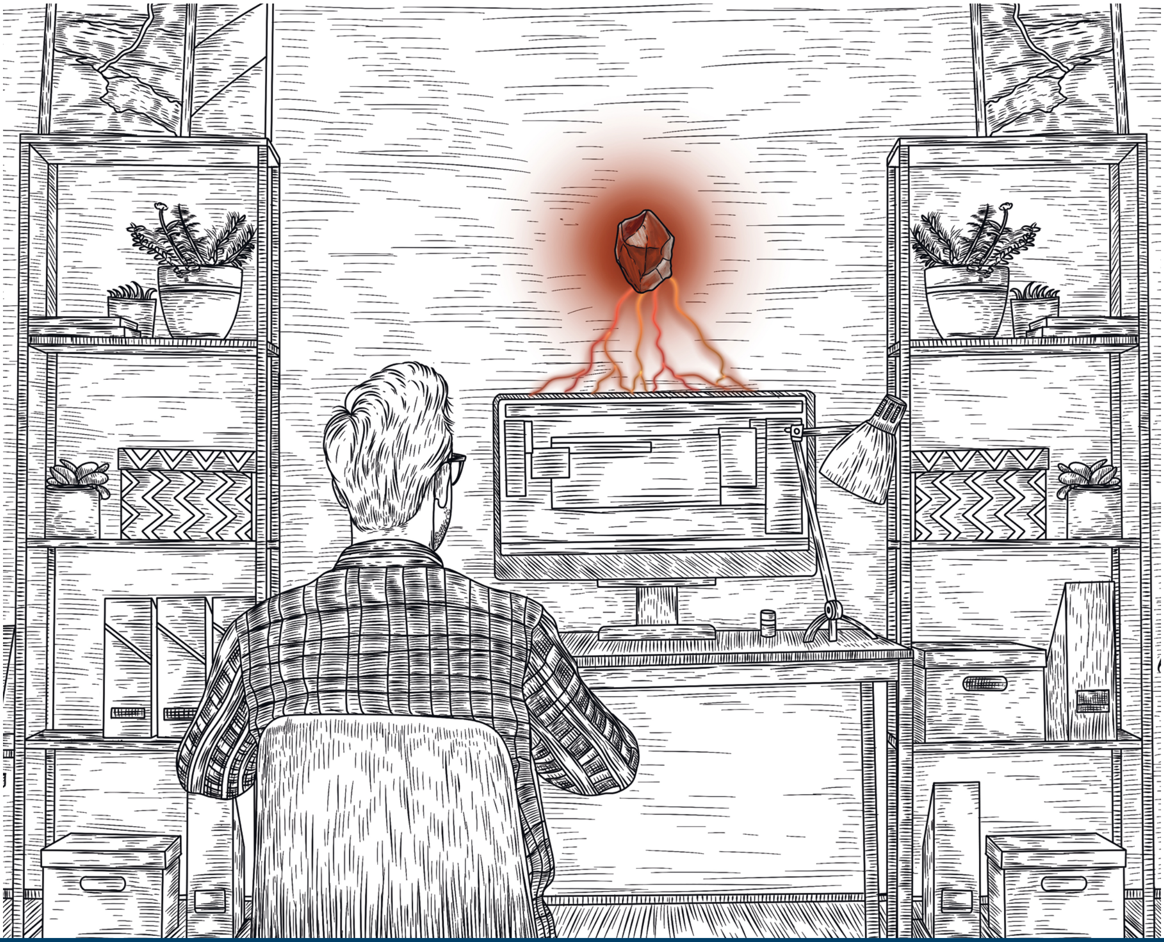

High-Performance Computing Approach to Hybrid Functionals in the All-Electron DFT Code FLEUR

Matthias Redies

Schlüsseltechnologien / Key Technologies
Band / Volume 257
ISBN 978-3-95806-639-7

Mitglied der Helmholtz-Gemeinschaft

JÜLICH
Forschungszentrum

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies Band / Volume 257

Forschungszentrum Jülich GmbH
Peter Grünberg Institut (PGI)
Quanten-Theorie der Materialien (PGI-1/IAS-1)

High-Performance Computing Approach to Hybrid Functionals in the All-Electron DFT Code FLEUR

Matthias Redies

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies Band / Volume 257

ISSN 1866-1807 ISBN 978-3-95806-639-7

Bibliografische Information der Deutschen Nationalbibliothek.
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte Bibliografische Daten
sind im Internet über http://dnb.d-nb.de abrufbar.

Herausgeber und Vertrieb:	Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag 52425 Jülich Tel.: +49 2461 61-5368 Fax: +49 2461 61-6103 zb-publikation@fz-juelich.de www.fz-juelich.de/zb
Umschlaggestaltung:	Grafische Medien, Forschungszentrum Jülich GmbH
Druck:	Grafische Medien, Forschungszentrum Jülich GmbH
Copyright:	Forschungszentrum Jülich 2022

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies, Band / Volume 257

D 82 (Diss. RWTH Aachen University, 2022)

ISSN 1866-1807
ISBN 978-3-95806-639-7

Abstract

Virtual materials design attempts to use computational methods to discover new materials with superior properties within the vast space of all conceivable materials. Density-functional theory (DFT) is central to this field, enabling scientists to predict material properties from first principles, i.e. without relying on external parameters or experimental values. While standard DFT is capable of predicting many materials with satisfying accuracy, it struggles with some properties such as details of the electronic structure or certain material classes, e.g. materials exhibiting strongly correlated electrons. This has created a need for methods with greater predictive power. One such class of methods are hybrid exchange-correlation functionals which combine the exact Hartree-Fock exchange with local exchange-correlation functionals, resulting in highly accurate predictions for many insulating or semiconductor materials. However, the computational cost of hybrid functionals increases rapidly with system size and limits their application to small systems. This thesis aims to solve the computational challenge posed by hybrid functionals in large systems by utilizing the massive computational power of today's supercomputers.

This thesis presents the improved implementation of hybrid exchange-correlation functionals in FLEUR, an all-electron full-potential linearized augmented planewave code. The improved CPU and a new GPU implementations allow users to make efficient use of modern compute nodes and a highly-scalable MPI implementation distributes calculations with a single $\hat{k}$-point to 3000 cores or 64 GPUs and far beyond that for calculations with multiple $\hat{k}$-points. This work promotes hybrid functionals to systems with hundreds of atoms, opening up their application to many new material classes and properties. We demonstrate the power of this algorithm by applying it to garnets, a class of complex magnetic materials with large unit-cells, which have promising applications in fields such as spintronics or quantum computing. Garnets are rare-earth oxides that exhibit strongly correlated electrons in localized $3d$- or $4f$-states, making the combination of hybrid functionals and FLAPW ideally suited to investigate these materials. After benchmarking our method against other highly predictive methods and experimental results using yttrium iron garnet as a reference system, we shift our focus the rare-earth-iron garnets gadolinium iron garnet and thulium iron garnet. For these materials we perform the first-ever hybrid exchange-correlation functional calculations of their electronic structure and

magnetic moments, establishing the predictive power of the hybrid functionals in FLEUR for large complex magnets.

Zusammenfassung

Virtuelles Materialdesign versucht mithilfe computergestützter Methoden neue Materialien mit überlegenen Eigenschaften innerhalb des riesigen Raums aller denkbaren Materialien zu entdecken. Zentral in diesem Bereich ist die Methode der Dichtefunktionaltheorie (DFT) die es Wissenschaftlern ermöglicht Materialeigenschaften aus ersten Prinzipien vorherzusagen, dass heisst ohne sich auf externe Parameter oder experimentelle Werte zu beziehen. Obwohl Standard-DFT in der Lage ist viele Materialien mit zufriedenstellender Genauigkeit vorherzusagen, hat sie Probleme mit einigen Eigenschaften wie etwa Details der elektronischen Struktur oder bestimmten Materialklassen, wie etwa Materialien mit stark korrelierten Elektronen. Dies hat einen Bedarf an Methoden mit grösserer Vorhersagekraft geschaffen. Eine Klasse von solchen Methoden sind hybride Austauschkorrelationsfunktionale, die den exakten Hartree-Fock-Austausch mit lokalen Austauschkorrelationsfunktionalen kombinieren, was zu hochgenauen Vorhersagen für viele Isolatoren oder Halbleitermaterialien führt. Der Rechenaufwand hybrider Funktionale nimmt jedoch schnell mit der Systemgrösse zu und beschränkt ihre Anwendung auf kleine Systeme. Diese Arbeit zielt darauf ab, das Problem des massiven Rechenaufwands von Hybridfunktionalen in grossen Systemen zu lösen, indem die enorme Rechenleistung heutiger Supercomputer genutzt wird.

Diese Dissertation präsentiert eine verbesserte Implementierung hybrider Austauschkorrelationsfunktionale in FLEUR, einem Programm das die *full-potential linearized augmented planewave* (FLAPW) Methode auf alle Elektronen anwendet. Die verbesserte CPU und eine neue GPU-Implementierung ermöglicht Wissenschaftlern die effiziente Nutzung moderner Rechenknoten und eine hochskalierbare MPI Implementierung verteilt Berechnungen mit einem einzigen $\hat{k}$-Punkt auf 3000 Kerne oder 64 GPUs und weit darüber hinaus für Berechnungen mit mehreren $\hat{k}$-Punkten. Diese Arbeit ermöglicht Berechnungen mit Hybridfunktionale in Systemen mit Hunderten von Atomen und erlaubt ihre Anwendung in vielen neuen Materialklassen. Wir demonstrieren die Leistungsfähigkeit dieses Algorithmus, indem wir ihn auf Granate anwenden, eine Klasse komplexer magnetischer Materialien mit grossen Elementarzellen, die vielversprechende Anwendungen in Bereichen wie Spintronik oder Quantencomputern haben. Granate sind Oxide die seltene Erden enthalten welche, stark korrelierte Elektronen in lokalisierten $3d$- oder $4f$-Zuständen aufweisen, wodurch

die Kombination von Hybridfunktionalen und FLAPW ideal für die Untersuchung dieser Materialien geeignet ist. Nachdem wir unsere Methode mit anderen vorhersagestarken Methoden und experimentellen Ergebnissen mit Yttrium-Eisen-Granat als Referenzsystem verglichen haben, verlagern wir unseren Fokus auf die Eisen-Granate mit seltenen Erden: Gadolinium-Eisen-Granat und Thulium-Eisen-Granat. Für diese Materialien führen wir die allerersten Hybrid-Funktionalrechnungen ihrer elektronischen Struktur und magnetischen Momente durch und demonstrieren so die Vorhersagekraft der Hybridfunktionale in FLEUR für grosse komplexe Magnete.

Contents

List of Abbreviations

AMD	Advanced Micro Devices, Inc.
ARM	Advanced RISC Machines Ltd.
AVX2	Advanced Vector Extensions - *Vectorized instruction set for x86*
BLAS	Basic Linear Algebra Subprograms
CPU	Central Processing Unit
DFT	Density Functional Theory
DOS	Density of States
FFT	Fast Fourier Transformation
FLAPW	Full-Potential Linearized Augmented Planewave
FLEUR	*DFT code, that implements the FLAPW method*
FLOP	Float-Point Operation
FLOPS	Float-Point Operations per Second
GdIG	Gadolinium Iron Garnet
GEA	Gradiant Expansion Approximation
GGA	Generalized Gradient Approximation
GGA+U	*GGA with an added Hubbard U*
GPU	Graphics Processing Unit
HPC	High-Performance Computing
HSE	*A hybrid exchange-correlation functional by J. Heyd et al* [1]
IS	Interstitial
LAPACK	Linear Algebra Package
LAPW	Linearized Augmented Planewave
LDA	Local Density Approximation
LDA+U	*LDA with an added Hubbard U*
LibXC	*A library that implement a large number of exchange-correlation functionals*
MetaGGA	*A GGA that includes higher-order terms, such as the kinetic energy density*
MKL	Math Kernel Library - *Intel's LAPACK implementation*
MPB	Mixed-Product Basis
MPI	Message Passing Interface - *A framework for distributed-memory parallelization*
MT	Muffin-Tin Sphere
OpenACC	Open Accelerators - *A shared-memory parallelization framework with focus on accelerators*
OpenMP	Open Multi-Processing - *A shared-memory parallelization framework*
PAW	Projector Augmented Wave

PBE	*A GGA by J. P. Perdew et al* [2]
PBE0	*A hybrid exchange-correlation functional by J. P. Perdew et al* [3]
QSGW	Quasi-Particle Self-Consistent GW
RIG	Rare-Earth Iron Garnet
RISC-V	Reduced Instruction Set Computer - *A patent free instruction set*
SCAN	*A MetaGGA by J. Sun et al* [4]
TmIG	Thulium Iron Garnet
YIG	Yttrium Iron Garnet
[. . .]	*General citation*
[@ . . .]	*Citation of a Website*

Introduction

1

Materials science aims to understand and predict material properties more and more accurately, so that new sophisticated materials can be discovered to drive innovation in the domains that rely on them. While materials science has been around for millennia, it was only at the beginning of the last century that the arrival of quantum mechanics enabled the exact description of the microscopic properties in materials. However, the cost of calculating the exact solution to the Schrödinger equation grows exponentially with the size of the system and is therefore limited to very small systems. This problem was addressed rigorously by Hohenberg, Kohn and Sham [5, 6] in the 1960s, with the development of density functional theory (DFT). DFT replaces the $3N$-dimensional wave function as the central quantity with the 3-dimensional ground-state density and thereby reduces the exponential computational cost to a polynomial one. While DFT is in principle exact, a key ingredient, the so-called *exchange-correlation energy*, has no known analytical expression. The approximations used for this term determine the accuracy with which material properties can be predicted. While the most commonly used approximations, the local density approximation (LDA) and the generalized gradient approximation (GGA) can predict certain properties with a high precision at a very low computational cost, they fail to predict some essential electronic properties [7].

Despite this, DFT has been amazingly successful. In 2020 alone almost 19.000 papers have been published that reference density functional theory in their title or abstract. Additionally, DFT is increasingly being used in the context of high-throughput calculations, where hundreds of thousands of material candidates are screened using automated workflows [8–10]. However, all of these calculations are limited to material classes and properties for which the underlying exchange-correlation functionals have a good predictive power. In order to enhance these calculations with materials classes and properties for which LDA and GGA fail, it is necessary to rely on more accurate methods producing high quality results. One class of accurate methods are the hybrid exchange-correlation functionals which are particularly suited to predicting electronic properties such as the band gap, the degree of charge localization and the polarization in materials with a stronger electron correlation [11–15].

Hybrid exchange-correlation functionals mix an orbital dependent exact exchange with the accurate correlation of other approximations, such as LDA or GGA. Their reliance on the orbital dependent exact exchange makes them computationally considerably more expensive than LDA or GGA. While an LDA or a GGA calculation grows with the 3rd power of the number of atoms, a hybrid exchange-correlation functional calculation typically grows with the 4th power of the number of atoms. Additionally, the computational cost of a hybrid calculation grows quadratically with the number of $\hat{k}$-points used to sample the Brillouin zone, whereas for an LDA or a GGA calculation it only grows linearly. This large computational cost has prohibited precise predictions for systems with large unit cells, including a number of interesting material classes such as garnets [16, 17] or materials of interest for solid-state batteries [18]. Additionally, investigations which rely on supercell calculations, such as the effects of defects in the crystal structure or alloyed materials have also been out of reach for these highly accurate methods.

There is an abundance of methods that implement hybrid functionals and DFT in general. Gaussian basis sets [19] are commonly used in quantum chemistry and the projector augmented wave (PAW) method [20] has been widely adopted for periodic systems. Even basis-set-free methods, relying on real-space grids, have been proposed [21]. In this thesis, however, we will focus on the full-potential augmented-plane-wave (FLAPW) method as it is implemented in the open-source code FLEUR [@22]. Unlike for example PAW, the FLAPW method treats all electrons explicitly and does not employ any approximations to represent the potential or density. It is therefore well suited for a wide range of systems, including systems containing heavy atoms that exhibit a lot of d- and f-states. It is considered the most accurate DFT method and has been used as a benchmark for other methods and codes [23].

While there have been significant advances in bringing hybrid functionals to systems with hundreds of atoms in other methods, such as the PAW [24, 25], hybrid functionals within FLAPW have been constrained to very small systems [26–28]. The aim of this thesis is to enable FLEUR's hybrid functional implementation to run on the world's most advanced supercomputers and use their immense computational power to investigate these large and interesting systems.

Utilizing modern supercomputers has become increasingly challenging. While the rapid miniaturization of transistors has provided developers with a stream of ever better performing CPUs for many decades, nowadays the most advanced supercomputers rely on specialized hardware, so-called accelerators, to deliver the majority of their computational power. In June 2017 three of the top ten supercomputers relied on accelerators [@29], now only four years later it is twice as many [@30]. Therefore, as part of this thesis we created not only a scalable CPU implementation of the hybrid functionals, but also a GPU version that makes use of these modern accelerators. Furthermore, it seems that the diversity of hardware used in high-performance computing is increasing. While the processor market was dominated by Intel just a few years ago, now AMD has a considerable market share, the fastest computer in the world relies on ARM [@30], and the European Processor Initiative is exploring the use of the open RISC-V architecture in their accelerators [@31]. In order to be prepared for this plethora of architectures we paid particular attention to writing code easily adaptable to new, unknown architectures.

Building on the work previously done on hybrid functionals in the FLAPW basis and in FLEUR specifically [27, 28, 32, 33], we analyzed the performance of this legacy implementation and its bottlenecks and explored algorithmic improvements needed to calculate hundreds of atoms with the accuracy that FLAPW and hybrid functionals offer. Achieving a good performance on a supercomputer can be split into two main tasks. First, the hardware of single node has to be used efficiently, regardless of whether the node has one or more CPUs or a number of additional GPUs; and second, the communication between nodes has to be orchestrated in such a way that the communication itself does not become a bottleneck limiting the speed of the calculation.

In order to solve the first task we expressed the most computationally costly routines in terms of large standard math problems, such as matrix multiplication or Fourier transformations. This approach has two advantages over custom code. First, the libraries provided for these standard problems are tuned with an immense effort that is far beyond the scope of any PhD thesis and therefore their performance is excellent. Second, each hardware platform has libraries specifically optimized for it, therefore we do not need to re-optimize FLEUR for new upcoming platforms. For parts of the code that do not fall within the mold of a standard math problem we created custom code for the intra-node parallelization. For the parallelization on CPU systems we used OpenMP and for the offloading to GPUs we used the OpenACC framework. We perform a detailed investigation of the performance achieved on single nodes as well as the performance of a single GPU. We measure the scalability of the code with an increasing number of cores and investigate its performance on different GPU generations. For the most computationally demanding routines we create a so-called roofline model, which compares the performance of certain code parts to the theoretical maximum for a given hardware.

For the second task, creating an efficient inter-node communication scheme, we designed three independent levels of parallelism, which enable us to distribute the large computational problem posed by the hybrid functionals to thousands of nodes. This mechanism allows us to distribute the problem over different $\hat{k}$- and $\hat{q}$-points as well as over different occupied bands. We measure the scalability of this communication pattern on SuperMUC-NG, calculating systems with almost 200 atoms in the unit cell. Additionally, we perform similar tests for the GPU implementation on Jureca DC.

Finally, we use our algorithm to investigate a number of large complex magnets. Yttrium iron garnet (YIG) has emerged as a key material for a number of fields, such as magnonics, spin transport or quantum computing [34–36]. Due to its large primitive unit cell, containing 80 atoms, its electronic structure has only recently been calculated accurately [37]. In YIG, like in other garnets, the iron atoms appear in two kinds of environments: its oxygen nearest neighbours are either arranged in a tetrahedron or an octahedron. Using the accurate treatment of the iron $3d$-states provided by FLEUR's hybrid exchange-correlation functionals, we investigate the effect of these environments. We predict a number of key material parameters for YIG and compare them to a series of PBE+U calculations performed in FLEUR as well as QSGW [37], another method known for accurate predictions of the electronic structure.

Lastly, we shift our focus to a related class of materials, the so-called *rare-earth-iron* garnets which share the same structure as YIG but the Yttrium is replaced with a rare-earth atom from the lanthanide series. We investigate Gadolinium iron garnet and Thulium iron garnet, which have applications as Faraday rotators in optical communication [38] and spintronics [39]. For these materials we perform the first known accurate calculations of the electronic structure. We predict their magnetic moments as well as provide the first accurate predictions of their band gaps.

This thesis is organized as follows. Chapter 2 gives an introduction to density functional theory and focuses on the hybrid exchange-correlation functionals at the heart of this thesis. In Chapter 3 the FLAPW method is presented and the *mixed-product basis*, an extension of the LAPW basis necessary to calculate the exact exchange, is derived. Chapter 4 specifies the changes made in order to tune the single node performance of the hybrid functional calculations. Afterwards, a detailed analysis of the resulting performance is presented. In Chapter 5 the same is done for the intra-node scalability. First we present the algorithms used to scale the calculation of the exact exchange to thousands of nodes and then we analyze their performance. In Chapter 6 we apply this algorithm to the three garnets discussed above. Finally, in Chapter 7 we reflect on the achievements of this work and the remaining challenges.

Density functional theory

2

2.1 Introduction

The advent of quantum mechanics in the early twentieth century facilitated the theoretical prediction of atomic systems with an accuracy never seen before [40, 41]. However, it soon became obvious that calculating exact solutions to the Schrödinger equation is not practically feasible. Even today's computational power is insufficient to calculate the exact solution for any quantum mechanical system of meaningful size.

In this chapter we introduce the many-body Hamiltonian and discuss problems with obtaining an exact solution to it. Then we motivate the so-called Born-Oppenheimer approximation [42] and introduce the frameworks at the foundation of density functional theory (DFT) [5, 6]. Finally, we shift our focus to the different classes of exchange-correlation potentials used in DFT. We provide an overview of the most commonly used local potential types and take a deeper dive into the non-local hybrid exchange-correlation functionals at the heart of this thesis. As a matter of simplicity we assume that the spin channels are degenerate throughout this chapter.

The first section of this chapter, discussing fundamental concepts of density functional theory is going to follow along the lines of previous introductions [32, 43–46].

2.1.1 Born-Oppenheimer approximation

The ground-state of a system of electrons and nuclei can be accurately described by the time-independent Schrödinger equation

$$\mathcal{H}\left|\Psi\right\rangle = E\left|\Psi\right\rangle , \tag{2.1}$$

where $|\Psi\rangle$ is the many-body wave function and the Hamiltonian $\mathcal{H}$ is defined as

$$\mathcal{H} = \underbrace{-\sum_{i}^{N_{\text{elec}}} \frac{\boldsymbol{\nabla}_i^{\,2}}{2}}_{\mathcal{T}_{\text{e}}} \underbrace{-\sum_{\alpha}^{N_{\text{atom}}} \frac{\boldsymbol{\nabla}_\alpha^{\,2}}{2m_\alpha}}_{\mathcal{T}_{\text{i}}} + \underbrace{\sum_{\substack{i,j \\ i\neq j}}^{N_{\text{elec}}} \frac{1}{2\|\hat{r}_i - \hat{r}_j\|}}_{\mathcal{V}_{\text{ee}}} + \underbrace{\sum_{\substack{\alpha,\beta \\ \alpha\neq\beta}}^{N_{\text{atom}}} \frac{Z_\alpha Z_\beta}{2\|\hat{r}_\alpha - \hat{r}_\beta\|}}_{\mathcal{V}_{\text{ii}}} \underbrace{-\sum_{i}^{N_{\text{elec}}} \sum_{\alpha}^{N_{\text{atom}}} \frac{Z_\alpha}{\|\hat{r}_i - \hat{r}_\alpha\|}}_{\mathcal{V}_{ei}} . \tag{2.2}$$

Throughout this thesis we will use atomic units ($e = m_{\text{e}} = \hbar = 1$) as defined by Hartree [47]. In Eq. 2.2 the Latin letters i and j label the electrons, while the Greek letters α and β label the nuclei. N_{elec} and N_{atom} indicate the total number of electrons and nuclei, while $\hat{r}_i$ and $\hat{r}_\alpha$ denote their positions. Z_α and m_α are the charge and mass of the nuclei.

The first two terms of the Hamiltonian $\mathcal{T}_{\text{e}}$ and $\mathcal{T}_{\text{i}}$ describe the kinetic energy of the electrons and nuclei, while the last three terms $\mathcal{V}_{\text{ee}}$, $\mathcal{V}_{\text{ii}}$ and $\mathcal{V}_{\text{ei}}$ represent the coulomb interaction among electrons, among the nuclei and lastly between electrons and nuclei.

Since the nuclei are at a minimum three orders of magnitude heavier than the electrons, the movement of the nuclei is much slower and appears frozen compared to that of the electrons. This is due to the fact that the same forces act on the electrons and nuclei, but the moment of inertia of the electrons is lower and therefore they experience greater accelerations. Hence, we assume that the nuclei are stationary on the time scale of the electrons. This effective decoupling of the movement of the electrons and nuclei is called the *Born-Oppenheimer approximation* [42].

By neglecting the kinetic energy of the nuclei $\mathcal{T}_{\text{i}}$ and assuming that the nuclei are frozen we can express Eq. 2.2 as a Hamiltonian, where the nuclei only appear as part of the external potential

$$\mathcal{H}_{\text{BO}} = \mathcal{T}_{\text{e}} + \mathcal{V}_{\text{ee}} + \mathcal{V}_{\text{ext}}, \tag{2.3}$$

where

$$\mathcal{V}_{\text{ext}} = -\sum_{i}^{N_{\text{elec}}} \sum_{\alpha}^{N_{\text{atom}}} \frac{Z_\alpha}{\|\hat{r}_i - \hat{r}_\alpha\|} + \mathcal{V}_{\text{ii}}. \tag{2.4}$$

Eq. 2.3 represents a big simplification compared to the original many-body Hamiltonian, but the solution to Eq. 2.3 still is a $3\,N_{\text{elec}}$-dimensional function $|\Psi\left(\hat{r}_1, ..., \hat{r}_{N_{\text{elec}}}\right)\rangle$, making it impossible to store, let alone to calculate, the solution to this equation even for very moderate values of N_{elec}.

The storage space requirement of the many-body wave function, growing exponentially with the number of electrons, makes a practical application of this approach unfeasible even if we assume the most optimistic future computational developments, as Walter Kohn pointed out in his Nobel lecture in 1999 [48]. In the next two sections we will introduce an auxiliary problem of independent particles, that in principle will share the charge density and total energy of the many-body system and thus eliminates the problem of $3N_{\text{elec}}$-dimensional many-body wave functions.

2.1.2 Hohenberg-Kohn Theorem

The idea at the heart of density functional theory is to substitute the ground-state many-body wave function with the many-body charge density

$$n(\hat{r}) = \langle\Psi|\sum_i \delta(\hat{r}-\hat{r}_i)|\Psi\rangle \tag{2.5}$$

as the fundamental quantity of any calculation. The Hohenberg-Kohn theorems [5] put this idea on a solid theoretical foundation. They state that

1. The ground-state density n_0 uniquely determines the external potential V_{ext} up to a constant shift
   $$V_{\text{ext}}(\hat{r}) = V_{\text{ext}}[n_0](\hat{r}). \tag{2.6}$$
   Therefore, the ground-state wave function and all observables are also functionals of the ground-state density.

2. The functional for the total energy is minimized by the ground-state density with the constraint that the number of electrons is kept fixed. So that
   $$E_0 = E[n_0] \leq E[n] \quad \forall n \quad \text{such that} \int n(\hat{r}) d^3r = \int n_0(\hat{r})\, d^3r = N_{\text{elec}} \tag{2.7}$$
   and
   $$E[n_0] < E[n] \quad \forall \quad n \neq n_0. \tag{2.8}$$
   Therefore, the ground-state density can be found by minimizing the total energy functional.

For a proof of these theorems the reader is referred to the literature [5, 43, 44].

2.1.3 Kohn-Sham System

The Hohenberg-Kohn theorems tell us that we can use the ground-state density to calculate every observable and that we get the ground-state density by minimizing the total energy functional

$$E[n] = F[n] + \int V_{\text{ext}}(\hat{r})\, n(\hat{r})\, d^3r, \tag{2.9}$$

where the functional F, the so-called *universal functional*,

$$F[n] = \min_{n[\Psi]} \langle \Psi | \mathcal{T} + \mathcal{V}_{\text{ee}} | \Psi \rangle \tag{2.10}$$

is the same regardless of the external potential. This alone however, is of limited practical application, since no analytical expression of this functional is known.

In their famous paper [6] Kohn and Sham suggest to split this functional into three terms

$$F[n] = T_e[n] + U_{\text{H}}[n] + E_{\text{xc}}[n], \tag{2.11}$$

where T_e is the kinetic energy of non-interacting electrons, U_{H} is the coulomb energy and $E_{\text{xc}}[n]$ is the so-called exchange-correlation functional. The coulomb energy U_{H} is defined as

$$U_{\text{H}}[n] = \frac{1}{2} \int \frac{n(\hat{r})\, n(\hat{r}')}{\|\hat{r} - \hat{r}'\|}\, d^3r d^3r'. \tag{2.12}$$

E_{xc} describes all the exchange and correlation interactions, that are not part of T_e or U_{H} and is therefore implicitly defined by Eq. 2.11

$$E_{\text{xc}} := F[n] - T_e[n] - U_{\text{H}}[n]. \tag{2.13}$$

While we just presented the exact expression for U_{H} and we present an expression for T_e late in this section, no analytic expression for E_{xc} is known and a long list of approximations to it have been proposed [1–4, 49–52]. One class of approximations are the so-called *hybrid functionals*, the speedy calculation of which is the topic of this thesis. Section 2.2 gives an overview of the different types of exchange-correlation functionals and Section 2.4 gives a more detailed introduction into hybrid functionals.

Splitting the universal functional F like this leads to very good results, because the contributions of T_e and U_{H}, which are calculated exactly, are quite large so that only the comparatively small, but important, E_{xc} needs to be approximated [53, p. 37].

