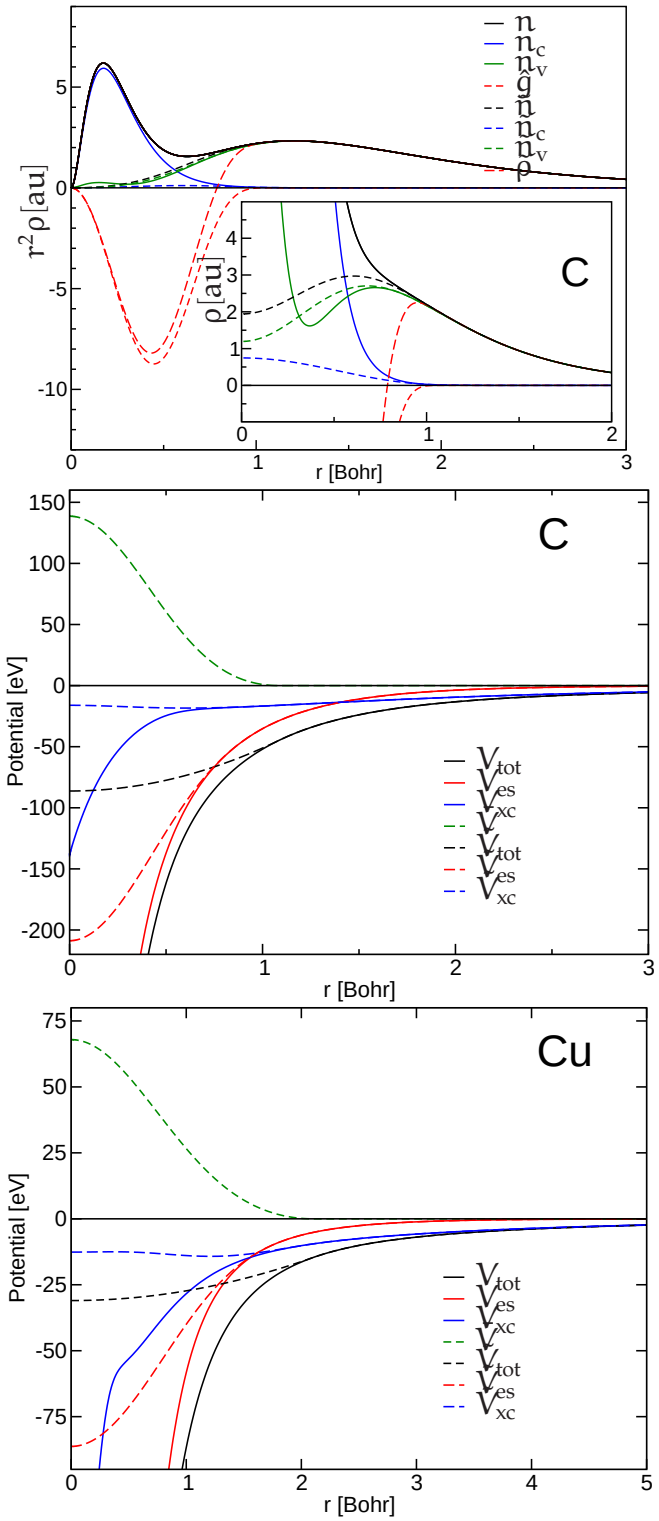


Figure A.4.: Examples of the PAW data generation process. Shown are the true and smooth densities for Carbon, the true and smooth potentials for Carbon (r_{cut}=1.1 Bohr) and for comparison the true and smooth potentials for Copper (r_{cut}=2.4 Bohr).

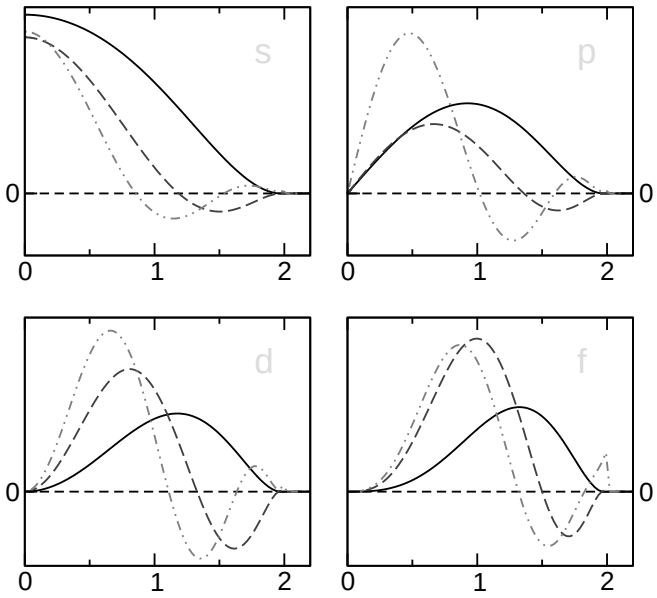


Figure A.5.: Projector functions for Copper (r_{cut}=2.1 Bohr).

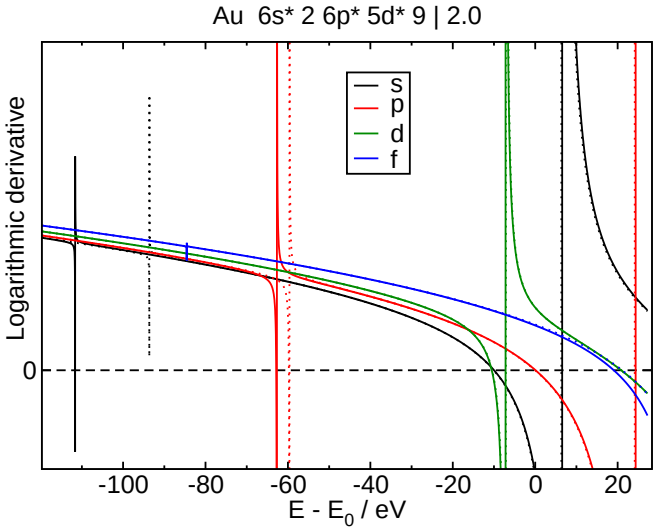


Figure A.6.: Logarithmic derivative as a function of energy for Gold with an indication of ghost states. The augmentation radius of 2.0 Bohr is too small so false resonances (dotted lines) appear at -93.6 eV and -59.6 eV in the *s*- and p-channel, respectively. Elsewhere, the scattering properties of the PAW setup coincide with those of the full-potential (solid lines).

APPENDIX B

GRID VS. BASIS

A real-space grid for the representation of a wave function is no basis set *per se* but can be considered equivalent to a basis set. The simplest example for their equivalence is the Fourier transform being the transformation from a grid in real-space into a plane-wave basis set. However, it is in general difficult to determine a localized function that describes a grid point in the fashion of a basis function. This can be seen considering the following problem arising with real-space grids.

Density Grids

During DFT calculations, the local effective potential needs to be evaluated on a real-space grid since usual density functionals are local (LDA) or semi-local (GGA) functions of the density. The real-space grid is at least $2\times$ denser (in each spatial direction) than the underlying equivalent wave function grid. For plane wave methods, the larger grid is obviously needed since the calculation of the density of a wave function in Fourier space is a convolution of two wave functions. If now the wave functions are given in Fourier coefficients up to G_{max}, the resulting density representation will hold Fourier coefficients up to $2\,G_{max}$. Real-space grid methods that sample the wave functions on a coarse grid with grid spacing h usually compute the resulting smooth density on the same grid, as

$$\tilde{n}_\sigma(\mathbf{r}_i) = \sum_{n\mathbf{k}} f_{n\sigma\mathbf{k}} |\Psi_{n\sigma\mathbf{k}}(\mathbf{r}_i)|^2. \tag{B.1}$$

Thinking in Fourier space, the wave function grid can represent frequency components up to $\frac{\pi}{h}$. Representing the density on the same grid, means that all possible frequency components in the range $\frac{\pi}{h} < |\mathbf{G}| \leqslant 2\frac{\pi}{h}$ are lost information. We can therefore identify $\frac{\pi}{h}$ to be the equivalent to a plane-wave cutoff of the density, not

of the wave function representation. In order to reach the same level of accuracy, we therefore have to apply two times higher cutoff frequencies, equivalent to four times higher cutoff energies or half the grid spacing. This, of course, is connected to a huge increase in the total workload, which usually does scale even less fortunate than linear with the total number of grid points.

A reason why real space methods still work well even though some higher frequency information of the density are lost is an advantageous error cancellation. The density is constructed from occupied orbitals of the KS Hamiltonian. Each two wave functions are mutually orthogonal, such that the extrema of a wave function coincide locally with the nodes (nodal surfaces in 3D, nodal lines in 2D and nodes in 1D) of another wave function. The simplest and extreme case is the homogeneous electron gas. The occupied eigenstates are plane waves up to the Fermi wave vector k_F and their densities add up to a constant, i.e. all Fourier coefficients of the density vanish except for $\mathbf{G} = 0$.

The implementation of the PAW method on real-space grid methods demands that the potential generation happens on a denser grid than the representation of the coarse wave functions, even though the local effective potential is finally applied on the coarse grid again. This is due to the multipole compensators that make up for the deficit of electrostatic multipole moments inside the augmentation sphere. These localized distribution functions require a much denser grid to be represented accurately. A strict localization inside the augmentation sphere is equivalent to arbitrarily high Fourier components. Their direct influence on the results is stronger than the influence of the localized atomic projector functions $|\tilde{p}^a\rangle$ since the multipole compensators enter the Poisson equation and, thus, the electrostatic potential. In order to keep the errors arising from these irrepresentably high frequency components small, we expand the multipole compensators on an $M\times$ denser grid with $M \geqslant 2$.

The cutoffs of the coarse grid can be exploited more efficiently by applying the interpolation to the wave function beforehand.

$$\tilde{n}_\sigma(\mathbf{r}_d) = \sum_{n\mathbf{k}} f_{n\sigma\mathbf{k}} \left| \sum_i w_{di} \Psi_{n\sigma\mathbf{k}}(\mathbf{r}_i) \right|^2 \tag{B.2}$$

where d is a grid index of an $M\times$ denser grid, i is a coarse grid index and w the interpolation operation connecting both grids. This way the density $\tilde{n}$ may contain frequency components up to $M\frac{\pi}{h}$. The normalization of the wave functions in the traditional scheme was simply given by

$$\sum_i |\Psi_{n\sigma\mathbf{k}}(\mathbf{r}_i)|^2 = 1 \tag{B.3}$$

and will thus change to a normalization on the dense grid, i.e.

$$1 = \sum_{d} \left| \sum_{i} w_{di} \Psi_{n\sigma\mathbf{k}}(\mathbf{r}_i) \right|^2 = \sum_{i_1 i_2} s_{i_1 i_2} \Psi^*_{n\sigma\mathbf{k}}(\mathbf{r}_{i_1}) \Psi_{n\sigma\mathbf{k}}(\mathbf{r}_{i_2}) \tag{B.4}$$

where the new overlap operator $\hat{s}$ is determined by the convolution of interpolation operators

$$s_{i_1 i_2} = \sum_{d} w_{di_1} w_{di_2}. \tag{B.5}$$

The complex conjugation was dropped here since usual interpolation weights are real-valued.

	matched		*between*	
M	s_0	$s_{\pm 1}$	s_0	$s_{\pm 1}$
1	1.000000	0.000000	0.500000	0.250000
2	0.750000	0.125000	0.625000	0.187500
3	0.703704	0.148148	0.648148	0.175926
4	0.687500	0.156250	0.656250	0.171875
6	0.675926	0.162037	0.662037	0.168981
8	0.671875	0.164062	0.664062	0.167969
10	0.670000	0.165000	0.665000	0.167500
12	0.668981	0.165509	0.665509	0.167245
16	0.667969	0.166016	0.666016	0.166992
⋮				
∞	0.666667	0.166667	0.666667	0.166667

Table B.1.: Weights of the grid point overlap operator $\hat{s}$ for 1D linear interpolation and various integer grid refinement factors M. The weights for $M = \infty$ are computed by analytical integration.

During the development of pseudopotential techniques and also later when the PAW method was invented, the main goal was to achieve a faster convergence of the results with respect to the number of degrees of freedom that span a representation of the smooth wave functions. This representation may be based on plane waves, local orbitals or real-space grids. An important step forward was introduced by loosening the norm-conservation constraint as in Vanderbuilt's US-PP and the PAW method. A consequence is that the KS eigenvalue equation is cast into a generalized eigenvalue problem. The general expression for the overlap operator is given as

$$\hat{S}_{\mathrm{PAW}} = \hat{s} + \sum_{aij} |\tilde{p}^a_i\rangle \Delta q^a_{ij} \langle \tilde{p}^a_j| \tag{B.6}$$

where for a traditional grid method $\hat{s}$ is diagonal $s_{i_1 i_2} = \delta_{i_1 i_2}$ in the grid points as implicitly given in Equation B.1. Here Δq_{ij}^{a} is the norm deficit due to the augmentation of the partial waves in the augmentation sphere of atom a and $|\tilde{p}^{a}\rangle$ are the localized atomic projector functions. The new overlap operator for grid points thus introduces only a modest overhead into the implementation of a real-space grid method using PAW.

B.0.5. Implementation of the non-local overlap operator

Data locality is the most essential ingredient to an efficient scaling behavior in massive parallelization. The real-space grid approach for DFT calculations provides an excellent way of exploiting data locality and short-sightedness of the operators can be introduced with only little approximations.

Besides the kinetic energy operator and the dyadic non-local projector operator $|\tilde{p}\rangle\langle\tilde{p}|$, the overlap operator $\hat{s}$ brings also non-locality into the equation. It relates grid points at a distance of two times the range of the interpolation scheme. We therefore choose the lowest order interpolation i.e. linear interpolation in between grid points in the coarse grid to find the wave function on the $M\times$ (usually $2\times$) denser grid. The new non-locality can be simply incorporated into the existing finite difference kinetic energy operator that requires non-process-local information. However, a minor change is needed. The finite difference operator only requires data from the six cartesian nearest neighbor processes, whereas $\hat{s}$ also relates grid points via the spatial diagonal, i.e. from all 27 surrounding processes. A way out of this extra expense in communication is to introduce a strict ordering of the communication operations in x, y, z-direction. We can thus let the information flow via the cartesian neighbors as explained on Section 5.3.1.

B.0.6. Choice of dense grid alignment

For an integer grid refinement we can choose from two highly symmetric ways to align the dense grid to the coarse grid. One alternative is to choose every M-th dense grid point to match in position with a coarse grid point an locate the others in between. The other way is to align the grids such that M dense grid points fall in between any two coarse grid points and each coarse grid points is centered between dense grid points. Table B.2 illustrates these two cases.

The difference between the two schemes becomes visible in the weights of $\hat{s}$ on a three-dimensional real-space grid. Table B.3 shows the elements of the overlap operator $\hat{s}$, s_{000}, s_{100}, s_{110} and s_{111}. We can see that the alignment *between* has a weaker on site weight s_{000} and, hence, more weight at the non-local elements compared to *matched*. For comparison, $M = \infty$ is shown here which results from one-

coarse grid	index i	*between*		1		2		3	
dense grid	index d		0	1	2	3	4	5	6
coarse grid	index i	*matched*		1		2		3	

Table B.2.: Two highly symmetric possibilities to align an M$\times$ denser grid to a coarse grid, shown for a grid refinement factor $M = 2$.

dimensional integrals. The one-site and off-site weights are found in between the two grid alignment schemes.

#			*matched* M=2	*between* M=2	M=∞
1	s_{000}	center	421.88	244.14	293.58
6	s_{100}	faces	70.31	73.24	73.39
12	s_{110}	edges	11.72	21.97	18.35
8	s_{111}	vertices	1.95	6.59	4.59

Table B.3.: 1000$\times$Weights of the grid point overlap operator $\hat{s}$ for linear interpolation in 3D. One can observe that the *matched* stencil shows a stronger on-site weights than for *between*. However, the *matched* grid alignment is disadvantageous when it comes to non-periodic boundaries. We therefore apply the *between* scheme. The weights for $M = \infty$ are products of the 1D-weights in Table B.1.

APPENDIX C

IMPLEMENTATION DETAILS

C.0.7. Bandstructure calculation

For a given (at best converged) local potential, jüRS can calculate the bandstructure. Therefore, a densely sampled path is generated passing along a set of predefined edge points in the Brillouin zone. The path edges are defined in the input file via the block keyword jüRS **Input Syntax:** kpath
followed by a list of edges specifying the three internal coordinates $\in [-\frac{1}{2}, \frac{1}{2}]$ and an optional label. A repetition of the same block keyword closed the list of **k**-edges. The calculation is parallelized over the available processes in σ**k**-parallelization. As an example, Figure C.1 shows the band structure of a hydrogen-saturated graphene ribbon $C_{14}H_2$ along the transport direction. A red dashed line indicates the Fermi level and a band gap of 4.8 *e*V at the Γ-point and 1 *e*V at the X-point.

C.1. Accuracy Benchmark

As a test of the accuracy of the implementation of jüRS, we performed two equivalent calculations with the implementation of the full-potential LAPW method FLEUR [20] and jüRS for graphene sheets with an interlayer separation of 6.7 Å and a bond length of 1.415 Å. jüRS is restricted to rectangular unit cells such that the smallest calculation contains four atoms which are symmetry equivalent. Although the LAPW code could exploit the symmetry better, the setup with four atoms per unit cell has been chosen for a simpler comparison. Figure C.2 shows the bands structures calculated with the two implementations (black jüRS, red FLEUR) in excellent agreement.