For U_H we already presented an explicit expression in Eq. 2.12 and here we derive an expression for T_e. This expression however is not directly dependent on the density, but instead relies on a set of non-interacting wave functions, called *Kohn-Sham orbitals*, that construct the density

$$n(\hat{r}) = 2\sum_{i=1}^{N_\mathrm{occ}} \phi_i^*(\hat{r})\,\phi_i(\hat{r}), \tag{2.14}$$

where N_occ is the number of occupied states. We refer to the full set of Kohn-Sham orbitals as $\mathbb{K} := \{\phi_1, ..., \phi_{N_\mathrm{occ}}\}$. The factor of two is added to account for the two degenerate spin channels. We can use this orbital dependent expression to derive the kinetic energy as

$$T_\mathrm{e}\left[\mathbb{K}[n]\right] = -2\sum_{i=1}^{N_\mathrm{occ}} \phi_i^*(\hat{r})\boldsymbol{\nabla}^2\phi_i(\hat{r})\,d^3r. \tag{2.15}$$

Introducing the Kohn-Sham orbitals makes solving our many-body problem considerably more complicated, because unlike $U_\mathrm{H}[n]$ or $E_\mathrm{xc}[n]$, the kinetic energy $T_\mathrm{e}[\mathbb{K}[n]]$ now depends on the Kohn-Sham orbitals, which in turn depend on on the density indirectly.

Since we now have to minimize $T_e[\mathbb{K}[n]]$ with respect to the wave function instead of the density, we have to replace the constraint on the number of electrons with a constraint to normalize the wave functions

$$\int \phi_i^*(\hat{r})\,\phi_i(\hat{r})\,d^3r = 1 \quad \forall i. \tag{2.16}$$

We enforce this constraint through a Lagrange multiplier ε_i and minimize the total energy by taking the functional derivative to obtain the so-called *Kohn-Sham equations*

$$\left[-\boldsymbol{\nabla}^2 + V_\mathrm{eff}(\hat{r})\right]\phi_i(\hat{r}) = \varepsilon_i\phi_i(\hat{r}), \tag{2.17}$$

which have the form of the Schrödinger equation for a single particle in the effective potential

$$V_\mathrm{eff}(\hat{r}) = V_\mathrm{ext}(\hat{r}) + V_\mathrm{H}(\hat{r}) + V_\mathrm{xc}(\hat{r}). \tag{2.18}$$

By having N_elec single-particle wave functions, rather than a single many-body wave function for N_elec particles we have reduced the storage requirement from an exponential $\mathcal{O}((N_\mathrm{bas})^{N_\mathrm{elec}})$ to an effortless $\mathcal{O}(N_\mathrm{elec} \cdot N_\mathrm{bas})$, where N_bas is the number of basis functions. This reduction in storage needed enables the calculation of systems with thousands of atoms [54]. In Eq. 2.17 we use the Hartree potential

$$V_{\mathrm{H}}[n](\hat{r}) = \int \frac{n(\hat{r}')}{\|\hat{r} - \hat{r}'\|} \, d^3r' \tag{2.19}$$

and the exchange-correlation potential

$$V_{\mathrm{xc}}[n] = \frac{\delta E_{\mathrm{xc}}[n]}{\delta n}. \tag{2.20}$$

With $V_{\mathrm{xc}}[n]$ and $V_{\mathrm{H}}[n]$ depending on the charge density $n(\hat{r})$ (Eq. 2.20, Eq. 2.19), which in turn depends on the Kohn-Sham orbitals ϕ_i (Eq. 2.14), which then again depend on $V_{\mathrm{xc}}[n]$ and $V_{\mathrm{H}}[n]$ (Eq. 2.17) we have created a self-consistency problem. The correct charge density will generate a potential, which reproduces the same density by applying it to the Kohn-Sham equations. All charge densities that do not fulfill this criterion are not solutions of the Kohn-Sham equations. Section 2.5 is going to introduce how we obtain a solution to this self-consistency problem.

Lastly we need to point out that while the density of the Kohn-Sham system is identical to that of the many-body system, the Kohn-Sham orbitals ϕ_i have limited physical meaning. While there is no obvious connection to the many-body wave function the Kohn-Sham orbitals have shown to be remarkably useful: Their band structure for example often reproduces experimental measurements surprisingly well.

2.2 Exchange-Correlation functionals

In Eq. 2.13 we defined E_{xc}, but no exact expression is known and we therefore need to find an appropriate approximation for it. While E_{xc} might only be a small contribution to the total energy it encapsulates a lot of the most interesting physics such as the Pauli principle or screening effects. Additionally, if we consider the Coulomb energy in Eq. 2.12 for the case of a single electron is becomes obvious that U_{H} introduces a nonphysical coulomb repulsion of a single electron with itself. Hence, a *self-interaction correction* should also be part of the exchange-correlation functional E_{xc}.

In this section we attempt to shine light on the nature of E_{xc}. First we introduce a series of exchange-correlation functional classes and discuss the performance of their most commonly used proponents. Then we discuss the coupling constant integration, a method that does not generate an applicable functional but instead furthers our understanding of the exchange-correlation functional and motivates the construction of hybrid exchange-correlation functionals. This section regarding the exchange-correlation functionals follows along the lines of [32, 33, 53].

Discussing the accuracy of any exchange-correlation functional is tricky a endeavor, since the performance of an exchange-correlation functional is usually measured by taking a set of systems and comparing a number of quantities, such as formation energies, lattice constants, bulk modulus or band gaps to experimental results. While this can give an indication about the performance of a certain exchange-correlation functional for a specific property, this kind of analysis heavily depends on the choice of the set of materials. Therefore, the performance statements in this chapter should be treated as rough indications rather than absolute truth.

2.2.1 Local Density Approximation (LDA)

This thesis deals with two kinds of approximations to E_{xc}: Local and non-local exchange correlation potentials, where the local type has the typical form

$$E_{\text{xc}}^{\text{loc}}[n] = \int n(\hat{r})\, \epsilon_{\text{xc}} \left(n(\hat{r}), \boldsymbol{\nabla} n(\hat{r}), \chi_1(\hat{r}), ..., \chi_n(\hat{r})\right)\, d^3r, \tag{2.21}$$

where $\chi_i(\hat{r})$ can be any quantity that is only evaluated at $\hat{r}$. In this form the exchange-correlation energy density ϵ_{xc} only depends on quantities at $\hat{r}$, hence the name *local* exchange-correlation potential. The most simple approximation of this kind is the so-called *local density approximation* (LDA), which only relies on the charge density $n(\hat{r})$. In the LDA ϵ_{xc} is calculated for a set of homogeneous electron gases of different densities and then $\epsilon_{\text{xc}}^{\text{HEG}}(n(\hat{r}))$ is used according to the local density at $\hat{r}$. For the exchange we can derive an exact expression in a homogeneous electron gas

$$\epsilon_{\text{x}}^{\text{HEG}}(n(\hat{r})) = -\frac{3\sqrt[3]{3\pi^2 n}}{4\pi}, \tag{2.22}$$

whereas the correlation is obtained by fitting a function to quantum Monte-Carlo calculations of a homogeneous electron gas [55]. While there is a number of parametrizations [56–59], they all describe the same functional and therefore the choice of the parametrization does not effect the DFT calculation in a meaningful way.

Despite being a seemingly crude approximation the LDA has been quite successful in solid state physics, often correctly predicting structural parameters within a few percent. However, the LDA systematically underestimates the lattice constant for example in the test set used by [60] a mean relative error (MRE) of -1.3% and mean absolute relative error (MARE) of 1.3% was reported. Other properties such as the band gap are much harder to predict using a LDA. Here the LDA underestimates the band gap with a MRE of -58% and a MARE of 58% in the set of materials used by [61]. Additionally transfer energies between orbitals with different l-quantum numbers are quite inaccurate [62].

2.2.2 Including gradients

While the LDA works reasonably well for a number of systems it is still a little crude in the sense that it neglects all non-homogenous components of the density. This approximation does not represent systems with rapidly changing densities well. The most obvious way to remedy this is to include the gradient of the density $\boldsymbol{\nabla} n(\hat{r})$ into the exchange-correlation functional [63]

$$E_{\text{xc}}^{\text{GEA}}[n] = \int n(\hat{r})\, \epsilon_{\text{xc}}^{\text{LDA}}\,(n(\hat{r}))\ d^3r + \int C_{\text{xc}}(n(\hat{r}))\frac{|\boldsymbol{\nabla} n|^2}{n^{4/3}}, \tag{2.23}$$

where $C_{\text{xc}}(n(\hat{r}))$ is the prefactor of second-order term gradient expansion of exchange-correlation energy [64]. The first-order term is simply the exchange-correlation energy used in a LDA. However, this so-called gradient expansion approximation (GEA) is much less accurate than the LDA [64].

Formal properties of exchange-correlation functionals

To understand why the LDA outperforms the GEA although it undoubtedly contains more information about the system we need to consider a few constraints that we know a true physical system to have. While 17 of these constraints are known [4] we will only introduce a few of them.

Consider a parameter $\gamma > 0$, with which we scale our wave function

$$\Psi_\gamma(\hat{r}_1, ...\hat{r}_N) = \gamma^{3N/2}\,\Psi(\gamma\hat{r}_1, ...\gamma\hat{r}_N) \tag{2.24}$$

uniformly. This scaling satisfies both the normalization

$$\langle\Psi_\gamma|\Psi_\gamma\rangle = \langle\Psi|\Psi\rangle = 1 \tag{2.25}$$

constraint and it conserves the number of electrons

$$n_\gamma(\hat{r}) = n(\hat{r}) = N_{\text{elec}}. \tag{2.26}$$

It can be shown that under this transformation a number of identities hold true [53, 65]

$$\begin{aligned}
U_{\text{H}}[n_\gamma] &= \gamma\, U_{\text{H}}[n] && (2.27)\\
T_{\text{e}}[n_\gamma] &= \gamma^2\, T_{\text{e}}[n] && (2.28)\\
E_{\text{x}}[n_\gamma] &= \gamma\, E_{\text{x}}[n] && (2.29)\\
\lim_{\gamma\to\infty} \frac{E_{\text{x}}[n_\gamma]}{E_{\text{xc}}[n_\gamma]} &\to 1 && (2.30)\\
\lim_{\gamma\to\infty} E_{\text{c}}[n_\gamma] &\to \text{constant} && (2.31)
\end{aligned}$$

and similarly we can define a non-uniform scaling

$$n^x_\gamma(x, y, z) = \gamma\, n(\gamma\, x, y, z) \tag{2.32}$$

which leads to constraints like

$$\begin{aligned}
\lim_{\gamma\to\infty} T_e[n^x_\gamma] &\to \int \frac{|\boldsymbol{\nabla} n|^2}{8n}\, d^3r && (2.33)\\
\lim_{\gamma\to\infty} E_{\text{x}}[n^x_\gamma] &\to \text{constant} && (2.34)\\
\lim_{\gamma\to\infty} E_{\text{c}}[n^x_\gamma] &\to \text{constant.} && (2.35)
\end{aligned}$$

Since the LDA is based on a real physical system (the homogeneous electron gas) it inherently satisfies all of these constraints. This is not the case for the GEA. GEA does not satisfy the constraints on E_{c} (Eq. 2.31, Eq. 2.35) nor the non-uniform constraint on E_{x} in Eq. 2.34. Therefore, the LDA often outperforms the GEA for calculations of physical systems, despite being constructed with less information.

Generalized gradient approximation

To improve upon the accuracy of LDA we therefore need the *generalized gradient approximation* (GGA), which has a more general form of

$$E^{\text{GGA}}_{\text{xc}}[n] = \int n(\hat{r})\epsilon^{\text{GGA}}_{\text{xc}}(n(\hat{r}), \boldsymbol{\nabla} n(\hat{r})). \tag{2.36}$$

Using this higher degree of freedom, functional developers could suggest non-empirical GGAs that fulfil more of the constraints than the GEA such as the widely used PBE [2]. While empirical exchange-correlation functionals which determine these parameters through experimental fitting are more commonly used for specific scientific domains.

Commonly used GGA functionals such as PBE [2] and its solid-state variant PBEsol [50] improve on the accuracy of LDA with a MRE/MAE of 1.05%/1.18% (PBE) and -0.17%/0.67% (PBEsol) for the lattice constant with the testsets used in [60]. PBE tends to slightly overestimate the lattice constant, while the band gap is still underestimated by quite a lot with a MRE of -53% and a MARE of 53% for PBE in [61].

2.2.3 MetaGGA

In 1998, two years after the publication of the PBE functional, Becke noted that the degrees of freedom within GGAs were insufficient and "...that further accuracy will not be won by continued tinkering with GGA parametrizations, but that new beyond-GGA directions need to be investigated." [66]

The question arises how one can go beyond GGA and an obvious choice might be to use the Laplacian of the density $\boldsymbol{\nabla}^2 n$, but this quantity can cause challenging numerical problems in the calculation of the potential $\boldsymbol{\nabla}^2 \left({}^{\partial \epsilon_{\mathrm{xc}}}/_{\partial \boldsymbol{\nabla}^2 n} \right)$ [67]. Becke suggests [66, 68] that one should rather use the closely related occupied kinetic energy density

$$\tau = \frac{1}{2} \sum_{i}^{N_{\mathrm{occ}}} |\boldsymbol{\nabla} \Psi_i(\hat{r})|^2 \tag{2.37}$$

and a series of MetaGGA exchange-correlation functionals, that rely on this quantity have been suggested [4, 69]. Most notably the SCAN functional which adheres to all 17 known constraints and which reduces the errors to 0.3% (MRE) and 0.6% (MARE) for the lattice constant the test set used by F. Tran et al [70] and -43% (MRE) / 43% (MARE) for the band gaps of the systems used in [71].

As a way to gain familiarity with the FLEUR code I created an interface to the LibXC library [72], which includes support for MetaGGAs in the exchange-correlation energy, but not the potential. An implementation of MetaGGA potentials would involve much bigger refactoring of our code, since the kinetic energy density τ is not density-, but orbital-dependent and therefore would require an implementation of the optimised effective potential method (OEP) [73, 74].

While we were able to verify our implementation of the SCAN functional with a few reference systems, numerical problems persist. We believe this has two root causes. Firstly, numerical problems with the SCAN functional are not unique to our implementation, but instead have been extensively discussed in the literature [75–78]. Secondly, we also notice numerical problems with LibXC if we compare their PBE implementation to our native implementation. While for some systems the agreement is excellent, we discovered that numerical problems often occur in areas of very low density and in areas very close to the core, where we rely on a relativistic Hamiltonian. While MetaGGAs seem like a good path towards greater accuracy, their performance in some key material classes, such as transition metal magnets [79] or alkali metals [80] is still being investigated.

2.2.4 Exact exchange

While the local exchange-correlation functionals in the previous sections deliver reasonable results and they are very quick to evaluate, there is no reason to believe that the true exchange-correlation functional is of the local type given in Eq. 2.21. This section introduces an exchange-correlation functional, that attempts to achieve greater accuracy by breaking with the local restriction. Hence, we call this a *non-local* exchange-correlation potential.

We can further decompose the exchange-correlation energy into an exchange and a correlation part

$$E_{\mathrm{xc}}[n] = E_{\mathrm{x}}[n] + E_{\mathrm{c}}[n]. \tag{2.38}$$

We define $E_{\mathrm{x}}[n]$ similarly to the exchange used in the Hartree-Fock formalism

$$E_{\mathrm{x}}^{\mathrm{HF}} = -\frac{1}{2}\sum_{i,j}^{N_{\mathrm{occ}}} \int\int \frac{\phi_i^*(\hat{r})\phi_j^*(\hat{r}')\phi_j(\hat{r})\phi_i(\hat{r}')}{\|\hat{r}-\hat{r}'\|} d\hat{r}d\hat{r}' \tag{2.39}$$

as

$$E_{\mathrm{x}}^{\mathrm{exact}}[n] = \langle\Phi_{\mathrm{KS}}|\mathcal{V}_{\mathrm{ee}}|\Phi_{\mathrm{KS}}\rangle - U_{\mathrm{H}}[n], \tag{2.40}$$

where $|\Phi_{\mathrm{KS}}\rangle$ is the product of the Kohn-Sham orbitals and we have simply applied the Kohn-Sham orbitals to Eq. 2.39. This expression differs from the exchange used for Hartree-Fock calculations only by the fact that the Kohn-Sham wave function are used here, rather than the Hartree-Fock wave functions. In special cases where the Kohn-Sham and the Hartree-Fock wave functions are the same, this expression yields the same exchange energy. We refer to this quantity as the *exact exchange*. In practice the numerical differences between the Hartree-Fock exchange energy and the exact exchange energy in a Kohn-Sham system are minor [81].

Combining Eq. 2.10 and Eq. 2.11 yields

$$\langle \Phi_{\mathrm{KS}} | \mathcal{T} + \mathcal{V}_{\mathrm{ee}} | \Phi_{\mathrm{KS}} \rangle = T_{\mathrm{e}}[n] + U_{\mathrm{H}}[n] + E_{\mathrm{x}}[n], \tag{2.41}$$

which in the case of a single electron ($\mathcal{V}_{\mathrm{ee}} = 0$, $\mathcal{T} = \mathcal{T}_{\mathrm{e}}$) this turns into

$$E_{\mathrm{x}}[n] = -U_{\mathrm{H}}[n], \tag{2.42}$$

meaning that the exact exchange derived in Eq. 2.40 and its potential $V_{\mathrm{x}}[n](\hat{r}) = \frac{\delta E}{\delta n(\hat{r})}$ correctly cancel out the self-interaction of the Hartree potential for a single electron.

With an exact expression for E_{x} this leaves the correlation energy as the difference between the total energy of the true physical system and the Kohn-Sham total energy

$$\begin{aligned} E_{\mathrm{c}}[n] &= F[n] - \left(T_e[n] + U_{\mathrm{H}}[n] + E_x[n] \right) & (2.43)\\ &= \langle \Psi | \mathcal{T} + \mathcal{V}_{\mathrm{ee}} | \Psi \rangle - \langle \Phi_{\mathrm{KS}} | \mathcal{T} + \mathcal{V}_{\mathrm{ee}} | \Phi_{\mathrm{KS}} \rangle \,. & (2.44) \end{aligned}$$

While it is possible to perform the exact exchange calculations in FLEUR and in DFT in general, it suffers the same drawbacks as Hartree-Fock calculations do. The neglect of correlation leads to inaccurate results. In the following sections we therefore try to shine some light on E_{xc} as a whole by introducing the coupling constant integration method.

2.3 Coupling constant integration

Since we were able to derive the exact exchange in Eq. 2.40 only E_{c} remains unknown. Eq. 2.44 provides a definition, but it is not useful for practical calculations, since it relies on the unknown universal potential $F[n]$. In this section we are going to try to use the so-called *coupling-constant integration* or adiabatic-connection method to investigate the nature of E_{xc} and therefore E_{c}.

Consider a set of Hamiltonians where we use a parameter $0 \leq \lambda \leq 1$ to tune the electron-electron interaction

$$H^{\lambda} = T + \lambda V_{\mathrm{ee}} + V^{\lambda}. \tag{2.45}$$

Here we adjust V^λ such that the ground-state density $n(\hat{r})$ for each system is the same, which is possible due to the Hohenberg-Kohn theorems. In the extreme cases for $\lambda = 0$ the potential V^λ will simply turn into the effective Kohn-Sham potential with the Kohn-Sham wave function $|\Phi_{\mathrm{KS}}\rangle$, while for $\lambda = 1$ the potential V^λ is the external potential for the real physical system with the wave function $|\Psi\rangle$.

Now we combine Eq. 2.40 and Eq. 2.44 to get

$$E_{\mathrm{xc}}[n] \tag{2.46}$$

$$= E_{\mathrm{x}}[n] + E_{\mathrm{c}}[n] \tag{2.47}$$

$$= \langle \Phi_{\mathrm{KS}} | \mathcal{V}_{\mathrm{ee}} | \Phi_{\mathrm{KS}} \rangle - U_{\mathrm{H}}[n] + \langle \Psi | \mathcal{T} + \mathcal{V}_{\mathrm{ee}} | \Psi \rangle - \langle \Phi_{\mathrm{KS}} | \mathcal{T} + \mathcal{V}_{\mathrm{ee}} | \Phi_{\mathrm{KS}} \rangle \tag{2.48}$$

$$= \langle \Psi | \mathcal{T} + \mathcal{V}_{\mathrm{ee}} | \Psi \rangle - \langle \Phi_{\mathrm{KS}} | \mathcal{T} | \Phi_{\mathrm{KS}} \rangle - U_{\mathrm{H}}[n] \tag{2.49}$$

$$\tag{2.50}$$

and as previously introduced we can reinterpret the Kohn-Sham and the real potential as the cases for $\lambda = 0$ and $\lambda = 1$ giving us

$$= \left\langle \Psi^\lambda \middle| \mathcal{T} + \lambda \mathcal{V}_{\mathrm{ee}} \middle| \Psi^\lambda \right\rangle \Big|_{\lambda=1} - \left\langle \Psi^\lambda \middle| \mathcal{T} + \lambda \mathcal{V}_{\mathrm{ee}} \middle| \Psi^\lambda \right\rangle \Big|_{\lambda=0} - U_{\mathrm{H}}[n] \tag{2.51}$$

$$= \int_0^1 d\lambda \frac{d}{d\lambda} \left\langle \Psi^\lambda \middle| \mathcal{T} + \lambda \mathcal{V}_{\mathrm{ee}} \middle| \Psi^\lambda \right\rangle - U_{\mathrm{H}}[n] \tag{2.52}$$

using the Hellmann-Feynman theorem [82, 83] we get

$$E_{\mathrm{xc}}[n] = \int_0^1 d\lambda \frac{dE_{\mathrm{xc}}^\lambda}{d\lambda} = \int_0^1 d\lambda \left[\left\langle \Psi^\lambda \middle| \mathcal{V}_{\mathrm{ee}} \middle| \Psi^\lambda \right\rangle - U_{\mathrm{H}}[n] \right]. \tag{2.53}$$

Here, only $\lambda = 0$ (the Kohn-Sham system) and $\lambda = 1$ (the real physical system) have meaning. All intermediate systems are only mathematical constructs. Eq. 2.53 gives us yet another exact expression for $E_{\mathrm{xc}}[n]$, that we can not calculate analytically, but it will prove useful in order to motivate the ansatz for the hybrid functionals in the next section.

2.4 Hybrid exchange-correlation functionals

So far we have seen several approaches to approximate E_{xc}. In analogy to Hartree-Fock calculations we were able to derive an exact expression for E_{x}, but exact exchange calculations suffer from notorious overbinding, due to their lack of any correlation [81, 84]. The LDA and GGAs are widely used exchange-correlation functionals, that approximate both exchange and correlation. We know however that the contribution of the exchange to E_{xc} is much larger than that of the correlation and therefore the question arises if there is a way to utilize the exact exchange together with the correlation given by the LDA or GGAs.

The naive approach of

$$E_{\text{xc}} = E_{\text{x}}^{\text{exact}} + E_{\text{c}}^{\text{LDA/GGA}} \tag{2.54}$$

has been investigated [85], but it fails in practice. It was only with the work of Becke [81] that this mixing was put on a firm theoretical foundation and his work also explains why this naive approach fails.

Recall Eq. 2.53, which we derived through coupling-constant integration as

$$E_{\text{xc}}[n] = \int_0^1 d\lambda \frac{dE_{\text{xc}}^{\lambda}}{d\lambda} = \int_0^1 d\lambda \left[\left\langle \Psi^{\lambda} \middle| \mathcal{V}_{\text{ee}} \middle| \Psi^{\lambda} \right\rangle - U_{\text{H}}[n] \right].$$

As discussed in Section 2.3 we know that the integrand is the effective potential of the Kohn-Sham system at $\lambda = 0$ and the true physical potential at $\lambda = 1$. Using this we can approximate the integrand over the whole range $0 \leq \lambda \leq 1$ as a linear function

$$\frac{dE_{\text{xc}}^{\lambda}}{d\lambda} = \left.\frac{dE_{\text{xc}}^{\lambda}}{d\lambda}\right|_{\lambda=0} + \lambda \left(\left.\frac{dE_{\text{xc}}^{\lambda}}{d\lambda}\right|_{\lambda=1} - \left.\frac{dE_{\text{xc}}^{\lambda}}{d\lambda}\right|_{\lambda=0} \right), \tag{2.55}$$

which we the can use to evaluate Eq. 2.53 as

$$E_{\text{xc}} = \frac{1}{2} \left[\left.\frac{dE_{\text{xc}}^{\lambda}}{d\lambda}\right|_{\lambda=0} + \left.\frac{dE_{\text{xc}}^{\lambda}}{d\lambda}\right|_{\lambda=1} \right]. \tag{2.56}$$

Here we know the first term is simply the exact-exchange of the non-interacting Kohn-Sham system and for the latter term we can use the LDA as an approximation, resulting in

$$E_{\text{xc}} = {}^{1}\!/_{2} \left(E_{\text{x}}^{\text{exact}} + E_{\text{xc}}^{\text{LDA}} \right), \tag{2.57}$$

for which Becke coined the term half-and-half mixing. This functional is also known as B3PW91.

Perdew generalized this approach by arguing, that rather than using linear interpolation in Eq. 2.55 any polynomial

$$E_{\text{xc}}^{\lambda} = E_{\text{xc}}^{\text{LDA/GGA}} + \left(E_{\text{x}} - E_{\text{x}}^{\text{LDA/GGA}} \right) (1 - \lambda)^{n-1} \tag{2.58}$$

could be used to interpolate E_{xc}^{λ} between $0 \leq \lambda \leq 1$, where the lowest possible n is chosen, such that

$$E_{\text{xc}}^{\lambda} = c_0 + c_1\,\lambda + ... + c_{n-1}\,\lambda^{(n-1)} \tag{2.59}$$

reproduces the λ-dependence of E_{xc}^{λ} reasonably well [3]. In this scheme the case of $n = 2$ reproduces Beckes approximation (Eq. 2.55), while $n = 1$ reproduces the naive approximation of Eq. 2.54, which fails because $E_{\text{xc}}^{\lambda} = \text{const.}$ is simply too crude of an approximation for the behaviour of E_{xc}^{λ}. Perdew argues, that $n = 4$ is the appropriate choice, since fourth-order Møller-Plesset perturbation theory [86] is the lowest order to give accurate atomization energies. This results in

$$E_{\text{xc}}^{\text{PBE0}} = E_{\text{xc}}^{\text{PBE}} + {}^{1}\!/\!{}_{4}\left(E_{\text{x}}^{\text{exact}} - E_{\text{x}}^{\text{PBE}}\right), \tag{2.60}$$

which is the PBE0 [3] hybrid functional. Finally, we should mention the HSE-functionals [1, 87], where the authors argue that in large periodic systems screening effects suppress long-range exchange interactions. Therfore, they split the coulomb interaction into a short-range (SR) and a long-range (LR) part using the error-function $\text{erf}(x)$ and its complement $\text{erfc}(x) = 1 - \text{erf}(x)$

$$v_{\text{coul}} = \frac{1}{r} = \frac{\text{erf}(r\,\omega)}{r} + \frac{\text{erfc}(r\,\omega)}{r} = v_{\text{coul}}^{\text{LR}}(r) + v_{\text{coul}}^{\text{SR}}(r). \tag{2.61}$$

After discarding the long-range exchange we get the HSE functionals

$$E_{\text{xc}}^{\text{HSE}} = E_{\text{xc}}^{\text{PBE}} + {}^{1}\!/\!{}_{4}\left(E_{\text{x}}^{\text{exact,SR}} - E_{\text{x}}^{\text{PBE,SR}}\right), \tag{2.62}$$

where ω is determined by fitting to experimental data. The HSE03 [1] functional uses $\omega = 0.15$, which was later improved upon with HSE06 [87] by using $\omega = 0.11$. While disregarding the long-range interactions can speedup the HSE calculations compared PBE0 calculations with certain basis sets [1], the basis set used by FLEUR does not allow for this.

2.4.1 Performance of hybrid functionals

Hybrid functionals provide a significant boost in terms of accuracy for certain material classes. While they provide a moderate improvement to the accuracy of geometric parameters such as the lattice constant in the tests done in [70, 88] (PBE0: MRE=0.6, MARE=0.8; HSE06 MARE 0.71), they provide drastically improved predictions for the band gap. PBE0 overestimates the band gap by a mean error of 0.43 eV and a MARE 45%, while HSE provides a more accurate prediction with a slight mean underestimation of 0.23 eV and a MARE of 15% with the test set calculated in [11].

This strong predictive power of hybrid functionals not only for structural but also for electronic properties makes them key ingredients for material screening procedures relying on accurate band gap predictions [89–91].

2.5 Self-consistency cycle

In Section 2.1.3 we saw that the Kohn-Sham equations pose a self-consistency problem, requiring that the Kohn-Sham orbitals ϕ_i reproduce the ground-state density n_0 and vice versa. When solving this self-consistency problem in FLEUR we have to consider two cases. The self-consistency cycles for a local and a non-local exchange-correlation potential are different.

If we are using a local exchange-correlation potential, FLEUR generates an initial guess for the charge density $n_1(\hat{r})$ by solving individual DFT problems for each atom. Then a repeating self-consistent cycle is started with the current charge density n_i. With this density the Kohn-Sham equations are solved resulting in a new density n_{new}. The start density for the next iteration n_{i+1} is then generated by combining n_{new} with the previous densities n_i, n_{i-1}, ..., n_1 using one of a selection of different solvers for self-consistency problems such as straight- [92], Broyden- [93] or Pulay-mixing [94]. Finally a “charge density distance” $\|n_{i+1} - n_i\|$ is calculated to determine if the calculation is converged and if not the cycle is restarted with n_{i+1} as an input. This is sketched in the blue box in Fig. 2.1.

If on the other hand a *non-local* exchange-correlation potential is used the self-consistency cycle becomes a little more complicated. Since the time for the calculation of the non-local potential dwarfs all other parts of a DFT calculation, the non-local potential is not updated together with the local potential, but instead the previously described DFT cycle is performed for a fixed non-local potential and the non-local potential is only updated once the density is converged [27]. In Fig. 2.1 this scheme is indicated as another green cycle around the blue density converging cycle. The goal of these nested self-consistency cycles is to minimize the number of iterations in the outer, more expensive cycle. Due to the lack of an initial guess for the non-local potential, a local potential is used for the first density cycle. Using the first converged charge density and the corresponding Kohn-Sham orbitals the non-local potential is calculated and a non-local potential is used in all following cycles.