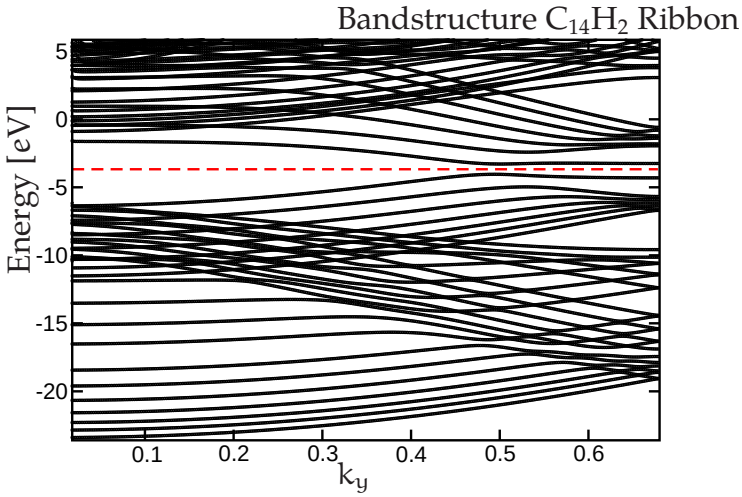


Figure C.1.: Bandstructure in y-direction of a graphene ribbon with an interlayer separation of $d = 6.7$ Å and a bond length of $a = 1.415$ Å, computed with the `PZ81` functional. The cell of this setup contains $C_{14}H_2$, so 16 atoms, and is strongly elongated in the x-direction: $22.64 \times 2.45 \times 6.7$ Å.

C.1.1. Fuzzy-Cell Decomposition

Performing molecular dynamics (MD) or structural relaxation with full self-consistent for each set of atomic positions is very expensive, and unfortunately neither the MD steps nor the SC cycles can be parallelized. However, information about the density can be reused as a better start guess density for the next iteration providing that the core did not move or relax much. This leads necessarily to a small time step and thus many MD steps, keeping the system close to self-consistency and thus electronically always in the ground state. The method proposed by Car and Parinello [38] does not iterate to find the electronic ground state at each MD step due to the underlying assumption that starting from a self-consistent electronic state an adiabatic movement of the ions will not excite the system electronically. This might hold in many cases, but is a potential risk. Applying a large time step in order to find, for example, the ground state structure by damped MD, we have to iterate in the SC density scheme. Therefore we need an educated way of reusing the density and maybe also wave functions of the last MD step as a better starting point. A useful but expensive way is to change from the uniform real space grid to a set of radial grids inside a certain radius that covers e.g. the nearest bond neighbors. This radial grid would represent the radial component of the function for different $Y_{\ell m}$ so $\rho_{\ell m}(r)$. In a sense, this basis is similar to those used in KKR [103, 104], except that a useful shape function would require a smoother transition from one Voronoi polyhedron to the next one than a well defined plane in space. One approach is the so called fuzzy-cell method [50]. Each spatial point is assigned weights $w^{\alpha}_{\text{fuzzy}}(\mathbf{r})$

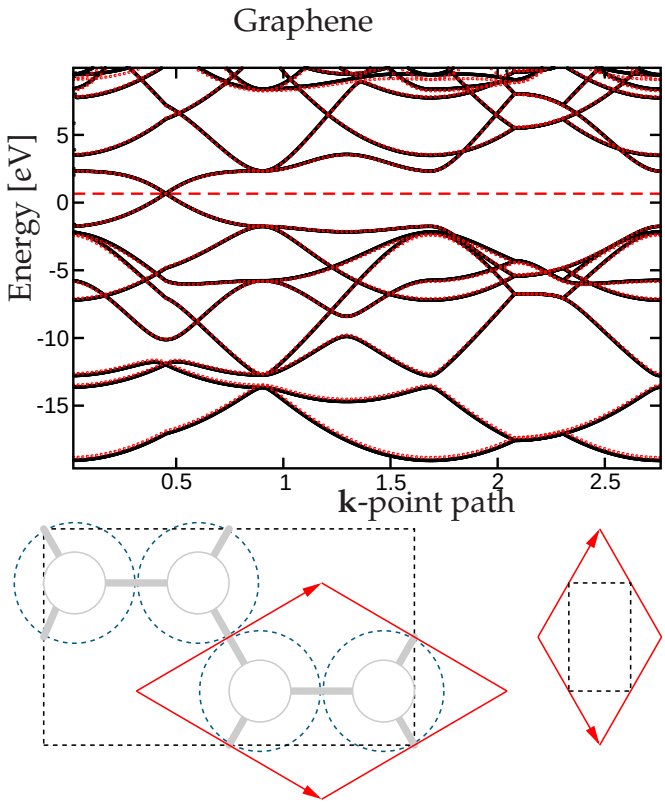


Figure C.2.: Structure and bandstructure of a graphene sheet with an interlayer separation of $d = 6.7$ Å and a bond length of $a = 1.415$ Å, computed with the `PZ81` functional. The smallest unit cell of this setup contains 2 atoms and has non-orthogonal unit vectors (indicated in red in the left pictogram). The structure shows the orthorhombic real-space cell of $4.25 \times 2.45 \times 6.7$ Å containing 4 atoms. The black bandstructure has been calculated with `jüRS` whereas the red bandstructure has been evaluated using the `FLEUR` code for comparison, an implementation of the full-potential linearized augmented plane-wave method [20]. Due to the doubled unit cell, folding of the reciprocal space leads to 16 occupied bands, twice as many as in the smallest unit cell.

of each atom a. One already sees, that this is not efficient in terms of memory and time for the setup, storing all weights explicitly. Depending on how the weight functions decay rate for larger distances from the atomic centers, only the nearest atoms will have a significant contribution to a grid point at $\mathbf{r}$. The smoothness of this weight function controls the smoothness of these mask functions. If one uses a sharp maximum function, such that each grid points only belongs to one atom, the Voronoi construction is recovered.

The transformed density on the radial grids thus is $\sum_a \rho^a_{\ell m}(|\mathbf{r} - \mathbf{R}^a|)\, w^a_{\text{fuzzy}}(\mathbf{r})$. We expect that this representation of the density is very suitable for having a good start guess after moving the atomic positions $\mathbf{R}^a$, if we just update $w^a_{\text{fuzzy}}(\mathbf{r})$ and keep the radial functions constant. Especially some kind of atomic occupations are preserved. Therefore, the cutoff in ℓ should at least include the highest occupied angular momentum of the valence state. The sampling of the radial components can be done in Bessel-space (c.f. q-space) or in real-space on an exponential grid which is good for the full density whereas for the smooth density, equidistant grid or even a grid that get coarser towards the origin are suitable.

Displacement interpolation

During structural relaxation, most of the force calculations are preformed with very similar atomic coordinates, i.e. the atoms move only little between to self-consistent calculations, especially close to convergence. An aim of the displacement interpolation is used to adopt a function in space to new atomic coordinates, knowing their displacement $\Delta\mathbf{R}^a$. This is particularly useful because using the old density (which is certainly better than atomic densities) as start density with new positions would lead to a p-like contribution of the density inside the atomic sphere. The p-like character arises from the displacement of the sphere with respect to the mostly spherical, s-like density. This can be seen by expanding the gradient operator in spherical harmonic functions. The p-like density would disturb the system such that it will require more self-consistency iterations until it converges again.

The local displacement field is found by

$$\mathbf{d}(\mathbf{r}) = \frac{\sum_a w^a(\mathbf{r})\Delta\mathbf{R}^a}{\sum_a w^a(\mathbf{r})} \tag{C.1}$$

where the weights $w^a_{\text{fuzzy}}(\mathbf{r})$ create a fuzzy mask defining the vicinity of each atom. Various shapes for this mask can be used. Gaussian and Weinert functions with the finite range $R^{\text{mask}}_{\text{max}}$ are implemented. Only points in space that are within $R^{\text{mask}}_{\text{max}}$ of the atomic position will have a non-zero weight for the associated displacement.

The new function is then evaluated at the displaced grid point positions using interpolation.

$$n_{\text{new}}(\mathbf{r}) = s \cdot \mathcal{I}\{n_{\text{old}}(\mathbf{r} - \mathbf{d}[\mathbf{r}])\} \tag{C.2}$$

with the scale factor s. The interpolation operator $\mathcal{I}$ is needed here because the displacements are of continuous length and direction so that they will most likely not match an integer multiple of the grid spacing. The implementation for this non-local operation is limited to a maximum displacement given by the maximal halo thickness (c.f. g%nh) times the grid spacing h (c.f. g%h) which again is limited by the sizes of the neigboring domains.

One can observe, how the density moves with the atoms that are closest. The density value of e.g. a covalent bond between two atoms will not be modified by this algorithm, even though we expect an increase when shrinking the atomic separation of the bond partners (which should than be the outcome of the next self-consistent density). Note that this algorithm is so far not norm-conserving. In order to find a density with the same norm, we could simply apply a global scale factor s such that

$$\int_V d^3\mathbf{r}\, n_{new}(\mathbf{r}) = \int_V d^3\mathbf{r}'\, n_{old}(\mathbf{r}') \tag{C.3}$$

The global character of this scaling is then equivalent to a non-local charge transfer that again would imply an unnecessary perturbation of the system. We therefore demand the scale factor to be local and fullfill the differential form of the equation

$$d^3\mathbf{r}\, n_{new}(\mathbf{r}) = d^3\mathbf{r}'\, n_{old}(\mathbf{r}') = d^3\mathbf{r}\, n_{old}(\mathbf{r} - \mathbf{d}(\mathbf{r})) \cdot \frac{d}{d\mathbf{r}} (\mathbf{r} - \mathbf{d}(\mathbf{r})) \tag{C.4}$$

where we can read off the local scaling factor $s(\mathbf{r})$ as

$$s(\mathbf{r}) = 1 - \operatorname{div} \mathbf{d}(\mathbf{r}). \tag{C.5}$$

To compute the divergence of the displacement field $\mathbf{d}(\mathbf{r})$ we weight the displacement vector of each atom with the atomic weights and normalize at each grid point $\mathbf{r}$:

$$\mathbf{d}[\mathbf{r}] = \frac{\sum_a w^a(\mathbf{r}) \Delta\mathbf{R}^a}{\sum_a w^a(\mathbf{r})}. \tag{C.6}$$

Then, the divergence of the displacement field is given by

$$\nabla \cdot \mathbf{d}(\mathbf{r}) = \frac{[\sum_a \nabla w^a(\mathbf{r}) \cdot \Delta\mathbf{R}^a] \, [\sum_a w^a(\mathbf{r})] - [\sum_a w^a(\mathbf{r}) \Delta\mathbf{R}^a] \cdot [\sum_a \nabla w^a(\mathbf{r})]}{[\sum_a w^a(\mathbf{r})]^2}. \tag{C.7}$$

For the weight function $w^a(\mathbf{r})$ we use a function that depends on the distance from the atom a only, i.e. $w^a(\mathbf{r}) = w(|\mathbf{r}^a|)$ where $\mathbf{r}^a = \mathbf{r} - \mathbf{R}^a$. The shape of $w(r)$ can be chosen differently and could, in principle also be varied depending on the species of atom a. However, for $w(r)$ we use either Weinert-like strictly localized function

$$w(r) = \left(1 - \frac{r^2}{R^2}\right)^{N_W} \qquad \text{for} \qquad |\mathbf{r}| < R \tag{C.8}$$

where the Weinert exponent N_W and the localization radius R can be customized according to the atomic species. The gradient expansion of $w^a(\mathbf{r})$ then reads

$$\nabla w^a(\mathbf{r}) = N_W \left(1 - \frac{\mathbf{r}\cdot\mathbf{r}}{R^2}\right)^{N_W-1} \left(-\frac{2\mathbf{r}}{R^2}\right) \tag{C.9}$$

Alternatively, we can use a strongly decaying Gaussian function for $w(r)$:

$$w(r) = \mathbf{e}^{-\frac{r^2}{2R^2}}, \tag{C.10}$$

$$\nabla w^a(\mathbf{r}) = w^a \left(-\frac{\mathbf{r}}{R^2}\right) \tag{C.11}$$

where R is chosen as 2.0 by default.

C.1.2. Band Reordering

During the application of the CG-method for solving the KS-equation, it is necessary to reorthogonalize the KS-state at hand to the states lower in energy. However, the changing effective potential can affect that two states may change their energy order. To keep the explicit Gram-Schmidt reorthogonalization simple, we want all states lower in energy to be stored in a continuous fashion which requires reordering of the states after the CG-band update. Wave functions, however, tend to have a rather large memory footprint, so we try to minimize the number of copying operations. The first step is to sort the given list of energy eigenvalues $[\epsilon]_{\text{unsorted}}$ and find $[\epsilon]_{\text{sorted}}$. During sorting, we keep the permutation information P, such that $P[\epsilon]_{\text{sorted}} = [\epsilon]_{\text{unsorted}}$. We then invert the permutation. The reordering process acting on the states requires a single temporary slot. The process starts off by copying the first element i in P^{-1} that is off-diagonal, i.e. $P^{-1}(i) \neq i$ and copies this to the temporary slot. Further, the memory position i is overwritten by the elements $P^{-1}(i)$ that needs to be moved there. The next index i is the given by the last memory position that has been copied, i.e. $P^{-1}(i)$ in this case. Simultaneously, a copy P' of the permutation P is modified the same way as the memory block of KS-states. This iteration continues until the n-fold permutation has been resolved. However, an unsorted list may contain more than one string of permutations, so that we have to restart at finding remaining off-diagonal entries in the modified permutation P'.

C.1.3. Combindices

Combindex stands for *combined index* and is an efficient method to address non-rectangular memory. An example is storing a triangularly shaped portion of memory as the upper triangular part of a symmetric/hermitian matrix that contains

only $\frac{1}{2}N(N+1)$ independent elements rather than N^2, where N is the matrix dimension. Combindices are used for several quantities related to the atomic setup, such as quantum numbers. Especially in the code parts where PAW quantities are addressed, many quantities depend on quantum numbers.

The inl-Combindex

The `inl` index combines the principle quantum number n that starts from 1 and the angular momentum quantum number $\ell \in [0, n-1]$, usually associated with the angular momentum characters $(s, p, d, f, \dots)$. Explicit values are listed in Table C.1.

```
inl=0
for n in 1...7
  for ell in 0...n-1
    inl++
```

	ℓ	0	1	2	3	4	5	6	…
n		s	p	d	f	g	h	i	…
1		1							
2		2	3						
3		4	5	6					
4		7	8	9	10				
5		11	12	13	14	15			
6		16	17	18	19	20	21		
7		22	23	24	25	26	27	28	
⋮									

Table C.1.: Combindex `inl` for indexing atomic orbitals of an isolated, spherically symmetric potential (m-degeneracy). `inl` $= \frac{1}{2}n(n-1)+\ell+1$

The ilm-Combindex

The `ilm` index combines the the angular momentum quantum number $\ell \in [0, n-1]$, usually associated with the angular momentum character $(s, p, d, f, \dots)$, with the so-called magnetic quantum number $m \in [-\ell, \ell]$. This index is used to address the spherical harmonic function $Y_{\ell m}$, especially their real-valued linear combinations. Explicit values are listed in Table C.2.

```
ilm=0
```

```
for ell in 0...6
  for emm in -ell...ell
    ilm++
```

ℓ	m	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
0	s							1						
1	p						2	3	4					
2	d					5	6	7	8	9				
3	f				10	11	12	13	14	15	16			
4	g			17	18	19	20	21	22	23	24	25		
5	h		26	27	28	29	30	31	32	33	34	35	36	
6	i	37	38	39	40	41	42	43	44	45	46	47	48	49
⋮														

Table C.2.: Combindex `ilm` for indexing $Y_{\ell m}$. `ilm` $= \ell^2 + \ell + m + 1$

The iln-Combindex and ilnm-Combindex

The `iln` index combines the the angular momentum quantum number $\ell \in [0, n-1]$, associated with the angular momentum character $(s, p, d, f, \dots)$, and the pseudo quantum number `enn`. `enn` counts the number of partial wave that are involved in the PAW-transformation. Since more than one partial waves are often associated to projectors of states higher in energy, `enn` > 1 is denoted by `enn`-1 stars $*$ following the angular momentum character. The maximum of `enn` is stored in `s%nn(ell)` and may differ for each `ell`, such that the `iln`-combindex will be generated by the following pseudo-code, but we cannot display a constant table here:

```
iln=0
for ell in 0...maxell
  for enn in 1...s%nn(ell)
    iln++
```

The `ilnm` extends the `iln` combindex by the `emm`-quantum number:

```
ilnm=0
for ell in 0...ellmax
  for enn in 1...s%nn(ell)
    for emm in -ell...ell
      ilnm++
```

C.1.4. LCAO Start Guess

The diagonalization of the Kohn-Sham Hamiltonian in a large basis representation as given in real-space is usually performed by iterative diagonalization schemes and is applied to the lowest states only. However, start vectors are required in any iterative method. Atomic orbitals from the radially symmetric all-electron calculation obviously are a simple and good choice since they show the correct asymptotic behavior at both, the origin and the tail. Furthermore, atomic orbitals are orthogonal if belonging to the same atomic center and exhibit a strong degree of localization in the region of the atom. The chance of two atomic orbitals on two different atoms leading to a linearly dependent set of start vectors is tiny. Now, a setup of initial guess wave functions that does not prefer any spatial direction requires a full shell of atomic orbitals on each atom hosts electrons of that corresponding ℓ-character, i.e. 2ℓ+1 orbitals. In many cases, the optimal number of states then is $N_{\text{bands}}^{\text{opt}} = \sum_a \sum_{\{\ell_a\}} 2\ell + 1$. A strong physical limitation to large calculations are the number of bands for two reasons. In jüRS, all KS wave functions are kept in memory during the computation such that we easily hit a hard limit given by memory. Solutions including temporary storage on the file system are thinkable, but will produce a lot of IO. Secondly, the scaling of the diagonalization methods is quadratic with respect the number of bands. Hence, we only want to work with the number of occupied states.

A cheap way to generate a set of start vectors that covers only the number of occupied states includes information about the structure of the system for the linear combinations of atomic orbitals. This means, the diagonalization in the 0th SCF iteration is performed in a basis of local orbitals - which is cheaper than the fully extended wave functions.

The ingredients to a LCAO step are described below.