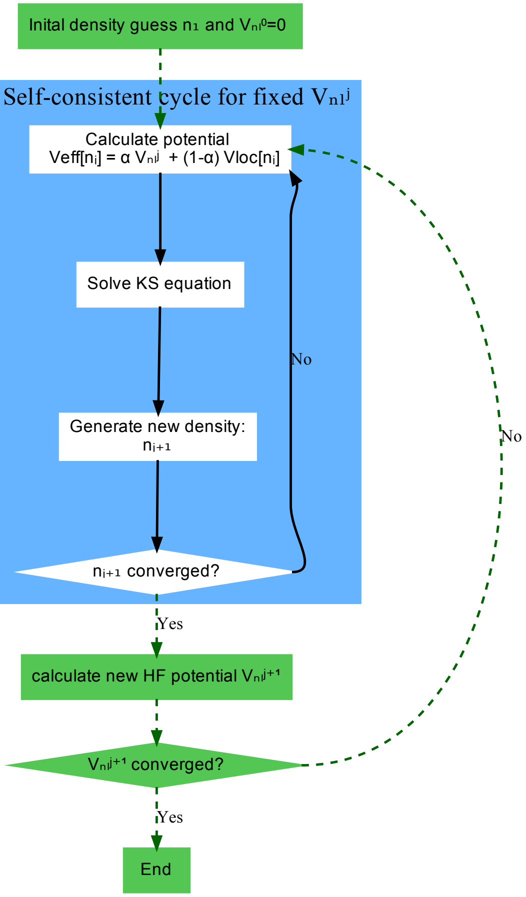

Fig. 2.1.: The self-consistent DFT cycle for a local potential is contained in the blue box. This cycle only converges the density. The whole plot indicates the self-consistent DFT cycle with a non-local potential, with two nested cycles converging the non-local potential and the density. At the beginning of the cycle we start with a guess for the density and a purely local potential. Using the density we set up the effective potential and solve the Kohn-Sham equations to generate a new density, which we mix with the previous iteration until it converges. Once a first density is converged we calculate a non-local potential, which we then use together with the converged density to set up a non-local effective potential and restart the density cycle. This is repeated until both the density and non-local potential are converged.

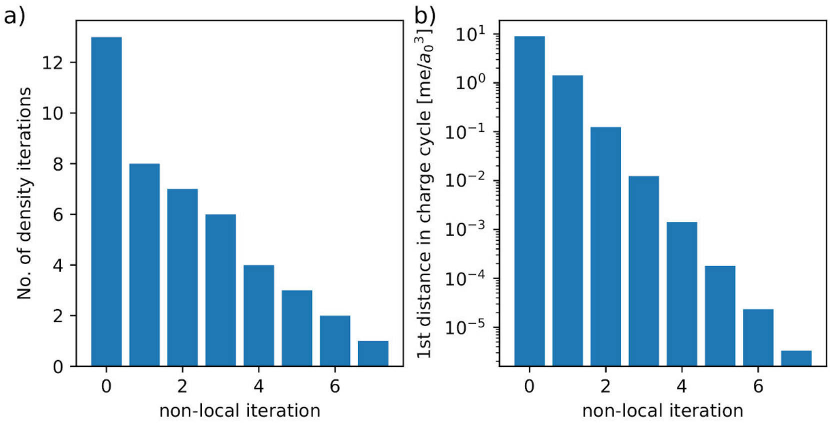

Fig. 2.2.: The number of iterations in each density converging cycle is shown in a) and in b) the first density distance is for each of these cycles is shown. The x-axes indicate the non-local potential iterations performed so far. The underlying calculation is a silicon system with $3 \times 3 \times 3$ $\hat{k}$-points.

Currently the non-local potential in FLEUR does not use a fast mixing scheme such as Broyden or Pulay, but instead uses a straight mixing with a mixing parameter of $\alpha = 1$, i.e. using the newly calculated potential and disregarding all older non-local potentials. This scheme only requires a single non-local potential to be stored at a time which is important because the non-local potential requires much more space than a charge density. The storage size of the charge density grows linearly with the system size and is independent of the number of $\hat{k}$-points used. The non-local potential on the other hand grows quadratically with the system size and linearly with the number of $\hat{k}$-points.

The number of iterations required for either cycle can vary widely depending on the material calculated. In general however, the number of iterations needed to converge the density decreases with an increasing number of non-local potential cycles performed. Fig. 2.2 shows this typical behaviour for a silicon calculation.

3

Full-Potential linearized augmented plane-wave method

In order to calculate solutions to the eigenvalue problem posed by the Kohn-Sham equations we need to express quantities such as the potentials, the density or the wavefunctions in some appropriate basis. FLEUR relies on the so-called *full-potential linearized augmented plane-wave method* (FLAPW) [95], which is among the most accurate methods known [23]. This accuracy is due to the fact that FLAPW can represent the potential everywhere without the need for further approximations. This is in contrast to e.g. plane-wave basis sets, that need to rely on pseudo-potentials to represent the diverging Coulomb potential close to the nuclei. Here the accuracy of the method strongly depends on the underlying pseudo-potentials. FLAPW in contrast allows for first-principles calculations, meaning that the choice of the exchange-correlation functional is the only free parameter in a FLAPW calculation.

In this chapter we briefly introduce the LAPW basis and then use it to construct the so-called *mixed product basis*, which is essential for the calculation of the non-local potential. Finally, we present the Coulomb matrix in this basis. This chapter will follow along the lines of [32, 43].

3.1 LAPW basis

As previously stated plane-wave basis sets are unable to represent the Coulomb potential $V_{\text{coul}}(r) \propto -{}^1\!/_r$ close to the core. This is because accurately representing this potential in a plane-wave basis set would require such a large number of basis functions that this treatment is practically not feasible. Therefore, plane-wave basis sets need to rely on further approximations such as pseudo-potentials. The FLAPW method solves this problem by splitting the unit cell into two kind of domains: In spherical regions centered around each nucleus a *muffin-tin* (MT) orbital basis [96], relying on the products of spherical harmonics and radial functions is used. In between these spheres, in the so-called *interstitial* (IS), a plane-wave basis, is used. This is depicted in Figure 3.1.

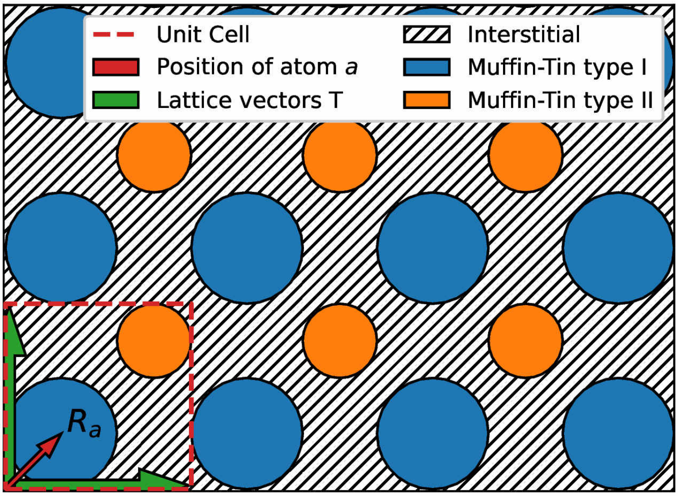

Fig. 3.1.: The unit cell is divided into two kinds of domains. The interstitial (striped area), where plane-waves are used as a basis and the muffin-tins (solid areas), where atomic bases are used. These two domains are matched at the boundaries so that the overall wave functions are continuous and smooth. Picture: [97]

The basis functions of the LAPW basis are defined as

$$\varphi^{\sigma}_{\hat{k}\hat{G}}(\hat{r}) = \begin{cases} \frac{1}{\sqrt{\Omega}} \exp\left[i\left(\hat{k}+\hat{G}\right)\cdot\hat{r}\right] & \text{if } \hat{r}\in \mathrm{IS} \\ \sum\limits_{lm}\left[\alpha^{a\sigma}_{lm}(\hat{k},\hat{G})\,u^{a\sigma}_l(\|\hat{r}_a\|) + \beta^{a\sigma}_{lm}(\hat{k},\hat{G})\,\dot{u}^{a\sigma}_l(\|\hat{r}_a\|)\right] Y_{lm}(\hat{e}_a) & \text{if } \hat{r}\in \mathrm{MT}_a \end{cases}, \tag{3.1}$$

where $\hat{k}$ and $\hat{G}$ are the Bloch- and reciprocal lattice-vectors. $\hat{r}$ denotes the position, while $\hat{r}_a := \hat{r} - \hat{R}_a$ is the position relative to the center of the muffin-tin and Ω is the volume of the unit cell. $\hat{e}_a = {}^{\hat{r}_a}/_{\|\hat{r}_a\|}$ is the unit vector in direction of $\hat{r}_a$. The angular momentum l and magnetic quantum number m are the indices of the spherical harmonic Y_{lm}. In this chapter we will explicitly denote the spin with σ. u and $\dot{u}$ are radial functions, where u is the solution to the radial Schrödinger equation and $\dot{u}$ is its energy derivative. In order to shorten Eq. 3.1 we omitted the energy dependence of $u^{a\sigma}_l$, $\dot{u}^{a\sigma}_l$, $\alpha^{a\sigma}_{lm}(\hat{k},\hat{G})$ and $\beta^{a\sigma}_{lm}(\hat{k},\hat{G})$.

In practice we define cutoff parameters $\mathcal{K}_{\max} > \left\|\hat{k}+\hat{G}\right\|$ and $l^a_{\max} > l$ to limit the size of this basis. Since this cutoff is applied to $\left\|\hat{k}+\hat{G}\right\|$ rather than just to $\left\|\hat{G}\right\|$ the basis set size at each $\hat{k}$-point might differ slightly.

Since FLEUR calculates periodic systems, its wave functions need to have the form of Bloch functions [98]:

$$\psi_{\hat{k}}(\hat{r}) = s_{\hat{k}}(\hat{r})\, e^{i\hat{k}\cdot\hat{r}}, \tag{3.2}$$

where $s_{\hat{k}}(\hat{r})$ is a lattice periodic function and $\hat{k}$ is the so-called *wave-* or *Bloch-vector*. In Eq. 3.1 $s_{\hat{k}}(\hat{r}) = {}^1\!/\!\sqrt{\Omega}\exp\left(i\hat{G}\cdot\hat{r}\right)$ is lattice periodic part in the interstitial, since the $\hat{G}$s are the lattice vectors. The coefficients $\alpha_{lm}^{a\sigma}(\hat{k},\hat{G})$ and $\beta_{lm}^{a\sigma}(\hat{k},\hat{G})$ are chosen such that the wave function at the boundary between the muffin-tins and the interstitial is continuous and smooth. This choice also ensures, that Bloch's theorem holds inside the muffin-tins. The exact choices for these coefficients are given in [32].

Additionally, we restrict the wave-vectors or $\hat{k}$*-points* to the first Brillouin zone. Different $\hat{k}$-points set up largely independent computational problems, that are only combined to calculate final properties such as the density or density of states. Therefore, $\hat{k}$-points offer an easy and efficient way to parallelize calculations.

With this we can require u to be normalized

$$\langle u|u\rangle = \int_0^{R_{\mathrm{MT}}} u^2\, r^2 dr = 1 \tag{3.3}$$

and can derive u by solving the radial Schrödinger equation

$$\left[\frac{1}{2}\frac{\partial^2}{\partial r^2} + \frac{l(l+1)}{2r^2} + V_{\text{eff,0}}^{a\sigma}(r) - E_l^{a\sigma}\right] r\, u_l^{a\sigma}(r, E_l^{a\sigma}) = 0. \tag{3.4}$$

$\dot{u}$ is derived by solving the energy derivative of the radial Schrödinger equation

$$\left[-\frac{1}{2}\frac{\partial^2}{\partial r^2} + \frac{l(l+1)}{2r^2} + V_{\text{eff,0}}^{a\sigma}(r) - E_l^{a\sigma}\right] r\, \dot{u}_l^{a\sigma}(r, E_l^{a\sigma}) = r\, u_l^{a\sigma}(r, E_l^{a\sigma}), \tag{3.5}$$

where $V_{\text{eff,0}}^{a\sigma}(r)$ refers to the spherical component of the potential around the center of MT_a. The functions u and $\dot{u}$ are solved on a logarithmic grid, so that the regions close to the core are sampled more densely.

By including u as well as its derivative $\dot{u}$ we can solve the radial Schrödinger equation (Eq. 3.4) up to first order accuracy with respect to $\Delta E := E - E_l^{a\sigma}$

$$u_l^{a\sigma}(r, E) = u_l^{a\sigma}(r, E_l^{a\sigma}) + \dot{u}_l^{a\sigma}(r, E_l^{a\sigma})\,\Delta E + \mathcal{O}(\Delta E^2). \tag{3.6}$$

This first order accuracy is the reason why we refer to this basis as a *linearized* augmented plane-wave basis. Unlike the augmented plane-wave basis [99] this extended basis allows us to choose an energy such that a single $E_l^{a\sigma}$ can be used for the whole valence band [100]. Therefore, only a single diagonalization is required for each $\hat{k}$-point and spin σ, making practical calculations much easier to perform and faster to run.

Kohn-Sham equations

We use these basis functions to express the n-th Kohn-Sham orbital at a certain $\hat{k}$-point as

$$\phi^{\sigma}_{n\hat{k}}(\hat{r}) = \sum_{\hat{G}}^{\|\hat{G}+\hat{k}\|<\mathcal{K}_{\max}} z^{\sigma}_{\hat{G}\hat{k}n}\,\varphi^{\sigma}_{\hat{k}\hat{G}}(\hat{r}). \tag{3.7}$$

Using this expression we can write the Kohn-Sham equations as a linear algebra problem

$$\sum_{\hat{G}}\left[H^{\sigma}_{\hat{G}\hat{G}'\hat{k}} - \epsilon^{\sigma}_{n\hat{k}}\,S^{\sigma}_{\hat{G}\hat{G}'\hat{k}}\right]\,z^{\sigma}_{\hat{G}'\hat{k}n} = 0, \tag{3.8}$$

where $z^{\sigma}_{\hat{G}'\hat{k}n}$ are the eigenvectors and the Hamiltonian is

$$H^{\sigma}_{\hat{G}\hat{G}'\hat{k}} = \int_{\Omega} d^3r \quad \varphi^{\sigma}_{\hat{k}\hat{G}}{}^{*}(\hat{r})\left[-\frac{1}{2}\boldsymbol{\nabla}^2 + V^{\sigma}_{\text{eff}}(\hat{r})\right]\varphi^{\sigma}_{\hat{k}\hat{G}'}(\hat{r}). \tag{3.9}$$

Notice, that Eq. 3.8 is not a regular eigenvalue problem, but rather a *generalized* eigenvalue problem, since we included an overlap matrix $S^{\sigma}_{\hat{G}\hat{G}'}(\hat{k})$. While plane-waves that span the whole unit cell are orthogonal, plane-waves in the LAPW basis are restricted to the interstitial (c.f. Fig. 3.1) and therefore they have a non-diagonal overlap matrix

$$S^{\sigma}_{\hat{G}\hat{G}'\hat{k}} = \int_{\Omega} d^3r \quad \varphi^{\sigma}_{\hat{k}\hat{G}}{}^{*}(\hat{r})\,\varphi^{\sigma}_{\hat{k}\hat{G}'}(\hat{r}), \tag{3.10}$$

which we need to include into our eigenvalue problem. In practice we can map the generalized eigenvalue problem to an auxiliary regular eigenvalue problem by performing a Cholesky decomposition of the overlap matrix:

$$\begin{aligned}
H\hat{v} &= \varepsilon S\hat{v}, && (3.11)\\
H\hat{v} &= \varepsilon LL^{\dagger}\hat{v}, && (3.12)\\
\underbrace{L^{-1}HL^{-\dagger}}_{=:H'}\,\underbrace{L^{\dagger}\hat{v}}_{=:\hat{v}'} &= \varepsilon\,\underbrace{L^{\dagger}\hat{v}}_{\hat{v}'}, && (3.13)\\
H'\hat{v}' &= \epsilon\,\hat{v}'. && (3.14)
\end{aligned}$$

Here we perform a back substitution algorithm to solve the system of linear equations, rather than performing an explicit inversion.

Core states and local orbitals

Another reason why FLEUR is one of the most accurate DFT codes is that it is an *all-electron* code, meaning that all electrons are treated explicitly and we do not rely on approximations such as frozen cores [101]. However, the LAPW basis functions introduced in Eq. 3.1 only cover the valence electrons and are not suitable for the treatment of core electrons, since the wave functions of the core electrons are orthogonal to the LAPW basis [102].

In FLEUR the core electrons are therefore treated separately from the LAPW basis and individually for each muffin-tin. This has a few additional benefits. First, it reduces the size of the basis, enabling faster calculations and second this separate treatment makes it possible to calculate the core electrons fully relativistic using the Dirac equation, while the valence electrons can get a more lightweight scalar- or non-relativistic treatment.

Finally, certain systems can exhibit states that are too low in energy to be suitably represented by the LAPW basis, but are too high in energy so that their wave functions are not orthogonal to the LAPW basis anymore. For these states FLEUR can extend the LAPW basis by adding so-called *local orbitals* (LO) [103]. These are fully contained within the muffin-tins and are zero in the interstitial.

With this we can define a simplified notation

$$u_{lp}^{a\sigma}(r) = \begin{cases} u_l^{a\sigma}(r) & p = 1 \\ \dot{u}_l^{a\sigma}(r) & p = 2 \\ u_l^{a\sigma}(r, E_{\mathrm{LO}}^{a\sigma}) & p \geq 3, \end{cases} \tag{3.15}$$

where $u_l^{a\sigma}(r, E_{\mathrm{LO}}^{a\sigma})$ is the radial function used for the local orbital with energy $E_{\mathrm{LO}}^{a\sigma}$.

3.2 Mixed product basis

Recall the exact exchange introduced in Eq. 2.39, which expressed in terms of Kohn-Sham orbitals is a six-dimensional integral

$$E_{\mathrm{x}}^{\mathrm{HF}} = -\frac{1}{2} \sum_{i,j}^{N_{\mathrm{occ}}} \int \int \frac{\phi_i^*(\hat{r})\phi_j^*(\hat{r}')\phi_j(\hat{r})\phi_i(\hat{r}')}{\|\hat{r} - \hat{r}'\|} d\hat{r} d\hat{r}' \tag{3.16}$$

However, evaluating this integral numerically in the given form is too costly. To reduce the algorithmic complexity of this integration we utilize the so-called *mixed product basis* (MPB) [104]. This basis is designed to represent products of the LAPW basis functions. By employing this basis we can reduce the calculation of the exact exchange from a term involving four Kohn-Sham orbitals to a term involving two MPB functions. Additionally, by representing products of the LAPW basis more efficiently we also significantly reduce the number of basis functions needed to represents these products from $\mathcal{O}({N_{\text{bas}}}^2)$ to $\mathcal{O}(N_{\text{bas}})$.

In the MT spheres a product of two basis functions is given as

$$\begin{aligned} &u_{lp}^{a\sigma}(\|\hat{r}_a\|)Y_{lm}^*(\hat{e}_a) \quad u_{l'p'}^{a\sigma}(\|\hat{r}_a\|)Y_{l'm'}(\hat{e}_a) \\ &= u_{lp}^{a\sigma}(\|\hat{r}_a\|)\, u_{l'p'}^{a\sigma}(\|\hat{r}_a\|) \sum_{\substack{L=|l-l'|,\ldots,|l+l'| \\ M=-L,\ldots,L}} G_{LM,l'm',lm} Y_{LM}(\hat{e_a}), \end{aligned} \tag{3.17}$$

where $G_{LM,l'm',lm}$ are the Gaunt coefficients, defined as

$$G_{LM,l'm',lm} = \int Y_{LM}(\hat{r})\, Y_{l'm'}(\hat{r})\, Y_{lm}^*(\hat{r}) \quad d\Omega. \tag{3.18}$$

For a given L we define a set of radial functions

$$U_{LP}^{a\sigma}(\|\hat{r}_a\|) = u_{lp}^{a\sigma}(\|\hat{r}_a\|)\, u_{l'p'}^{a\sigma}(\|\hat{r}_a\|), \tag{3.19}$$

where P counts over all p,p',l and l', for which the Gaunt coefficients are non-zero. Notice, that the span of this set lies outside of the space spanned by the set of $\{u_{lp}^{a\sigma}\}$. Since set of $U_{LP}^{a\sigma}$ can be highly linear dependent, we reduce the size of this basis further by finding a set of orthogonal vectors that span almost the same space.

This is done by employing the procedure described in [105]. First we calculate the overlap matrix for a given L and a

$$O_{L,PP'}^{a\sigma\sigma'} = \int U_{LP}^{a\sigma}\, U_{LP'}^{a\sigma'}\, r^2\, dr. \tag{3.20}$$

We diagonalize this matrix and the set of eigenvectors whose associated eigenvalues exceed a certain cutoff $\kappa \approx 10^{-4}$ build a minimal orthonormal representation of the space spanned by the set of $U_{LP}^{a\sigma}$. We will refer to this set as $M_{LP}^{a}(\hat{r})$.

The LAPW basis explicitly depends on the spin $\sigma = \uparrow, \downarrow$. In this new mixed product basis the LAPW basis functions of both spin channels are mixed together, so that the resulting MPB builds a single spin-independent basis. Finally, we can construct the Bloch functions inside the muffin-tins as

$$M^a_{\hat{k},LMP} = \frac{1}{\sqrt{N}} \sum_T M^a_{LP} \left(\left\| \hat{r}_a - \hat{T} \right\| \right) Y_{LM} \left(\frac{\hat{r}_a - \hat{T}}{\left\| \hat{r}_a - \hat{T} \right\|} \right) \exp\Big(i\hat{k}(T + \hat{R}_a) \Big), \quad (3.21)$$

where $\hat{R}_a$ denotes the center of the muffin-tin a, while T are the lattice vectors and N denotes the number of unit cells.

In the interstitial region the construction of the MPB is much simpler: The product of two plane-waves is still a plane-wave.

$$\frac{\Theta(\hat{r})}{\sqrt{\Omega}} \exp\left[-i \left(\hat{k} + \hat{G} \right) \hat{r} \right] \frac{\Theta(\hat{r})}{\sqrt{\Omega}} \exp\left[i \left(\hat{k}' + \hat{G}' \right) \hat{r} \right] = \frac{\Theta(\hat{r})}{\Omega} \exp\left[i \left(\hat{q} + \hat{G}'' \right) \cdot \hat{r} \right], \quad (3.22)$$

where $\hat{q} = \hat{k}' - \hat{k}$ and $\hat{G}'' = \hat{G}' - \hat{G}$. The step function Θ restricts the plane-waves to the IS

$$\Theta(\hat{r}) = \begin{cases} 1 & \hat{r} \in \text{IS} \\ 0 & \hat{r} \in \text{MT}. \end{cases} \quad (3.23)$$

Therefore, the normalized MPB in the interstitial region is given as

$$M_{\hat{k},\hat{G}} = \frac{1}{\sqrt{N\Omega}} \exp\Big(i(\hat{k} + \hat{G})\hat{r} \Big) \, \Theta(\hat{r}), \quad (3.24)$$

In order to be able to exactly represent the products of the LAPW basis in the IS, we need to introduce a new cutoff $\tilde{\mathcal{K}}_{\text{max}} = 2\,\mathcal{K}_{\text{max}} + \mathcal{K}^{\Theta}_{\text{cut}}$, where $\mathcal{K}^{\Theta}_{\text{cut}}$ is the cutoff for the step function in reciprocal space. In practice lesser values $\tilde{\mathcal{K}}_{\text{max}} \approx \frac{2}{3} \left(2\,\mathcal{K}_{\text{max}} + \mathcal{K}^{\Theta}_{\text{cut}} \right)$ have proven themselves to deliver accurate results as well.

We combine the MPB in the MT and IS to a single basis, but unlike the LAPW basis, the MPB is not continuous or smooth at the boundary between the MT and the IS. While it would be possible to construct a MPB that is smooth and continuous [32], this is not necessary for the calculation of the non-local potential. This newly constructed basis is also not orthogonal, due to the fact the plane-waves don't extend over the whole unit cell. Therefore, we calculate the overlap matrix

$$M^*_{\hat{k},I} O_{IJ}(\hat{k}) = \int_\Omega M^*_{\hat{k},I} \, M_{\hat{k},J} \quad d^3r, \quad (3.25)$$

where we use I and J as superindices counting over all MPB functions in the IS and MT. This non-diagonal overlap matrix slightly complicates the completeness relation of the mixed product basis to

$$\begin{aligned} 1 &= \sum_J \underbrace{\sum_I \left|M_{\hat{k},I}\right\rangle O^{-1}_{IJ}(\hat{k})}_{=:\left|\tilde{M}_{\hat{k},I}\right\rangle} \left\langle M_{\hat{k},J}\right| \\ &= \sum_J \left|\tilde{M}_{\hat{k},J}\right\rangle \left\langle M_{\hat{k},J}\right| = \sum_J \left|M_{\hat{k},J}\right\rangle \left\langle \tilde{M}_{\hat{k},J}\right| \end{aligned} \tag{3.26}$$

3.2.1 Coulomb matrix

Using this completeness relation we can write the non-local exact exchange from Eq. 3.16 as

$$V^{\text{exact}}_{\sigma,nn'}(\hat{k}) = -\sum_{n''}^{N_{\text{occ}}} \sum_{\hat{q}}^{\text{BZ}} \sum_{IJ} \left\langle \phi^{\sigma}_{n\hat{k}} \middle| \phi^{\sigma}_{n''\hat{k}-\hat{q}} M_{\hat{q},I} \right\rangle \mathcal{C}_{IJ}(\hat{q}) \left\langle M_{\hat{q},J} \phi^{\sigma}_{n''\hat{k}-\hat{q}} \middle| \phi^{\sigma}_{n'\hat{k}} \right\rangle, \tag{3.27}$$

where the Coulomb matrix is

$$\mathcal{C}_{IJ}(\hat{q}) = \iint \frac{\tilde{M}^{*}_{\hat{q},I}(\hat{r}) \tilde{M}_{\hat{q},J}(\hat{r})}{\|\hat{r} - \hat{r}'\|} d^3r d^3r'. \tag{3.28}$$

The evaluation of Eq. 3.27 is the most time-intensive part of typical hybrid functional calculations. It consists of two major parts:

The calculation of the wave function products in the MPB $\left\langle \phi^{\sigma}_{n\hat{k}} \middle| \phi^{\sigma}_{n''\hat{k}-\hat{q}} M_{\hat{q},I} \right\rangle$ is the most time consuming part. Here the calculation in the interstitial is especially expensive, because this calculation relies on a large number of fast Fourier transforms with the large cutoff $\tilde{\mathcal{K}}_{\text{max}}$. The other time consuming part is the vector-matrix-vector products between the Coulomb matrix and the wave function products from the left and the right. In FLEUR the vector-matrix-vector product has been expressed as a matrix-matrix-matrix-product by stacking the vectors of the MPB into matrices. Throughout this thesis we refer to this step as the *triple matrix product*.

Eq. 3.27 dictates a complicated access pattern: While all the $\hat{k}$-points are in the irreducible Brillouin zone, this is not true for $\hat{q}$- or $\hat{k}-\hat{q}$-points. Their wave functions are calculated by applying the appropriate rotations to the wave functions of the equivalent $\hat{k}$-point inside of the irreducible Brillouin zone. This sampling of the Brillouin zone at $\hat{k}, \hat{q}$, and $\hat{k} - \hat{q}$, limits the choice of $\hat{k}$-point meshes to equidistant meshes, so that every $\hat{k} - \hat{q}$ is also on the mesh. Additionally, the Γ-point needs to be part of this mesh in order to accurately treat the diverging Coulomb matrix at $\hat{q} = 0$. Lastly, it should be pointed out that by applying appropriate unitary transformations to the Coulomb matrix it can be made sparse [104]. An example of the sparsity pattern is shown in Fig. 4.1. This greatly accelerates the calculation of the triple matrix products.

In the current and the previous chapter we have introduced the formalisms needed to evaluate and apply hybrid functionals in the LAPW basis. Chapters 4 and Chapter 5 will introduce how this is done in practice for systems containing hundreds of atoms.

4

Node-level performance and its portability

4.1 Introduction

FLEUR is increasingly being using in the context of high-throughput screening calculations [106] or calculations of very large systems [54]. These types of calculations require a colossal amount of compute resources and therefore it is essential that FLEUR uses these resources efficiently. In this chapter we are going to discuss how we attempt to implement a good performance for FLEURs hybrid functional code on a single node and after this we are going to presents measurements of the achieved performance.

Furthermore, it is of little use to accomplish a perfect performance on a single system at the expense of the performance on other machines. The goal is to have a code with a strong performance on all major types of machines, regardless whether it is a laptop or any of the supercomputers available around the world. This goal is called *performance portability*.

If you compare the current list of the top 500 supercomputers [@30] with its four year old predecessor [@29], you will notice that achieving performance portability has become significantly more difficult. Five years ago 92.8% of systems relied on Intel CPUs and only 18.2% of systems used an accelerator architecture. On the current iteration of the list Intels share has shrunk to 86.4%, while the share of systems using accelerators has grown to 29.4%. Even though the accelerator market is currently dominated by NVIDIA, this is also likely to change with AMD supplying accelerators for a number of very large systems in the near future [@107, @108]. This increasing diversity in hardware makes it increasingly difficult to plan for a future-proof performance portable code.

Considering this challenge we decided not to tune custom code for specific machines, but rather to express our code in terms of standard linear algebra and to rely on linear algebra libraries, such as BLAS and LAPACK. By doing this we can utilize the performance tuning efforts the vendors have or will put into their linear algebra libraries. This approach gives us on the one hand highly efficient code and on the other it ensures performance portability, since each vendor has to provide highly efficient libraries for their hardware.

While we managed to replace a lot of custom code with BLAS or LAPACK expressions, not every part of the code can be expressed in terms of linear algebra. In these code sections we used OpenMP [109] to parallelize code on the CPU. OpenMP is an established standard for CPU parallization and every major compiler supports it well. Hence, it seems like a safe assumption to think that OpenMP will still be supported in a decade or more.

Picking a future-proof framework for accelerator programming is a much more difficult task. Here only NVIDIA has hardware widely available and therefore many of the frameworks, such as CUDA, are vendor specific. We decided to implement the custom GPU kernels using OpenACC. While OpenACC is mainly supported by NVIDIA the hope is that it only requires little effort to replace one directive based framework such as OpenACC with another more vendor agnostic framework such as OpenMP.

4.2 CPU implementation

4.2.1 Employing high-level basic linear algebra routines (BLAS)

One example of a transition from custom code to a standard linear algebra expression is the final calculation of the exchange matrix. In order to calculate the exchange matrix the wave function products expressed in the MPB need to be multiplied from the left and right to the coulomb matrix.

As discussed in Sec. 3.2.1 the coulomb matrix is largely occupied with zeros, except for a number of smaller blocks along the diagonal, a big block in the bottom right corner and a few additional very sparse off-diagonal terms. FLEUR makes use of this sparsity by only storing the big and the small blocks as separate arrays of the minimal required size. The additional terms are also stored in a sparse scheme, so that no memory is wasted on storing a large amount of zeros. This has two major advantages. First the size of the matrix is reduced for communication as well as storage allowing for larger calculations and second it improves the speed of any calculation involving the coulomb matrix drastically, since only the big block in the bottom right corner is significant for the runtime. All other parts have a vanishingly small runtime compared to it.

In the legacy implementation the wave function products were applied to the coulomb matrix a single pair of vectors at a time, whereas in the new implementation the wave function products are stacked into matrices and then applied to the coulomb matrix in bulk. This is sketched in Fig. 4.1.

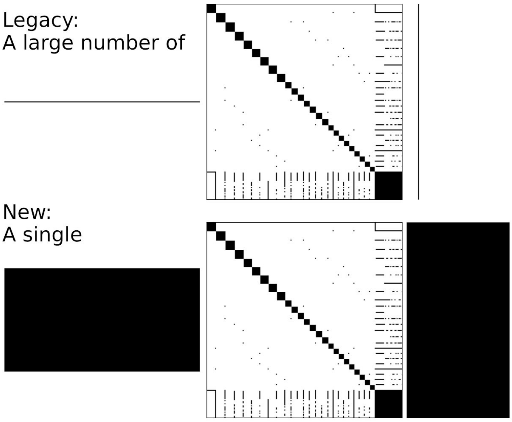

Fig. 4.1.: This graphic sketches the previous and the new method of calculation the exchange matrix. While the previous method relied on calculating single matrix elements, the new method stacks several wave function products into a matrix and calculates large blocks of the exchange matrix at once.

By replacing a large number of matrix-vector- and dot-products with a single triple matrix multiplication we shift the parallelization and optimization of overall calculation from our custom code to a vendor tuned BLAS implementation. The resulting performance improvement is shown in Fig. 4.2.

We compared these implementations for two cases: First we used the MKL without any changes on AMD hardware and second we manipulated MKL so that it assumes it runs on Intel hardware [@110]. In the first case no vectorization is used, while in the second case AVX2 is used. On a single core without vectorization the new implementation is only slightly faster than the legacy implementation relying on matrix-vector products. This improvement is likely due to the better memory access pattern of a matrix multiplication compared to that of a legacy implementation. The true advantage of the new implementation is visible for multiple cores. The matrix vector product is such a short calculation, that the BLAS library gets almost no speedup by using multiple cores. For the matrix multiplication however the BLAS library gets much more work at once and can parallelize it very efficiently. Even for 64 cores the matrix multiplication is paralellized with a parallel efficiency of 53%.

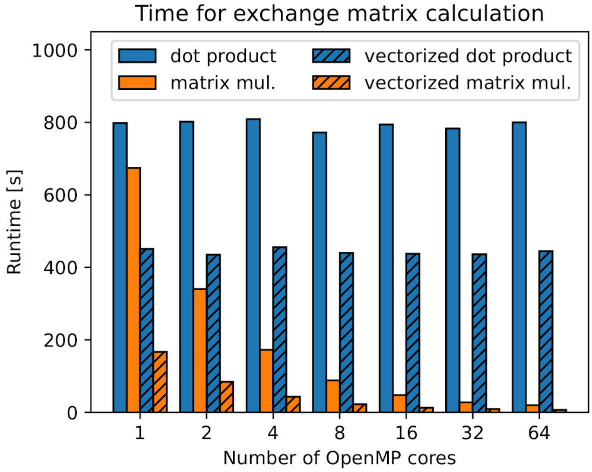

Fig. 4.2.: This graph shows a comparison of four modes to calculate the exchange matrix for a system KAlCl with 24 atoms. All calculations were done on an AMD EPYC 7742 64-core processor with Intels MKL as a BLAS library. For the solid bars MKL was used without any changes, so that MKL does not use vectorized routines on AMD hardware. For the striped bars a trick [@110] was used to suggest to MKL that it is running on Intel hardware and therefore employs vectorized routines. Lower is better.

This effect becomes even stronger when the vectorization is used, where even on a single core the new implementation is more than twice as fast as the legacy one.

This case study shows that our strategy of relying on vendor libraries is correct. The new implementation not only has a better parallelization, but also makes better use of the vectorization and cache sizes. The latter two are very hardware specific and it is therefore difficult to create a single code, that employs them equally well on many different platforms.

4.2.2 Single node scaling

In order to test the overall performance of our shared-memory parallelization we performed strong scaling tests for three tests systems on a single AMD EPYC 7742 64 core processor: A KlAlCl sytem with 24 atoms, a GaTeCl system with 44 atoms and a NaCl supercell with a K defect and with 64 atoms overall. The resulting speedups and parallel efficiencies are shown in Fig. 4.3, while the absolute times for a selection of systems is shown in Fig. 4.4.

For the KAlCl and the GaTeCl system the parallel efficiency is still $> 50\%$ for calculations with 16 OpenMP threads and for the NaKCl system this is still the case with 8 OpenMP threads. In addition to this all systems still experience speedups from 32 to 64 OpenMP threads.

Most parts of the calculation of the non-linear potential show a nice scaling behaviour, with the muffin-tin part of the wave function products and the coulomb matrix setup exhibiting a parallel efficiency of $> 50\%$ even for full usage of the CPU. The triple matrix multiplication also has a strong performance. For the first two systems in Fig. 4.3 it has a good scalability up to 64 cores, and for the last system it still scales to 16 cores. Why the wave function products in the interstitial exhibit a plateau in the speedup will be discussed in the next section.

For a real world calculation the user has to find a balance between the number of OpenMP threads and MPI processes per node. Experience has shown that the extreme cases of all OpenMP or all MPI parallelization within a node usually show a lower parallel efficiency compared to an intermediate setup.

4.2.3 Roofline model

The underlying assumptions for the “ideal” line in Fig. 4.3 are purely based on considerations of the float-point operations (FLOP). In this model we did not consider other factors such as memory transfer. On modern compute architectures these assumptions do not represent reality anymore. In this section we will introduce a model with more realistic assumptions, the so-called *roofline model*.

The *arithmetic intensity* describes the ratio of FLOPs per byte of memory transfer. It can be used to quantify whether a certain routine is limited by memory bandwidth or whether it is limited by the number of FLOPs per second (FLOPS) a CPU can execute. An algorithm with a low arithmetic intensity will be limited by the memory bandwidth without fully utilizing the CPUs compute capabilities. An algorithm with a high arithmetic intensity on the other hand will be purely be limited by the number of FLOPS.

The transition between these two limits can be described in the so-called *roofline model*. In Fig. 4.5 roofline models are shown for a three-dimensional fast Fourier transform and for a matrix matrix multiplication. The x-axis indicates the arithmetic intensity of each algorithm, while the y-axis shows the performance in terms of FLOPS. Additionally we can see two kinds of limits drawn into this roofline model. The memory bandwidth limits linearly increase with the arithmetic intensity and

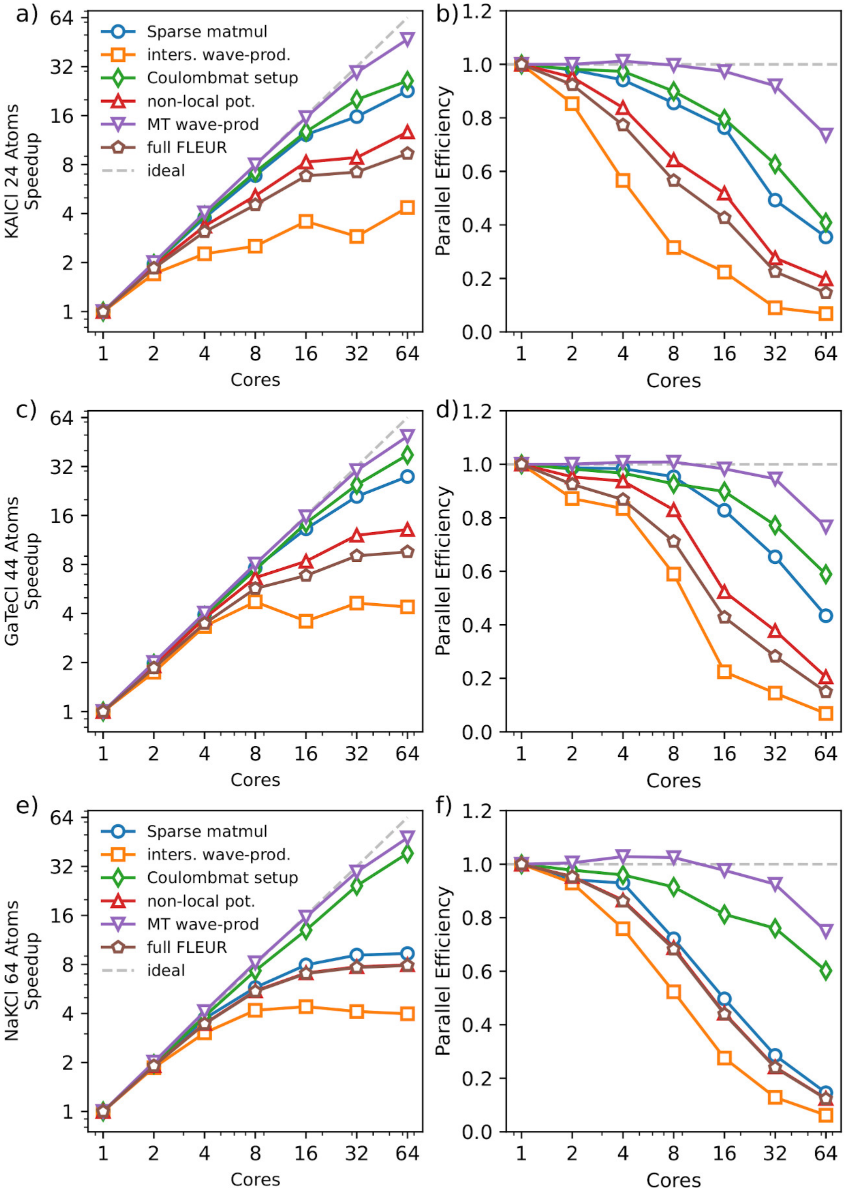

Fig. 4.3.: Here the strong scaling behaviour with OpenMP on a single AMD EPYC 7742 64 core processor is shown. The overall FLEUR iteration is shown with brown pentagons, while the calculation of the non-local potential is shown in red triangles. The four remaining lines show the major parts of the non-local potential.

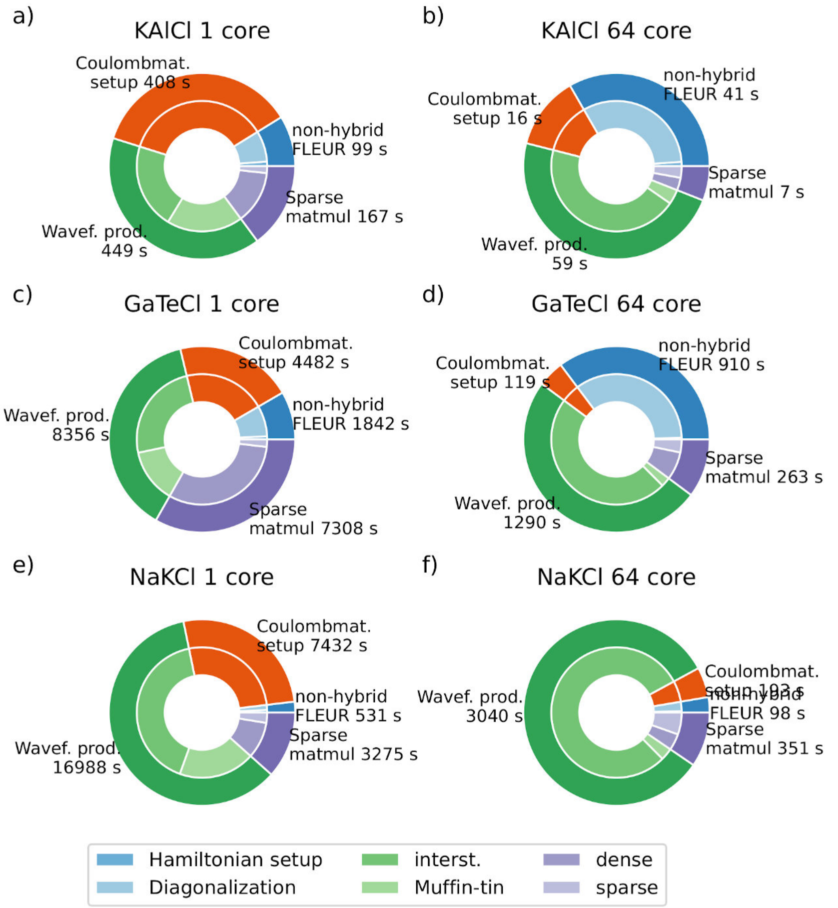

Fig. 4.4.: These pie charts indicate the absolute times of the most relevant code parts. The coulomb matrix setup is shown in orange, all other parts are indicated in the legend. The unit cell size are 24 in the first, 44 in the second and 64 in the third row. For bigger unit cells the non-hybrid part of FLEUR becomes less significant, since is only scales with $\mathcal{O}({n_{\text{atom}}}^3)$, while for large systems the wave function products become the dominant part of the calculation.

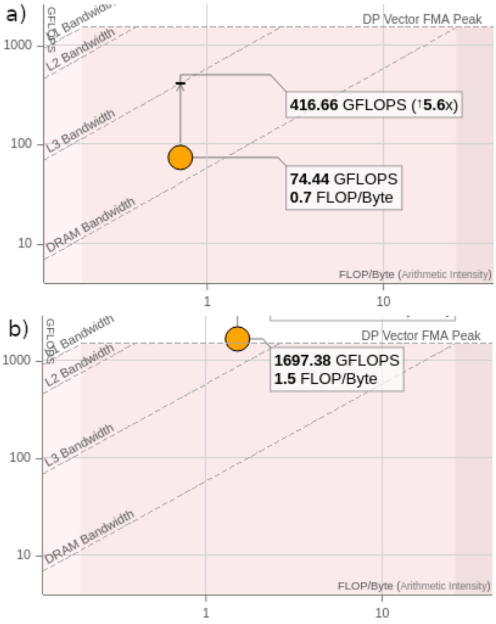

Fig. 4.5.: These roofline models show the limiting factors for two selected routines during the calculation of the 24 atom KAlCl system on a single Intel Xeon Platinum 8160 processor using all 24 cores. The roofline was measured using Intel VTune. In a) the fast Fourier transforms used for the mixed-product basis in the interstitial and in b) the large matrix-matrix multiplications used for the triple matrix product are investigated. Clearly, the fast Fourier transforms are limited by the DRAM bandwidth, while the matrix matrix multiplication is compute bound.

build the “hip” of the roof, if we decide to stick with the roof analogy, while a vertical ridge line is given by the peak FLOPS of the CPU. These theoretical peak FLOPS consider all cores of the CPU as well as all vector units. Due to the complex memory architecture of a CPU with a RAM and three levels of increasingly smaller and faster caches, there are several lines indicating different memory bandwidth limits. By measuring the arithmetic intesity and the FLOPS of a certain algorithm we can use the roofline to determine the factors that limit the execution of this algorithm.

The roofline model in Fig. 4.5 explains why the wave function products in the interstitial do not scale as well as the other code parts. In the interstitial the wave function products are calculated using a fast Fourier transform (c.f. Appendix C), which has an arithmetic intensity of 0.7 FLOP/byte and is limited by the memory bandwidth of the main memory as you can see in Fig. 4.5 a).

Looking at Fig. 4.3 we can see that the speedup of the wave function products in the interstitial reaches a plateau after eight or more cores are used. This is due to the limiting memory bandwidth. If we use four or less cores we reduce the peak performance ridge so much, that this limit is lower than the memory bandwidth limit at 0.7 FLOP/byte. Once we go to eight or more cores the bandwidth limit is lower than the peak performance limit and therefore we can not see any further speedup beyond eight cores.

The matrix multiplication, dominant in the application of the coulomb matrix, on the other hand has an arithmetic intensity of more than twice that. In Fig. 4.5 b) you can see that it not restricted by the memory bandwidth, but rather by the peak performance of the CPU. This is partly because it has a higher arithmetic intensity, but also because Intel MKLs matrix multiplication is so well optimized, that it is not bound by either the DRAM bandwidth nor by the L3 cache bandwidth.

Achieving this with the FFTs is more difficult, because there we have to calculate a large number of small FFTs, unlike the matrix multiplication, where we have a few very big matrix multiplications. This means the time spent in the routine is too small to take advantage of the libraries cache optimizations.

4.3 GPU implementation

In addition to the tuning efforts on CPUs discussed in the previous sections we also created an OpenACC implementation of the hybrid functionals in order to be able to utilize modern accelerators.

In this chapter we will investigate the performance of this implementation on two specific accelerators. We will be comparing the performance of a NVIDIA V100-SXM2 on the CLAIX 2018 [@111] supercomputer or a NVIDIA A100 SXM 40GB on Jureca DC [@112] with the AMD EPYC 7742 CPU from the previous section. Here the A100 GPU represents the most high-end data center hardware NVIDIA currently has to offer and the V100 is the A100s four year old predecessor.

Two key differences distinguish these GPU accelerators from their CPU competitors. Firstly, they have a much larger number of cores, running at a slightly lower frequency, but overall they deliver a much higher peak performance (19.5 TFLOPS on A100, 7.8 TFLOPS on V100, compared to 3.48 TFLOPS on the AMD EPYC 7742) and secondly they utilize a high-bandwidth memory, which outperforms that of the

CPU(206GB/s spec, 64 GB/s measured) by almost one order of magnitude (V100 900 GB/s, A100 1555 Gb/s).

We should note that in addition to differences in the hardware there are also big differences in the performance of the compilers provided by the vendors. This is particularly evident in routines that mostly run on the CPU, such as the Coulomb matrix setup. While Jureca DC, the system running the A100 GPUs, has the same exact CPUs as our CPU reference machine its setup of the Coulomb matrix is drastically slower as evident in Fig. 4.6. If we compare the CPU-only runtime for this routine it is about $2.4\times$ faster (KAlCl: Intel$\rightarrow$ 15.6s; NVIDIA$\rightarrow$ 37.8s) if FLEUR is compiled with the Intel compiler compared to NVIDIAs Fortran compiler.

The implementation for the GPUs mainly focused on the most time consuming parts of the non-local potentials: the wave function products and the triple matrix products are entirely performed on the GPU, while for the coulomb matrix setup only a small part of the calculation is performed on the GPU. One reason for this is, that the coulomb matrix setup occupies only a small part of the total run time and that is scales linearly with the number of $\hat{k}$-points, while the wave function products and triple matrix product scales quadratically with it. The parts of FLEUR, that are unrelated to the non-local potential have also been ported to the GPU, but this effort is not the subject of this thesis and therefore, this chapter neglects the GPU performance of these parts.

For the port to a GPU architecture we paid particular attention to the data movement between the host and the GPU. While the data movement on the GPU and even CPU is very fast, the PCIe Gen4 connection between the GPU and the CPU only has a bandwidth of 64GB/s representing the weakest link in the data moving chain. Therefore, we tried to minimize the amount of data movement across this connection. However, a balance has to be found, since leaving all the data on the GPU during the whole calculation would waste available memory and reduce the size of the biggest systems we can calculate on a given GPU.

During the calculation of the wave function products we therefore only copy the relatively small eigenvector matrix to the GPU and then only copy the final result out to the host. No data movement is required during the calculation, avoiding potentially throttling data dependencies. The triple matrix products work similarly, reducing the communication between host and GPU to input and output arrays at the beginning and end of the routine. Additionally, we created dedicated timers to measure the data movement between host and GPU. Here we never found that any significant time was spent on the data transfers.

Fig. 4.6 the compares the performance for the major parts of the non-local potential calculation on these two types of GPUs to their CPU counterparts. The runtime of the coulomb matrix setup fluctuates wildly even though most of its runtime is spent on the CPU. This is due to the fact, that each of the systems shown have different underlying CPUs and use different libraries for the linear algebra und Fourier transforms. However, the speedup experienced for the whole non-local potential calculation with a single A100 is between $1.8\times$ and $3.9\times$, depending on the system size. For the older V100 the speedup was between $\approx 1.0\times$ and $2.6\times$. In Table 4.1 the runtime and speedup of the major parts is given explicitly.

System	Part	CPU [s]	V100 [s]	A100 [s]	V100 speedup	A100 speedup
KAlCl	sparse matmul	7.34	8.30	3.41	0.88	2.15
	Coulomb setup	15.59	61.79	37.79	0.25	0.41
	wavef. prod. MT	4.49	2.86	2.14	1.56	2.09
	wavef. prod. IS	54.43	9.29	2.06	5.91	26.4
GaTeCl	sparse matmul	263.17	179.89	72.27	1.46	3.64
	Coulomb setup	118.87	540.85	315.37	0.22	0.38
	wavef. prod. MT	60.34	54.55	42.89	1.11	1.41
	wavef. prod. IS	1229.62	91.48	43.01	13.44	28.59
NaKCl	sparse matmul	350.81	308.32	170.26	1.14	1.96
	Coulomb setup	192.87	849.77	518.50	0.23	0.37
	wavef. prod. MT	111.24	140.98	93.19	0.79	1.19
	wavef. prod. IS	2928.41	177.24	89.00	16.52	32.9

Tab. 4.1.: Here the absolute runtimes and the speedups of the GPUs over the CPU implementation of the four major parts of the non-local potential are given.

Some parts such as the interstitial wave function products experience drastic speedups, while other parts such as the muffin-tin wave function products might even slow down in some cases. This is because the very large problem of the interstitial is easily distributed over a GPU and it benefits from the GPUs high bandwidth memory, while smaller problems are more difficult to distribute so that a GPU is fully utilized. However, executing these parts on the GPU, all be it a little slower than on a CPU, is still much faster than copying the data off the GPU and back in order to perform the calculation on the CPU. Additionally, these code parts don't contribute meaningfully to the overall runtime and we can therefore accept a speedup less than $1\times$.

In this section we focused on the performance of a single GPU. Typically a single node will host multiple GPUs. For example CLAIX [@111] uses two per node (one per socket) and Jureca DC [@112] has four per node (two per socket). FLEURs hybrid functionals can utilize multiple GPUs, but this is done using the MPI implementation, which is the subject of the next chapter. Therefore, the reader is referred to Chapter 5 for the discussion of multi GPU scaling.

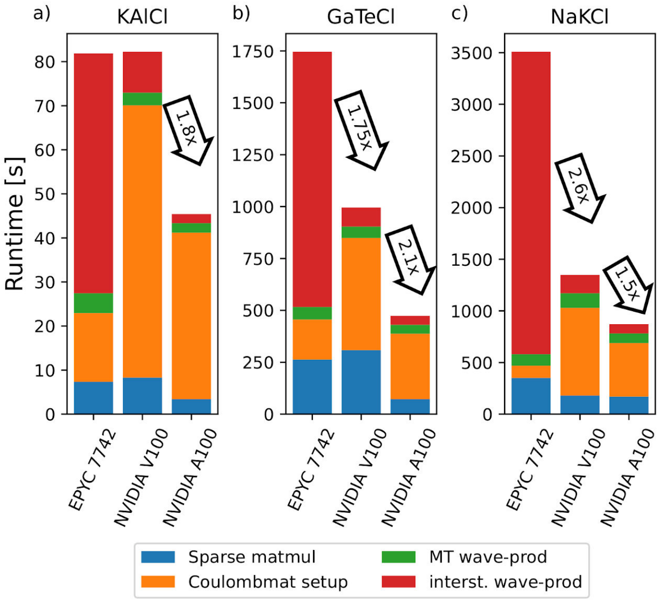

Fig. 4.6.: Here we compare the runtime of the non-local potential calculation of three test systems on three different architectures. The 24 atom KAlCl, the 44 atom GeTeCl and the 64 atom NaKCl previously introduced in this chapter are calculated on three systems: On a AMD EPYC 7742 CPU (cf. Fig. 4.3 and Fig. 4.4), a NVIDIA V100 and a NVIDIA A100 card.

4.4 Conclusion

In this chapter we discussed the efforts to improve the single node performance of FLEURs hybrid functional code as part of this thesis. Additionally, we were able to demonstrate a strong performance on GPUs utilizing tuned external libraries as well as custom OpenACC code, enabling the code to run on future exascale machines. Finally, we were able to demonstrate performance portability, showing that our single code base is capable of efficiently running of vastly different hard- and software architectures.

Multi node scalability

5

In the last chapter we demonstrated, that the hybrid functionals in FLEUR can utilize a single node efficiently, but with this only half the battle is won. In order to calculate large magnetic systems with a fine $\hat{k}$-mesh, we need to combine the compute power of hundreds or thousands of nodes to perform our calculations. This chapter is going do discuss the MPI algorithm we developed for inter node communication and its scalability.

5.1 Data flow for non-local potential calculations

The calculation of the non-local potential is the most time consuming part of a FLEUR calculation with hybrid functionals. This is due to the fact, that it scales with $\mathcal{O}\left({n_{\text{atom}}}^4\right)$, while the rest of the DFT cycle scales with $\mathcal{O}\left({n_{\text{atom}}}^3\right)$. For small systems with less than $\lessapprox 10$ atoms other parts of the algorithm such as the diagonalization or potential generation require a significant share of the runtime, while they become increasingly negligible for larger systems. For large systems with more than fifty atoms the calculation of the non-local potential becomes computationally dominant. Since smaller systems don't need to be calculated on large supercomputers this chapter is focused on the scaling of the non-local potential.

In Chapter 3 (Eq. 3.27) we presented the expression for the non-local potential in the FLAPW basis, which was

$$V^{\text{exact}}_{\sigma,nn'}(\hat{k}) = -\sum_{n''}^{N_{\text{occ}}} \sum_{\hat{q}}^{\text{BZ}} \sum_{IJ} \left\langle \phi^{\sigma}_{n\hat{k}} \middle| \phi^{\sigma}_{n''\hat{k}-\hat{q}} M_{\hat{q},I} \right\rangle \mathcal{C}_{IJ}(\hat{q}) \left\langle M_{\hat{q},J} \phi^{\sigma}_{n''\hat{k}-\hat{q}} \middle| \phi^{\sigma}_{n'\hat{k}} \right\rangle .$$

This equation dictates quite a complex data flow. The mixed product basis combines all wave functions at $\hat{k}$, with the *occupied* wave functions at $\hat{k} - \hat{q}$. The $(\hat{k} - \hat{q})$-points might be outside of the irreducible Brillouin zone and therefore need to be mapped on symmetry equivalent points within the irreducible Brillouine zone. Finally, they are combined with the coulomb matrix for the $\hat{q}$-point. Due to these complex requirements, a good communication pattern is key in order to create a scalable algorithm for the non-local potential.

The MPI algorithm described in this section is designed to achieve a balance between a number of different goals. Firstly, rather to than calculate each matrix element individually we ported the code to rely on large matrix-matrix multiplications (cf. Chapter 4.2.1). However, in order to calculate large unit cells on computing platforms with limited memory such as accelerators, we need to be able to adjust the matrix size to be as big as possible but still within memory constraints. To achieve this we introduced three levels of MPI parallelism that enable us to distribute the problem among a large number of nodes.

A FLEUR calculation with a purely local potential has two levels of MPI-parallelisim: Firstly, a parallelization over $\hat{k}$-points allows different $\hat{k}$-points to be calculated as largely independent problems, which makes this parallelization very efficient. Secondly, each $\hat{k}$-point problem can be further subdivided, for example into groups of bands. This subdivision however requires substantially more communication compared to that of the $\hat{k}$-points.

For non-local potential calculations, we introduced an additional MPI level in order to be able to distribute the problem to more nodes. Fig. 5.1 depicts the distribution of the MPI ranks into subgroups for the non-local potential. Similar to the local potential calculation the highest level of MPI parallelization is between different largely independent $\hat{k}$-point problems. The calculation of the non-local potential at a single $\hat{k}$-point requires a summation over a set $\hat{q}$-points. Therefore, the work for each $\hat{k}$-point is further divided into $\hat{k}\hat{q}$-pairs, that are also largely independent with only a single reduction at the end of the non-local potential calculation. Finally, for the last MPI level the work within a single $\hat{k}\hat{q}$-pair is split into groups of occupied bands. This parallelization requires more communication, but it allows us to limit the size of the largest matrix to a minimum of $n_{\text{basis size}} \cdot n_{\text{total bands}}$ which then has a size on the order of $\mathcal{O}({n_{\text{atoms}}}^2)$. Since this matrix has to be stored on a single node, it turns out to be the bottleneck that determines the largest system we can calculate on a given computing platform. On SuperMUC-NG we were able to calculate systems with up to twohundred atoms, even though SuperMUC-NG only has 90 GB of memory per node.

The complicated symmetries of the extended irreducible brillouin zone determine the number of $\hat{q}$-points associated with each $\hat{k}$-point and their matching. This leads to an equally complicated requirement for the dataflow. Here we have to distinguish between two different cases. The eigenvectors for the $\hat{k}$-points are distributed to a known set of ranks and each rank needs the full set of eigenvectors for its $\hat{k}$-points. For the $\hat{q}$-points on the other hands only a few eigenvectors are needed on each rank and their communication pattern is much less predictable.

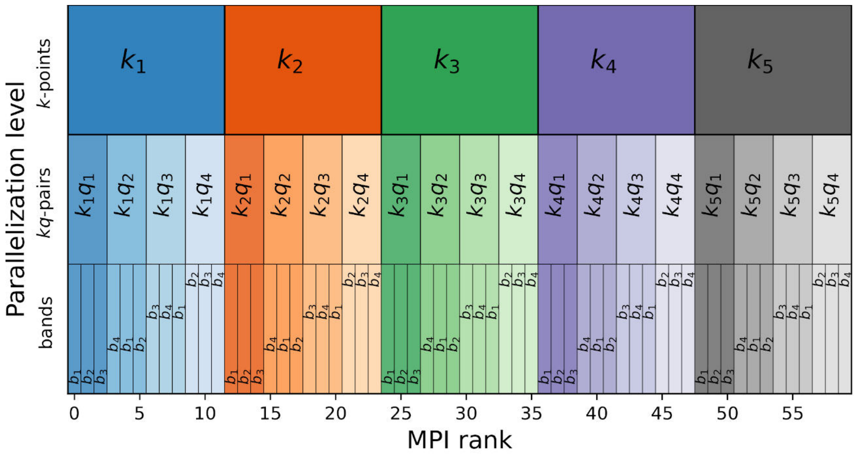

Fig. 5.1.: Here an idealized distribution of 60 MPI ranks for a calculation with 5 $\hat{k}$-points is shown. Groups of 12 ranks work on each $\hat{k}$-point, these groups of are subdivided into 4 groups for each $\hat{k}\hat{q}$-pair. Finally, the occupied bands of each $\hat{k}\hat{q}$-pair are distributed over 3 ranks. In a realistic calculation the $\hat{k}$-points might not have the same amount of $\hat{k}\hat{q}$-pair and the number of occupied bands will also differ depending on the $\hat{k}\hat{q}$-pair. If necessary the algorithm will assign more ranks to certain $\hat{k}$-points and $\hat{k}\hat{q}$-pairs in order to balance the computational load between different ranks.