Slater-Koster Integrals for an LCAO Start Guess

In an LCAO-type (linear combination of atomic orbitals) ansatz for finding a reasonable start guess wave functions, several convolution integrals of two functions given on a radial grid times a single spherical harmonic function which are not centered at the same origin are needed [105]. Let us assume the two function $f_0(\mathbf{r}) = f_0(|\mathbf{r}|)\, Y_{\ell_0 m_0}(\hat{\mathbf{r}})$ and $f_1(\mathbf{r}) = f_1(|\mathbf{R}-\mathbf{r}|)\, Y_{\ell_1 m_1}(\widehat{\mathbf{R}-\mathbf{r}})$

One way to compute these integrals is to sample to functions on a 3dim real space grid in that part of space where both functions are non-zero. A lot of computational effort can be avoided when we consider the displacement vector $\mathbf{R}$ only to be pointing in the z-direction of the coordinate system ($\mathbf{R} = R\,\mathbf{e}_z$). Then the integrals can be solved in cylindrical coordinates exploiting that the dependence on

the horizontal angle ϕ gives a $\delta_{m_0 m_1}$ (in both notations: real and complex spherical harmonics). Note that this δ does not come together with a $\delta_{\ell_0 \ell_1}$ since spherical harmonic functions are only orthogonal when there are centered at the same origin. The advantage of the cylindrical coordinates is that the integration left is now 2-dimensional:

$$\int d^3\mathbf{r}\, f_0(\mathbf{r}) f_1(\mathbf{r}) = 2\pi\delta_{m_0 m_1} \int_{z_{min}}^{z_{max}} dz \int_0^{r_{max}} dr_{xy}\, r_{xy}\, f_0(r_{xy}, z) f_1(r_{xy}, z) \tag{C.12}$$

with

$$f_i(r_{xy}, z) = f_i(r_i) Y_{\ell|m|}(r_{xy}, z_i), \qquad i \in \{0, 1\} \tag{C.13}$$

where $z_0 = z - \frac{1}{2}R$, $z_1 = z + \frac{1}{2}R$, $r_0 = \sqrt{r_{xy}^2 + (z - \frac{1}{2}R)^2}$ and $r_1 = \sqrt{r_{xy}^2 + (z + \frac{1}{2}R)^2}$. Now R is the length of the displacement vector. This integration can be executed easily by sampling a rectangular box of the two coordinates z and r_{xy} in the overlap region only. The memory requirement of this procedure is especially modest since the sampled functions do not need to be stored.

Now we have found an expression for the overlap integral of two atomic orbitals when the displacement between the two origins points along the z-direction in the definition of the spherical harmonic functions. Then only the following integrals have to be evaluated:

ℓ		0	1	1	2	2	2
	$\lvert m\rvert$	0	0	1	0	1	2
0	0	$s_0 s_0(R)$	$p_1 s_0(R)$		$d_2 s_0(R)$		
1	0	$s_0 p_1(R)$	$p_1 p_1(R)$		$d_2 p_1(R)$		
1	1			$p_0 p_0(R)$		$d_1 p_0(R)$	
2	0	$s_0 d_2(R)$	$p_1 d_2(R)$		$d_2 d_2(R)$		
2	1			$p_0 d_1(R)$		$d_1 d_1(R)$	
2	2						$d_0 d_0(R)$

Table C.3.: Non-zero Slater-Koster integrals to be evaluated in cylindrical coordinates. The indices $_{0,1,2}$ label the powers of the associated Legendre polynomials contained in the spherical harmonic functions and so equal $\ell - |m|$. R is the scalar distance between the two atomic centers.

In general, our displacement vector $\mathbf{R}$ will not point into the z-direction of the coordinate system. Therefore, we have to apply the Wigner rotation matrices as transformation for each ℓ-subspace of spherical harmonics. In addition, the analytical considerations above were made in terms of complex spherical harmonics whereas the main code uses real spherical harmonics. The indices of the real spherical harmonics will here be labelled with μ and complex with m, respectively. Then,

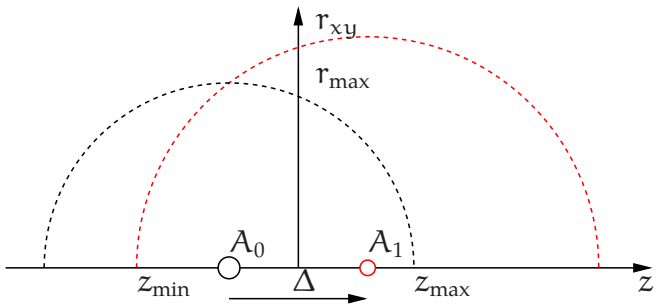


Figure C.3.: Two-center integral in cylindrical coordinates z and r_{xy}.

the overlap of atomic orbitals is given by

$$\langle f_0 \cdot Y_{\ell_0\mu_0}(0)|f_1 \cdot Y_{\ell_1\mu_1}(\mathbf{R})\rangle = \langle Y_{\ell_0\mu_0}|Y_{\ell_0 m_0}\rangle\langle Y_{\ell_0 m_0}|Y_{\ell_0 m_0'}(\hat{\mathbf{R}})\rangle \tag{C.14}$$

$$\times \langle f_0 \cdot Y_{\ell_0 m_0'}(0)|f_1 \cdot Y_{\ell_1 m_1'}(|\mathbf{R}|)\rangle \tag{C.15}$$

$$\times \langle Y_{\ell_1 m_1'}(\hat{\mathbf{R}})|Y_{\ell_1 m_1}\rangle\langle Y_{\ell_1 m_1}|Y_{\ell_1\mu_1}\rangle \tag{C.16}$$

where the indices m_0, m_0', m_1' and m_1 are contracted. The Wigner rotation matrix elements only depend on the direction of the displacement vector $\hat{\mathbf{R}}$ and the overlap integral in the rotated coordinate system only depends on the length $|\mathbf{R}|$. The transformation matrices $m \to \mu$ (complex to real) are constants (the same as used in Section C.1.8).

The integral in cylindrical coordinates favors the reduction of the complex spherical harmonics $r^\ell Y_{\ell m}$ to the following simplified spherical harmonics:

ℓ	m	ilm	Prefactor	Formula
0	0	1	$\sqrt{\frac{1}{4\pi}}$	1
1	-1	2	$\sqrt{\frac{3}{8\pi}}$	r_{xy}
1	0	3	$\sqrt{\frac{3}{4\pi}}$	z
1	1	4	$\sqrt{\frac{3}{8\pi}}$	$-r_{xy}$
2	-2	5	$\sqrt{\frac{15}{32\pi}}$	r_{xy}^2
2	-1	6	$\sqrt{\frac{15}{8\pi}}$	$r_{xy}z$
2	0	7	$\sqrt{\frac{5}{16\pi}}$	$2z^2 - r_{xy}^2$
2	1	8	$\sqrt{\frac{15}{8\pi}}$	$-r_{xy}z$
2	2	9	$\sqrt{\frac{15}{32\pi}}$	r_{xy}^2

Table C.4.: Simplified expressions of reduced $r^\ell Y_{\ell m}(\mathbf{r})$ in cylindrical coordinates z and r_{xy}.

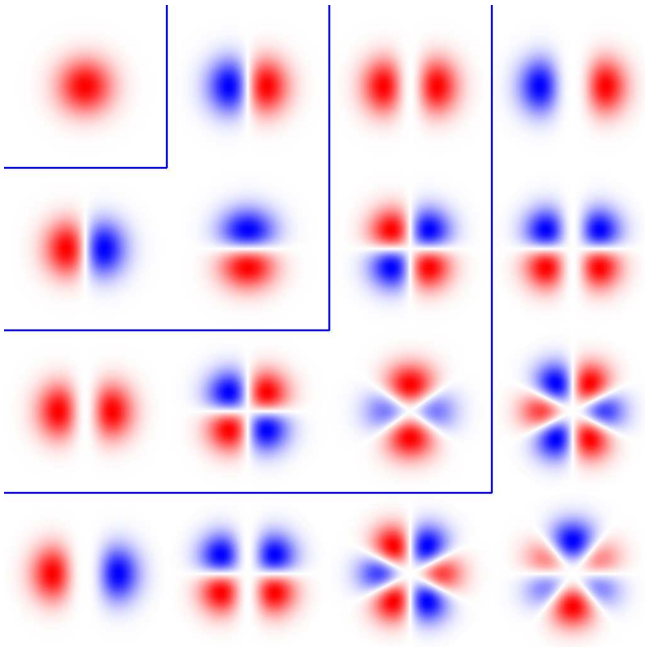


Figure C.4.: Expressions $r^{\ell}Y_{\ell m}$ in cylindrical coordinates r_{xy} and z (vertical). Although $r_{xy} > 0$, it is shown for both signs to demonstrate the symmetry. A Gaussian radial decay function and arbitrarily scaling have been employed here.

Gaussian overlap

A well suited basis for the radial functions are Gaussian functions $\mathbf{e}^{-k_i \mathbf{r}^2}$ since then, Equation (C.12) can be computed analytically and without r_{max} from the basis integrals

$$I_z = \int_{-\infty}^{\infty} dz\, \mathbf{e}^{-(k_0+k_1)z^2+R(k_0-k_1)z} z^{P_z} \tag{C.17}$$

and

$$I_r = \int_{0}^{\infty} dr\, \mathbf{e}^{-(k_0+k_1)r^2} r^{P_r+1} \tag{C.18}$$

with the integer polynomial powers $P_z, P_r \in [0, \ell_0 + \ell_1]$. The original integral reads

$$I = \iiint_{-\infty}^{\infty} dx\, dy\, dz\, \mathbf{e}^{-k_0(\mathbf{r}-\mathbf{R}_0)^2} [r_0^{\ell} Y_{L_0}](\mathbf{r}-\mathbf{R}_0) \cdot \mathbf{e}^{-k_1(\mathbf{r}-\mathbf{R}_1)^2} [r_1^{\ell} Y_{L_1}](\mathbf{r}-\mathbf{R}_1) \tag{C.19}$$

where we chose the origin of the coordinate system such that $\mathbf{R}_0 = -\frac{k_1}{k_0+k_1}\left(\mathbf{R}^1 - \mathbf{R}^0\right)$ and $\mathbf{R}_1 = \frac{k_0}{k_0+k_1}\left(\mathbf{R}^1 - \mathbf{R}^0\right)$. This effects that the linear term $k_0\mathbf{R}_0 + k_1\mathbf{R}_1$ in the expo-

nent vanishes. Then

$$I = \underbrace{\mathbf{e}^{-\frac{k_0 k_1}{k_0+k_1}(\mathbf{R}_1-\mathbf{R}_0)^2}}_{\text{decay factor}} \iiint_{-\infty}^{\infty} dx\, dy\, dz\, \mathbf{e}^{-(k_0+k_1)\mathbf{r}^2} [r_0^\ell Y_{L_0}](\mathbf{r}-\mathbf{R}_0) \cdot [r_1^\ell Y_{L_1}](\mathbf{r}-\mathbf{R}_1) \tag{C.20}$$

Now we are left with linear combinations of integrals of the type

$$\int dx\, \mathbf{e}^{-kx^2} x^p = \begin{cases} \sqrt{\frac{\pi}{k}} & p = 0 \\ 0 & p \text{ odd} \\ \frac{p-1}{2k} \int dx\, \mathbf{e}^{-kx^2} x^{p-2} & p > 0 \text{ and } p \text{ even} \end{cases} \tag{C.21}$$

C.1.5. APW Method of Soler & Williams

The implementation of the Linearized Augmented Plane Wave (LAPW) method by Soler and Williams [37] shows a striking similarity with the PAW method. Especially, a fast convergence of the results with respect the cutoff parameter ℓ_{cut} for the angular momentum expansion in the augmentation spheres is reported. A similar $\ell_{cut} = \ell_{phys.}+1$, as used in the PAW method, seems sufficient here. $\ell_{phys.}$ denotes the physical maximum ℓ, i.e. 0 for elements of group one in the periodic table, 1 for p-elements, 2 for transition metals and 3 for rare earth elements. The difference to conventional implementations of the LAPW method is that the plane waves are allowed to penetrate the spheres which brings this method closer to PAW in combination with plane waves (PW). Therefore, we discuss the action of the PAW transformation onto PWs as basis functions for the smooth waves.

PAW Transformation on Plane Waves

The PAW transformation has been defined by Blöchl [22] as

$$\mathcal{T} = \mathbf{1} + \sum_a \mathcal{T}^a = \mathbf{1} + \sum_{aLp} |\phi^a_{pL} - \tilde{\phi}^a_{pL}\rangle\langle \tilde{p}^a_{pL}|.$$

Applied to a PW $\mathbf{e}^{i\mathbf{K}\mathbf{r}}$, we find a set of basis functions χ for PW-PAW

$$\chi_{\mathbf{K}}(\mathbf{r}) = \frac{1}{\sqrt{\Omega}} \mathbf{e}^{i\mathbf{K}\mathbf{r}} + \sum_{aLp} c^a_{pL\mathbf{K}} \left(\phi^a_{p\ell\epsilon}(|\mathbf{r}^a|) - \tilde{\phi}^a_{p\ell}(|\mathbf{r}^a|) \right) Y_L(\hat{\mathbf{r}^a})$$

where $\mathbf{r}^a = \mathbf{r} - \mathbf{R}^a$ and the projection coefficients c with

$$c^a_{pL\mathbf{K}} = \langle \tilde{p}^a_{pL} | \mathbf{e}^{i\mathbf{K}\mathbf{r}} \rangle = 4\pi i^\ell \mathbf{e}^{i\mathbf{K}\mathbf{R}^a} Y^*_L(\hat{\mathbf{K}}) \int_0^{R^a} dr r^2\, j_\ell(|\mathbf{K}|r) \tilde{p}^a_{p\ell}(r)$$

Usually, the smooth partial waves $\tilde{\phi}^a_{p\ell}(r)$ are chosen according to $a_{1p} j_\ell(k_{1p} r) + a_{2p} j_\ell(k_{2p} r)$ where the four degrees of freedom are determined such that $\tilde{\phi}^a_{p\ell}(R^a)$ matches $\phi^a_{p\ell}(R^a)$ in value, first, second and third derivative. A usual choice for precise PAW calculations are two projectors per ℓ.

An alternative way of matching is assuming the smooth partial wave $\tilde{\phi}^a_{\ell|\mathbf{K}|} = j_\ell(|\mathbf{K}|r)$ and finding a true partial wave $\phi^a_{\ell|\mathbf{K}|\epsilon}$, with ϵ such that the logarithmic derivative coincides. The value can be match by simple scaling. In case the true spherical potential $V^a_0(r)$ that is applied in the outwards integration for $\phi^a_{\ell|\mathbf{K}|\epsilon}$ is aligned such that $V^a_0(R^a) = 0$ vanishes at the sphere boundary, ϵ should coincide with $\frac{1}{2}|\mathbf{K}|^2$. An approximation to simplify this procedure is the well known energy linearization, where we use the solution of the full spherical potential at a fixed energy parameter ϵ_0 and its energy derivative to find a linear combination of the two functions that matches the logarithmic derivative of the Bessel function.

In the original PAW formalism smooth partial waves, true partial waves and projector functions are not $|\mathbf{K}|$-dependent. However, following the ideas proposed above, we find an APW basis function

$$\chi_{\mathbf{K}}(\mathbf{r}) = \frac{1}{\sqrt{\Omega}} e^{i\mathbf{K}\mathbf{r}} + \sum_{aLp} \frac{4\pi}{\sqrt{\Omega}} i^\ell e^{i\mathbf{K}\mathbf{R}^a} Y^*_L(\hat{\mathbf{K}}) \left(\phi^a_\ell[\epsilon_\ell(|\mathbf{K}|)](|\mathbf{r}^a|) - j_\ell(|\mathbf{K}||\mathbf{r}^a|)\right) Y_L(\hat{\mathbf{r}^a})$$

where $\epsilon_\ell(|\mathbf{K}|)$ has to be determined such that the logarithmic derivative of the outwards integrated solution of the spherical potential $V^a_{\text{ref}}(r)$ matches the logarithmic derivative of the Bessel function $j_\ell(|\mathbf{K}|r)$ at $r = R^a$. Proper scaling of ϕ^a is assumed.

We introduce a set of basis of functions χ that consist each of one PW augmented with a local orbital (LO) which is tailor made for this PW, i.e.

$$\chi_{\mathbf{k}+\mathbf{G}}(\mathbf{r}) = e^{i(\mathbf{k}+\mathbf{G})\mathbf{r}} + \sum_{aL} \Phi^{aL}_{\mathbf{k}+\mathbf{G}}(\mathbf{r}) \tag{C.22}$$

where $\Phi^{aL}_{\mathbf{K}}(\mathbf{r}) = \sum_{p=0}^{2} c^{aLp}_{\mathbf{K}} u^{K}_{aLp}(|\mathbf{r}^a|) Y_L(\hat{\mathbf{r}}^a)$ with $K = |\mathbf{K}|$ and $\mathbf{r}^a = \mathbf{r} - \mathbf{R}^a$. The index $p \in [0, 2]$ samples three special radial functions. For p=1, $u^K_{aLp}(r)$ is the homogeneous solution $u^a_\ell(r)$ of the spherical potential in the sphere of atoms a, $u^K_{aL2}(r)$ is the corresponding energy derivative $\dot{u}^a_\ell(r)$ (inhomogeneous solution) and $u^K_{aL0}(r) = j_\ell(Kr)$ the spherical Bessel function that results from a Rayleigh expansion of the PW $e^{i\mathbf{K}\mathbf{r}}$. Note that $u^K_{aLp}(r)$ for p=1 and p=2, i.e. $u^a_\ell(r)$ and $\dot{u}^a_\ell(r)$, are actually independent of K and m (L is short for ℓm). The latter leads to the effect that all three radial functions are independent of m and thus we can write $u^K_{a\ell p}(r)$ instead of $u^K_{aLp}(r)$. Furthermore, $j_\ell(Kr)$ is in principle independent of the atom type of a, however in practice we might use radial support grids that are customized for each atom type.

We define the atomic sphere S^a by $|\mathbf{r} - \mathbf{R}^a| < R^a$ the position of the nucleus $\mathbf{R}^a$ and a sphere radius R^a chosen such that spheres show no spatial overlap. Now the

coefficients $c_{\mathbf{K}}^{aLp}$ are determined such that $\Phi_{\mathbf{K}}^{aL}(\mathbf{r})$ can be restricted to the sphere $\mathcal{S}^a$ and $\chi_{\mathbf{K}}(\mathbf{r})$ is continuous in value and first derivative. To achieve this, it is required that the value of the LO vanishes at the sphere boundary, i.e.

$$\sum_{p=0}^{2} c_{\mathbf{K}}^{aLp}\, u_{a\ell p}^{K}(r)\big|_{r=R^a} = 0 \tag{C.23}$$

and so the derivative with respect to the radius

$$\sum_{p=0}^{2} c_{\mathbf{K}}^{aLp} \frac{d}{dr}\, u_{a\ell p}^{K}(r)\big|_{r=R^a} = 0. \tag{C.24}$$

This can be ensured by the matching procedure that is standart in APW methods:

$$c_{\mathbf{K}}^{aLp=1} = 4\pi i^{\ell} e^{iKR^a} Y_L^*(\hat{\mathbf{K}}) \frac{1}{W} \left[\dot{u}_\ell(R^a) j_\ell'(KR^a) - \dot{u}_\ell'(R^a) j_\ell(KR^a)\right] \tag{C.25}$$

$$c_{\mathbf{K}}^{aLp=2} = -4\pi i^{\ell} e^{iKR^a} Y_L^*(\hat{\mathbf{K}}) \frac{1}{W} \left[u_\ell(R^a) j_\ell'(KR^a) - u_\ell'(R^a) j_\ell(KR^a)\right] \tag{C.26}$$

Here, W is the Wronskian determinant defined as

$$W = \dot{u}_\ell(R^a) u_\ell'(R^a) - u_\ell(R^a) \dot{u}_\ell'(R^a) \tag{C.27}$$

and the p=0-coefficient is given by

$$c_{\mathbf{K}}^{aLp=0} = -4\pi i^{\ell} e^{iKR^a} Y_L^*(\hat{\mathbf{K}}) \tag{C.28}$$

Observe the minus sign here which cancels out the plane wave parts in the sphere for $\ell \leqslant \ell_{cut}$. Optionally, we could further demand that the second radial derivative also vanishes at the sphere boundary. This would require that we take a third atomic solution $\ddot{u}_\ell(r)$ (p=3) into account.

Now regarding the overlap matrix $S_{ij} = \langle \chi_{\mathbf{K}_i} | \chi_{\mathbf{K}_j} \rangle$ we have to evaluate three different expressions: PW-PW, PW-LO and LO-LO. Let us start by PW-PW:

$$S_{ij}^{PW-PW} = \int_V d^3\mathbf{r}\; e^{i(\mathbf{K}_j - \mathbf{K}_i)\mathbf{r}} = V\, \delta_{ij}$$

Fairly simple are also the LO-LO terms

$$S_{ij}^{LO-LO} = \int_V d^3\mathbf{r} \sum_{aL} \sum_{a'L'} \Phi_{\mathbf{K}_i}^{a'L'}(\mathbf{r}) \Phi_{\mathbf{K}_j}^{aL}(\mathbf{r}) \tag{C.29}$$

The localization of the Φs to the non-overlapping atomic spheres gives us $\delta_{aa'}$. Being left with an integration over each sphere $\mathcal{S}^a$, we can switch to spherical coordinates

$$= \sum_{a} \sum_{LL'pp'} c_{\mathbf{K}_i}^{*aLp} c_{\mathbf{K}_j}^{aL'p'} \int_0^{R^a} dr\, r^2\, u_{aLp}^{K_i}(r) u_{aL'p'}^{K_j}(r) \int d\Omega\, Y_{L'}^*(\Omega) Y_L(\Omega) \tag{C.30}$$

which simplifies to

$$= \sum_{a} \sum_{Lpp'} c^{*aLp}_{\mathbf{K}_i} c^{aLp'}_{\mathbf{K}_j} \langle u_{K_j} | u_{K_i} \rangle^{a\ell}_{pp'} \delta_{mm'} \tag{C.31}$$

Where the inner part (p=1 and p=2) of the metric $\langle u_{K_j} | u_{K_i} \rangle^{a\ell}_{pp'}$ does not depend on Ks:

$$\langle u_{K_j} | u_{K_i} \rangle^{a\ell} = \begin{pmatrix} \langle j_\ell(K_j) | j_\ell(K_i) \rangle & \langle u^a_\ell | j_\ell(K_i) \rangle & \langle \dot{u}^a_\ell | j_\ell(K_i) \rangle \\ \langle j_\ell(K_j) | u^a_\ell \rangle & \langle u^a_\ell | u^a_\ell \rangle & \langle \dot{u}^a_\ell | u^a_\ell \rangle \\ \langle j_\ell(K_j) | \dot{u}^a_\ell \rangle & \langle u^a_\ell | \dot{u}^a_\ell \rangle & \langle \dot{u}^a_\ell | \dot{u}^a_\ell \rangle \end{pmatrix} \tag{C.32}$$

Note that the integral $\langle j_\ell(K_j) | j_\ell(K_i) \rangle$ depends in principle only on R^a.

Now let us consider the tricky part: PW-LO.

$$S^{PW-LO}_{ij} = \int_V d^3\mathbf{r}\, e^{-iK_i r} \sum_{aL} \Phi^{aL}_{\mathbf{K}_j}(\mathbf{r}) \tag{C.33}$$

Here again, we find the Rayleigh expansion for the PW and exploit the orthogonality of the spherical harmonics

$$= \sum_{aLp} 4\pi(-i)^\ell e^{-iK_i R^a} Y_L(\hat{\mathbf{K}}_i)\, c^{aLp}_{\mathbf{K}_j} \int_0^{R^a} dr\, r^2\, j_\ell(K_i r) u^{K_j}_{aLp}(r) \tag{C.34}$$

with the definition from above, we can identify our p=0-coefficient and the metric

$$= -\sum_{aLp} c^{*aLp}_{\mathbf{K}_i}\, c^{aL0}_{\mathbf{K}_j} \langle u_{K_j} | u_{K_i} \rangle^{a\ell}_{p0} \tag{C.35}$$

and thus the LO-PW term (PW and LO interchanged) is

$$S^{LO-PW}_{ij} = -\sum_{aLp} c^{aL0}_{\mathbf{K}_i}\, c^{*aLp}_{\mathbf{K}_j} \langle u_{K_j} | u_{K_i} \rangle^{a\ell}_{0p} \tag{C.36}$$

We can thus cancel out terms and find

$$S_{ij} = V\delta_{ij} + \sum_{aL} \left(\sum_{pp'=1}^{2} c^{*aLp}_{\mathbf{K}_i} c^{aLp'}_{\mathbf{K}_j} \langle u_{p\ell} | u_{p'\ell} \rangle_a - c^{*aL0}_{\mathbf{K}_i} c^{aL0}_{\mathbf{K}_j} \langle j_\ell(K_j) | j_\ell(K_i) \rangle_a \right) \tag{C.37}$$

The overlap integral of two spherical Bessel functions is given by

$$\mathcal{J}_\ell(q_1, q_2) = R_a^3 \int_0^1 dr\, r^2\, j_\ell(q_1 r)\, j_\ell(q_2 r) \tag{C.38}$$

$$\mathcal{J}_0(q_1, q_2) = R_a^3\, \frac{q_1\, q_2}{q_2^2 - q_1^2}\, (q_1 \cos(q_1) \sin(q_2) - q_2 \cos(q_2) \sin(q_1)) \tag{C.39}$$

$$\mathcal{J}_1(q_1, q_2) = R_a^3\, (\mathcal{J}_0(q_1, q_2) - \sin(q_1) \sin(q_2)) \tag{C.40}$$

$$\mathcal{J}_2(q_1, q_2) = R_a^3\, (\mathcal{J}_0(q_1, q_2) - 3\,(\cos(q_1) - \sin(q_1)/q_1)\,(\cos(q_2) - \sin(q_2)/q_2)) \tag{C.41}$$

where $q_1 = K_1 R_a$ and $q_2 = K_2 R_a$, respectively.

These expressions appear to be irregular for the limit $q_1 \to q_2$, but we can find simplified regular expressions for these cases:

$$\mathcal{J}_0(q, q) = R_a^3 \left(\frac{1}{2} q\, (q + \cos(q) \sin(q)) \right) \tag{C.42}$$

$$\mathcal{J}_1(q, q) = R_a^3 \left(\frac{1}{2} q\, (q + \cos(q) \sin(q)) - \sin^2(q) \right) \tag{C.43}$$

$$\mathcal{J}_2(q, q) = R_a^3 \left(\frac{1}{2} q\, (q + \cos(q) \sin(q)) - 3 \right) \tag{C.44}$$

$$\tag{C.45}$$

C.1.6. Convergence Acceleration

Instability of Metallic Systems

The SCF convergence of systems with a non-vanishing density of states at the Fermi level E_F may be difficult due to the instability arising from reoccupation. Let us assume a simpler system that is found by neglecting orbital relaxation, kinetic energy and the exchange correlation potential. We are left with the external potential (e.g. of atomic cores) and the Hartree potential. Let us further assume that the external potential has eigenstates with a high degeneracy close to the Fermi level. Then the Hartree potential is defined by

$$V_H[n](\mathbf{r}) = \int_V d^3\mathbf{r}'\, \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \qquad \text{(isolated)}, \tag{C.46}$$

$$V_H[n](\mathbf{G}) = \frac{4\pi}{\mathbf{G}^2} n(\mathbf{G}), \qquad |\mathbf{G}| > 0 \qquad \text{(periodic)} \tag{C.47}$$

where $\mathbf{G}$ are reciprocal lattice vectors. The solution of the KS-equation returns the eigenstates $\Psi_i(\mathbf{r})$ and eigenenergies E_i and the new density is found by

$$n(\mathbf{r}) = \sum_i f_i(E_i - E_F, k_B T) |\Psi_i(\mathbf{r})|^2. \tag{C.48}$$

Now let us assume that the density is self consistent. To analyze the stability, we can introduce a slight perturbation of any of these quantities, $V_H(\mathbf{r})$, $\Psi_i(\mathbf{r})$, E_i, f_i or $n(\mathbf{r})$. We will start by a change in the local potential $\Delta V_{pert}(\mathbf{r})$. In first order, the energy eigenvalues will change according to

$$\Delta E_i = \int d^3\mathbf{r}\, |\Psi_i(\mathbf{r})|^2\, \Delta V_{pert}(\mathbf{r}) \;+\; \mathcal{O}(\Delta^2) \tag{C.49}$$

so if the potential change is mostly positive (negative) where most of the state's weight is located, we will have a positive (negative) change of the eigenvalue, respectively. The change of the occupation numbers f_i is directly related to the change in the cooresponding energies, i.e.

$$\Delta f_i = \frac{df(E_i - E_F, k_B T)}{dE_i} \Delta E_i \;+\; \mathcal{O}(\Delta^2). \tag{C.50}$$

We neglected the first order relaxation of the orbitals $\Delta\Psi_i(\mathbf{r})$ here, since the change in density $\Delta n_r(\mathbf{r})$ that is induced by reoccupation (index $_r$) can be much stronger than the effect of orbital relaxation. We thus find a new density $n^{new}(\mathbf{r}) = n(\mathbf{r}) + \alpha\Delta n_r(\mathbf{r})$ where α is the straight mixing coefficient. The change of the Hartree potential is thus given by Equation C.46. The non-local Coulomb kernel $|\mathbf{r} - \mathbf{r}'|^{-1}$ is difficult to interpret in a real-space representation, however, we can say that values are high (low) for short (long) distances, respectively. So we are left with the coupled equations

$$\Delta n_r(\mathbf{r}) = \sum_i \Delta f_i |\Psi_i(\mathbf{r})|^2 \tag{C.51}$$

$$\approx \sum_i \frac{df(E_i - E_F, k_B T)}{dE_i} |\Psi_i(\mathbf{r})|^2\, \Delta E_i \tag{C.52}$$

and

$$\Delta E_i = \alpha \int_V d^3\mathbf{r} |\Psi_i(\mathbf{r})|^2 V_H\left[\Delta n_r\right](\mathbf{r}) \tag{C.53}$$

$$= \alpha \int_V d^3\mathbf{r} |\Psi_i(\mathbf{r})|^2 V_H\left[\sum_j \Delta f_j |\Psi_j|^2\right](\mathbf{r}) \tag{C.54}$$

We see that the change in density $\Delta n_r(\mathbf{r})$ due to reoccupation is present on both sides of the equation, whereas on the right side it is involved in the integration over $\mathbf{r}'$ for the Hartree potential. So for the non-periodic case this non-linear equation involves the integrals

$$C_{ij} = \iint_V d^3\mathbf{r} d^3\mathbf{r}' \frac{|\Psi_i(\mathbf{r})|^2 |\Psi_j(\mathbf{r}')|^2}{|\mathbf{r} - \mathbf{r}'|} \tag{C.55}$$

Further, we see that the coupling involves the derivative of the Fermi-Dirac distribution function which is negative and proportional to the inverse temperature $\beta = (k_B T)^{-1}$. The couplings therefore lead to faster re-occupations, the smaller the smearing temperature T is chosen.

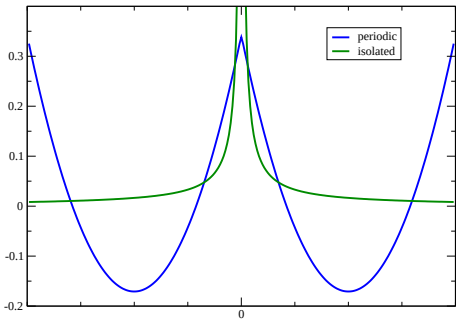


Figure C.5.: Bare Coulomb potential in a 1DEG for isolated and periodic boundary conditions.

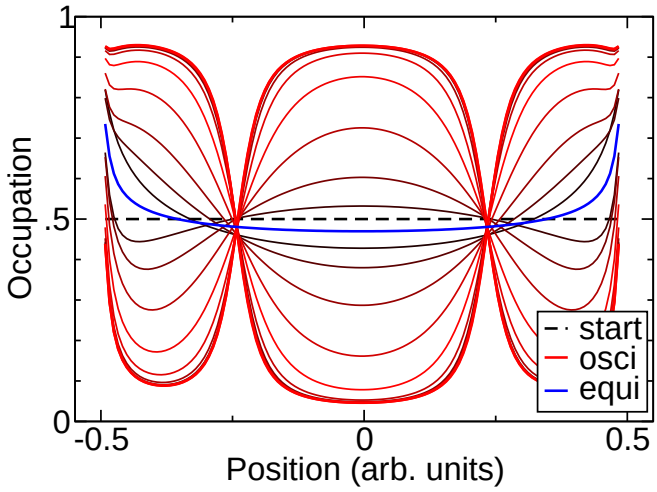


Figure C.6.: Simplified model for reoccupation dynamics. Starting from an even distribution of occupation $\frac{1}{2}$ (black dashed line), the undamped system tends to oscillate between two states (red solid lines). The damped system tends to the equilibrium solution shown in blue.

Density Mixing

There are several ways to lead the self-consistency cycle towards convergence. An important aspect certainly is the mixing of old and new potentials or densities. The simplest form herein is straight mixing of the (true) density

$$n_v^{\text{next}}(\mathbf{r}) = (1-\alpha) n_v^{(\text{in})}(\mathbf{r}) + \alpha n_v^{(\text{out})}(\mathbf{r}) \tag{C.56}$$

where in many situations the value of α needs to be kept very small in order to converge which also results in a huge number of steps needed until the residual falls below a given threshold. However, in practice we do not deal with the true density as a single quantity but rather the smooth density and the atomic density matrices D^a_{ij}

$$\tilde{n}_v^{(\text{next})}(\mathbf{r}) = (1-\alpha)\tilde{n}_v^{(\text{in})}(\mathbf{r}) + \alpha\tilde{n}_v^{(\text{out})}(\mathbf{r}) \qquad \forall \mathbf{r}, \tag{C.57}$$

$$D^a_{(\text{next})} = (1-\alpha)D^a_{(\text{in})} + \alpha D^a_{(\text{out})} \qquad \forall a. \tag{C.58}$$

Quasi-Newton methods for convergence acceleration

In higher order methods, inner products of density changes Δn are required. Also here, the true densities are important such that we need to define a metric for the smooth quantities $\tilde{n}_v(\mathbf{r})$ and D^a_{ij} that reproduces the inner product of true quantities:

$$\langle \Delta n_1 | \Delta n_2 \rangle = \int_V d^3\mathbf{r}\, \Delta n_1(\mathbf{r})\, \Delta n_2(\mathbf{r}) \tag{C.59}$$

$$= \int_V d^3\mathbf{r}\, \Delta \tilde{n}_1(\mathbf{r})\, \Delta \tilde{n}_2(\mathbf{r}) + \sum_{aijkl} \Delta D^a_{1,ij} \Delta D^a_{2,kl} \left(Q^a_{ijkl} - \tilde{Q}^a_{ijkl} \right) \tag{C.60}$$

$$Q^a_{ijkl} = \int_{S^a} d^3\mathbf{r}\, \phi^a_i(\mathbf{r})\phi^a_j(\mathbf{r})\, \phi^a_k(\mathbf{r})\phi^a_l(\mathbf{r}) \tag{C.61}$$

$$\tilde{Q}^a_{ijkl} = \int_{S^a} d^3\mathbf{r}\, \tilde{\phi}^a_i(\mathbf{r})\tilde{\phi}^a_j(\mathbf{r})\, \tilde{\phi}^a_k(\mathbf{r})\tilde{\phi}^a_l(\mathbf{r}) \tag{C.62}$$

Broyden mixing scheme

The combination of the iterative self-consistency cycles with the iterative and approximate solution of the eigenstates of the Kohn-Sham Hamiltonian leads to a very sensitive interplay in the convergence behavior of the SC convergence. Starting from a superposition of atomic densities and atomic orbitals as start wave functions, it is obvious that these states are no eigenstates of the Hamiltonian with the effective potential generated from the start density. In fact, we can find an LCAO-type Hamiltonian in this first cycle as described in more detail in Section 4.4.3. The number of self consistency iterations needed to converge the density residual depends strongly on the method of density mixing. One can also mix the potentials in a similar fashion but we will describe density mixing in this text. The simplest method is straight (simple) mixing. The density of the next iteration $i+1$ is a combination of the input density of iteration i and the output density of iteration i

$$n^{\text{in}}_{[i+1]}(\mathbf{r}) = n^{\text{in}}_{[i]}(\mathbf{r}) + \alpha \left(n^{\text{out}}_{[i]}(\mathbf{r}) - n^{\text{in}}_{[i]}(\mathbf{r}) \right). \tag{C.63}$$

Straight mixing can only achieve a linear convergence behavior, i.e. the decay of the residual $R_{[i]}$ defined by

$$R^2_{[i]} = \frac{1}{V} \int d^3\mathbf{r} \left(n^{out}_{[i]}(\mathbf{r}) - n^{in}_{[i]}(\mathbf{r}) \right)^2 \tag{C.64}$$

is limited by α according to $R_{[i+1]} \geqslant (1-\alpha)\, R_{[i]}$. The residual is shown by the dashed lines in Figure C.7 for an α of 1 %, 5 %, 10 %, 20 % and 30 %. In this semi-logarithmic plot the linear convergence shows straigt lines converging faster for larger values of α. However, there is a critical, system dependent mixing parameter α^{crit}. Exceeding $\alpha > \alpha^{crit}$ will decelerate the convergence as visible for $\alpha = 50\,\%$ or might even spoil the convergence completely as for $\alpha = 80\,\%$ (not shown). The dependence of α^{crit} of the system is mostly governed by the density of states at the Fermi level, E_F, and the degree of localization of the states around E_F. This situation is discussed in detail in Section C.1.6.