Hence, we employ two different algorithms for these two cases. The eigenvectors corresponding to the $\hat{k}$-kpoints are broadcast directly from the node where they are stored after the diagonalization to all nodes, that work on this $\hat{k}$-point. This is very straightforward and has almost no performance impact.

For the distribution of the eigenvectors corresponding to the $\hat{q}$-points we rely on MPIs one-sided communication feature. This is easy to implement and if the code is changed it requires very little work to adapt it. However, achieving a good performance is not as straightforward. In the next section we describe the problems associated with one-sided communication and two different approaches how we achieved good performance.

5.2 Problem with one-sided communication

The performance of the one-sided communication within MPI strongly depends on its implementation, which creates challenges for the performance portability of FLEUR. We implemented two different strategies in order to achive good scaling performance regardless of the platform FLEUR is executed on and which MPI library is provided on this platform.

5.2.1 Serialization

Most modern HPC platforms do not support one-sided communication through direct memory access on a hardware level. Therefore, unbeknownst to the developers MPIs one-sided communication still involves communication from the sender and receiver. For most MPI implementations, the node that stores the data only can react to a request, while it is in some routine of the MPI runtime but not if it is in a unrelated routine, such as a large matrix manipulation calculation. If this is the case this can lead to an effective serialization of the code, as is shown in the schematic in Fig. 5.2. After one of the ranks gets all his requests full-filled the rank will go into a, at times lengthy, calculation without any calls to MPI, so that the rank can not respond to incoming requests. This leads all other ranks that depend on data from the rank, which already received its data, to wait. Consequently, in the worst case the ranks will execute their calculations sequentially. Fig. 5.3 shows timing data from a calculation where this serialization occurred. In the following two subsections we will introduce two different solutions to this issue.

Progress thread

The root-cause of the serialization is that, during a lengthy calculation the code will not have any MPI calls and therefore MPI is not able to respond to any incoming requests. This can be circumvented by creating a separate thread, that calls the non-blocking `MPI_IPROBE` every few hundred milliseconds in an infinite loop. In FLEUR we implemented a so-called *progress thread* in C using a separate thread spun up by pthreads. The disadvantage of a progress thread, is that is requires MPI to support multi-threading at least to the level `MPI_THREAD_MULTIPLE`, which is not available on every supercomputer, such as the GPU machine discussed later in this chapter.

Barriers

On machines, where the MPI library does not support a progress thread MPI barriers can be used as an alternative. A global MPI barrier is placed after each set of one-sided communication requests, so that the ranks only start their calculation, after

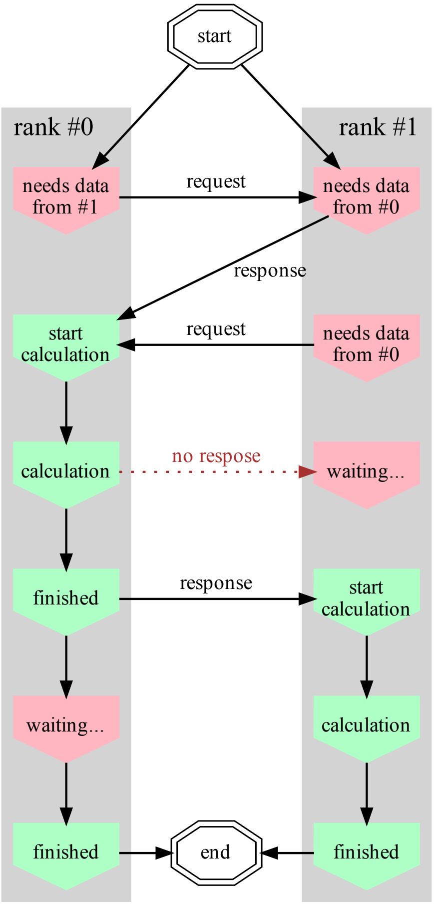

Fig. 5.2.: This schematic shows how one-sided communication can cause serialization. In this scenario both ranks need data from each other. Rank #0 gets its request fulfilled first and immediately starts a lengthy calculation. During this calculation rank #0 is not able to respond to rank #1's request. Rank #1 only gets its request fulfilled after the calculation on rank #0 is finished. Rank #1 only starts its calculation after rank #0 is finished, so that the calculations are not run in parallel, but sequentially.

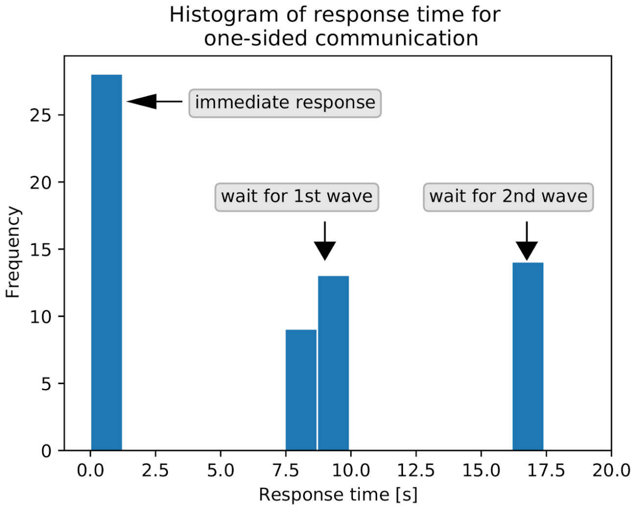

Fig. 5.3.: This plot depicts a typical signature of serialization. A test system was calculated on 64 MPI ranks and the duration of the one-sided communication calls was measured and plotted as a histogram. The x-axis shows the time that passed until the one-sided request was answered and the y-axis show how many ranks fall within a certain time bin. There seem to be three groups, the first group of 27 ranks gets a response to their requests in less than a second, a second group of 24 after 7.5 seconds - 10 seconds and a last group of 15 gets ther requests answered after more than 16 seconds. This is a very typical signature of the serialization thas occurs if neither a progress thread or a barrier is used. The second group only gets their request fulfilled after the first group finishes their calculations of $\sim$ 8 seconds and the last group another $\sim$ 8 seconds later.

every ranks request has been fulfilled. This has a few disadvantages. First of all, this scheme assumes, that all ranks issue one-sided communication requests almost at the same time. Luckily this is the case for our algorithm. Secondly, not every $\hat{k}$-point has the same amount of $\hat{q}$-points, which means, that the ranks, that issue fewer request, need to run through an additional set of "empty" barriers, so that a deadlock is avoided.

Additionally the barriers will force the code execution to be synchronized, even if the workload of intermediate steps is distributed unevenly. Therefore, using barriers creates an unnecessary drag on the performance and a progress thread should be used if possible.

5.2.2 Distributing large arrays

Another issue with the one-sided communication is, that it relies on point-to-point communication. Due to the symmetry in the unit cell it can happen, that a lot of ranks need to access the same eigenvector. If this is done using point-to-point communication, the rank holding the eigenvector would sequentially send it to each node. This communication pattern is particularly ineffcient for large arrays, that are distributed to a lot of ranks.

Because each MPI rank needs the full eigenvector matrix for the $\hat{k}$-point it is working on, we decided not to rely on the sequential one-sided point-to-point communication, but rather distribute these large arrays using MPIs collective communication. For the smaller arrays of the $\hat{q}$-point, each eigenvector is only used a few times. Therefore, the cost of using sequential point-to-point communication is much lower compared to the $\hat{k}$-points.

5.3 Multi-node scaling

In order to be able to calculate large complex materials with the accuracy, that hybrid functionals offer, the calculation of the non-local potential needs to be fast, but more importantly it needs to scale to a lot of computing resources efficiently. In this section we will discuss the scalability we achieved for the calculation of the non-local potential using the parallelization scheme discussed above. Every timing result shown in this section is for a single PBE iteration, followed by a single PBE0 iteration. This is sufficient to show the scaling, since a realistic hybrid calculation will just repeatedly calculate these two iteration types in a self-consistent loop (cf. Fig. 2.1).

5.3.1 Single $\hat{k}$-point performance

As mentioned previously, we implemented three levels of MPI parallelism. In this section we will investigate the lowest level of MPI parallelism, the occupied bands, separately. This is the most difficult level to parallelize because, there are a lot of inter-dependencies between the different occupied bands of a single $\hat{k}\hat{q}$-pair. The parallelization between different $\hat{q}$-point is easier to scale because it has less dependiencies on other $\hat{q}$-points, while the $\hat{k}$-points require even less communication with other $\hat{k}$-points. After a thorough investigation of the scaling of a single $\hat{k}\hat{q}$-pair, we will investigate the scaling of all three MPI levels together.

Strong scaling

The scalability of a HPC applications can be characterized under two different assumptions [113]: *strong scaling* and *weak scaling*. Strong scaling describes the scaling for a fixed size problem, while weak scaling deals with a problem that grows with the computational resources used. First we are going to focus on strong scaling. For strong scaling we investigate how the time to solution decreases for three different test systems when we add more computational resources. We are going to use two measures to asses the strong scalability. First we define the speedup as

$$S_n = \frac{T_{n_\text{min}}}{T_n}, \tag{5.1}$$

where T_n is the time to solution on n computational units and T_{n_min}, is the T_n with the smallest n. In this chapter a single computational units is going to be either a full node on CPU systems or a single GPU on systems with accelerators. Usually n_min is chosen to be 1 but in some cases this is it not feasible for large systems because of memory or time constraints. In these cases we chose the smallest number of computes units possible. Assuming perfect strong scalability the time to solution is inverse proportional to the number of compute units

$$T_n = \frac{T_{n_\text{min}}}{n/n_\text{min}}. \tag{5.2}$$

With this we can calculate the ideal speedup of

$$S_n^{\text{ideal}} = \frac{n}{n_{\text{min}}}. \tag{5.3}$$

This however is only a theoretical concept. In real applications algorithmic limitations, communication and startup overhead [113] will cause the real speedup to be less than ideal. However, we can use the ideal speedup to define the second measure of scalability the *parallel efficiency* of our application as

$$\tau_n = \frac{S_n}{S_n^{\text{ideal}}}. \tag{5.4}$$

In general we will consider a parallel efficiency of $\tau_n \geq 50\%$ as a threshold for a good efficiency and as long as additional resources still add to the speedup we consider this to be useful for most scientific applications, even if the parallel efficiency is lower than 50%.

We measured the scaling of FLEURs hybrid calculations on two different systems: For CPU based scaling we used SuperMUC-NG, which is a 19 PetaFLOP supercomputer consisting of 6480 nodes [@114] where each node has two sockets with a 24-core Intel Xeon Platinum 8174 processor. Here we use two MPI ranks per node. To investigate GPU based scaling we used Jureca-DCs GPU partition, which has 192 nodes, each of which has two AMD EPYC 7742 processors and four Nvidia A100 accelerators [@112]. On this machine we used one MPI rank per GPU.

Fig. 5.4 shows the strong scaling behaviour of three test systems on SuperMUC-NG, while Fig. 5.6 show it for the Jureca-DC module. Fig. 5.5 and Fig. 5.7 show the absolute run times for a few selected calculations. On SuperMUC-NG we tested the strong scaling upto 256 nodes, while the job restrictions on Jureca DC limited us to a maximum of 64 GPUs.

For the GPU implementation we ported three major parts of the calculation to the GPU: The calculation of the wave function products both in the muffin-tins and the interstitial and the triple matrix multiplication between the coulomb matrix and the wave function products from both sides. For the setup of the coulomb matrix only a minor part has been moved to the GPU. All other parts of the non-local potential calculation are done on the CPU. A detailed discussion of the work done for the GPUs can be found in Chapter 4.3.

On the CPU system the scaling measurements were done before we implemented the changes described in Chapter B. Therefore, we manually subtracted the run

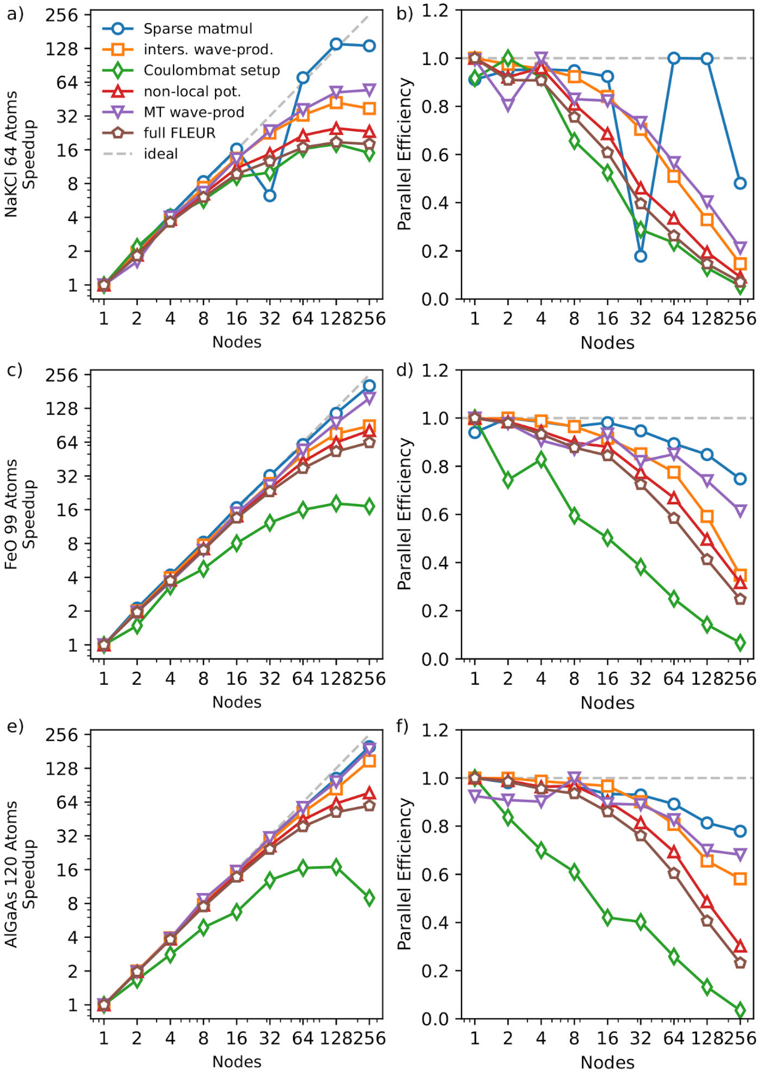

Fig. 5.4.: This figure shows the scaling behaviour of systems with a single $\hat{k}$-point for three different systems on SuperMUC-NG. The top row shows NaCl supercell with 64 Atoms and a K defect, the middle row a 99-atom FeO supercell and a vacancy defect and the last row shows a 120-atom GaAs supercell with an Al defect. The left column shows the speedup, while the right columns shows the corresponding parallel efficiency. In plots a) and b) you can see that the data point for the triple matrix multiplication with 32 nodes has a strange behaviour. This seems to be due to a faulty measurement. In all other calculations of the triple matrix product the runtime is dominated by the multiplication of the dense block at the bottom of the coulomb matrix with the MPB from the right. In this specific run the runtime for this part is reasonable on every MPI rank, but the application of the left hand side wave function products takes an unreasonably long time on a few select MPI ranks. While all of the rank supposedly do the same work some of the ranks only finish after > 8 seconds, while the majority finishes after < 100 milliseconds. The dip in the speedup and efficiency is driven by this artifact. We believe this to be due to a temporary glitch with the supercomputer.

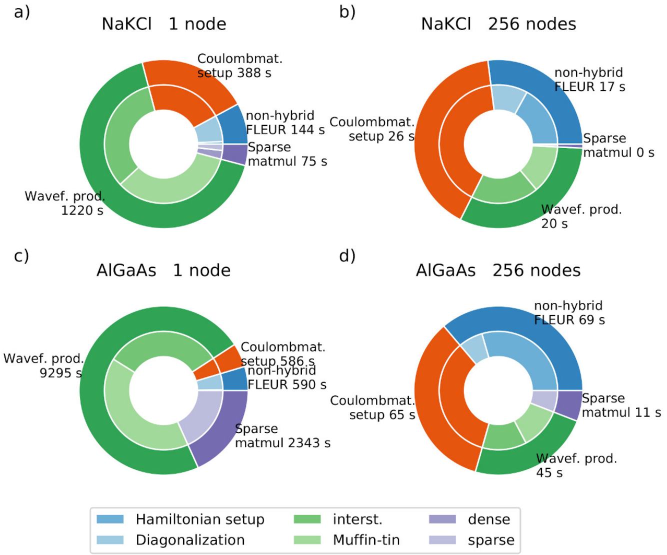

Fig. 5.5.: The absolute run times for a subset of calculations shown in Fig. 5.4 are depicted here. The top row shows the smallest system, a NaCl supercell with 64 atoms and the bottom row shows the biggest system, a GaAs supercell with 120 atoms. The left column shows the run time for calculations run on a single node, while the right shows the run times of the maximum number of nodes. The outer ring splits the total runtime into four main parts: Calculation of the wave function products (green), the setup of the coulomb matrix (orange), the triple matrix product between the wave function products and the coulomb matrix (purple) and everything unrelated to non-local potential is shown in blue. The inner circle subdivides the wave function products into an interstitial (darker green) and muffin-tin part (light green), the triple matrix product into the application of the sparse block diagonal parts and the large dense block in the interstitial.

time of the routine responsible of subtracting the local exchange-correlation potential in order to provide the realistic scaling behaviour.

Fig. 5.5 a) and c) show that on a single node the calculation of the wave function products is computationally dominant requiring $\sim 70\%$ of the total run time. Luckily both routines for the wave function products show a scaling behaviour superior to most other code parts as is shown in Fig. 5.4. The wave function products in the interstitial region are depicted as orange squares while they are shown as purple triangles for in the muffin-tin regions. For the smallest test system with 64 atoms both routines still have a parallel efficiency of $> 50\%$ for 64 nodes or 3072 CPU cores. In the medium sized tests system of 99 atoms, the muffin-tin wave function products have a parallel efficiency $> 50\%$ even for 256 nodes while it drops to

$\sim 40\%$ for the interstitial setup. For the largest test system both routines have a parallel efficiency of $> 60\%$ even for the full 256 nodes. The observed improved parallel efficiency with a growing problem size is due to the fact, that the share of unparallelized code or code with bad scaling shrinks for larger problems. This is a well known fact of high-performance computing [115]. For larger systems the triple matrix multiplication between the wave function products from the left, the sparse coulomb matrix and the wave function products from the right becomes increasingly important. This routine also has a scaling performance with a parallel efficiency of $> 50\%$ for almost all cases.

The setup of the coulomb matrix exhibits the lowest parallel efficiency throughout. This is due to the fact that the setup of the coulomb matrix is very communication intensive and that some of its kernels only can be parallelized over loops, that iterate over short quantities such as the number of atoms or the quantum numbers l and m. In these cases we can only observe a speedup when $n_{\text{MPI}} < n_{\text{atoms}}$ or $n_{\text{MPI}} < n_{lm}$. However this scaling behaviour does not cause a significant slow down of the calculation of the hybrid functionals as a whole. In Fig. 5.5 it is clear that the calculation of the coulomb matrix is not dominant even for 256 nodes and in Fig. 5.4 the scaling of the whole FLEUR iteration scales much better than the setup of the coulomb matrix, further indicating this scaling behaviour is not dominant. Table 5.1 shows, that regardless of system size the end user can pick a parallelization such that a reasonable balance between iteration time and parallel efficiency is achieved.

On the GPU machine the weights of the different routines are shifted. This is partly, because some routine have been ported to the GPU and are therefore accelerated, but it is also due to the fact that the code for the CPU sections is generated by a different compiler and utilizes different libraries. The latter effect is discussed in detail in Section 4.3.

The scaling on the GPUs is quite different compared to that of the CPUs. The scalibility of the sparse triple matrix multiplication is nearly perfect with a parallel efficiency of $> 90\%$ through out. The muffin-tin part of the wave function product scales almost equally well, achieving a parallel efficiency $> 80\%$ throughout. We believe this is due to the fact that here the problem is distributed over 64 GPUs on 16 nodes, compared to the 256 node maximum on SuperMUC. This has two advantages. Firstly, with fewer MPIs it is possible to distribute the problem more evenly. If we assume as an example, that our problem has 300 occupied bands, than we would split it into 64 groups of 4 and 5 or into 256 groups of 1 or 2 bands. In the first case some nodes have to spend 20% of their time idling, while in the latter

Iteration time [s]	CPU			GPU		
Nodes/GPUs	1	64	256	1	16	64
64 NaCl	2145	165	156	2159	363	120
99 FeO	18411	643	490	13508	1024	384
120 GaAs	21718	727	533	13577	1060	375

Tab. 5.1.: Here the time for a single iteration is shown for the three tests systems on different architectures with different numbers of compute resources. Keep in mind that FeO is a magnetic system and that we need to calculate this system for both spins, while we only need to calculate it once for NaCl and GaAs.

case some nodes spend 50% of their time idling. Secondly, distributing the problem over 16 nodes rather than 256 nodes requires less communication, comparable to that of SuperMUC at 16 nodes where the parallel efficiency equally strong. Here however we have $2.1\times$(NaCL), $3.9\times$(FeO) and $4.6\times$(GaAs) faster iterations times on 16 nodes compared to that of SuperMUC with 16 nodes.

5.3.2 $\hat{k}$-point parallelization

In a realistic calculation with hybrid functionals it is not sufficient to calculate a single $\hat{k}$-point and hence only a single $\hat{k}\hat{q}$-pairs. Depending on the system size many $\hat{k}$-points may be required to converge the calculation. This section is going to focus on the scalability of calculations with multiple $\hat{k}$-points. Unlike a calculation with a local potential, which scales $\mathcal{O}\left(n_{\hat{k}}\right)$, a hybrid functional calculation will scale $\mathcal{O}\left({n_{\hat{k}}}^2\right)$ because the exchange matrix is a sum over all $\hat{q}$-points associated with a certain $\hat{k}$-point. In Fig. 5.8 the scaling behaviour of the previously used NaCl supercell is shown for 6 and 10 $\hat{k}$-points.

The scaling behaviour of the parallelization over different $\hat{k}$-points or $\hat{q}$-points is excellent as can be seen in Fig. 5.8. Every part except the setup of the coulomb matrix has a parallel efficiency $> 80\%$. This is expected because the $\hat{k}\hat{q}$-pairs present fairly independent problems and therefore only need very limited communication. In the beginning every rank participates in the distribution of the wave functions through a broadcast and one-sided communication and at the end there is a reduction of the exchange matrix over all $\hat{q}$-points within a single $\hat{k}$-point. There is no communication among $\hat{k}\hat{q}$-pairs in between this.

The key to achieving a good parallelization is to pick the number of ranks so that the $\hat{k}\hat{q}$-pairs can be distributed evenly. In Fig. 5.8 the number of nodes were deliberately chosen to enable an even distribution of the $\hat{k}\hat{q}$-pairs. However in Fig. 5.8 c) and Fig. 5.8 d), the system has 10 $\hat{k}$-points and 205 $\hat{k}\hat{q}$-pairs. Therefore, an even

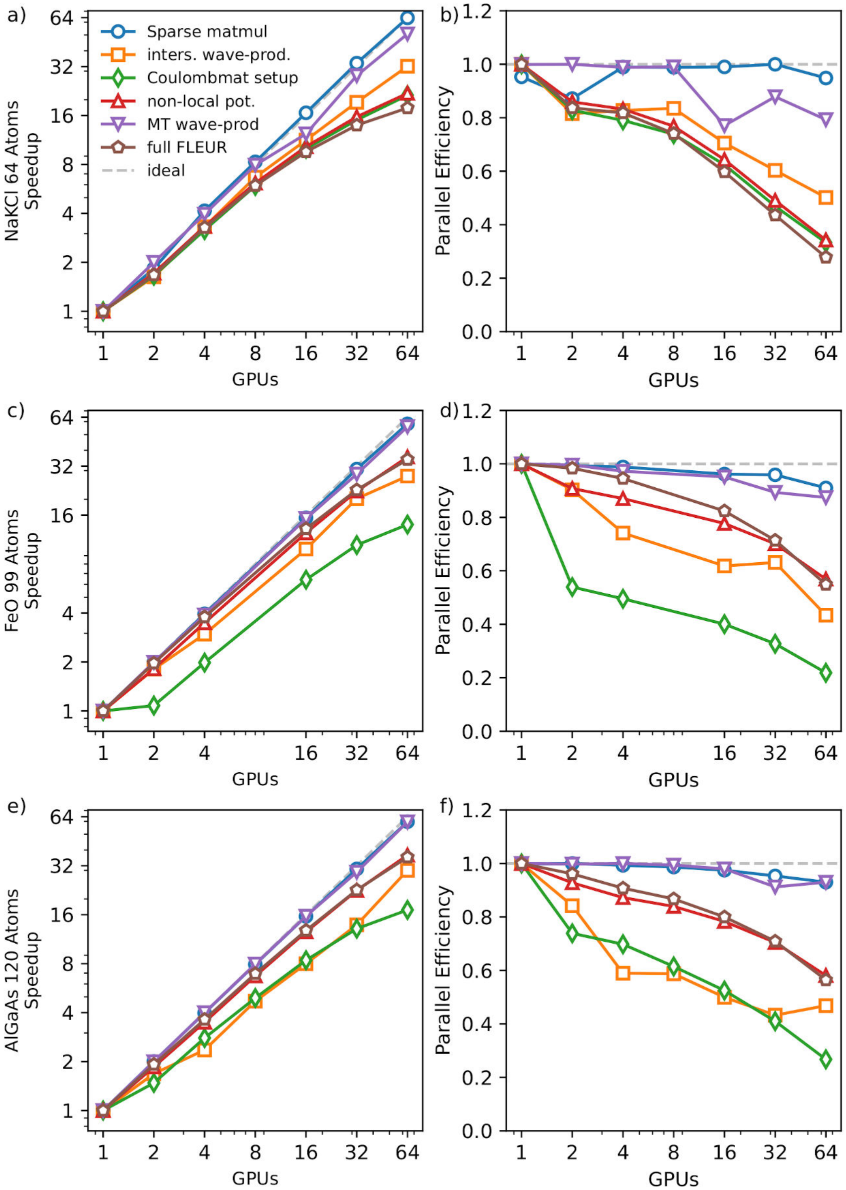

Fig. 5.6.: In these graphs the scaling behaviour for a single $\hat{k}$-point of same systems as Fig. 5.4 are plotted for a GPU architecture. The top row shows NaCl supercell with 64 Atoms and a K defect. The middle row shows a FeO supercell with 99 atoms and a vacancy defect and the last row shows a GaAs supercell with an Al defect with 120 atoms. In the left column you can see the speedup on a double logarithmic scale, while the right columns you can find the corresponding parallel efficiency.

Fig. 5.7.: This figure shows a selection of the absolute run times from Fig. 5.6. The color code and the arrangement are the same as in Fig. 5.5. While the parts that were ported to the GPU such as the wave function products and the sparse triple matrix multiplication are much faster, the parts that remained on the CPU, such as the diagonalization or the coulomb matrix setup are much slower due a different compiler and library set.

distribution over 10 nodes (20 MPI ranks) is not possible. This suboptimal distribution for the calculation with fewer nodes causes the superscalar behaviour in the speedup and a perceived parallel efficiency of $> 100\%$. For all calculation with less than 205 nodes some nodes have to calculate more $\hat{k}\hat{q}$-pairs than others and therefore some are idling. This effect disappears for calculations with 205 and 410 nodes.

The scaling of the coulomb matrix setup is comparatively poor. This is due to the fact that this problem only scales with $\mathcal{O}\left(n_{\hat{k}}\right)$ and is therefore over-parallelized when spread over the same amount of resources as the calculation of the non-local potential. However, this behaviour also causes the run time of the coulomb matrix setup to become negligible with an increasing number of $\hat{k}$-points as can be seen in Fig. 5.9.

Because the number of $\hat{k}\hat{q}$-pairs grows so rapidly with the number of $\hat{k}$-points there only is a band parallization with 40, 80 and 160 nodes for the system with 6 $\hat{k}$-points and for the system with 10 $\hat{k}$-points only for 410 nodes.

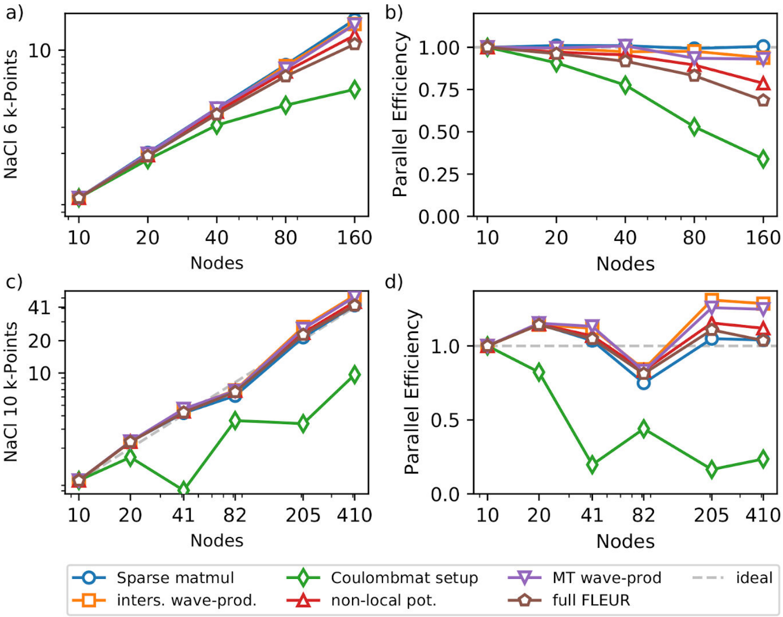

Fig. 5.8.: Here the strong scaling behaviour of multiple $\hat{k}$-points for a 64 atom NaCl supercell with a K defect is shown. The top row shows the strong scaling for 6 $\hat{k}$-points and in the bottom row for 10 $\hat{k}$-points. The number of $\hat{k}\hat{q}$-pairs is 40 in the top row and 205 in the bottom row. The super scalar behaviour for 10 $\hat{k}$-points is due to the fact, that the 205 $\hat{k}\hat{q}$-pairs are divided unevenly by number of MPI ranks.