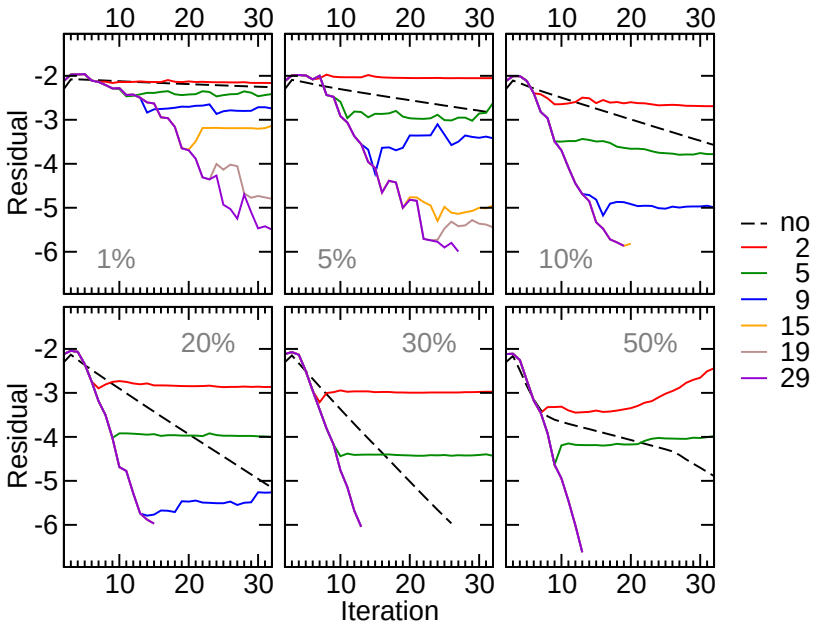


Figure C.7.: Reduction of the residuals of the density over the SCF cycles for various mixing parameters α. The test system is an insulating Oligothiophene molecule, $S_2C_8H_8$. Black dashed lines show the linear mixing scheme. Solid lines are color-coded according to their number of Broyden history steps.

The class of second order methods promises an improved convergence behavior in the quadratic region, i.e. close to the converged solution [106]. In principle, second order methods should lead to a quadratic convergence. However, this is not the case since the derivation of these methods assumes that the functional to be minimized possesses a multi-dimensional quadratic minimum. For any den-

sity that is not the converged ground-state density, the residual vector is thus assumed to be a linear function of the response vector. Due the the non-linearity in the exchange-correlation potential and reoccupation of orbitals, this does not hold. However, the linearization is a better approximation the closer the system is to convergence. Second order methods are based on constructing a history of changes of the density over the previous SCF cycles. Each cycle extends the information about the total energy surface in the phase space of all possible densities a bit further. An approximate Jacobian of the density response function is computed in a finite subspace, i.e. the densities of the previous cycles span the explored portion of the phase space. In order to avoid IO-operations `jüRS` keeps the previous densities in memory. This, however, requires an limiting maximum of history steps, H^{max}. Figure C.7 shows the reduction of the residual for the mixing parameters $\alpha \in \{1\,\%, 5\,\%, 10\,\%, 20\,\%, 30\,\%, 50\,\%\}$ and $H^{max} \in \{2, 5, 9, 15, 19, 29\}$, colored red, green blue, orange, brown and violet, respectively. At short histories we can observe that the residual does not decay any further after the iteration number has passed 4+H^{max}. This becomes very clear for H^{max} of 2 and 5 in the range 20 % to 30 %. The offset of four iterations comes from the first iteration that does not involve density mixing and three initial iterations of straight mixing. Comparing only the cases of unreached history limitations (violet) we may observe that too small values for α lead to a non-monotonous and slower convergence. This might be caused by a deterioration of the signal-to-noise ratio of the response $n_{[out]} - n_{[in]}$ when multiplied with values as small as 1 %.

`jüRS` **Input Syntax:** `mixing 0.4 Broyden 19`

The SCF convergence within the Broyden mixing scheme [107, 108] is best for our test system at $\alpha = 50\,\%$ and an unreached history limitation $H^{max} \geqslant 9$ since convergence ($R_{[i]} < 10^{-6}\, e/\text{Bohr}^3$) is reached after 13 SCF iterations.

The aforementioned approximate Jacobian needs to be inverted in order to guess the next density change such that we get as close to the self-consistent solution as possible with the given information about the total-energy surface. For this Jacobian it is of major importance that the density response function $n^{out} - n^{in}$ as a function of n^{in} is accurate. The output density is generated from the eigenstates of the Hamiltonian which are computed with an iterative method. We thus have to find a reasonable criterion for the quality of the eigenstates that is required to have a sufficiently accurate response function. Too large errors in the response function will lead to a erroneous and probably even singular Jacobian that will let the SCF cycles diverge rather than converge.

Preconditioning

The DIIS band update method optimizes the eigenvectors by adding a linear combination of resdual vectors. For a faster convergence of this method, preconditioning of the residual vectors is useful. The best preconditioner is, of course, the inverse of the problem. However, this is obviously not a practical ansatz. We want to find a preconditioner which is independent of the problem and whose application is cheap. With respect to the parallelization of the real-space grid in domain decomposition, only a localized preconditioner is leads to favorable scaling. A very simple and cheap precondition of this kind is given by two weights only, an on-site weight of $\frac{1}{2}$ and an off-site weight of $\frac{1}{12}$ for each of the six cartesian nearest-neighbor grid point. Figure C.8 shows the transmission of this preconditioner in terms of frequencies. The transmission for low frequencies is unity and decays quadratically. The maximum frequency in each direction, (100), (110) and (111), which can be represented on an equidistant grid with grid spacing h is $\frac{\pi}{h}$, $\frac{\pi}{h}\sqrt{2}$ and $\frac{\pi}{h}\sqrt{3}$, respectively. The transmission functions for these values assumes $\frac{2}{3}$, $\frac{1}{3}$ and 0, respectively. This means, that higher frequencies are suppressed than lower ones.

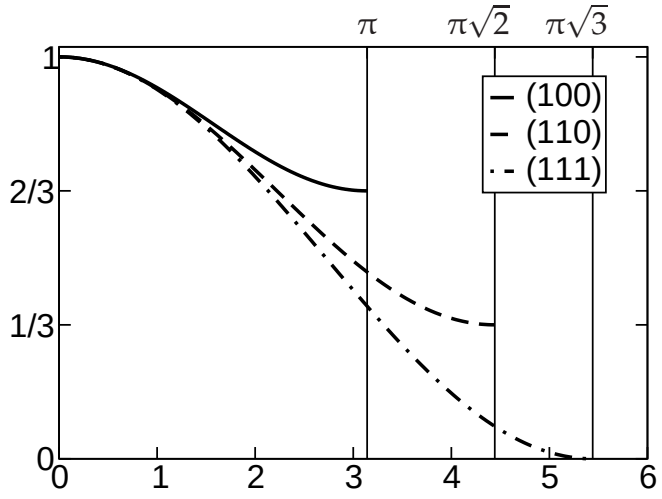


Figure C.8.: Transmission of the sparse nearest-neighbor preconditioner.

C.1.7. Localized Functions

The PAW formalism and the pseudocharge construction lead to two kinds on localized functions, atomic projector functions and multipole compensator functions. Localized functions $F(\mathbf{r})$ in a sphere can always be expanded as a radial functions times spherical harmonics, i.e.

$$F(\mathbf{r}) = \sum_{\ell m} f_{\ell m}(|\mathbf{r}-\mathbf{R}|) Y_{\ell m}(\widehat{\mathbf{r}-\mathbf{R}}) \tag{C.65}$$

where $\mathbf{R}$ denotes the sphere center.

In the special case of projector functions that are generated from a spherically symmetric pre-calculation, and for the compensation charges, we can drop the m-index at the radial function f_ℓ due to m-degeneracy. The aim of projector function is to project only onto one ℓ-symmetry and thus the expression reduces to a radial function times a full shell of spherical harmonic functions belonging to that particular ℓ.

The spherical harmonics $Y_{\ell m}(\hat{\mathbf{r}})$ depend on sin- and cos-functions of the two angles ϑ and φ. However, the evaluation of inverse trigonometric functions is required to find ϑ and φ from the components of the normalized vector $\hat{\mathbf{r}}$. The quantity $|\mathbf{r}|^\ell Y_{\ell m}(\hat{\mathbf{r}})$ on the other hand contains only powers of the vector components of $\mathbf{r}$ and does not even require normalization by the vector length. For speeding up the calculation, several division operations and evaluations of inverse trigonometric functions can be avoided, if a separated representation is introduced as

$$F(\mathbf{r}) = \sum_{\ell m} \left[f_\ell r^{-\ell}\right] (|\mathbf{r}-\mathbf{R}|)[r^\ell Y_{\ell m}](\mathbf{r}-\mathbf{R}). \tag{C.66}$$

The function `Ylmax_rl(ellmax, v)` of the module `harmonics` returns the proper values for ℓ less than or equal `ELLMAX_IMPLEMENTED`. Then all radial functions have to be divided by the factor r^ℓ. This procedure can only take place, if the radial function stays regular at the origin. However this is the case for all projector functions, since these behave as r^ℓ in the limit $r \to 0$. The same applies for the true and smooth partial waves, whereas the smooth partial waves are only represented on to the cartesian grid for start guess wave functions. True partial waves can never be represented properly on coarse real-space grids.

Derivative of localized functions

The evaluation of the force acting on the nuclei requires the computation of the gradient with respect to the atomic origin $\mathbf{R}^a$ of several quantities. Especially the localized function are to be derived. Assume a localized function $F(\mathbf{r})$ consisting of a radial function f_ℓ and one single spherical harmonic $Y_{\ell m}$.

$$\frac{\partial F(\mathbf{r})}{\partial \mathbf{R}} = \frac{\partial}{\partial \mathbf{R}} \left(f_\ell(|\mathbf{r}-\mathbf{R}|) Y_{\ell m}(\widehat{\mathbf{r}-\mathbf{R}}) \right) \tag{C.67}$$

$$= \frac{\partial}{\partial \mathbf{R}} \left([f_\ell r^{-\ell}](|\mathbf{r}-\mathbf{R}|)[r^\ell Y_{\ell m}](\mathbf{r}-\mathbf{R}) \right) \tag{C.68}$$

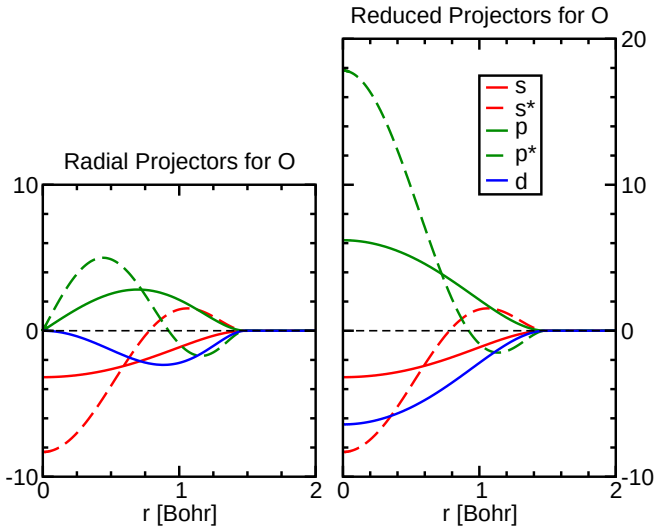


Figure C.9.: Example for the reduction of localized radial projector functions to $[f_\ell r^{-\ell}]$ for Oxygen configured `element O 2s* 2 2p* 2 3d | 1.3`

separation as defined in Section C.1.7

$$\frac{\partial F(\mathbf{r})}{\partial \mathbf{R}} = \frac{\partial [f_\ell r^{-\ell}](r)}{\partial r} \cdot \frac{\partial r}{\partial \mathbf{R}} \cdot [r^\ell Y_{\ell m}](\mathbf{r}-\mathbf{R}) + [f_\ell r^{-\ell}](r) \cdot \frac{\partial [r^\ell Y_{\ell m}](\mathbf{r})}{\partial \mathbf{r}} \cdot \frac{\partial (\mathbf{r}-\mathbf{R})}{\partial \mathbf{R}} \tag{C.69}$$

$$= -[f_\ell r^{-\ell}]'(|\mathbf{r}-\mathbf{R}|) \cdot \frac{\mathbf{r}-\mathbf{R}}{|\mathbf{r}-\mathbf{R}|} \cdot [r^\ell Y_{\ell m}](\mathbf{r}-\mathbf{R}) - [f_\ell r^{-\ell}](|\mathbf{r}-\mathbf{R}|) \cdot [r^\ell Y_{\ell m}]'(\mathbf{r}-\mathbf{R}) \tag{C.70}$$

exploiting that

$$\frac{\partial r}{\partial \mathbf{R}} = \frac{\partial |\mathbf{r}-\mathbf{R}|}{\partial \mathbf{R}} = -\frac{\mathbf{r}-\mathbf{R}}{|\mathbf{r}-\mathbf{R}|}. \tag{C.71}$$

The derived spherical harmonics $[r^\ell Y_{\ell m}]'(\mathbf{r})$ are again three simple expressions of the vector components of $\mathbf{r}$. The function `d_Ylmax_rl_dri(ellmax, v, derive2i)` of the module `harmonics` returns the corresponding values in analogy to the functions introduced in C.1.7.

Implementation of spherical harmonics

The expression $r^\ell Y_{\ell m}$ of the real spherical harmonics are regular expressions of the power ℓ in the components of $\mathbf{r} = [x, y, z]$. Replacing $\cos(\theta)$ by $\frac{z}{r}$, $\sin(\theta)\cos(\varphi)$ by $\frac{x}{r}$ and $\sin(\theta)\sin(\varphi)$ by $\frac{y}{r}$ we get the expression for the real spherical harmonic functions shown in Table C.5.

ℓ	m	ilm	Prefactor	Formula	Name
0	0	1	$\sqrt{\frac{1}{4\pi}}$	1	s
1	-1	2	$\sqrt{\frac{3}{4\pi}}$	x	px
1	0	3	$\sqrt{\frac{3}{4\pi}}$	z	pz
1	1	4	$\sqrt{\frac{3}{4\pi}}$	y	py
2	-2	5	$\sqrt{\frac{15}{16\pi}}$	$x^2 - y^2$	eg
2	-1	6	$\sqrt{\frac{15}{4\pi}}$	zx	t2g
2	0	7	$\sqrt{\frac{5}{16\pi}}$	$3z^2 - r^2$	eg
2	1	8	$\sqrt{\frac{15}{4\pi}}$	yz	t2g
2	2	9	$\sqrt{\frac{15}{4\pi}}$	xy	t2g
3	-3	10	$\sqrt{\frac{35}{32\pi}}$	$x(x^2 - 3y^2)$	f
3	-2	11	$\sqrt{\frac{105}{16\pi}}$	$z(x^2 - y^2)$	f
3	-1	12	$\sqrt{\frac{21}{32\pi}}$	$x(5z^2 - r^2)$	f
3	0	13	$\sqrt{\frac{7}{16\pi}}$	$z(5z^2 - 3r^2)$	f
3	1	14	$\sqrt{\frac{21}{32\pi}}$	$y(5z^2 - r^2)$	f
3	2	15	$\sqrt{\frac{105}{4\pi}}$	xyz	f
3	3	16	$\sqrt{\frac{35}{32\pi}}$	$y(3x^2 - y^2)$	f

Table C.5.: Simplified expressions of $Y_{\ell m} r^{\ell}(\mathbf{r})$

This way of implementing the spherical harmonic functions is advantageous in terms of computational execution speed and simplicity of the force implementation: forces usually involve the three spatial derivatives of localized functions such as the projector functions in the PAW method. The implementation of $r^\ell Y_{\ell m}(\mathbf{r})$ in powers of the vector components of $\mathbf{r}$ makes it particularly simple to derive the the cartesian coordinates.