Combining all three MPI levels gives us an outlook on the scaling potential of this algorithm. If we were for example to calculate the GaAs system with 120 atoms and we would use 8 $\hat{k}$-points we would get 125 $\hat{k}\hat{q}$-pairs. Fig. 5.4 shows that for this system a single $\hat{k}\hat{q}$-pair has a good parallel performance even if distributed over 64 nodes. Therefore, it is reasonable to assume that the calculation of the non-local potential for a system with 8 $\hat{k}$-points would still have good scaling with $64 \cdot 125 = 8000$ nodes, which is ~ 1500 more than the 19 PetaFLOP SuperMUC-NG has. The expected time per iteration would be the slightly more than it is for a single $\hat{k}\hat{q}$-pairs on 64 nodes: 12 minutes 7 seconds.

5.3.3 Weak scaling

While the meaning of strong scaling is very intuitive, it doesn't necessarily reflect real life applications. Being able to calculate a system with twenty atoms in a

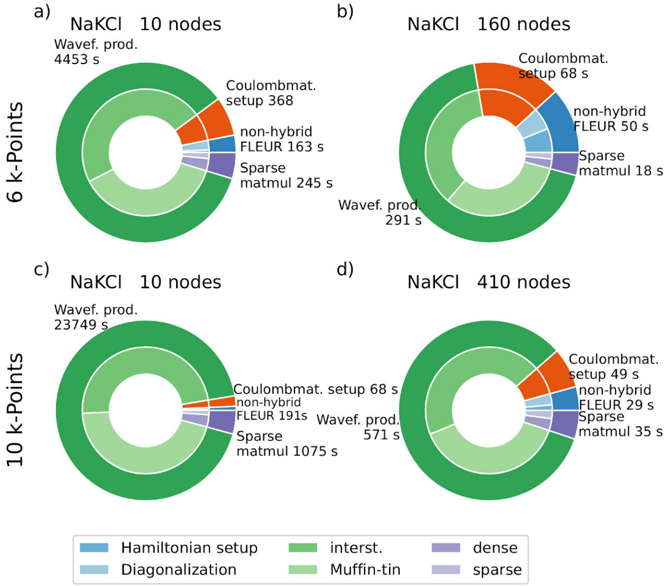

Fig. 5.9.: This chart shows the relation of the run times of the main code parts for calculations with multiple $\hat{k}$-points. The top row show timings for calculations with 6 $\hat{k}$-points, the bottom row with 10 $\hat{k}$-points. The color scheme corresponds to Fig. 5.5 and Fig. 5.7: The wave function products are green, the coulomb matrix setup is orange, the triple matrix product is purple and the remaining parts of FLEUR are summarized in blue.

minute or less does not advance science a lot. Science is advanced by being able to calculate increasingly bigger and more complex systems in a reasonable time frame.

Weak scaling deals with the latter. As discussed previously, the computational demand of a hybrid functional calculation scales with

$$\mathcal{O}\left({n_{\text{atom}}}^4\right). \tag{5.5}$$

We are going to focus on a single $\hat{k}\hat{q}$-pair because the number of required $\hat{k}$-points reduces with system size but is also highly dependent on other external factors, such as the symmetries of the system or the quantities calculated. In Fig. 5.10 two systems, one boron phosphate (BP), and one gallium arsenide (GaAs) were scaled

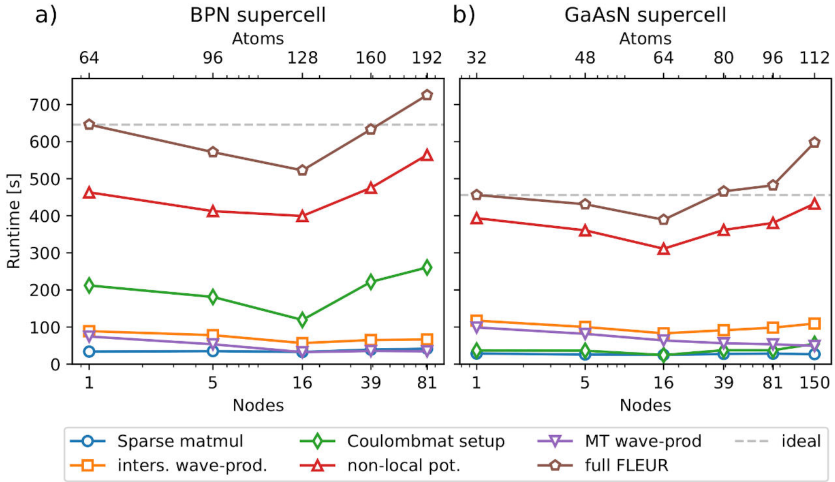

Fig. 5.10.: The weak scaling behaviour of FLEURs hybrid functional calculations is shown for two test systems. In a) a boron phosphate system is scaled into a supercell and a nitrogen defect is added. In b) the same is done for GaAs. The y-axis shows the runtime for different code parts, while the lower x-axis shows the number of nodes used. The upper x-axis shows the number of atoms in that particular system.

into supercells with a single nitrogen defect. Then they were calculated with the parallelization chosen such that

$$n_{\text{nodes}} = \left(\frac{n_{\text{atoms}}}{\min(n_{\text{atoms}})} \right)^4 . \tag{5.6}$$

With ideal weak scaling the run time should be constant regardless of the size of the unit cell, since the computational cost in Eq. 5.5 is canceled out by the additional compute resources chosen in Eq. 5.6. In Fig. 5.10 we show, that the hybrid functionals in FLEUR have a good weak scaling performance. With the BP supercell the runtime of a full FLEUR run rises 12% above its run time on a single node and the calculation of the non-local potential rises 20% about the single node calculation. In the case of GaAs we get similar values a 9% increase for the non-local potential and a 30% increase for a full iteration.

For both systems the runtime does not monotonously increase as you would expect for the weak scaling of a simple algorithm performing a single task. In FLEUR the situation is more complicated. In FLEUR some parts of the code grow with $\mathcal{O}\left({n_{\text{atom}}}^3\right)$ and others with $\mathcal{O}\left({n_{\text{atom}}}^4\right)$. For larger systems the latter will be dominant, but in small systems the first take up a significant share of the run time. In these cases the choice of Eq. 5.6 is not suitable, because the compute resources are increased faster than the computational complexity. This explains the initial dip in the runtime in Fig. 5.10.

5.4 Conclusion

In this section we demonstrated the scalibility of FLEURs hybrid functional implementation. First we used the criterion of strong scaling to show that our algorithm scales efficiently to a large number of compute units. Later we demonstrated, that we are capable of performing calculations with a suitable iteration time using weak scaling. The MPI algorithm used enables a good scalability within as single $\hat{k}\hat{q}$-pair as well as over different $\hat{k}$-points. Taken together this allows us to perform calculations, that fully utilize state-of-the-art supercomputers efficiently and investigate interesting materials such as large complex magnets with a high degree of precision. This ability is key for the emerging field of virtual materials design. A field that attempts to theoretically predict interesting materials by calculating large groups of candidate materials with increasing accuracy, in order to short cut time consuming experiments.

6

Garnets, their electronic structure and magnetic moments

Yttrium iron garnet ($Y_3Fe_5O_{12}$ or short *YIG*) is a complex ferrimagnetic insulator with a number of remarkable applications, such as in magnonics [34], ultra low temperature physics [116] and quantum computing [36]. This success has sparked interest in a related class of materials, the so-called *rare-earth-iron garnets* (RIGs), where the yttrium atom in the YIG structure is replaced with an element of the lanthanide series. Here applications range from materials with giant magnetorestriction [117] to spin Seebeck insulators [118]. Despite great interest in these materials there is only a limited number of theoretical studies of their electronic structure. While YIG has been studied with the electronically rather accurate quasiparticle self-consistent GW (QSGW) [37] method, the whole range of RIG has to our knowledge only been studied with LDA+U [17, 119]. The hybrid functional HSE06 [87] has been applied to Bismuth iron garnet, but only with plane-wave basis sets [120]. This is most likely due to the large unit cells with 160 atoms in the conventional and 80 atoms in the primitive unit cell.

In correlated materials, the Coulomb repulsion becomes increasingly important and competes with the kinetic energy, leading to states that are more localized. Local exchange-correlation functionals such as LDA or GGA struggle to describe this phenomenon accurately. LDA+U attempts to rectify this by adding two additional terms to the exchange-correlation energy

$$E_{\text{xc}}^{\text{LDA+U}}[n(\hat{r})] = E_{\text{xc}}^{\text{LDA}}[n(\hat{r})] + E_{\text{Hub}}[n_{mm'}^{I\sigma}] - E_{\text{dc}}[n^{I\sigma}], \tag{6.1}$$

where $E_{\text{Hub}}[n_{mm'}^{I\sigma}]$ adds an repulsive energy U for electrons present at the same atom and in the same orbital, imitating the treatment of the Coulomb interaction in the Hubbard model [121], and the so-called double-counting term $E_{\text{dc}}[n^{I\sigma}]$ which subtracts the contributions contained in both $E_{\text{xc}}^{\text{LDA}}$ and E_{Hub}. The larger U is chosen, the stronger the localization of the associated states becomes. Additionally, a parameter J can be introduced to model exchange splitting. While the values for U and J can be calculated from LDA [122] or RPA [123], they are often chosen to match experimental results [17, 124]. Hence, such LDA+U results are not calculated strictly from first principles. In complex solids such as RIGs with Fe atoms in different environments and rare-earth atoms with $4f$ electrons more localized than the $3d$ orbitals of Fe, the number of parameters and choice of their size quickly becomes quickly unwieldy. Additionally, the U-parameters are not transferable between electronic structure methods since the length scale of orbital projection is different between different methods.

Materials containing elements with strongly localized f-states, such as the rare-earth elements present in RIGs, are common applications of LDA+U. However, FLEUR's hybrid functionals have in the past been used to accurately calculate exactly these kind of system purely from first principles [28]. While the LAPW basis allows for an accurate description of the localized f-states, hybrid functionals provide an improved ab-initio prediction of their correlated behaviour. Furthermore, hybrid functionals have proven to be suited to predict rare-earth oxides in combination with other codes [125].

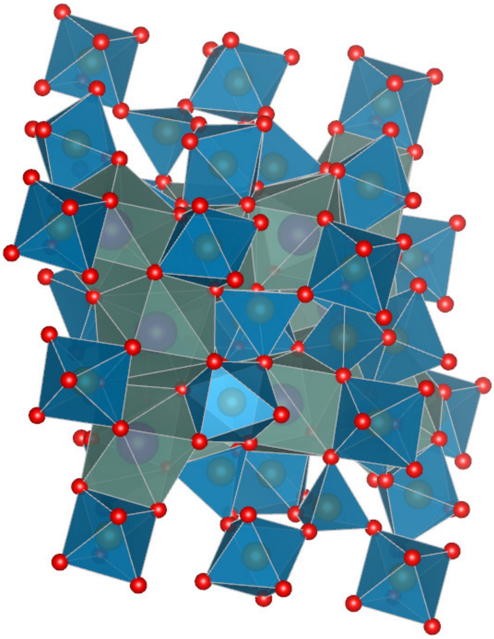

Fig. 6.1.: Unit cell of a garnet. Oxygen nuclei are shown in red, while iron atoms are shown golden inside the blue polyhedra. The rare-earth or yttrium atoms are shown purple inside the grey dodecahedra. While the yttrium or rare-earth nuclei are all symmetry equivalent the iron is present in two different environments as shown in Fig. 6.2. Structure from [126] and plotted with [127].

The typical unit cell of a garnet is shown in Fig. 6.1. The iron atoms in this structure have two types of environments. They are either in the centre of an octahedron or a tetrahedron spanned by neighbouring oxygen atoms. These different iron environments, depicted in Fig. 6.2, have a strong effect on the electronic structure, which is discussed in detail later in this chapter. YIG and most RIGs are ferrimagnets, such that the magnetic moments of the 8 octahedral iron atoms point in the negative direction of the 12 tetrahedral iron atoms, which, for the RIGs discussed here are aligned in parallel with the rare-earth elements. Only a very minor magnetic moment is induced in the yttrium atom.

At the beginning of this chapter we compare the predictions made by a number of different methods for yttrium iron garnet. First we use PBE, a local exchange-correlation functional, and then we perform a series of PBE+U calculations for different value of U and J. Lastly, we calculate this system using PBE0, a hybrid exchange-correlation functional. We discuss the emerging trends and compare our results to previous literature work. Afterwards, we shift our focus to two representative members of the RIGs. We calculate them using PBE as well as our highly-scalable hybrid exchange-correlation functionals. We compare these results to literature calculations performed using LDA+U and then we reflect on the differences and similarities between these RIGs and YIG.

6.0.1 Numerical setup

In previous studies LDA+U was used to describe the Fe $3d$-states or the $4f$-states in the lanthanide series. Here however, there is disagreement on what values for U are appropriate, while Nakamoto et al [17] use a $U = 4$ eV for the iron $3d$-states to determine structural properties, other studies [124] uses a $U = 4.7$ eV. For the rare-earth elements Nakamoto et al [17] suggests a number of values for $U = \{0, 4, 6\}$ eV, depending on which rare-earth is used. Given this uncertainty in a key parameter, it is questionable what predictive power these calculations have.

In order to understand how the choice of the exchange-correlation functional effects the electronic structure of YIG we calculated the density-of-states (DOS) with PBE, PBE+U and with PBE0. For PBE+U calculations we used a series of values for $U = \{1, 2, 3, 4, 5, 6\}$ eV with $J = 0$ eV as well as a calculation with $U = 5$ eV and $J = 2$ eV. All calculations were performed on a $2 \times 2 \times 2$ $\hat{k}$-point grid. We confirmed that the DOS is converged on this grid by comparing the PBE results to results on a denser $\hat{k}$-point grid. We use a smearing of $\sigma = 0.136$ eV for all DOS calculations shown in this chapter. The structural information, e.g. the unit cell and the atom positions used in this chapter are based on the experimental ones, exhibiting a $Ia\bar{3}d$-structure [126, 128, 129]. Unless specified otherwise the muffin-tin radii used in this chapter are $r_{\mathrm{Fe}} = 2.14\ a_0$, $r_{\mathrm{O}} = 1.21\ a_0$ and $r_{\mathrm{Y/Gd/Tm}} = 2.8\ a_0$.

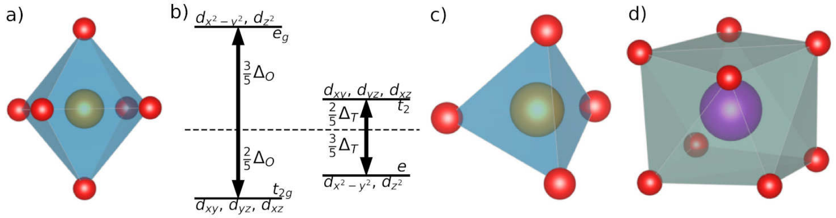

Fig. 6.2.: Unlike the yttrium or rare-earth atom, iron atoms appear in two different environments. In a) the iron atom is positioned at the centre of an octahedron spanned by oxygen atoms at the corners, while in c) it is surrounded by oxygen atoms at the corners of a tetrahedron. In d) the environment of the yttrium atom is shown. In b) the crystal field splitting for Fe in an octahedral environment is shown on the left and in a tetrahedral environment on the right. The magnitude of the crystal field splitting in the tetrahedral environment Δ_{T} is $\approx \frac{4}{9}\Delta_{\mathrm{O}}$, assuming equal distances between the iron and oxygen atoms. Structure in a), c) and d) from [126] and plotted with [127].

6.1 Electronic structure of YIG

As expected, with a value of 0.44 eV, PBE massively underestimates the experimental band gap of 2.8 eV [130]. However, the experimental value relies on optical measurements which are not sensitive to all transitions, potentially missing certain states and thus overestimating the real band gap. The band gap in PBE+U calculations is strongly dependent on the choice of U, where the literature choice of $U = 4$ eV gives a band gap of 1.7 eV, while PBE0 predicts slightly better band gap of 1.83 eV.

In Fig. 6.3 the DOS of YIG is calculated using PBE as an exchange-correlation functional. In this figure, the antiferromagnetic alignment of the iron atoms is visible: the occupied states associated with the tetrahedral iron atoms are mainly in the spin-up channel and the unoccupied ones are in the spin-down channel, while for the octahedral iron atoms the situation is reversed: below the Fermi level the octahedral iron states are mostly in the spin-down channel and above it in the spin-up. Most states associated with the oxygen atoms are occupied, while the yttrium states are largely unoccupied. Below the Fermi level the DOS in the interstitial closely follows the oxygen DOS. Additionally, the DOS associated with both iron types also

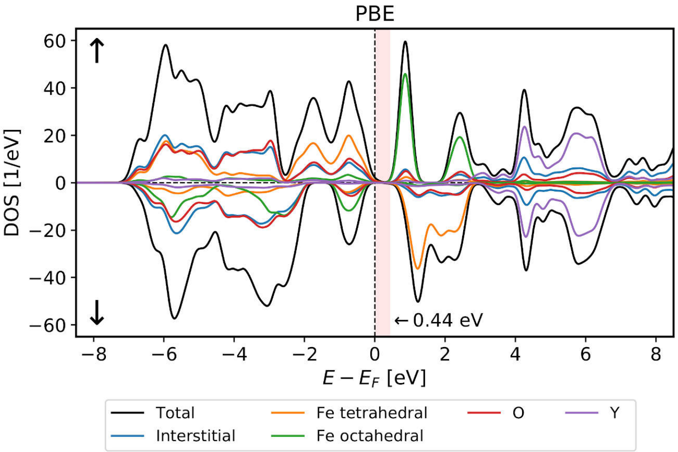

Fig. 6.3.: DOS of YIG calculated with the PBE exchange-correlation function on a $2 \times 2 \times 2$ $\hat{k}$-grid with a smearing of $\sigma = 0.136$ eV. The occupied atoms for the tetrahedral iron are mostly in the spin-up channel, while for the octahedral iron they are mostly in the spin down channel. This leads to the magnetic moments with opposite signs. The states below the Fermi level associated with the oxygen, the iron and the interstitial coincide indicating hybridzation between the $2p$-states in the oxygen and the $3d$-states in iron. For the unoccupied octahedral iron a signature of crystal field splitting is visible with a large peak at 1.5 eV and a smaller one at 2.5 eV.

coincide with the oxygen and interstitial DOS. This indicates that the $2p$-states of the oxygen and the $3d$-states of iron hybridize for both iron environments. This analysis is supported by number of valence electrons found in the different muffin-tin spheres which are 24.5 and 24.2 electrons for iron atoms in the center of the tetrahedral and octhedral environments, respectively, 37.143 electrons in the sphere of yttrium, an average of 5.7 electrons in the spheres of oxygen and 164.1 electrons in the interstitial region. For the unoccupied octahedral iron states we can see a clear signature of crystal field splitting. In crystal field theory, the oxygen atoms are assumed to be negative charge points, which repel the electrons of the iron. Depending on the arrangement of the oxygen atoms around the iron ones, this shifts the energy of the iron $3d$-orbitals differently. In the case of an octahedral oxygen environment this leads the three t_{2g}-states to shift down and the two e_g-states to shift up as is indicated in Fig. 6.1 b). For the octahedral iron atoms, above the Fermi level, there are two distinct peaks, with the lower one containing three and the higher one containing two states, indicating crystal field splitting. For a tetrahedral arrangement of the oxygen atoms, crystal field theory predicts the opposite effect: the e-states are shifted down, while the t_2-states are shifted up. This is visible at an energy around 2 eV in the spin-down channel, however, the peaks are not fully separated and still overlap. This might be explained by the fact that the splitting for tetrahedral arrangement is typically smaller by a factor of $\sim \frac{4}{9}$ compared to an octahedral environment with equal distances. In YIG however, the distance between the iron and the oxygen atoms is 2.0 Å in the octahedral environment, while it is 1.8 Å in the tetrahedron. In contrast to the octahedral case we do not see two distinct peaks but rather two groups of smaller peaks for the tetrahedral iron states. This subsplitting of the t_2-states might be explained by interactions with the second nearest neighbour yttrium atom. While the oxygen atoms at the corners of the tetrahedron are all the same distance from the iron atom in the centre, this is not the case for the yttrium atoms in the vicinity, causing the splitting within the t_2- and e-states. For the occupied iron states, crystal field splitting is not visible for either iron type. The strong hybridization between the oxygen and iron states causes the simple picture of distinct orbitals and point charges at the heart of crystal field theory to break down.

In Fig. 6.4 the DOS of YIG was calculated using PBE+U with $J = 0$ eV and values for U between 1 eV and 6 eV. As discussed in the beginning of this chapter larger U values lead to a stronger localization of the associated states. This is evident for U values between $U = 3$ eV and $U = 5$ eV where the occupied iron states partially loose their hybridization and localized iron d-states appear below the p-states, while the $2p$-states of the oxygen get pushed towards the Fermi level. With a $U = 6$ eV the iron states are localized to an almost atomic-like state around -6 eV. With increasing U the band gap also grows larger. In Fig. 6.5 we calculated the DOS of YIG with a J of 2 eV and a $U = 5$ eV to investigate the effect of exchange interaction on the different iron environments. However, the resulting DOS is not substantially different from that seen for $U = 3$ eV, $J = 0$ eV, which is the corresponding effective $U' = U - J$.

In Fig. 6.6 we plotted the band structure of a PBE+U calculation with a $U = 4$ eV and $J = 0$ eV. By examining this band structure we can explain a phenomena visible in Fig. 6.4, Fig. 6.5 and Fig. 6.7. In these plots the calculated band gap (red area) seemingly ends before there are any states in DOS. In fact however, if once pays close attention there is a state visible at the top of the band gap in each of these figures in the spin-down channel. In the band structure shown in Fig. 6.6 a single state in the spin-down channel is clearly visible in this energy range. Together with a state of the opposite spin, this state builds a direct band gap at the Γ-point. In the energy range of this state the DOS is dominated by contributions from the tetrahedral iron, the yttrium and the interstitial. The large dispersion of this state suggest that it is the highly delocalized yttrium $5s$-state, which seems to hybridize

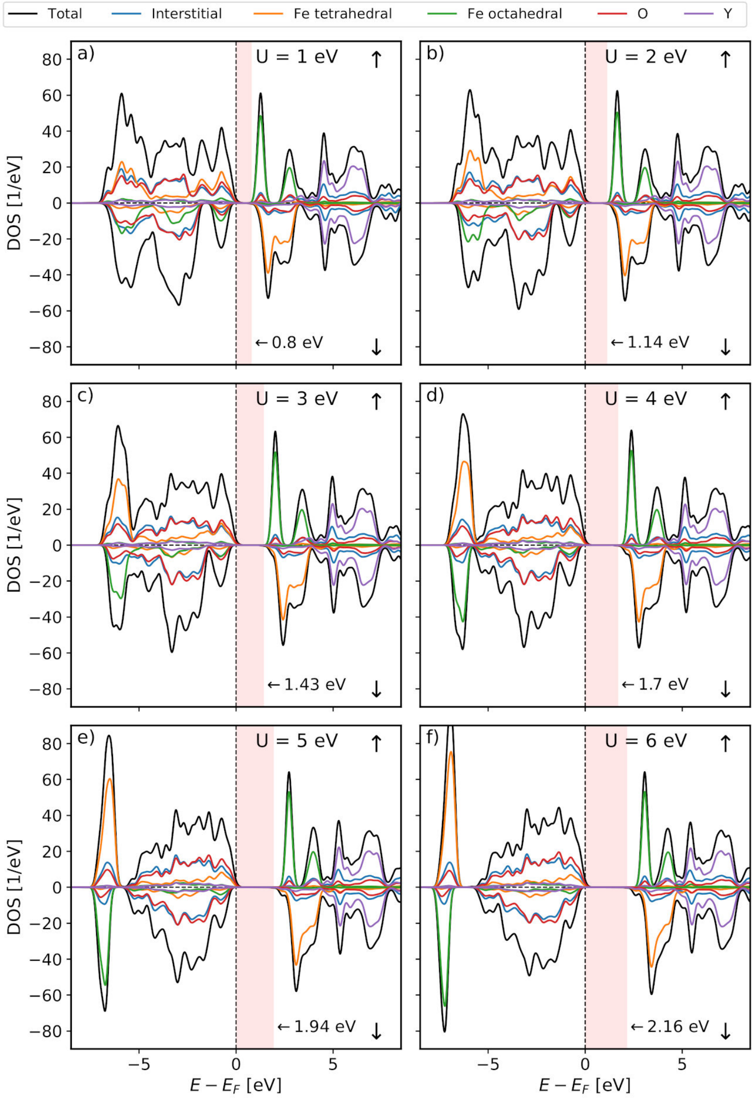

Fig. 6.4.: DOS for YIG calculated with the PBE+U exchange-correlation functional on a $2\times2\times2$ $\hat{k}$-grid for a series of Us with a smearing of $\sigma = 0.136$ eV. With increasing U the hybridization between the iron and the oxygen atoms is suppressed and the states associated with iron become very localized.

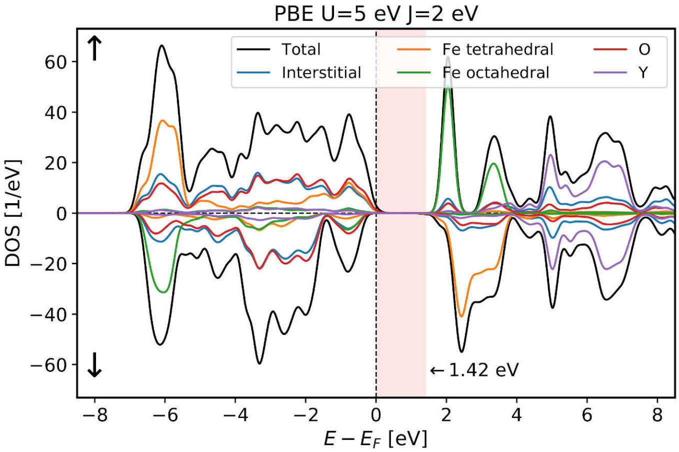

Fig. 6.5.: DOS for YIG of a PBE+U calculation with $U = 5$ eV and $J = 2$ eV on a $2 \times 2 \times 2$ $\hat{k}$-point grid with a smearing of $\sigma = 0.136$ eV. The DOS shown is quite similar to the one seen for $U = 3$ eV and $J = 0$ eV in Fig. 6.4.

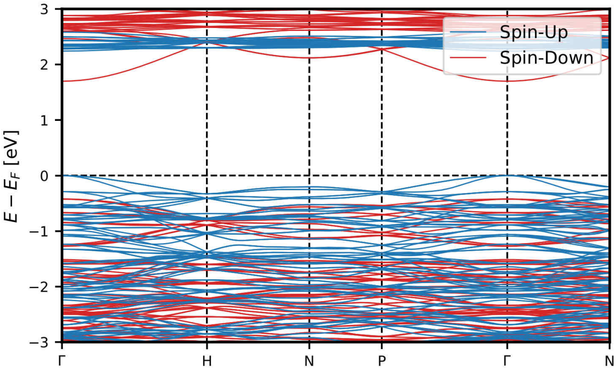

Fig. 6.6.: Bandstructure calculated with PBE+U using a U of 4 eV and $J = 0$ eV. A single spin-down band appears in the region 1 eV $\leq E \leq 2$ eV. A close examination of the DOS reveals, that this state is a hybridization between second nearest neighbours of yttrium and the tetrahedral iron.

with the $2p$-state of the tetrahedral iron atom, even though these atoms are only second nearest neighbours. The transition between the d-orbitals at the bottom of the band gap and the s-orbitals at the top of it is hard to detect using optical measurements, since it violates the selection rule $l - l' = \pm 1$. A detailed discussion of the optical transitions in YIG can be found in [131]. This might explain why the experimentally measured band gap significantly exceeds that of both PBE0 and

QSGW [130]. The distance between the tetrahedral iron and the oxygen nearest neighbour is 1.8 Å, the distance between the yttrium atom and its nearest neighbour oxygen is 2.4 Å and the shortest distance between the yttrium and the tetrahedral iron is 3.1 Å.

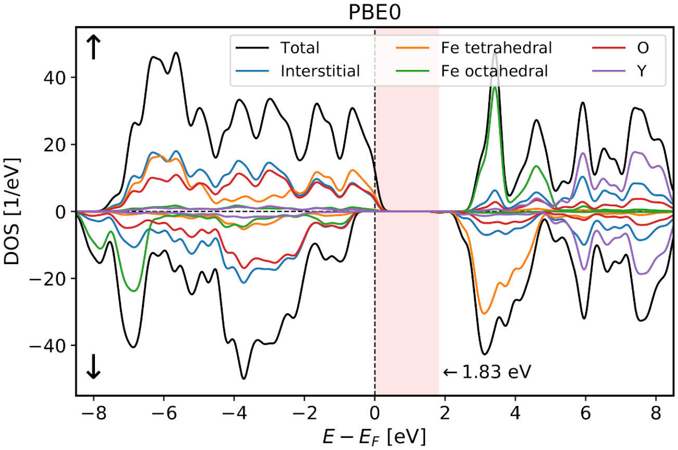

Fig. 6.7.: The DOS of YIG calculated with the hybrid exchange-correlation functional PBE0 on a $2 \times 2 \times 2$ $\hat{k}$-point grid with a smearing of $\sigma = 0.136$ eV. While the tetrahedral iron states (orange) still hybridize, the octahedral iron states (green) below the fermi level are localized at ~ -7 eV.

Finally, in Fig. 6.7 we performed a DOS calculation using the hybrid exchange-correlation functional PBE0. The results are qualitatively different from any of the PBE or PBE+U results. With PBE0 the two types of iron behave quite differently. While the tetrahedral iron $3d$-states hybridize with the $2p$-states of the surrounding oxygen atoms, the octahedral iron $3d$-states are strongly localized around -6.5 eV. The DOS of the tetrahedral iron states resembles its counterpart calculated using PBE, while the octahedral iron DOS exhibits much more localized states similar to the ones seen in calculations with larger $U \geq 4$ eV values. This suggests that using the same Us for the two iron types might not be the correct approach for this system. The Us model a screened *effective* Coulomb interaction, which depends on the electronic environment of the atom. This demonstrates the advantage of a fully ab-initio method such as the hybrid functionals.

For the unoccupied octahedral iron states the typical signature of crystal field splitting is still visible. Between 2 eV and 6 eV there are two distinct peaks in the DOS, corresponding to the t_{2g}- and e_g-states. The unoccupied $3d$-states of tetrahedral iron are also split by the crystal field, with the e-states being lower in energy than the t_2-states. However, while the nearest neighbour environment of the iron atom,

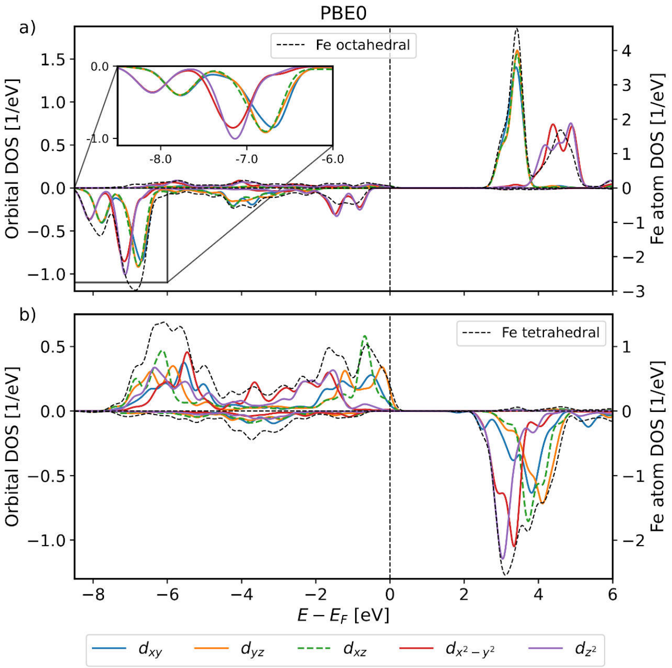

Fig. 6.8.: Orbital decomposition of the density of states shown in Fig. 6.7. In a) the states associated with an octahedral iron atom are shown and in b) they are shown for a tetrahedral iron atom. The y-axis on the left applies to the orbitals given in the legend at the bottom. The y-axis on the right applies to the black dashed line indication the DOS for the whole iron atom. In the octahedral case crystal field splitting is clearly visible for the states above the Fermi level. For the tetrahedral iron atoms the magnitude of this splitting is weaker (cf. Fig. 6.2) and there is a further subspliting of the t_2- and e-states next-nearest neighbour effects.

namely the oxygen atoms, is equivalent along local x-, y- and z-directions, this is not the same for the second nearest neighbour, the yttrium. The distance between the iron and the yttrium is either 3.10 Å or 3.80 Å, depending on the direction. This lifts the degeneracy of the t_2- and e-states, visible in Fig. 6.8. Below the Fermi energy we see another splitting for the octahedral iron. Here however crystal field splitting seems not to be the cause, since contrary to typical crystal field splitting the t_{2g}-states are higher in energy than the e_g-states. Additionally, a stronger effect causing a splitting with a large peak at -7 eV and a minor peak at -8 eV is visible. Therefore, it seems that crystal field splitting is not the dominating effect for occupied states in this system. Rather than this the splittings of the occupied states appear to be driven by different degrees of localization.

6.1.1 Lattice parameter

In order to quantify the effect of the localization of the octahedral iron states, seen with PBE0, we compared the lattice constant predicted for YIG by PBE and by PBE0. This is shown in Fig. 6.9. Both results are very precise and reproduce the experimental lattice parameter within $\lesssim 1\%$. The PBE result is in good agreement with [17].

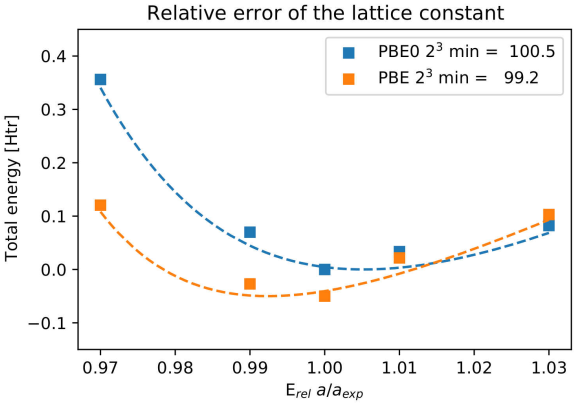

Fig. 6.9.: The lattice constant for YIG calculated using PBE0 (blue) and PBE (orange) on a $2 \times 2 \times 2$ $\hat{k}$-point grid. In this plot all muffin-tin radii were reduced by 5%. The Birch-Murnaghan equation of state is fitted to the data to determine the global minimum.

For PBE0 we can see a trend towards a slightly larger lattice parameter, overestimating it compared to the experimental lattice by about 0.5%, while PBE underestimates the lattice constant by 0.8%. This might be explained with the behaviour seen in the density of states: with PBE0 the octahedral iron atoms hybridize less with the oxygen atoms compared to PBE and therefore the bond strength is reduced, leading to larger lattice parameters.

6.1.2 Magnetic moment

In the introduction we discussed that some of the key applications of YIG rely on its magnetic properties. Therefore, we want to investigate the precision of predictions for magnetic properties with different exchange-correlation functionals. In Table 6.1 we compare the magnetic moments predicted for the different iron atom types. The magnetization calculated for the oxygen and yttrium atoms is negligible regardless of the computational method used. The total magnetization per unit formula was $5\mu_\mathrm{B}$ for every functional. This agreement is expected, since YIG is a magnetic insulator which constrains the total magnetization per unit cell to integer values.

	Fe tetra. $[\mu_\mathrm{B}]$	**Fe octa.** $[\mu_\mathrm{B}]$
PBE	3.52	-3.64
PBE+U=1 J=0	3.70	-3.85
PBE+U=2 J=0	3.81	-3.96
PBE+U=3 J=0	3.90	-4.06
PBE+U=4 J=0	3.98	-4.13
PBE+U=5 J=0	4.04	-4.19
PBE+U=6 J=0	4.11	-4.25
PBE+U=5 J=2	3.90	-4.05
PBE0	3.83	-4.01
PBE+U [17]	4.10	-4.20
QSGW [37]	3.93	-4.17
exp. $R\bar{3}$ [16]	3.95	-4.01
exp. $Ia\bar{3}d$ [16]	5.37	-4.11

Tab. 6.1.: The magnetic moments within the different muffin-tins of both Fe types in units of μ_B. The experimental results [16] in the $Ia\bar{3}d$ symmetry have been called into question [37].

Once again the magnetic moment predicted with PBE+U strongly depends on the choices of U and J. While PBE predicts the magnetic moment of the two iron types within only $0.5\mu_\mathrm{B}$ of the experimental value, the predictions by PBE0 are remarkable close to the experimental results for $R\bar{3}$. If we compare the PBE0-predicted magnetic moment of the octahedral iron to the one predicted by Barker et al [37] obtained using QSGW, another highly precise method, we can see that their calculation predicts a magnetic moment which is $\sim 4\%$ too small, while PBE0 is very precise. Note however, that the magnetic moment of single atoms is never well defined: in FLEUR for example we use the magnetic moment within the muffin-tin and therefore the magnetic moment depends on the choice of muffin-tin radius. The improved accuracy of PBE0 might be explained by comparing the DOS obtained by these two methods where QSGW, unlike PBE0, does not predict the localization of the octahedral iron states. This highly accurate result for the octahedral atoms supports the localization of the $3d$-states seen there with PBE0. The discrepancy between the QSGW and the hybrid functional DOS might be explained by a QSGW calculation that converged into a false local minima, but further experimental work is needed to conclusively determine the electronic structure of this material. The experimental values based on the assumption of a $Ia\bar{3}d$ symmetry cause inaccurate results in the original paper [16] and the method of obtaining this result has been called into question by others [37].

Our calculations of YIG with PBE0 have shown that our implementation is not only correct, but also enables us to calculate predictions in a remarkable agreement with experimental results.

6.2 Rare-Earth-Iron garnets

In this section we investigated two representatives of the rare-earth-iron garnet group. We chose to examine $Gd_3Fe_5O_{12}$ (GdIG) and $Tm_3Fe_5O_{12}$ (TmIg) more closely. We selected these materials, because a lot of interesting experimental [119, 132–136] and even some theoretical work using the FLAPW method [119] has been published for these materials. Additionally, it is currently not possible to combine hybrid exchange-correlation functionals with spin-orbit coupling (SOC) in FLEUR. We believe the electronic configuration these elements (cf. Table 6.2) allows for a fairly accurate description without the need to include SOC into these calculations. In gadolinium the $4f$-states in the spin-down channel are fully occupied and they therefore are not close to the band gap, so that the splitting of the f-states is less consequential. Thulium is missing only one electron to full occupation of the $4f$-states, which we hope is close enough to full occupation to allow for a reasonable description without taking SOC into account.

Element	**Electronic configuration**
Yttrium	[Kr] $5s^2$ $4d^2$
Gadolinium	[Xe] $6s^2$ $4f^7$ $5d^1$
Thulium	[Xe] $6s^2$ $4f^{13}$

Tab. 6.2.: Electronic configuration of the yttrium, gadolinium and thulium.

6.2.1 Electronic structure

In Fig. 6.10 and Fig. 6.11 we present the density of states for GdIG and TmIG, both with PBE and PBE0 as an exchange-correlation functional. Reaching numerical self-consistency (cf. Chapter 2.5) for TmIG was challenging for PBE. We achieved self-consisting by using a few hundred straight mixing iterations with a low mixing parameter, followed by a set of Anderson mixing iterations until convergence was

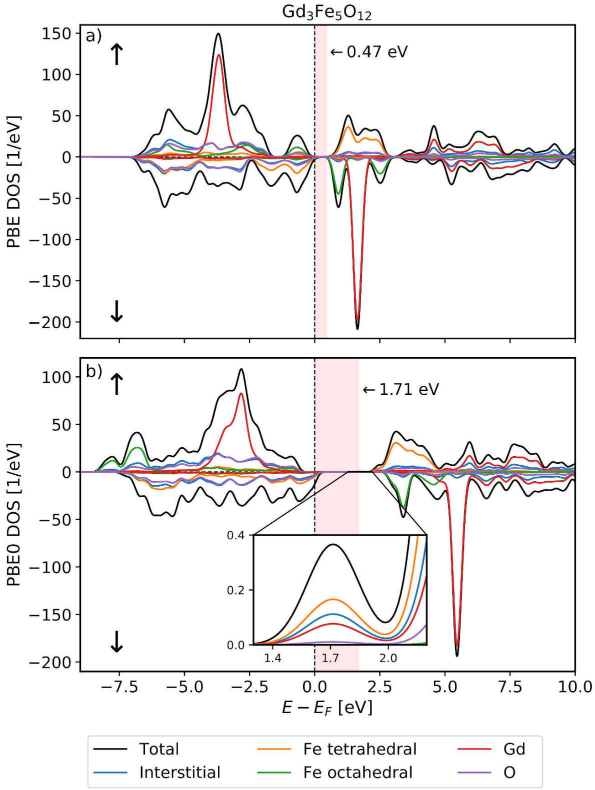

Fig. 6.10.: DOS of gadolinium iron garnet calculated using PBE in a) and PBE0 in b) on a $2 \times 2 \times 2$ $\hat{k}$-point grid. The inset shows the top of the bandgap.

reached. With a converged PBE as a starting density the convergence of PBE0 is straight forward. This difficult convergence is caused by the metallic behaviour of TmIG with PBE as a functional. After the non-local potential is included a gap opens up and all later density convergence cycles do not exhibit this problematic metallic behaviour. GdIG converged without problems both for PBE and PBE0.

For GdIG the band gap was calculated to be 1.7 eV with PBE0. Literature values obtained using PBE+U produce a gap of 1.6 eV [17]. For TmIG we also predict a band gap of 1.7 eV using PBE0. To our knowledge this is the first prediction for the band gap of TmIG. We are not aware of any experimental results regarding the band gap in either system.

The electronic structure of these two systems has a few striking similarities with the ones calculated for YIG. The $3d$-states of both types of iron atoms hybridize with the

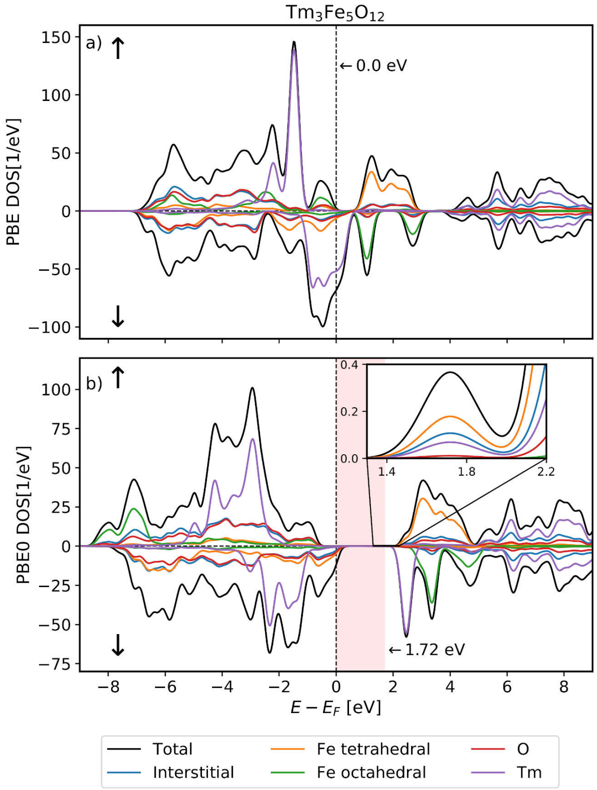

Fig. 6.11.: The DOS of thulium iron garnet calculated using PBE in a) and PBE0 in b) on a $2 \times 2 \times 2$ $\hat{k}$-point grid. The inset shows the top of the bandgap.

oxygen $2p$-states in PBE, while with PBE0 the octahedral iron states localize. This indicates that the tetrahedral and octahedral oxygen environment of the iron atoms cause different effective Us at these atoms, further casting doubt on simple PBE+U predictions for these garnet systems. For the unoccupied octahedral iron states we can see the typical signature of crystal field splitting and in the tetrahedral case this signature is weaker.

The additional $4f$-states of the rare-earth elements in the spin-up channel are strongly localized in PBE, while in PBE0 they hybridize with the freed oxygen $2p$-states. As expected Gd has no occupied $4f$-states in the spin-down channel, while the $4f$-states of Tm are partially occupied, causing a metallic behaviour in PBE. In PBE0 the Tm $4f$-states are separated by a band gap.

6.2.2 Magnetic moments

In Table 6.3 the magnetic moments of all atom types are given. For GdIG we predict a total magnetization per formula unit of $16.0\ \mu_\mathrm{B}$ and for TmIG we predict $1.75\ \mu_\mathrm{B}$ for PBE as well as PBE0. Notice, that the formula unit contains 20 atoms, while the primitive unit cell contains 80. This means, while the magnetic moment per formula unit is not integer, it is integer per unit cell as you would expect for an insulator.

The predicted total magnetic moments are in exact agreement with experimental results for GdIG [137], while they are in good agreement with the experimental value of $1.2\mu_\mathrm{B}$ for the TmIG. This experimental value would correspond to a total magnetic moment of $4.8\mu_\mathrm{B}$ for the primitive unit cell. PBE+U shows a tendency to predict larger magnetizations for almost all atoms: $4.2\mu_\mathrm{B}$ for the octahedral iron, $-4.1\mu_\mathrm{B}$ for the tetrahedral iron, $7.0\mu_\mathrm{B}$ for Gd and $1.9\mu_\mathrm{B}$ for Tm [17].

		Fe tetra.	**Fe octa.**	**Gd/Tm**	**O**
$Gd_3Fe_5O_{12}$	PBE	-3.54	3.69	6.88	-0.06
	PBE0	-3.85	4.04	6.94	-0.06
	PBE+U [17]	-4.1	4.2	7.0	-
$Tm_3Fe_5O_{12}$	PBE	-3.298	3.59	1.61	-0.04
	PBE0	-3.82	4.01	1.93	-0.05
	PBE+U [17]	-4.1	4.2	1.9	-

Tab. 6.3.: Predicted magnetic moments for each atom type are given in GdIG and TmIG.

6.3 Summary

In this chapter we have proven that FLEURs hybrid functionals are capable of predicting the electronic structure and magnetic properties of yttrium iron garnet with a high degree of precision. Giving us the confidence to apply our algorithm to more uncharted systems such as gadolinium iron garnet and thulium iron garnet. While hybrid functionals enable us to calculate these strongly correlated materials fully ab-initio this is not possible with LDA+U, however a recently proposed extension to LDA+U, the so-called *LDA+UV* method [138, 139], promises to have similar capability at lower computational cost. Using our hybrid functional implementation we were able to predict the electronic structure of GdIG, while for TmIG we report the first ever calculation of the electronic structure and the most accurate prediction of its magnetic properties so far.

This shows the potential of the method we have created for this work. Its predictive power and scalability allow for applications beyond the rare-earth-iron garnets discussed here. The calculations presented in this chapter significantly advance the understanding of the garnet materials discussed here. Our predicted electronic structures, specifically for the octahedral iron, significantly deviate from previous predictions. The improved prediction of the magnetic moment for this specific atom supports the results obtained using PBE0. However, in order to settle this question conclusively further experiments are needed.

Conclusion & Outlook

7

In this thesis we extended the application of density functional theory (DFT), currently the most successful approach to realistically describe and predict the properties of solids from the principles of quantum mechanics, towards electronically, structurally and chemically more complex materials. Throughout the thesis, we use the FLEUR code, an all-electron implementation of density functional theory in terms of the full-potential linearized augmented planewave (FLAPW) method known for its high precision and unbiased results. We have created a scalable implementation of FLEUR's hybrid exchange-correlation functionals, capable of utilizing the power of today's most advanced supercomputers. The number of atoms that can be treated with this method has been increased by two orders of magnitude, enabling FLEUR to calculate materials with unit cell sizes larger than $1\,\mathrm{nm}^3$ or more than 100 atoms, respectively, – opening up this high-precision method to the study of complex classes of materials, some of which are envisioned to solve the most pressing technological issues of our times. The innovations presented in this thesis thus represent a significant asset to the field of virtual materials design.

In Chapter 1 the reader is given an overview of the challenges currently facing materials sciences and the opportunities that high-performance computing provides to address them. This is followed by an introduction to density functional theory in Chapter 2, while Chapter 3 concludes the introductory remarks of this thesis with a discussion of the FLAPW method and the mixed-product basis. Chapter 4 discusses the work performed for the single-node performance of FLEUR's hybrid functional implementation. By reformulating the most costly computations as standard math problems, the corresponding routines can rely on widely available libraries rather than custom code, giving them an excellent single-node performance and making the code fairly hardware independent. For routines that are not suitable for this treatment custom parallel kernels using OpenMP on CPUs and OpenACC on GPUs were created. The resulting implementation shows an exceptional performance on both CPUs and on GPUs, both operating close to their theoretical peaks. Chapter 5 shifts the focus to multi-node parallelism and presents a novel parallel algorithm specifically designed for the calculation of FLEUR's hybrid exchange-correlation functionals. The three-level MPI implementation relies on one-sided point-to-point communication as well as collective communication directives. A major challenge of this thesis was the accidental serialization of the one-sided communication. Depending on the underlying MPI library this problem was solved either by using a progress thread or by introducing barriers that artificially synchronize all threads after each one-sided communication call. The resulting implementation shows a first-rate scaling behaviour. While the first two levels show hardly any loss of parallel efficiency, the lowest level scales the work of a single $\hat{k}\hat{q}$-pair to 64 nodes efficiently. The combination of these three levels enables FLEUR to distribute hybrid functional calculations over today's fastest supercomputers and prepares it for the next generation of exascale machines. Making good practical use of a supercomputer does not necessarily mean sending jobs which queue for weeks-on-end and then scale to every single core which the machine has to offer, but rather that we can efficiently use significant portions of the machine to investigate interesting and meaningful systems. To this end FLEUR's new hybrid functional implementation enables scientists to precisely predict properties of large complex systems containing over one hundred atoms with a very short iteration time of less than 10 minutes, allowing them to rapidly review results and build on their findings. In Chapter 6, FLEUR's new hybrid functional implementation is applied to a number of iron garnet materials, a type of synthetic ferrimagnetic materials with garnet crystal structure that has the chemical composition $\mathrm{RE_3Fe_2(FeO_4)_3}$, where RE stands for rare-earth chemical elements or yttrium. We benchmark the predictive power of our code against theoretical and experimental results in the literature, finding that our predictions are on par with most accurate calculations published for yttrium iron garnet (YIG). However, we find details in the electronic structure that differ significantly, such as the effect of the oxygen environment on the $3d$-states of the iron atom. For the gadolinium and thulium iron garnet we perform the first ever precise calculations of the electronic structure and confirm a remarkable similarity to YIG.

The present work pushes the accuracy of the description of the electronic properties of complex solids provided by the combination of hybrid functionals and the FLAPW method to new scientific domains. This is demonstrated by applying this algorithm to synthetic rare-earth-iron garnet materials, a key material class with applications ranging from spintronics to superconducting qubits. The treatment is challenging as the unit cell contains 80 atoms of localized $2p$, $3d$ and $4f$ elements partly at low-symmetry positions exhibiting a ferrimagnetic insulating state. A frequently pursued conceptual approach amending the treatment of the electronic structure by several local Coulomb parameters U is unsatisfactory and provides frequently unsatisfactory results. This work advances the understanding of these systems significantly. For yttrium iron garnet we predict, contrary to previous studies, that the oxygen environment surrounding the iron atoms has a strong effect on the electronic structure and is not merely limited to crystal field splitting. We predict that the $3d$-states of the octahedral iron are strongly localized, while the tetrahedral ones form bonds with the $2p$-states of the surrounding oxygen atoms. Our predictions for YIG are in good agreement with experimental results, further supporting these findings. Furthermore, we performed the most accurate electronic structure calculations for gadolinium iron garnet and thulium iron garnet, two systems that had previously only been studied with local exchange-correlation potentials. We provide the first ever predictions of TmIG's band gap. For both systems we observe that the iron atoms in the octahedral and tetrahedral oxygen environments behave similar to their counter parts in YIG, something also absent from previous studies.

Lastly, it should be pointed out that the applications of this work to the three garnet systems presented in Chapter 6 open a new vista to the treatment of synthetic garnets, natural garnet minerals and minerals in general. There are numerous garnet systems with interesting applications such as yttrium aluminium garnet ($Y_3Al_5O_{12}$) which is used in lasers or highly durable LEDs [140, 141]. Additionally garnets often appear as alloys, such as bismuth-substituted YIG [142], which require even larger unit cells to calculate. Outside of the garnet group, a number of rare-earth oxides have key applications inside solid-state [143, 144] or metal-air [145] batteries, both of which promise to improve the range and safety of electric vehicles and thus accelerate the transition away from fossil fuels. All of these materials have in common that they are rare-earth oxides with complex electronic properties driven by large Coulomb interactions due to localized electrons, have large unit cells and their applications directly depend on the electronic structure. Therefore, FLEUR equipped with the hybrid exchange-correlation functionals extended by the treatment of the spin-orbit interaction and noncollinear magnetism is the ideal method to study these materials classes in the future.

Bibliography

[1] Jochen Heyd, Gustavo E. Scuseria, and Matthias Ernzerhof. “Hybrid functionals based on a screened Coulomb potential”. In: *The Journal of Chemical Physics* 118.18 (2003), pp. 8207–8215. eprint: `https://doi.org/10.1063/1.1564060` (cit. on pp. x, 10, 21).

[2] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. “Generalized Gradient Approximation Made Simple”. In: *Phys. Rev. Lett.* 77 (18 Oct. 1996), pp. 3865–3868 (cit. on pp. xi, 10, 16).

[3] John P. Perdew, Matthias Ernzerhof, and Kieron Burke. “Rationale for mixing exact exchange with density functional approximations”. In: *The Journal of Chemical Physics* 105.22 (1996), pp. 9982–9985. eprint: `https://doi.org/10.1063/1.472933` (cit. on pp. xi, 10, 21, 106).

[4] Jianwei Sun, Adrienn Ruzsinszky, and John P. Perdew. “Strongly Constrained and Appropriately Normed Semilocal Density Functional”. In: *Phys. Rev. Lett.* 115 (3 July 2015), p. 036402 (cit. on pp. xi, 10, 14, 16).

[5] P. Hohenberg and W. Kohn. “Inhomogeneous Electron Gas”. In: *Phys. Rev.* 136 (3B Nov. 1964), B864–B871 (cit. on pp. 1, 7, 9).

[6] W. Kohn and L. J. Sham. “Self-Consistent Equations Including Exchange and Correlation Effects”. In: *Phys. Rev.* 140 (4A Nov. 1965), A1133–A1138 (cit. on pp. 1, 7, 10).

[7] Kirstin Alberi, Marco Buongiorno Nardelli, Andriy Zakutayev, et al. “The 2019 materials by design roadmap”. In: *Journal of Physics D: Applied Physics* 52.1 (Oct. 2018), p. 013001 (cit. on p. 1).

[8] Nicolas Mounet, Marco Gibertini, Philippe Schwaller, et al. “Two-dimensional materials from high-throughput computational exfoliation of experimentally known compounds”. In: *Nature Nanotechnology* 13.3 (Mar. 2018), pp. 246–252 (cit. on p. 1).

[9] Qimin Yan, Jie Yu, Santosh K. Suram, et al. “Solar fuels photoanode materials discovery by integrating high-throughput theory and experiment”. In: *Proceedings of the National Academy of Sciences* 114.12 (2017), pp. 3040–3043. eprint: `https://www.pnas.org/content/114/12/3040.full.pdf` (cit. on p. 1).

[10] Andrew S. Rosen, Justin M. Notestein, and Randall Q. Snurr. “Identifying promising metalorganic frameworks for heterogeneous catalysis via high-throughput periodic density functional theory”. In: *Journal of Computational Chemistry* 40.12 (2019), pp. 1305–1318. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/jcc.25787` (cit. on p. 1).

[11]Alejandro J. Garza and Gustavo E. Scuseria. “Predicting Band Gaps with Hybrid Density Functionals”. In: *The Journal of Physical Chemistry Letters* 7.20 (2016), pp. 4165–4170. eprint: `https://doi.org/10.1021/acs.jpclett.6b01807` (cit. on pp. 1, 22).

[12]Christopher J. Cramer and Donald G. Truhlar. “Density functional theory for transition metals and transition metal chemistry”. In: *Phys. Chem. Chem. Phys.* 11 (46 2009), pp. 10757–10816 (cit. on p. 1).

[13]Kieron Burke. “Perspective on density functional theory”. In: *The Journal of Chemical Physics* 136.15 (2012), p. 150901. eprint: `https://doi.org/10.1063/1.4704546` (cit. on p. 1).

[14]Axel D. Becke. “Perspective: Fifty years of density-functional theory in chemical physics”. In: *The Journal of Chemical Physics* 140.18 (2014), 18A301. eprint: `https://doi.org/10.1063/1.4869598` (cit. on p. 1).

[15]Cui Zhang, Davide Donadio, François Gygi, and Giulia Galli. “First Principles Simulations of the Infrared Spectrum of Liquid Water Using Hybrid Density Functionals”. In: *Journal of Chemical Theory and Computation* 7.5 (2011). PMID: 26610134, pp. 1443–1449. eprint: `https://doi.org/10.1021/ct2000952` (cit. on p. 1).

[16]D. Rodic, M. Mitric, R. Tellgren, H. Rundlof, and A. Kremenovic. “True magnetic structure of the ferrimagnetic garnet Y3Fe5O12 and magnetic moments of iron ions”. In: *Journal of Magnetism and Magnetic Materials* 191.1 (1999), pp. 137–145 (cit. on pp. 2, 78, 79).

[17]Ryan Nakamoto, Bin Xu, Changsong Xu, Hu Xu, and L. Bellaiche. “Properties of rare-earth iron garnets from first principles”. In: *Phys. Rev. B* 95 (2 Jan. 2017), p. 024434 (cit. on pp. 2, 67–69, 77, 78, 80, 82).

[18]Seungho Yu, Robert D. Schmidt, Regina Garcia-Mendez, et al. “Elastic Properties of the Solid Electrolyte Li7La3Zr2O12 (LLZO)”. In: *Chemistry of Materials* 28.1 (2016), pp. 197–206. eprint: `https://doi.org/10.1021/acs.chemmater.5b03854` (cit. on p. 2).

[19]W. J. Hehre, R. F. Stewart, and J. A. Pople. “SelfConsistent MolecularOrbital Methods. I. Use of Gaussian Expansions of SlaterType Atomic Orbitals”. In: *The Journal of Chemical Physics* 51.6 (1969), pp. 2657–2664. eprint: `https://doi.org/10.1063/1.1672392` (cit. on p. 2).

[20]P. E. Blöchl. “Projector augmented-wave method”. In: *Phys. Rev. B* 50 (24 Dec. 1994), pp. 17953–17979 (cit. on p. 2).

[21]Axel D. Becke. “Basis-set-free density-functional quantum chemistry”. In: *International Journal of Quantum Chemistry* 36.S23 (1989), pp. 599–609. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/qua.560360862` (cit. on p. 2).

[23]Kurt Lejaeghere, Gustav Bihlmayer, Torbjörn Björkman, et al. “Reproducibility in density functional theory calculations of solids”. In: *Science* 351.6280 (2016), aad3000. eprint: `https://www.science.org/doi/pdf/10.1126/science.aad3000` (cit. on pp. 2, 25).

[24] Taylor A. Barnes, Thorsten Kurth, Pierre Carrier, et al. “Improved treatment of exact exchange in Quantum ESPRESSO”. In: *Computer Physics Communications* 214 (2017), pp. 52–58 (cit. on p. 2).

[25] Ivan Carnimeo, Stefano Baroni, and Paolo Giannozzi. “Fast hybrid density-functional computations using plane-wave basis sets”. In: *Electronic Structure* 1.1 (Feb. 2019), p. 015009 (cit. on p. 2).

[26] Peter Blaha, Karlheinz Schwarz, Fabien Tran, et al. “WIEN2k: An APW+lo program for calculating the properties of solids”. In: *The Journal of Chemical Physics* 152.7 (2020), p. 074101. eprint: `https://doi.org/10.1063/1.5143061` (cit. on p. 2).

[27] Markus Betzinger, Christoph Friedrich, and Stefan Blügel. “Hybrid functionals within the all-electron FLAPW method: Implementation and applications of PBE0”. In: *Phys. Rev. B* 81 (19 May 2010), p. 195117 (cit. on pp. 2, 3, 22).

[28] Martin Schlipf, Markus Betzinger, Christoph Friedrich, Marjana Le ai, and Stefan Blügel. “HSE hybrid functional within the FLAPW method and its application to GdN”. In: *Phys. Rev. B* 84 (12 Sept. 2011), p. 125142 (cit. on pp. 2, 3, 68).