ℓ	m	ilm	Prefactor	$\frac{\partial}{\partial x}$	$\frac{\partial}{\partial y}$	$\frac{\partial}{\partial z}$
0	0	1	$\sqrt{\frac{1}{4\pi}}$			
1	-1	2	$\sqrt{\frac{3}{4\pi}}$	1		
1	0	3	$\sqrt{\frac{3}{4\pi}}$			1
1	1	4	$\sqrt{\frac{3}{4\pi}}$		1	
2	-2	5	$\sqrt{\frac{15}{16\pi}}$	$2x$	$-2y$	
2	-1	6	$\sqrt{\frac{15}{4\pi}}$	z		x
2	0	7	$\sqrt{\frac{5}{16\pi}}$	$-2x$	$-2y$	$4z$
2	1	8	$\sqrt{\frac{15}{4\pi}}$		z	y
2	2	9	$\sqrt{\frac{15}{4\pi}}$	y	x	

Table C.6.: Simplified expressions of $\frac{\partial}{\partial \mathbf{r}}[Y_{\ell m} r^\ell](\mathbf{r})$

C.1.8. Gaunt Coefficients

The Gaunt-coefficients allow to express the products of two spherical harmonic functions defined on the same angles (ϑ, φ) in the basis of single spherical harmonic functions again. Let us assume a function $f(\vartheta, \varphi)$ that is a product of only two spherical harmonic functions $Y_{\ell_1 m_1}(\vartheta, \varphi)$ and $Y_{\ell_2 m_2}(\vartheta, \varphi)$ and for simplicity choose the prefactor to be unity:

$$f(\vartheta, \varphi) = Y_{\ell_1 m_1}(\vartheta, \varphi) \cdot Y_{\ell_2 m_2}(\vartheta, \varphi) \tag{C.72}$$

then the expansion of this function f in spherical harmonics reads

$$f(\vartheta, \varphi) = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} f_{\ell m}\, Y_{\ell m}(\vartheta, \varphi) \tag{C.73}$$

with the expansion coefficients

$$f_{\ell m} = \langle Y_{\ell m} | Y_{\ell_1 m_1} \cdot Y_{\ell_2 m_2} \rangle \tag{C.74}$$

$$= \int_0^{2\pi} d\varphi \int_0^{\pi} \sin\vartheta d\vartheta \; Y^*_{\ell m}(\vartheta,\varphi) Y_{\ell_1 m_1}(\vartheta,\varphi) Y_{\ell_2 m_2}(\vartheta,\varphi) \tag{C.75}$$

$$:= G\left(\ell_1, m_1, \ell_2, m_2; \ell, m\right)_{\text{complex}} \tag{C.76}$$

With the definition of the complex spherical harmonics

$$Y_{\ell m}(\vartheta,\varphi) = \sqrt{\frac{(2\ell+1)}{4\pi}\frac{(\ell-m)!}{(\ell+m)!}}\frac{(-1)^m}{2^\ell \ell!} e^{im\varphi}(x^2-1)^{\frac{m}{2}}\frac{d^{\ell+m}}{dx^{\ell+m}}(x^2-1)^\ell \tag{C.77}$$

where $x = \cos\vartheta$, the φ-integral can be solved separately:

$$\int_0^{2\pi} d\varphi \; e^{-im\varphi} \; e^{im_1\varphi} \; e^{im_2\varphi} = 2\pi \, \delta_{m,m_1+m_2}. \tag{C.78}$$

This means that all Gaunt coefficients vanish, unless $m = m_1 + m_2$. Furthermore, this simplifies the ϑ-integral since then, disregarding prefactors, only the following has to be solved

$$\int_{-1}^{1} dx \; (x^2-1)^{\frac{m+m_1+m_2}{2}} \; d_{\ell m}(x) \; d_{\ell_1 m_1}(x) \; d_{\ell_2 m_2}(x) \qquad \text{with} \tag{C.79}$$

$$d_{\ell m}(x) = \frac{d^{\ell+m}}{dx^{\ell+m}}\left\{(x^2-1)^\ell\right\} \tag{C.80}$$

Saving of computational resources is strongly enhanced by treating the real-valued linear combinations of the spherical harmonics as a basis set. The basis transformation that leads from complex-valued spherical harmonics $Y_{\ell m}$ to real ones (here denoted as $X_{\ell\mu}$, but throughout this document, the real spherical harmonics are denotes as $Y_{\ell m}$) relates only basis functions within the same ℓ-value, i.e. it is (block) diagonal because of the orthogonality of the associated Legendre polynomials. The full unitary transformation $U^{\ell,\ell'}_{m,m'}$ can thus be simplified to $U^{\ell}_{\mu,m'}\delta_{\ell,\ell'}$, so that

$$X_{\ell m} = \sum_{m'} U^{\ell}_{\mu,m'} Y_{\ell m'}$$

A possible choice for this transformation is proposed by Homeier *et al.* [109] shown here for the first three ℓ-values.

$$U^{\ell=0} = 1, \; U^{\ell=1} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 & i \\ 0 & \sqrt{2} & 0 \\ -1 & 0 & i \end{pmatrix}, \; U^{\ell=2} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 & 0 & 0 & i \\ 0 & 1 & 0 & i & 0 \\ 0 & 0 & \sqrt{2} & 0 & 0 \\ 0 & -1 & 0 & i & 0 \\ 1 & 0 & 0 & 0 & -i \end{pmatrix}$$

Then the Gaunt coefficient relating the real spherical harmonics $X_{\ell\mu}$ can be found by the transformed tensor

$$G\left(\ell_1,\mu_1,\ell_2,\mu_2;\ell,\mu\right)_r = \sum_{m,m_1,m_2} [U^{\ell}_{\mu,m}]^* U^{\ell_1}_{\mu_1,m_1} U^{\ell_2}_{\mu_2,m_2}\, G\left(\ell_1,m_1,\ell_2,m_2;\ell,m\right)_c \quad \text{(C.81)}$$

where $_r$ and $_c$ denote the real and complex Gaunt coefficients.

C.1.9. Density of States

Computation of Occupation Numbers

The new density is calculated from the eigenstates of the Hamiltonian. The Kohn-Sham states are occupied with electrons according to the Fermi-Dirac (FD) distribution f_{FD} which is a function of the Kohn-Sham energies E_i (i labels all states i.e. all bands, spins and $\mathbf{k}$-points) and the Fermi energy E_F which is unknown. The occupation numbers f_{FD} are defined as

$$f_{FD}(E - E_F, T_F) = \frac{1}{\mathbf{e}^{\beta(E-E_F)} + 1} \quad \text{(C.82)}$$

where $\beta = (k_B T_F)^{-1}$ corresponds to a fictitious smearing temperature T_F that will be explained below. The sum of all occupation number has to be equal to the number of electrons in the system, N_{ele}. This is achieved by tuning the Fermi level E_F until

$$\sum_i f_{FD}(E_i - E_F, T_F) = \sum_i \frac{1}{\mathbf{e}^{\beta(E_i-E_F)} + 1} \stackrel{!}{=} N_{ele} \quad \text{(C.83)}$$

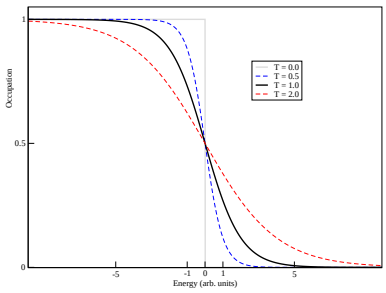
We have to find a null in the function $g(E_F) = \sum_i f_{FD}(E_i, E_F) - N_{ele}$ with respect to E_F.

The null of a function $g(e)$ can be found applying the bisection method. This is based on a search interval which is iteratively divided into two smaller intervals. If the function g is monotonous in the start interval $[e_1, e_2]$ and has a null, the method is supposed to always converge. One iteration of the bisection performs the following steps.

1. evaluate $g(e_1) = g_1$ and $g(e_2) = g_2$. The start results g_1 and g_2 are of opposite sign since the interval contains exactly one null and $g(e)$ is monotonous. If this is not the case, extend the start interval's limits until $g_1 \cdot g_2 < 0$.
2. bisect the interval $e_m = \frac{1}{2}(e_1 + e_2)$.
3. evaluate $g(e_m) = g_m$.
4. if g_1 and g_m are of opposite sign the null must be in $[e_1, e_m]$. restart with $[e_1, e_m]$, otherwise restart with $[e_m, e_2]$.

The convergence of this algorithm is known to be linear since the error on e is proportional to the interval length. Every iteration this length is divided in half.

The reason to use a finite temperature in the Fermi-Dirac distribution is not the systems real temperature. Rather it is the need for smeared out occupation numbers around the Fermi level. Systems with a band gap will not notice the finite temperature since the highest state below the gap will be occupied and the first states above the gap unoccupied. However, bands that cross the Fermi level produce a non-zero density of states at E_F. Also using a finite number of **k**-points will produce some eigenenergies close to E_F. Within the self-consistency cycle, these energies may change slightly. Consider the Fermi-Dirac distribution at $T_F = 0$ i.e. a step function that drops discontinuously at E_F. If the energy of an unoccupied state above the Fermi level falls below the energy of an occupied states, these states will exchange their occupancies. The new density, $\tilde{n}(\mathbf{r})$, is created from the occupied states only. Therefore the new density will deviate a lot from the old density, $n(\mathbf{r})$. The changed density may effect the eigenenergies to assume their previous ordering. One can see that there is an instability leading to oscillations and a constant change in density. In order to damp this out, the finite temperature produces partial occupancies around the Fermi level E_F, mainly in the interval $[E_F - k_B T_F, E_F + k_B T_F]$. The exchange of two energies E_i and E_j does then not induce a change of the occupancies f_i and f_j from 0 to 1 and vice versa but the change is proportional to $(E_i - E_j)/(k_B T_F)$ if both energies are close to the Fermi level, i.e. $|E_i - E_F| \ll k_B T$ and $|E_j - E_F| \ll k_B T$. Consequently, these oscillations may be damped by increasing the smearing temperature T_F. Temperature smearing is usually turned on by the user for metallic systems.



$$f(E_{KS}) = \frac{1}{e^{\beta(E_{KS}-E_F)} + 1} \tag{C.84}$$

Figure C.10.: Fermi-Dirac distribution function. A finite temperature T_F enables a linearization around E_F which is located at the origin in this plot.

Density of States at the Fermi Energy

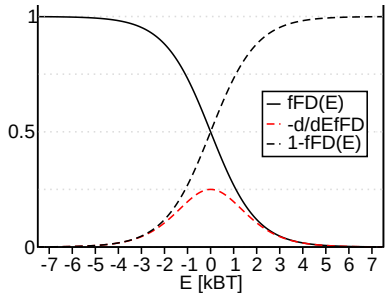


Figure C.11.: Fermi-Dirac distribution function and its derivative.

During the process of finding the correct Fermi energy given a spectrum of eigenenergies, we have to evaluate the Fermi-Dirac distribution function

$$f_{FD}(E - E_F, T_F) = \frac{1}{e^{\beta(E-E_F)} + 1}, \qquad \beta = (k_B T_F)^{-1} \tag{C.85}$$

where an artificial occupation temperature T_F has been introduced to generate temperature smearing. This is necessary in the case of metallic systems.

`jüRS` **Input Syntax:** `temp 345 Kel`

Especially in the situation of periodic samples it becomes important for the SCF convergence to apply smearing such that $k_B T_F$ is of the order of the *pseudo* band gap that arises from the finite **k**-point sampling. In order to quickly check if a system behaves metallic or insulating a quantity `DoS(0)` is displayed during the execution of `jüRS` that shows a measure of the Density of States around the Fermi energy E_F. This quantity is found also during the computation of the Fermi-Dirac distribution function. We can define a smearing or broadening function

$$b_{FD}(E, T_F) = -k_B T_F \frac{\partial}{\partial E} f_{FD}(E, T_F). \tag{C.86}$$

Once the Fermi-Energy is found, we evaluate `DoS(0)` $= b_{FD}(E_F, T_F)$.

Atom	Spin $f_{FD}(0, T_F)$	integrated DoS	Spin $f_{FD}(0, T_F)$	polarized DoS	Magn.$[\mu_B]$
B	1/3	5/6	1/3	2/3	1
C	2/3	4/3	2/3	2/3	2
N	3/3	3/2			3
O	4/3	4/3	1/3	2/3	2
F	5/3	5/6	2/3	2/3	1

In the special case of a discrete number of degenerate states at the Fermi energy this number is of interest. From table REF we can see that the classical value `DoS(0)`

does not give a direct estimate of the number of states since we expect B, C, O and F to show 3 states at the Fermi level in a spin-polarized calculation and 6 states in a spin-integrated run. Therefore we should look at the density of states at the corrected Energy $E_F + \Delta E$ where $\Delta E = k_B T_F \left(\ln(1-f) - \ln f\right)$. Evaluating the Fermi-Dirac function at this corrected Fermi level will give $f = \frac{1}{2}$ and `DoS(0)`=$\frac{1}{4}$ per state. However, this procedure does not apply for the situation of many states close to the Fermi energy and a large number of fractional occupation numbers as is a usual bulk calculation for a metal and hence it is not implemented.

The smearing function is also employed as broadening for the Density of States plots whereas a larger temperature might in many cases be good to achieve smoother graphs.

Density of States Plot

A graphical way to qualitatively analyze the Kohn-Sham spectrum of systems containing many atoms is the density of (KS-)states plot. In contrast to **k**-resolved band structure plots $E_\sigma(\mathbf{k})$ that may become more confusing the more bands are shown, the **k** dependency is fully integrated such that we are left with merely $D_\sigma(E)$ as a function of energy (one per spin).

$$D^\sigma(E) = \sum_n \int_{BZ} d^3\mathbf{k}\, \delta(E - E_{n\sigma\mathbf{k}})$$

To achieve a smooth function rather than a barcode, we replace the very sharp δ-function by a broadening function. Various types of broadening functions are possible, but we use the derivative of the Fermi-Dirac distribution $b_{FD}(E, T_F)$ as broadening function as outlined above and apply either the temperature of the calculation T_F or even a larger one to yield a continuous line.

Projected Density of States

The PAW method reveals the ℓ-character of each Kohn-Sham state inside of each augmentation sphere $\mathcal{S}^a$. Even though angular momentum can only be called a good quantum number if the potential is spherically symmetric (and without spin-orbit coupling), the projection of each state onto the localized atomic projector functions yields information about its local symmetry. The absolute of the projection coefficients $\langle \tilde{p}^a_{\ell 1 m} | \tilde{\Psi}_{n\sigma\mathbf{k}} \rangle$ tells us about the similarity of the KS-state $|\tilde{\Psi}_{n\sigma\mathbf{k}}\rangle$ to the smooth partial waves $|\tilde{\phi}^a_{1\ell m}\rangle$. We can thus use these as weights during the formation of the density of states:

$$D^\sigma_{a\ell m}(E) = \sum_n \int_{BZ} d^3\mathbf{k}\, \delta(E - E_{n\sigma\mathbf{k}}) \left| \langle \tilde{p}^a_{\ell 1 m} | \tilde{\Psi}_{n\sigma\mathbf{k}} \rangle \right|^2 \tag{C.87}$$

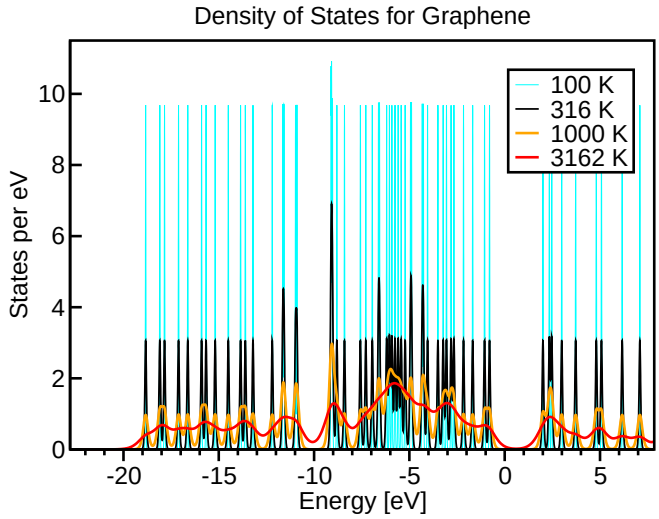


Figure C.12.: Total density of states of graphene for the smearing temperatures T_F=100 K, 316 K (default), 1000 K and 3162 K.

We only use the n=1 atomic partial waves and projectors since only those can be normalized to atomic eigenstates (compare Section A.0.2). If the directional character is not of interest, we may also look at the m-integrated DoS

$$D^{\sigma}_{a\ell}(E) = \sum_{m=-\ell}^{\ell} D^{\sigma}_{a\ell m}(E). \tag{C.88}$$

Similarly, also the ℓ quantum number can be integrated out, D^{σ}_{a}, such that we are left with the identification of the contributions of single atoms to the density of states.

Furthermore, we can perform statistical analysis e.g. by summing over all atoms of one species.

$$D^{\sigma}_{s\ell m} = \sum_{a \in s} D^{\sigma}_{a\ell m}. \tag{C.89}$$

Table C.7 gives an overview of the possibilities.

	atom		species		
ℓm-resolved	$D^{\sigma}_{a\ell m}$	`alm`	$D^{\sigma}_{s\ell m}$	`slm`	
ℓ-resolved	$D^{\sigma}_{a\ell}$	`al`	$D^{\sigma}_{s\ell}$	`sl`	m-integrated
	D^{σ}_{a}	`a`	D^{σ}_{s}	`s`	ℓm-integrated

Table C.7.: Projected density of states. To activate the generation of projected DoS plots choose `pdos` from { `alm`, `al`, `a`, `slm`, `sl`, `s` }.

Figure C.13.: Total density of states and projected density of states. To activate the generation of projected DoS plots in addition to the total DoS, choose `pdos` from { `alm`, `al`, `a`, `slm`, `sl` or `s` }.

ScaLAPACK interface workflow

- Initialize the BLACS grid of a largest possible square number of process elements $N_p \times N_p$ as a subset of the grid communicator `grid_comm`
- Setup BLACS grid descriptors for H_{nm}, S_{nm} and the eigenvectors X_{mk}
- Start here, if the module has been initialized before
- `init_matrix` allocates memory for the matrix chunks and clears all matrix and entries
- Fill matrices H_{nm} and S_{nm} via a setter-function `set_matrix_entry( Hnm, Snm, (n,m) )`
- Allreduce H_{nm} and S_{nm} via `band_comm` (only in band parallelization, see Section 5.4)
- `solve_matrix` allocates work memory and starts the ScaLAPACK diagonalization invoking `PDSYGVX` or `PZHEGVX`
- Broadcast the new eigenvalues E_n to all processes in `grid_comm`
- Get the eigenvector coefficients X_{mk} for the new linear combinations of bands via the getter-function `get_matrix_entry( (/m,k/), comm )`

The getter-function `get_matrix_entry` determines the rank of the process element which stores the eigenvector element X_{mk}. An `MPI_Broadcast` is performed along the `grid_comm` whose root is the element owning process element. The communicational overhead of a separate broadcast for each element in not tractable, therefore the function has been replaced by `get_matrix_block( (/m,k/), comm, b, (/ib,jb/) )`. This routine broadcasts the matrix block of BS $\times$ BS elements with block indices `ib` and `jb` that contains the requested matrix element. If the next request lies in the same block, communication is suppressed and the element is taken from the locally stored block `b(:,:)`.

APPENDIX D

USER MANUAL

D.0.10. Building the Application

jüRS is a pure Fortran code with coding standards newer than FORTRAN90, i.e. it uses some features of the '95 and 2003 standards. Before compiling, the directory build should contain the following items: Makefile, system.make, README and the directory src/ containing the .F90 and .F95 source files.

The Makefile usually does not need to be changed for building the application. All configurations such as preprocessor, compiler linker and their flags are defined in the architecture-specific include file system.make. If there is an example include file such as system.make.INTEL, you can copy that file or use a soft link to it:

```
ln -s system.make.INTEL system.make
```

For preprocessing the .F90 source files, a C-preprocessor (e.g. cpp) is applied. Some source files carry the extension .F95. These source files contain simultaneous code for real/complex and will be preprocessed twice with an additional flag R1_C2. The first time cpp -D R1_C2=1 $< generates the head part of the final .f90-source file with routines that accept real as data type for wave functions, the second time cpp -D R1_C2=2 $< generates routines for complex wave functions and the tail part of the preprocessed .f90-file. Each source file contains exactly one module.

The code has been developed for double precision that is activated via the compiler flags (Intel:-r8, IBM:-qrealsize=8). In principle, one could also compile in single precision, for that, -D SINGLE_PRECISION should be added to the preprocessor flags in order to replace library calls (MPI, LAPACK, ScaLAPACK[86]) with the proper interface.

On JUGENE and similar architectures, add -D JUGENE to the preprocessor flags.

Furthermore, the MPI library can be replaced with dummy routines leading to a serial version by defining `-D NOMPI`. On systems where ScaLAPACK is not installed, `-D NOScaLAPACK` will replace the ScaLAPACK calls by calls to the corresponding LAPACK routines. `-D NOLAPACK` can be used for testing the compilation and linking of all the others code parts but does not lead to a fully usable application. During development, some additional checks for NaN (Not a Number) can be included into the code with `-D NaN_SEARCH`.

The binary executable will be called `paw`.

D.0.11. Get Started

Once the application `./paw` was successfully compiled and linked, we may start with the help function `./paw --help` which explains how to generate an example input file and more command line options.

D.0.12. Input Files

`jüRS` (v5.*) tries to load the required element specific PAW data stored in `pawdata.00Z` where `00Z` is the three-digit atomic number Z^a, e.g. `pawdata.047` contains the PAW data for Silver. These files are `ASCII`-formatted. A description of the format can be found in the module `pawdatafile`.

The system to be calculated is described in a single input file. Three different elements of the input syntax are to be distinguished:

- Keyword statements D.0.13
- Variable definitions D.0.14
- Blocks D.0.15

The following subsection will describe each syntax element in detail.

D.0.13. Keyword statements

A keyword statement consists of one line in the input file that is lead by the keyword and followed by one or three values, depending on the quantity to be set. Blanks as separation are required here, no equality-sign. Lines starting from # are comments and will be ignored. For all keyword statements, the latter statement will overwrite the earlier. However, multiple mentioning of a keyword will launch a warning. In vector quantities (requiring three values) all vector entries are set to the same value if the keyword is followed only by one value.

A list of all keywords can be found by `./paw --keywords` and their syntax is explained further in Table D.1.

D.0.14. Variable definitions

jüRS is equipped with a built-in variable environment that helps to generate flexible input files. All variables are scalar real values. To define a variable yourself, simply write `$<var> = <expression>` in one line. The blanks to the left and right of = are important. A line that reads `$<var>` only will show the variable's value in the output. Here, `<expression>` must be one of float values, fractional values (with :), predefined constants `<cst>` (see Table D.2), defined variables, functions of values or math operations. Valid operators are +, -, *, : , ^ (or **). The /-slash for divisions cannot be used. No operator between two strings means multiplication. Operators only lead to valid expressions in a context `<value> <operator> <value>`. The minus sign therefore cannot be used as a unary operator or function, please use `0 - $<var>` instead. Valid functions are `exp`, `cos`, `sin`, `abs`, `sqrt` and `log`. Functions lead to valid expressions in a context `[<prefactor>] <function> <argument>`. An expressions may only consist of up to three strings. The usage of parenthesis is not supported. A string as `a:b` without blanks before and after the colon : will be interpreted as fractional number $\frac{a}{b}$.

D.0.15. Blocks

To input a list of dynamic length, blocks are defined by block keywords. A line containing only the block keyword opens the block or closes the block, depending on the application's previous state. In between block opening/closing keywords, block items are added to the corresponding list. Blocks are implemented for atoms and **k**-points: Block keywords are

- `kpoints` **k**-points for Brillouin zone sampling $x\ y\ z\ w$
- `kpath` edges for a band structure path sampling $x\ y\ z$
- `atoms` for absolute atomic positions $Z\ x\ y\ z\ [m_x\ m_y\ m_z\ \sigma]$
- `atoms_fractional` for relative positions $Z\ \frac{x}{L_x}\ \frac{y}{L_y}\ \frac{z}{L_z}\ [m_x\ m_y\ m_z\ \sigma]$

where Z may be either the atomic number (`integer`) or the chemical symbol (e.g. `Mg` instead of `12`, element symbols are case sensitive). L_i with $i \in \{x, y, z\}$ are the extends of the orthorhombic cell (see `cell`). For the input of atoms, we may optionally add three logic values $m_i \in \{\mathtt{T},\mathtt{F}\}$ for geometry relaxations. The default m_i is `T`, so atoms are usually free to move. If the three m_i are defined, the user may also define a spin-flip integer $\sigma \in \{-1, 0, +1\}$. If the element data contain a magnetic orbital occupation (run `grep PartialWave pawdata.*` to reassure a different occu-

<keyword>	#	Type	Default	Description
ascale	1	real	1.0	scale absolute atomic positions by this factor
bands	1	int	**[auto]**	total number of bands
boundary	3 or 1	int	0 0 0	boundary condition, 0:isolated 1:periodic
cell	3 or 1	real	**[required]**	extend of the orthorhombic cell
charged	1	real	0.0	additional electrons in the system
dg	2	int	2 5	DoubleGrid order N and meshrefinement M
domains	3 or 1	int	1 1 1	processes for domain decomposition
efield	3 or 1	real	.0 .0 .0	electric field (only directions where boundary is
element		string		PAW configuration, paw -e <ChemSym> shows d
forces	3 or 1	int	1 1 1	calculate forces 1:yes 0:no
hfield	1	real	.0	magnetic field, shifts local potential up/dn by t
kmesh	3 or 1	int	[32/cell]	automatic **k**-point sampling
kshift	3 or 1	real	.0 .0 .0	origion shift for kmesh
md	1	crit	0 0 1E-4	criteria for structural relaxation
mixing	1	real	.25	simple mixing ratio or ...
	3		.1 Broyden 19	higher order methods with history length
nf	2	int	4 4	FiniteDifference order N_f for KohnSham and Po
ngps	3 or 1	int	**[auto]**	total numbers of grid points per direction
nscale	1	int	1	ratio of potential grid over density grid
origin	3 or 1	real	.0 .0 .0	shift of the coordinate system in fractions of cel
pawpath	1	word	.	path to the pawdata.00Z files (use \ instead of /
pdos	1	word	none	projected DoS {alm,al,a,slm,sl,s,z}
poisson	1	crit	333 3 1E-7	criteria for CG iterations of electrostatics
scale	1	real	1.0	scale cell by this factor
scf	1	crit	33 3 1E-7	criteria for self-consistency iterations
shift			do not use!	
solver	1	word	diis	eigensolver method
spacing	3 or 1	int	0.25 Ang	grid spacing suggestion (to be adjusted)
spin	1	int	1	1:spin integrated 2:spin polarized
symmetry	1	word	none	do not use!
temp	1	real	.001 Ha	electronic smearing temperature
units	1 or 2	word	Ang eV	output units {aB,Ang,nm,pm} and {Ha,eV,Ry,kJ,...}
wfs	1	crit	1 1 .1	criteria for eigensolver iterations
xc	1	word	PZ81	XC-functional {VWN80,PZ81,...}

Table D.1.: Keywords and their syntax. The required key cell has no default values since the application will not run without its specification. Type crit stands for the input syntax of convergence criteria which are specified e.g. poisson max 333 min 3 < 1E-7 for the max. and min. number of iterations and the convergence threshold, short noted as 333 3 1E-7 in the column of default values.

<cst>		Internal Value	Name
Ang	Å	1.8897261247728	Ångström
nm	nm	18.897261247728	nanometer
pm	pm	0.0188972612477	picometer
Pi	π	3.1415926535898	
Deg	$\frac{2\pi}{360}$	0.0174532925199	degrees, $^\circ$
sqrth	$\sqrt{\frac{1}{2}}$	0.7071067811865	
sqrt2	$\sqrt{2}$	1.4142135623731	
sqrt3	$\sqrt{3}$	1.7320508075689	
sqrt5	$\sqrt{5}$	2.2360679774998	
Kel	K	$3.1668294 \cdot 10^{-6}$	Kelvin
eV	eV	27.210282768626	eVolt
meV	meV	0.0272102827686	milli eVolt
Ry	Ry	2	Rydberg
Ha	Ha	1	Hartree
aB	Bohr	1	Bohr radius

Table D.2.: Predefined constants for easy input.

pation of up and dn), the spin-flip determines the sign of the magnetization of the inital orbitals in a spin-resolved calculation. Default σ is +1.

A single block keyword followed by the name of an existing external file will open the file and try to read block items from it. Useful examples: atoms coords.xyz or kpoints kpts

In order to check the input file for syntax and perform some quick analysis of the atomic geometry we can run in CheckMode (most useful in serial). Invoke the executable as paw <inp> --CheckMode where <inp> is your input file. Then search the output for WARNING!s.

A continued calculation may load existing densities (--Load) and/or wave functions (--load) from the file system. See Section D.0.16 for their description.

D.0.16. Output Files

The name of the input file is the project name. All non-temporary output files will have the same name as the project plus an extension that depends on the type of output. Usual file name extensions are

- .dos Density of states (ASCII)
- .bst Plotable band structure (xmgrace v5.0+)
- .frc Atomic positions and forces (ASCII, atomic units)

- `.adm` Atomic density matrices $D^{a}_{ij\sigma}$ (`ASCII`)
- `.rho` Smooth valence density $\tilde{n}^{\sigma}_{v}(\mathbf{r})$ (`binary`)
- `.wfs` Smooth valence wave functions $\tilde{\Psi}_{n\sigma\mathbf{k}}(\mathbf{r})$ (`binary`)

In `CheckMode` some more files for the visualization are created

- `.xyz` Atomic positions (`ASCII`, Å)
- `.pov` Geometry for `povray` (`ASCII`)

Both types of binary files (density and wave functions) are created by `MPI`-collectives that write files in `BIG ENDIAN` [110, 111]. A leading header of 1024 Bytes proceeds the data arrays. The header is encoded in `ASCII` and holds information about the calculation such as file name, stored quantity, numbers of grid points, cell extends, number of bands, number of spins, number of **k**-points and a time-stamp of the file generation time. When densities are to be loaded, the number of grid points needs to match the number of grid points specified in the density file. Significant deviations of the cell sizes will launch `WARNING!`s. For loading wave functions, the number of bands, spins and **k**-points found in the file may be larger than the requested number, however, this will also be warned.

D.0.17. Global Application Configuration

Most configurations are defined in `mod_configurations.F90` and, if necessary, have to be adjusted before compiling.

D.0.18. Limitations

The restriction to nearest-neighbor communication during data-exchange introduces a limitation. The lower limit for the number of grid points per domain is the number of finite-difference neighbors N_f used for the kinetic energy stencil in the Kohn-Sham Hamiltonian. This defines a lower boundary to the parallelization in domain decomposition. The finite-difference order of the Laplacian in the Poisson equation of the electrostatic part may be twice as large since the density grid contains at least twice the number of grid points per direction in each domain compared to the wave function grid. The syntax for controlling N_f is explained in Table D.1 (`nf`).

D.0.19. Example input

Either run `./paw --example` to get an example input file
or create a file called `inp` with this content:

```
cell 12 Ang

$r = 76 pm
$a = 45 Deg
$x1 = $r cos $a
$y1 = $r sin $a
$x2 = 0 - $x1
$y2 = 0 - $y1

atoms
Li  $x1  $y1  0.0
Li  $x2  $y2  0.0
atoms
```

Now invoking `./paw inp --CheckMode` or equivalently `./paw inp -cm` tells the user that `./pawdata.003` could not be found. Re-running `./paw inp -cm --gen` generates a PAW data file for Lithium with the default PAW configuration for this species. Check `./paw -e Li` to see the default PAW configuration string for Lithium. Running again `./paw inp -cm` performs a check of the geometry specified in the input file. To start the calculation, remove the CheckMode flag `-cm` and add `-r` for `real` wave functions (faster). The file name does not need to be `inp` but `inp` is a placeholder for the project name. The input file of a project carries only the project name, `<projectname>`. Any output file will be named `<projectname>.<ext>` where `<ext>` depends on the kind of output.

Version 5.11 (2012-11-19)

BIBLIOGRAPHY

[1] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Phys. Rev.*, 136(3b):B864, 1964.

[2] W. Kohn and L.J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140(4a):A1133, 1965.

[3] P.A.M. Dirac. Quantum mechanics of many-electron systems. *Proc. R. Soc. Lond. A*, 123:714, 1929.

[4] M. Born and R. Oppenheimer. Zur quantentheorie der molekeln. *Annalen der Physik*, 389(20):457–484, 1927.

[5] M.E. Goodsite and M.S. Johnson, editors. *Advances in Quantum Chemisty*. Elsevier Inc., 2008.

[6] C. Fiolhais, F. Nohueira, and M. Marques, editors. *A Primer in Density Functional Theory*. Springer, Berlin, 2003.

[7] L.H. Thomas. The calculation of atomic fields. *Mathematical Proceedings of the Cambridge Philosophical Society*, 23:542–548, 1927.

[8] E. Fermi. Eine statistische methode zur bestimmung einiger eigenschaften des atoms und ihre anwendung auf die theorie des periodischen systems der elemente. *Zeitschrift für Physik A Hadrons and Nuclei*, 48:73–79, 1928. 10.1007/BF01351576.

[9] Ch. Huang and E.A. Carter. Toward an orbital-free density functional theory of transition metals based on an electron density decomposition. *Phys. Rev. B*, 85:045126, Jan 2012.

[10] Richard M. Martin. *Electronic structure*. Cambridge University Press, Apr 2004.

[11] C. Herring. A new method for calculating wave functions in crystals. *Phys. Rev.*, 57:1169–1177, Jun 1940.

[12] J.C. Slater. An augmented plane wave method for the periodic potential problem. *Phys. Rev.*, 92:603–608, Nov 1953.

[13] O.K. Andersen. Linear methods in band theory. *Phys. Rev. B*, 12(8):3060–3083, Oct 1975.

[14] E. Wimmer, H. Krakauer, M. Weinert, and A.J. Freeman. Full-potential self-consistent linearized-augmented-plane-wave method for calculating the electronic structure of molecules and surfaces: o_2 molecule. *Phys. Rev. B*, 24(2):864–875, Jul 1981.

[15] M. Weinert, E. Wimmer, and A.J. Freeman. Total-energy all-electron density functional method for bulk solids and surfaces. *Phys. Rev. B*, 26:4571–4578, Oct 1982.

[16] H.J.F. Jansen and A.J. Freeman. Total-energy full-potential linearized augmented-plane-wave method for bulk solids: Electronic and structural properties of tungsten. *Phys. Rev. B*, 30:561–569, Jul 1984.

[17] S. Sharma, J.K. Dewhurst, and C. Ambrosch-Draxl. All-electron exact exchange treatment of semiconductors: Effect of core-valence interaction on band-gap and d-band position. *Phys. Rev. Lett.*, 95:136402, Sep 2005.

[18] G.K.H. Madsen, P. Blaha, K. Schwarz, E. Sjöstedt, and L. Nordström. Efficient linearization of the augmented plane-wave method. *Phys. Rev. B*, 64:195134, Oct 2001.

[19] http://exciting-code.org. EXCITING FLAPW implementation.

[20] http://www.flapw.de. Full-Potential Linearized Augmented Plane-Wave.

[21] S.-H. Wei, H. Krakauer, and M. Weinert. Linearized augmented-plane-wave calculation of the electronic structure and total energy of tungsten. *Phys. Rev. B*, 32:7792–7797, Dec 1985.

[22] P.E. Blöchl. Projector augmented-wave method. *Phys. Rev. B*, 50(24):17953–17979, Dec 1994.

[23] L. Kleinman and D.M. Bylander. Efficacious form for model pseudopotentials. *Phys. Rev. Lett.*, 48(20):1425–1428, May 1982.

[24] D.R. Hamann, M. Schlüter, and C. Chiang. Norm-conserving pseudopotentials. *Phys. Rev. Lett.*, 43(20):1494–1497, Nov 1979.

[25] D.R. Hamann. Generalized norm-conserving pseudopotentials. *Phys. Rev. B*, 40:2980–2987, Aug 1989.

[26] D.C. Allan and M.P. Teter. Nonlocal pseudopotentials in molecular-dynamical density-functional theory: Application to sio_2. *Phys. Rev. Lett.*, 59:1136–1139, Sep 1987.

[27] N. Troullier and J.L. Martins. Efficient pseudopotentials for plane-wave cal-

culations. *Phys. Rev. B*, 43:1993–2006, Jan 1991.

[28] N. Troullier and J.L. Martins. Efficient pseudopotentials for plane-wave calculations. ii. operators for fast iterative diagonalization. *Phys. Rev. B*, 43:8861–8869, Apr 1991.

[29] M. Fuchs and M. Scheffler. Ab initio pseudopotentials for electronic structure calculations of poly-atomic systems using density-functional theory. *Computer Physics Communications*, 119(1):67 – 98, 1999.

[30] D. Vanderbilt. Soft self-consistent pseudopotentials in a generalized eigenvalue formalism. *Phys. Rev. B*, 41(11):7892–7895, Apr 1990.

[31] S. Goedecker, F. Lançon, and T. Deutsch. Linear scaling relaxation of the atomic positions in nanostructures. *Phys. Rev. B*, 64:161102, Sep 2001.

[32] M.A.L. Marques, A. Castro, G.F. Bertsch, and A. Rubio. octopus: a first-principles tool for excited electron-ion dynamics. *Computer Physics Communications*, 151(1):60–78, 2003.

[33] P. Giannozzi, S. Baroni, N. Bonini, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, G.L. Chiarotti, M. Cococcioni, I. Dabo, A. Dal Corso, S. de Gironcoli, S. Fabris, G. Fratesi, R. Gebauer, U. Gerstmann, C. Gougoussis, A. Kokalj, M. Lazzeri, L. Martin-Samos, N. Marzari, F. Mauri, R. Mazzarello, S. Paolini, A. Pasquarello, L. Paulatto, C. Sbraccia, S. Scandolo, G. Sclauzero, A.P. Seitsonen, A. Smogunov, P. Umari, and R.M. Wentzcovitch. Quantum espresso: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter*, 21(39):395502 (19pp), 2009.

[34] G. Kresse and J. Furthmüller. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Phys. Rev. B*, 54(16):11169–11186, Oct 1996.

[35] G. Kresse and D. Joubert. From ultrasoft pseudopotentials to the projector augmented-wave method. *Phys. Rev. B*, 59:1758–1775, Jan 1999.

[36] X. Gonze, B. Amadon, P.-M. Anglade, J.-M. Beuken, F. Bottin, P. Boulanger, F. Bruneval, D. Caliste, R. Caracas, M. Côté, T. Deutsch, L. Genovese, Ph. Ghosez, M. Giantomassi, S. Goedecker, D.R. Hamann, P. Hermet, F. Jollet, G. Jomard, S. Leroux, M. Mancini, S. Mazevet, M.J.T. Oliveira, G. Onida, Y. Pouillon, T. Rangel, G.-M. Rignanese, D. Sangalli, R. Shaltaf, M. Torrent, M.J. Verstraete, G. Zerah, and J.W. Zwanziger. Abinit: First-principles approach to material and nanosystem properties. *Computer Physics Communications*, 180(12):2582 – 2615, 2009.

[37] J.M. Soler and A.R. Williams. Simple formula for the atomic forces in the augmented-plane-wave method. *Phys. Rev. B*, 40:1560–1564, Jul 1989.

[38] R. Car and M. Parrinello. Unified approach for molecular dynamics and

density-functional theory. *Phys. Rev. Lett.*, 55(22):2471–2474, Nov 1985.

[39] F. Gygi and G. Galli. Real-space adaptive-coordinate electronic-structure calculations. *Phys. Rev. B*, 52(4):R2229–R2232, Jul 1995.

[40] 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(11):2745, 2002.

[41] D.R. Bowler, T. Miyazaki, and M.J. Gillan. Recent progress in linear scaling ab initio electronic structure techniques. *Journal of Physics: Condensed Matter*, 14(11):2781, 2002.

[42] C.-K. Skylaris, P.D. Haynes, A.A. Mostofi, and M.C. Payne. Introducing [small-caps onetep]: Linear-scaling density functional simulations on parallel computers. *The Journal of Chemical Physics*, 122(8):084119, 2005.

[43] V. Blum, R. Gehrke, F. Hanke, P. Havu, V. Havu, X. Ren, K. Reuter, and M. Scheffler. Ab initio molecular simulations with numeric atom-centered orbitals. *Computer Physics Communications*, 180(11):2175 – 2196, 2009.

[44] T.L. Beck. Real-space mesh techniques in density-functional theory. *Rev. Mod. Phys.*, 72(4):1041–1080, Oct 2000.

[45] E.L. Briggs, D.J. Sullivan, and J. Bernholc. Real-space multigrid-based approach to large-scale electronic structure calculations. *Phys. Rev. B*, 54(20):14362–14375, Nov 1996.

[46] J.R. Chelikowsky, N. Troullier, and Y. Saad. Finite-difference-pseudopotential method: Electronic structure calculations without a basis. *Phys. Rev. Lett.*, 72(8):1240–1243, Feb 1994.

[47] A. Castro, H. Appel, M. Oliveira, C.A. Rozzi, X. Andrade, F. Lorenzen, M.A.L. Marques, E.K.U. Gross, and A. Rubio. octopus: a tool for the application of time-dependent density functional theory. *physica status solidi (b)*, 243(11):2465–2488, 2006.

[48] J.J. Mortensen, L.B. Hansen, and K.W. Jacobsen. Real-space grid implementation of the projector augmented wave method. *Phys. Rev. B*, 71:035109, Jan 2005.

[49] 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(25):253202, 2010.

[50] K. Hirose, T. Ono, Y. Fujimoto, and Sh. Tsukamoto. *First-Principles Calculations in Real-Space Formalism*. Imperial College Press, London, 2005.

[51] T. Ono, M. Heide, N. Atodiresei, P.F. Baumeister, Sh. Tsukamoto, and S. Blügel. Real-space electronic structure calculations with full-potential all-electron precision for transition metals. *Phys. Rev. B*, 82(20):205115, Nov 2010.

[52] T. Ono and K. Hirose. First-principles study of peierls instability in infinite single row al wires. *Phys. Rev. B*, 68(4):045409, Jul 2003.

[53] T. Ono and K. Hirose. Timesaving double-grid method for real-space electronic-structure calculations. *Phys. Rev. Lett.*, 82(25):5016–5019, Jun 1999.

[54] Jugene: http://www.fz-juelich.de/jsc/jugene.

[55] Openmp: http://openmp.org/.

[56] http://top500.org/.

[57] M. Wuttig and N. Yamada. Phase-change materials for rewriteable data storage. *Nature Materials*, 6:824–832, Nov 2007.

[58] T. Zhang, Zh. Song, B. Liu, G. Feng, S. Feng, and B. Chen. Effect of structural transformation on the electrical properties for $ge_1sb_2te_4$ thin film. *Thin Solid Films*, 516(1):42 – 46, 2007.

[59] A. Klein, H. Dieker, B. Späth, P. Fons, A. Kolobov, C. Steimer, and M. Wuttig. Changes in electronic structure and chemical bonding upon crystallization of the phase change material $gesb_2te_4$. *Physical Review Letters*, 100(1), 2008. cited By (since 1996) 19.

[60] D. Lencer, M. Salinga, B. Grabowski, T. Hickel, J. Neugebauer, and M. Wuttig. A map for phase-change materials. *Nature Materials*, 7(12):972–977, Dec 2008.

[61] J.C. Slater. A simplification of the hartree-fock method. *Phys. Rev.*, 81(3):385–390, Feb 1951.

[62] J.C. Slater. Note on hartree's method. *Phys. Rev.*, 35:210, 1930.

[63] V. Fock. Näherungsmethode zur lösung des quantenmechanischen mehrkörperproblems. *Z. Physik*, 61:126, 1930.

[64] E. Teller. On the stability of molecules in the thomas-fermi theory. *Rev. Mod. Phys.*, 34:627–631, Oct 1962.

[65] Ch. Huang and E.A. Carter. Nonlocal orbital-free kinetic energy density functional for semiconductors. *Phys. Rev. B*, 81:045206, Jan 2010.

[66] U. von Barth and L. Hedin. A local exchange-correlation potential for the spin polarized case. i. *Journal of Physics C: Solid State Physics*, 5(13):1629, 1972.

[67] D.M. Ceperley and B.J. Alder. Ground state of the electron gas by a stochastic

method. *Phys. Rev. Lett.*, 45(7):566–569, Aug 1980.

[68] S.H. Vosko, L. Wilk, and M. Nusair. Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis. *Canadian Journal of Physics*, 58(8):1200–1211, 1980.

[69] J.P. Perdew and A. Zunger. Self-interaction correction to density-functional approximations for many-electron systems. *Phys. Rev. B*, 23(10):5048–5079, May 1981.

[70] J.C. Slater. Wave functions in a periodic potential. *Phys. Rev.*, 51(10):846–851, May 1937.

[71] P.P. Ewald. Die berechnung optischer und elektrostatischer gitterpotentiale. *Annalen der Physik*, 369(3):253–287, 1921.

[72] J.D. Jackson. *Classical electrodynamics*. Number 4 May 2011. Wiley & Sons, 1998.

[73] M. Weinert. Solution of poisson's equation: Beyond ewald-type methods. *J. Math. Phys.*, 22(11):2433–2439, 1981.

[74] C. Rostgaard. The projector augmented-wave method. online, 03 2010.

[75] J.-L. Fattebert, R.D. Hornung, and A.M. Wissink. Finite element approach for density functional theory calculations on locally-refined meshes. *Journal of Computational Physics*, 223(2):759 – 773, 2007.

[76] L. Kronik, A. Makmal, M.L. Tiago, M.M.G. Alemany, M. Jain, X. Huang, Y. Saad, and J.R. Chelikowsky. Parsec - the pseudopotential algorithm for real-space electronic structure calculations: recent advances and novel applications to nano-structures. *physica status solidi (b)*, 243(5):1063–1079, 2006.

[77] M. Tafipolsky and R. Schmid. A general and efficient pseudopotential fourier filtering scheme for real space methods using mask functions. *J. Comput. Phys.*, 124(17):174102, 2006.

[78] L.-W. Wang. Mask-function real-space implementations of nonlocal pseudopotentials. *Phys. Rev. B*, 64(20):201107, Nov 2001.

[79] T. Ono and K. Hirose. Real-space electronic-structure calculations with a time-saving double-grid technique. *Phys. Rev. B*, 72(8):085115, Aug 2005.

[80] M. Barad and Ph. Colella. A fourth-order accurate local refinement method for poisson's equation. *Journal of Computational Physics*, 209(1):1 – 18, 2005.

[81] P. Pulay. Convergence acceleration of iterative sequences. the case of scf iteration. *Chemical Physics Letters*, 73(2):393 – 398, 1980.

[82] G.M. Amdahl. Validity of the single-processor approach to achieving large scale computing capabilities. *AFIPS Conference Proceedings*, 30:483–485, Apr. 1967.

[83] M. Geimer, F. Wolf, B.J.N. Wylie, E. Ábrahám, D. Becker, and B. Mohr. The scalasca performance toolset architecture. *Concurrency and Computation: Practice and Experience*, 22(6):702–719, 2010.

[84] Scalasca: http://www.scalsca.org, Dec 2011.

[85] J.L. Gustafson. Reevaluating amdahl's law. *Commun. ACM*, 31:532–533, May 1988.

[86] L.S. Blackford, J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R.C. Whaley. Scalapack users' guide: http://netlib.org/scalapack/, May 1997.

[87] J. Poulson, B. Marker, J.R. Hammond, N.A. Romero, and R. van de Geijn. Elemental: a new framework for distributed memory dense matrix computations. *ACM Transactions on Mathematical Software*, 2010. submitted for publication.

[88] Intel xeon cluster: http://www.fz-juelich.de/jsc/juropa.

[89] R. Nishtala, P.H. Hargrove, D.O. Bonachea, and K.A. Yelick. Scaling communication-intensive applications on bluegene/p using one-sided communication and overlap. In *Parallel Distributed Processing, 2009. IPDPS 2009. IEEE International Symposium on*, pages 1 –12, May 2009.

[90] P. Micikevicius. 3d finite difference computation on gpus using cuda. In *Proceedings of 2nd Workshop on General Purpose Processing on Graphics Processing Units*, GPGPU-2, pages 79–84, New York, NY, USA, 2009. ACM.

[91] D. Michéa and D. Komatitsch. Accelerating a three-dimensional finite-difference wave propagation code using gpu graphics cards. *Geophysical Journal International*, 182(1):389–402, 2010.

[92] J. Hegedüs and S.R. Elliott. Microscopic origin of the fast crystallization ability of ge-sb-te phase-change memorymaterials. *Nature Materials*, 7(5):399–405, 2008. cited By (since 1996) 96.

[93] T. Matsunaga, J. Akola, Sh. Kohara, T. Honma, K. Kobayashi, E. Ikenaga, R.O. Jones, N. Yamada, M. Takata, and R. Kojima. From local structure to nanosecond recrystallization dynamics in aginsbtephase-changematerials. *Nature Materials*, 10:129–134, Feb 2011.

[94] W. Wełnic, A. Pamungkas, R. Detemple, C. Steimer, S. Blügel, and M. Wuttig. Unravelling the interplay of local structure and physical properties in phase-change materials. *Nature Materials*, 5(1):56–62, 2006. cited By (since 1996) 137.

[95] E. Prokhorov, E. Trapaga, and J. Gonzalez-Hernandez. Structural and electrical properties of $ge_1sb_2te_4$ face centered cubic phase. *Journal of Applied Physics*, 104(10):103712, 2008.

[96] J.P. Perdew and A. Zunger. Self-interaction correction to density-functional approximations for many-electron systems. *Phys. Rev. B*, 23(10):5048–5079, May 1981.

[97] A.R. Thieß. *Massively parallel KKR-Green function method for large scale systems*. PhD thesis, RWTH Aachen University, 2011.

[98] T. Fujiwara and T. Yokokawa. Universal pseudogap at fermi energy in quasicrystals. *Phys. Rev. Lett.*, 66:333–336, Jan 1991.

[99] P.F. Baumeister. jurs documentation wiki: http://www.real-space.de/.

[100] http://www.nist.gov/.

[101] J.J. Sakuari. *Modern Qantum Mechanics*. Addision Wesley Publishing Company, Inc., 1994.

[102] X. Gonze, P. Käckell, and M. Scheffler. Ghost states for separable, norm-conserving, ab initio pseudopotentials. *Phys. Rev. B*, 41:12264–12267, Jun 1990.

[103] J. Korringa. On the calculation of the energy of a bloch wave in a metal. *Physica*, 13(6-7):392–400, 1947.

[104] W. Kohn and N. Rostoker. Solution of the schrödinger equation in periodic lattices with an application to metallic lithium. *Phys. Rev.*, 94(5):1111–1120, Jun 1954.

[105] D.A. Papaconstantopoulos and M.J. Mehl. The slater-koster tight-binding method: a computationally efficient and accurate approach. *Journal of Physics: Condensed Matter*, 15(10):R413, 2003.

[106] P.H. Dederichs and R. Zeller. Self-consistency iterations in electronic-structure calculations. *Phys. Rev. B*, 28:5462–5472, Nov 1983.

[107] C.G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Math. Comp.*, 19:577–593, 1965.

[108] D.D. Johnson. Modified broyden's method for accelerating convergence in self-consistent calculations. *Phys. Rev. B*, 38:12807–12813, Dec 1988.

[109] H.H.H. Homeier and E.O. Steinborn. Some properties of the coupling coefficients of real spherical harmonics and their relation to gaunt coefficients. *Journal of Molecular Structure: THEOCHEM*, 368:31 – 37, 1996. Proceedings of the Second Electronic Computational Chemistry Conference.

[110] Mpi: A message-passing interface standard. University of Tennessee, Knoxville, Tennessee, 1995.

[111] Mpi-2: Extensions to the message-passing interface. University of Tennessee, Knoxville, Tennessee, 1997.

ACKNOWLEDGMENTS

First of all, I would like to express my sincere gratitude to Prof. Dr. Stefan Blügel for giving me this unique opportunity of undertaking my research and for providing the right amount of guidance. He gave me the chance to obtain insights into a large variety of computational methods and always provided the basis for productive work at the Institut für Festkörperforschung and later the Peter Grünberg Institut & Institute for Advanced Simulation.

Special thanks are due to Prof. Dr. Paolo Bientinesi for his willingness to act as co-examiner for this thesis.

I would also like to thank Prof. Dr. Tomoya Ono for his invaluable introduction to real-space methods. His supervision and collaboration have inspired me to take the real-space grid approach to new limits.

I wish to acknowledge the helpful cooperation of Prof. Dr. Ricardo Mazzarello and Wei Zhang for the phase-change material GeSbTe.

I am grateful to Dr. Daniel Wortmann for teaching me the various aspects of motivation. Furthermore, he contributed a lot of fruitful discussions to this work and helped me with his excellent supervision and proofreading.

I was lucky to share an office with Dr. Shigeru Tsukamoto. I couln't have a better office mate. He taught me a lot of Chinese characters on the white board which I want to acknowledge here. I also wish to thank Keiko Satō-Blügel for her well-prepared Japanese language classes.

For her helpful advice and active proofreading in combination with constructive criticism I would like to thank Dr. Nicole Helbig.

Many thanks go to the Japan Society for the Promotion of Science, Prof. Dr. Kuwahara (Osaka University), Joachim Schiffer, Akane Morinishi and Bernd Zimmermann for a great time and unforgettable experiences in Japan.

I am also indepted to all my colleagues at the institute for providing unfailing motiviation. I would like to thank all of them, however, some stand out in particular. I am grateful to Dr. Andreas Gierlich for listening to my personal issues in a tricky situation, to Dr. Andreas Dolfen for enduring inspiration in terms of physics and computational methods, to Dr. Alexander Thieß for his cooperative support and to Dr. Markus Betzinger for profound and invaluable discussions on a daily basis.

Finally, and most importantly, I am grateful to my parents for all that I am.

```
implicit none
```

The first 90 percent of the code accounts for the first 90 percent of the development time. The remaining 10 percent of the code accounts for the other 90 percent of the development time. – Tom Cargill, Bell Labs