[32] Markus Betzinger. “Orbital-dependent exchange-correlation functionals in density-functional theory realized by the FLAPW method”. englisch. PhD thesis. Jülich: Zugleich: Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2011, 2013, VI, 173 Seiten (cit. on pp. 3, 7, 13, 25, 27, 31).

[33] Martin Schlipf. “Heyd-Scuseria-Ernzerhof screened-exchange hybrid functional for complex materials : all-electron implementation and application [E-Book]”. englisch. PhD thesis. Jülich: Zugleich: Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2012, 2013, 1 Online–Ressource (XV, 170 Seiten) (cit. on pp. 3, 13).

[34] A A Serga, A V Chumak, and B Hillebrands. “YIG magnonics”. In: *Journal of Physics D: Applied Physics* 43.26 (June 2010), p. 264002 (cit. on pp. 4, 67).

[35] L. J. Cornelissen, J. Liu, R. A. Duine, J. Ben Youssef, and B. J. van Wees. “Long-distance transport of magnon spin information in a magnetic insulator at room temperature”. In: *Nature Physics* 11.12 (Dec. 2015), pp. 1022–1026 (cit. on p. 4).

[36] Y. Tabuchi, S. Ishino, A. Noguchi, et al. “Coherent coupling between a ferromagnetic magnon and a superconducting qubit”. In: *Science* 349.6246 (July 2015), pp. 405–408 (cit. on pp. 4, 67).

[37] Joseph Barker, Dimitar Pashov, and Jerome Jackson. “Electronic structure and finite temperature magnetism of yttrium iron garnet”. In: *Electronic Structure* 2.4 (Dec. 2020), p. 044002 (cit. on pp. 4, 67, 78, 79).

[38] W. A. Crossley, R. W. Cooper, J. L. Page, and R. P. van Stapele. “Faraday Rotation in RareEarth Iron Garnets”. In: *Journal of Applied Physics* 40.3 (1969), pp. 1497–1498. eprint: `https://doi.org/10.1063/1.1657740` (cit. on p. 4).

[39]Andy Quindeau, Can O. Avci, Wenqing Liu, et al. “Tm3Fe5O12/Pt Heterostructures with Perpendicular Magnetic Anisotropy for Spintronic Applications”. In: *Advanced Electronic Materials* 3.1 (2017), p. 1600376. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/aelm.201600376` (cit. on p. 4).

[40]E. Schrödinger. “An Undulatory Theory of the Mechanics of Atoms and Molecules”. In: *Phys. Rev.* 28 (6 Dec. 1926), pp. 1049–1070 (cit. on p. 7).

[41]Paul Adrien Maurice Dirac and Ralph Howard Fowler. “The quantum theory of the electron”. In: *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 117.778 (1928), pp. 610–624. eprint: `https://royalsocietypublishing.org/doi/pdf/10.1098/rspa.1928.0023` (cit. on p. 7).

[42]M. Born and R. Oppenheimer. “Zur Quantentheorie der Molekeln”. In: *Annalen der Physik* 389.20 (1927), pp. 457–484. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19273892002` (cit. on pp. 7, 8).

[43]Gregor Michalicek. “Extending the precision and efficiency of the all-electron full-potential linearized augmented plane-wave density-functional theory method”. englisch. PhD thesis. Jülich: Zugleich: Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2014, 2015, 195 Seiten (cit. on pp. 7, 9, 25).

[44]Richard M. Martin. *Electronic structure: basic theory and practical methods*. Cambridge University Press, 2004 (cit. on pp. 7, 9).

[45]Jan-Philipp Hanke. “Topological properties of complex magnets from an advanced ab-initio Wannier description”. englisch. PhD thesis. Jülich: Zugleich: Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2018, 2018, XI, 173 Seiten (cit. on p. 7).

[46]Philipp Kurz. “Non-collinear magnetism at surfaces and in ultrathin films [E-Book]”. englisch. PhD thesis. Jülich: Zugleich: Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2000, 2000, 1 Online–Ressource (IV, 207 Seiten) (cit. on pp. 7, 105).

[47]D. R. Hartree. “The Wave Mechanics of an Atom with a Non-Coulomb Central Field. Part I. Theory and Methods”. In: *Mathematical Proceedings of the Cambridge Philosophical Society* 24.1 (1928), pp. 89–110 (cit. on p. 8).

[48]W. Kohn. “Nobel Lecture: Electronic structure of matter—wave functions and density functionals”. In: *Rev. Mod. Phys.* 71 (5 Oct. 1999), pp. 1253–1266 (cit. on p. 9).

[49]J. P. Perdew and Alex Zunger. “Self-interaction correction to density-functional approximations for many-electron systems”. In: *Phys. Rev. B* 23 (10 May 1981), pp. 5048–5079 (cit. on p. 10).

[50]John P. Perdew, Adrienn Ruzsinszky, Gábor I. Csonka, et al. “Restoring the Density-Gradient Expansion for Exchange in Solids and Surfaces”. In: *Phys. Rev. Lett.* 100 (13 Apr. 2008), p. 136406 (cit. on pp. 10, 16).

[51] John P. Perdew, J. A. Chevary, S. H. Vosko, et al. “Atoms, molecules, solids, and surfaces: Applications of the generalized gradient approximation for exchange and correlation”. In: *Phys. Rev. B* 46 (11 Sept. 1992), pp. 6671–6687 (cit. on p. 10).

[52] Fabien Tran and Peter Blaha. “Accurate Band Gaps of Semiconductors and Insulators with a Semilocal Exchange-Correlation Potential”. In: *Phys. Rev. Lett.* 102 (22 June 2009), p. 226401 (cit. on p. 10).

[53] Jianmin Tao, ed. *Density functional theory of atoms, molecules, and solids : development of accurate semilocal approximations for exchange and correlations*. englisch. Saarbrücken: VDM Verl. Müller, 2010, IV, 125 S. (Cit. on pp. 10, 13, 15).

[54] Uliana Alekseeva, Gregor Michalicek, Daniel Wortmann, and Stefan Blügel. “Hybrid Parallelization and Performance Optimization of the FLEUR Code: New Possibilities for All-Electron Density Functional Theory”. In: *Euro-Par 2018: Parallel Processing*. Ed. by Marco Aldinucci, Luca Padovani, and Massimo Torquati. Cham: Springer International Publishing, 2018, pp. 735–748 (cit. on pp. 12, 35).

[55] D. M. Ceperley and B. J. Alder. “Ground State of the Electron Gas by a Stochastic Method”. In: *Phys. Rev. Lett.* 45 (7 Aug. 1980), pp. 566–569 (cit. on p. 13).

[56] Emil Proynov and Jing Kong. “Analytic form of the correlation energy of the uniform electron gas”. In: *Phys. Rev. A* 79 (1 Jan. 2009), p. 014103 (cit. on p. 13).

[57] Michele Ruggeri, Pablo López Ros, and Ali Alavi. “Correlation energies of the high-density spin-polarized electron gas to meV accuracy”. In: *Phys. Rev. B* 98 (16 Oct. 2018), p. 161105 (cit. on p. 13).

[58] John P. Perdew and Yue Wang. “Accurate and simple analytic representation of the electron-gas correlation energy”. In: *Phys. Rev. B* 45 (23 June 1992), pp. 13244–13249 (cit. on p. 13).

[59] J. P. Perdew and Alex Zunger. “Self-interaction correction to density-functional approximations for many-electron systems”. In: *Phys. Rev. B* 23 (10 May 1981), pp. 5048–5079 (cit. on p. 13).

[60] Philipp Haas, Fabien Tran, and Peter Blaha. “Calculation of the lattice constant of solids with semilocal functionals”. In: *Phys. Rev. B* 79 (8 Feb. 2009), p. 085104 (cit. on pp. 14, 16).

[61] Fabien Tran, Jan Doumont, Leila Kalantari, et al. “Semilocal exchange-correlation potentials for solid-state calculations: Current status and future directions”. In: *Journal of Applied Physics* 126.11 (2019), p. 110902. eprint: `https://doi.org/10.1063/1.5118863` (cit. on pp. 14, 16).

[62] O. Gunnarsson and R. O. Jones. “Extensions of the LSD approximation in density functional calculations”. In: *The Journal of Chemical Physics* 72.10 (1980), pp. 5357–5362. eprint: `https://doi.org/10.1063/1.439028` (cit. on p. 14).

[63] Frank Herman, John P. Van Dyke, and Irene B. Ortenburger. “Improved Statistical Exchange Approximation for Inhomogeneous Many-Electron Systems”. In: *Phys. Rev. Lett.* 22 (16 Apr. 1969), pp. 807–811 (cit. on p. 14).

[64]John P. Perdew. “Accurate Density Functional for the Energy: Real-Space Cutoff of the Gradient Expansion for the Exchange Hole”. In: *Phys. Rev. Lett.* 55 (16 Oct. 1985), pp. 1665–1668 (cit. on p. 14).

[65]Mel Levy and John P. Perdew. “Hellmann-Feynman, virial, and scaling requisites for the exact universal density functionals. Shape of the correlation potential and diamagnetic susceptibility for atoms”. In: *Phys. Rev. A* 32 (4 Oct. 1985), pp. 2010–2021 (cit. on p. 15).

[66]Axel D. Becke. “A new inhomogeneity parameter in density-functional theory”. In: *The Journal of Chemical Physics* 109.6 (1998), pp. 2092–2098. eprint: `https://doi.org/10.1063/1.476722` (cit. on p. 16).

[67]Fabien Tran, Péter Kovács, Leila Kalantari, Georg K. H. Madsen, and Peter Blaha. “Orbital-free approximations to the kinetic-energy density in exchange-correlation MGGA functionals: Tests on solids”. In: *The Journal of Chemical Physics* 149.14 (2018), p. 144105. eprint: `https://doi.org/10.1063/1.5048907` (cit. on p. 16).

[68]A. D. Becke. “HartreeFock exchange energy of an inhomogeneous electron gas”. In: *International Journal of Quantum Chemistry* 23.6 (1983), pp. 1915–1922. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/qua.560230605` (cit. on p. 16).

[69]Jianmin Tao, John P. Perdew, Viktor N. Staroverov, and Gustavo E. Scuseria. “Climbing the Density Functional Ladder: Nonempirical Meta–Generalized Gradient Approximation Designed for Molecules and Solids”. In: *Phys. Rev. Lett.* 91 (14 Sept. 2003), p. 146401 (cit. on p. 16).

[70]Fabien Tran, Julia Stelzl, and Peter Blaha. “Rungs 1 to 4 of DFT Jacobs ladder: Extensive test on the lattice constant, bulk modulus, and cohesive energy of solids”. In: *The Journal of Chemical Physics* 144.20 (2016), p. 204120. eprint: `https://doi.org/10.1063/1.4948636` (cit. on pp. 16, 21).

[71]Bikash Patra, Subrata Jana, Lucian A. Constantin, and Prasanjit Samal. “Efficient band gap prediction of semiconductors and insulators from a semilocal exchange-correlation functional”. In: *Phys. Rev. B* 100 (4 July 2019), p. 045147 (cit. on p. 16).

[72]Susi Lehtola, Conrad Steigemann, Micael J.T. Oliveira, and Miguel A.L. Marques. “Recent developments in libxc A comprehensive library of functionals for density functional theory”. In: *SoftwareX* 7 (2018), pp. 1–5 (cit. on p. 16).

[73]R. T. Sharp and G. K. Horton. “A Variational Approach to the Unipotential Many-Electron Problem”. In: *Phys. Rev.* 90 (2 Apr. 1953), pp. 317–317 (cit. on p. 16).

[74]James D. Talman and William F. Shadwick. “Optimized effective atomic central potential”. In: *Phys. Rev. A* 14 (1 July 1976), pp. 36–40 (cit. on p. 16).

[75]Daniel Meja-Rodrguez and S. B. Trickey. “Meta-GGA performance in solids at almost GGA cost”. In: *Phys. Rev. B* 102 (12 Sept. 2020), p. 121109 (cit. on p. 17).

[76] J. G. Brandenburg, J. E. Bates, J. Sun, and J. P. Perdew. “Benchmark tests of a strongly constrained semilocal functional with a long-range dispersion correction”. In: *Phys. Rev. B* 94 (11 Sept. 2016), p. 115144 (cit. on p. 17).

[77] Albert P. Bartók and Jonathan R. Yates. “Regularized SCAN functional”. In: *The Journal of Chemical Physics* 150.16 (2019), p. 161101. eprint: `https://doi.org/10.1063/1.5094646` (cit. on p. 17).

[78] James W. Furness, Aaron D. Kaplan, Jinliang Ning, John P. Perdew, and Jianwei Sun. “Accurate and Numerically Efficient r2SCAN Meta-Generalized Gradient Approximation”. In: *The Journal of Physical Chemistry Letters* 11.19 (2020). PMID: 32876454, pp. 8208–8215. eprint: `https://doi.org/10.1021/acs.jpclett.0c02405` (cit. on p. 17).

[79] Yuhao Fu and David J. Singh. “Applicability of the Strongly Constrained and Appropriately Normed Density Functional to Transition-Metal Magnetism”. In: *Phys. Rev. Lett.* 121 (20 Nov. 2018), p. 207201 (cit. on p. 17).

[80] Péter Kovács, Fabien Tran, Peter Blaha, and Georg K. H. Madsen. “Comparative study of the PBE and SCAN functionals: The particular case of alkali metals”. In: *The Journal of Chemical Physics* 150.16 (2019), p. 164119. eprint: `https://doi.org/10.1063/1.5092748` (cit. on p. 17).

[81] Axel D. Becke. “A new mixing of HartreeFock and local densityfunctional theories”. In: *The Journal of Chemical Physics* 98.2 (1993), pp. 1372–1377. eprint: `https://doi.org/10.1063/1.464304` (cit. on pp. 18, 20).

[82] Hans. Hellmann, ed. *Einführung in die Quantenchemie, : mit 35 Tab.* deutsch. Leipzig: Deuticke, 1937, 4 p. . 350 p. (Cit. on p. 19).

[83] R. P. Feynman. “Forces in Molecules”. In: *Phys. Rev.* 56 (4 Aug. 1939), pp. 340–343 (cit. on p. 19).

[84] D. Von-Eiff, S. Haddad, and M. K. Weigel. “Comparison of the relativistic Hartree-Fock approximation and its semiclassical expansion”. In: *Phys. Rev. C* 50 (2 Aug. 1994), pp. 1244–1246 (cit. on p. 20).

[85] Enrico Clementi and Subhas J. Chakravorty. “A comparative study of density functional models to estimate molecular atomization energies”. In: *The Journal of Chemical Physics* 93.4 (1990), pp. 2591–2602. eprint: `https://doi.org/10.1063/1.458899` (cit. on p. 20).

[86] Chr. Møller and M. S. Plesset. “Note on an Approximation Treatment for Many-Electron Systems”. In: *Phys. Rev.* 46 (7 Oct. 1934), pp. 618–622 (cit. on p. 21).

[87] Aliaksandr V. Krukau, Oleg A. Vydrov, Artur F. Izmaylov, and Gustavo E. Scuseria. “Influence of the exchange screening parameter on the performance of screened hybrid functionals”. In: *The Journal of Chemical Physics* 125.22 (2006), p. 224106. eprint: `https://doi.org/10.1063/1.2404663` (cit. on pp. 21, 67).

[88]Subrata Jana, Abhilash Patra, and Prasanjit Samal. "Efficient lattice constants and energy bandgaps for condensed systems from a meta-GGA level screened range-separated hybrid functional". In: *The Journal of Chemical Physics* 149.9 (2018), p. 094105. eprint: `https://doi.org/10.1063/1.5037030` (cit. on p. 21).

[89]Jin Yu, Bingbing Zhang, Xiaodong Zhang, et al. "Finding Optimal Mid-Infrared Nonlinear Optical Materials in Germanates by First-Principles High-Throughput Screening and Experimental Verification". In: *ACS Applied Materials & Interfaces* 12.40 (2020). PMID: 32924416, pp. 45023–45035. eprint: `https://doi.org/10.1021/acsami.0c15728` (cit. on p. 22).

[90]Bingbing Zhang, Xiaodong Zhang, Jin Yu, et al. "First-Principles High-Throughput Screening Pipeline for Nonlinear Optical Materials: Application to Borates". In: *Chemistry of Materials* 32.15 (2020), pp. 6772–6779. eprint: `https://doi.org/10.1021/acs.chemmater.0c02583` (cit. on p. 22).

[91]Bhawna Sahni, Vikram, Jiban Kangsabanik, and Aftab Alam. "Reliable Prediction of New Quantum Materials for Topological and Renewable-Energy Applications: A High-Throughput Screening". In: *The Journal of Physical Chemistry Letters* 11.15 (2020). PMID: 32702983, pp. 6364–6372. eprint: `https://doi.org/10.1021/acs.jpclett.0c01271` (cit. on p. 22).

[92]W. Kohn and L. J. Sham. "Self-Consistent Equations Including Exchange and Correlation Effects". In: *Phys. Rev.* 140 (4A Nov. 1965), A1133–A1138 (cit. on p. 22).

[93]C. G. Broyden. "A class of methods for solving nonlinear simultaneous equations". In: *Mathematics of Computation* 19.92 (1965), pp. 577–577 (cit. on p. 22).

[94]Phanisri P. Pratapa and Phanish Suryanarayana. "Restarted Pulay mixing for efficient and robust acceleration of fixed-point iterations". In: *Chemical Physics Letters* 635 (2015), pp. 69–74 (cit. on p. 22).

[95]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". In: *Phys. Rev. B* 24 (2 July 1981), pp. 864–875 (cit. on p. 25).

[96]O.K. Andersen and R.G. Woolley. "Muffin-tin orbitals and molecular calculations: General formalism". In: *Molecular Physics* 26.4 (1973), pp. 905–927. eprint: `https://doi.org/10.1080/00268977300102171` (cit. on p. 25).

[97]This figure or a derivative their-of has been published on Wikipedia to illustrate the article on FLAPW. However, the author of this thesis is also the original creator of this figure. (cit. on p. 26).

[98]Felix Bloch. "Über die Quantenmechanik der Elektronen in Kristallgittern". In: *Zeitschrift für Physik* 52.7 (July 1929), pp. 555–600 (cit. on p. 27).

[99]J. C. Slater. "Wave Functions in a Periodic Potential". In: *Phys. Rev.* 51 (10 May 1937), pp. 846–851 (cit. on p. 27).

[100] Gregor Michalicek, Markus Betzinger, Christoph Friedrich, and Stefan Blügel. "Elimination of the linearization error and improved basis-set convergence within the FLAPW method". In: *Computer Physics Communications* 184.12 (2013), pp. 2670–2679 (cit. on p. 28).

[101] Edward S. Sachs, Juergen Hinze, and Nora H. Sabelli. "Frozen core approximation, a pseudopotential method tested on six states of NaH". In: *The Journal of Chemical Physics* 62.9 (1975), pp. 3393–3398. eprint: `https://aip.scitation.org/doi/pdf/10.1063/1.430993` (cit. on p. 29).

[102] Lars Nordstrom David J. Singh. *Planewaves, Pseudopotentials, and the LAPW Method*. 2nd. Springer, 2005 (cit. on p. 29).

[103] Gregor Michalicek, Markus Betzinger, Christoph Friedrich, and Stefan Blügel. "Elimination of the linearization error and improved basis-set convergence within the FLAPW method". In: *Computer Physics Communications* 184.12 (2013), pp. 2670–2679 (cit. on p. 29).

[104] Christoph Friedrich, Arno Schindlmayr, and Stefan Blügel. "Efficient calculation of the Coulomb matrix and its expansion around k=0 within the FLAPW method". In: *Computer Physics Communications* 180.3 (2009), pp. 347–359 (cit. on pp. 30, 33).

[105] F. Aryasetiawan and O. Gunnarsson. "Product-basis method for calculating dielectric matrices". In: *Phys. Rev. B* 49 (23 June 1994), pp. 16214–16222 (cit. on p. 30).

[106] Jens Bröder, Daniel Wortmann, and Stefan Blügel. "Using the AiiDA-FLEUR package for all-electron ab initio electronic structure data generation and processing in materials science". In: *In Extreme Data Workshop 2018 Proceedings*. Vol. 40. IAS Series. Jülich: Forschungszentrums Jülich, 2019, p 43–48 (cit. on p. 35).

[109] R. Menon and L. Dagum. "OpenMP: An Industry-Standard API for Shared-Memory Programming". In: *Computing in Science & Engineering* v.01 (Jan. 1998), pp. 46–55 (cit. on p. 36).

[113] Georg Hager. *Introduction to high performance computing for scientists and engineers*. Boca Raton, FL: CRC Press, 2011 (cit. on pp. 54, 55).

[115] John L. Gustafson. "Reevaluating Amdahl's Law". In: *Commun. ACM* 31.5 (May 1988), pp. 532–533 (cit. on p. 58).

[116] S. O. Demokritov, V. E. Demidov, O. Dzyapko, et al. "Bose–Einstein condensation of quasi-equilibrium magnons at room temperature under pumping". In: *Nature* 443.7110 (Sept. 2006), pp. 430–433 (cit. on p. 67).

[117] F. Sayetat. "Huge magnetostriction in Tb3Fe5O12, Dy3Fe5O12, Ho3Fe5O12, Er3Fe5O12 garnets". In: *Journal of Magnetism and Magnetic Materials* 58.3 (1986), pp. 334–346 (cit. on p. 67).

[118] K. Uchida, J. Xiao, H. Adachi, et al. "Spin Seebeck insulator". In: *Nature Materials* 9.11 (Nov. 2010), pp. 894–897 (cit. on p. 67).

[119] H. Lassri, E.K. Hlil, S. Prasad, and R. Krishnan. "Magnetic and electronic properties of nanocrystalline Gd3Fe5O12 garnet". In: *Journal of Solid State Chemistry* 184.12 (2011), pp. 3216–3220 (cit. on pp. 67, 79).

[120]Federico Iori, Adrien Teurtrie, Laura Bocher, et al. “Bismuth iron garnet: Ab initio study of electronic properties”. In: *Phys. Rev. B* 100 (24 Dec. 2019), p. 245150 (cit. on p. 67).

[121]J. Hubbard and Brian Hilton Flowers. “Electron correlations in narrow energy bands”. In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 276.1365 (1963), pp. 238–257. eprint: `https://royalsocietypublishing.org/doi/pdf/10.1098/rspa.1963.0204` (cit. on p. 67).

[122]P. H. Dederichs, S. Blügel, R. Zeller, and H. Akai. “Ground States of Constrained Systems: Application to Cerium Impurities”. In: *Phys. Rev. Lett.* 53 (26 Dec. 1984), pp. 2512–2515 (cit. on p. 67).

[123]Ersoy a o lu, Christoph Friedrich, and Stefan Blügel. “Effective Coulomb interaction in transition metals from constrained random-phase approximation”. In: *Phys. Rev. B* 83 (12 Mar. 2011), p. 121101 (cit. on p. 67).

[124]Li-Shan Xie, Guang-Xi Jin, Lixin He, et al. “First-principles study of exchange interactions of yttrium iron garnet”. In: *Phys. Rev. B* 95 (1 Jan. 2017), p. 014423 (cit. on pp. 68, 69).

[125]Juarez L. F. Da Silva, M. Verónica Ganduglia-Pirovano, Joachim Sauer, Veronika Bayer, and Georg Kresse. “Hybrid functionals applied to rare-earth oxides: The example of ceria”. In: *Phys. Rev. B* 75 (4 Jan. 2007), p. 045121 (cit. on p. 68).

[126]Pierre Villars and Karin Cenzual, eds. *Y3Fe5O12 Crystal Structure: Datasheet from “PAULING FILE Multinaries Edition – 2012” in SpringerMaterials*. accessed 2021-09-02 (cit. on pp. 68, 70).

[127]Koichi Momma and Fujio Izumi. “VESTA3 for three-dimensional visualization of crystal, volumetric and morphology data”. In: *Journal of Applied Crystallography* 44.6 (Dec. 2011), pp. 1272–1276 (cit. on pp. 68, 70).

[128]Pierre Villars and Karin Cenzual, eds. *Gd3Fe5O12 Crystal Structure: Datasheet from “PAULING FILE Multinaries Edition – 2012” in SpringerMaterials (https://materials.springer.com/isp/crystallogra*
accessed 2021-10-04 (cit. on p. 70).

[129]Pierre Villars and Karin Cenzual, eds. *Tm3Fe5O12 Crystal Structure: Datasheet from “PAULING FILE Multinaries Edition – 2012” in SpringerMaterials (https://materials.springer.com/isp/crystallogra*
accessed 2021-10-04 (cit. on p. 70).

[130]P.K. Larsen and R. Metselaar. “Defects and the electronic properties of Y3Fe5O12”. In: *Journal of Solid State Chemistry* 12.3 (1975), pp. 253–258 (cit. on pp. 70, 75).

[131]Wei-Kuo Li and Guang-Yu Guo. “First-principles study on magneto-optical effects in the ferromagnetic semiconductors $\mathrm{Y_3Fe_5O_{12}}$ and $\mathrm{Bi_3Fe_5O_{12}}$”. In: *Phys. Rev. B* 103 (1 Jan. 2021), p. 014439 (cit. on p. 74).

[132]M H Phan, M B Morales, C N Chinnasamy, et al. “Magnetocaloric effect in bulk and nanostructured Gd3Fe5O12materials”. In: *Journal of Physics D: Applied Physics* 42.11 (May 2009), p. 115007 (cit. on p. 79).

[133] P. B. A. Fechine, R. S. T. Moretzsohn, R. C. S. Costa, et al. “Magneto-dielectric properties of the Y3Fe5O12 and Gd3Fe5O12 dielectric ferrite resonator antennas”. In: *Microwave and Optical Technology Letters* 50.11 (2008), pp. 2852–2857. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/mop.23824` (cit. on p. 79).

[134] G. L. S. Vilela, J. E. Abrao, E. Santos, et al. “Magnon-mediated spin currents in Tm3Fe5O12/Pt with perpendicular magnetic anisotropy”. In: *Applied Physics Letters* 117.12 (2020), p. 122412. eprint: `https://doi.org/10.1063/5.0023242` (cit. on p. 79).

[135] Nguyen M. Vu, Peter B. Meisenheimer, and John T. Heron. “Tunable magnetoelastic anisotropy in epitaxial (111) Tm3Fe5O12 thin films”. In: *Journal of Applied Physics* 127.15 (2020), p. 153905. eprint: `https://doi.org/10.1063/1.5142856` (cit. on p. 79).

[136] Aidan J. Lee, Adam S. Ahmed, Jose Flores, et al. “Probing the Source of the Interfacial Dzyaloshinskii-Moriya Interaction Responsible for the Topological Hall Effect in Metal/$\mathrm{Tm_3Fe_5O_{12}}$ Systems”. In: *Phys. Rev. Lett.* 124 (10 Mar. 2020), p. 107201 (cit. on p. 79).

[137] S. Geller, J. P. Remeika, R. C. Sherwood, H. J. Williams, and G. P. Espinosa. “Magnetic Study of the Heavier Rare-Earth Iron Garnets”. In: *Phys. Rev.* 137 (3A Feb. 1965), A1034–A1038 (cit. on p. 82).

[138] Nicolas Tancogne-Dejean and Angel Rubio. “Parameter-free hybridlike functional based on an extended Hubbard model: $\mathrm{DFT}+U+V$”. In: *Phys. Rev. B* 102 (15 Oct. 2020), p. 155117 (cit. on p. 82).

[139] Luis A. Agapito, Stefano Curtarolo, and Marco Buongiorno Nardelli. “Reformulation of $\mathrm{DFT}+U$ as a Pseudohybrid Hubbard Density Functional for Accelerated Materials Discovery”. In: *Phys. Rev. X* 5 (1 Jan. 2015), p. 011006 (cit. on p. 82).

[140] Zhiguo Xia and Andries Meijerink. “Ce3+-Doped garnet phosphors: composition modification, luminescence properties and applications”. In: *Chem. Soc. Rev.* 46 (1 2017), pp. 275–299 (cit. on p. 87).

[141] Tongyu Gao, Junhang Tian, Yuanhong Liu, Ronghui Liu, and Weidong Zhuang. “Garnet phosphors for white-light-emitting diodes: modification and calculation”. In: *Dalton Trans.* 50 (11 2021), pp. 3769–3781 (cit. on p. 87).

[142] P. Hansen, K. Witter, and W. Tolksdorf. “Magnetic and magneto-optic properties of lead- and bismuth-substituted yttrium iron garnet films”. In: *Phys. Rev. B* 27 (11 June 1983), pp. 6608–6625 (cit. on p. 87).

[143] Qian Zhang, Zhiqing Gao, Xiaomeng Shi, et al. “Recent advances on rare earths in solid lithium ion conductors”. In: *Journal of Rare Earths* 39.1 (2021), pp. 1–10 (cit. on p. 87).

[144] S. Ramakumar, C. Deviannapoorani, L. Dhivya, Lakshmi S. Shankar, and Ramaswamy Murugan. “Lithium garnets: Synthesis, structure, Li+ conductivity, Li+ dynamics and applications”. In: *Progress in Materials Science* 88 (2017), pp. 325–411 (cit. on p. 87).

[145] Zhong-Li Wang, Dan Xu, Ji-Jing Xu, and Xin-Bo Zhang. “Oxygen electrocatalysts in metalair batteries: from aqueous to nonaqueous electrolytes”. In: *Chem. Soc. Rev.* 43 (22 2014), pp. 7746–7786 (cit. on p. 87).

[147] Giovanni Pizzi, Valerio Vitale, Ryotaro Arita, et al. “Wannier90 as a community code: new features and applications”. In: *Journal of Physics: Condensed Matter* 32.16 (Jan. 2020), p. 165902 (cit. on p. 101).

[148] James W. Cooley and John W. Tukey. “An algorithm for the machine calculation of complex Fourier series”. In: *Mathematics of Computation* 19.90 (May 1965), pp. 297–297 (cit. on p. 108).

Webpages

[@22] Forschungszentrum Jülich - PGI-1 & IAS-1. 2021. URL: `https://www.flapw.de` (cit. on p. 2).

[@29] Jülich Supercomputing Centre. *TOP500 June 2017*. 2017. URL: `https://www.top500.org/lists/top500/2017/06/highlights/` (cit. on pp. 3, 35).

[@30] Jülich Supercomputing Centre. *TOP500 June 2021*. 2017. URL: `https://www.top500.org/lists/top500/2021/06/highs/` (cit. on pp. 3, 35).

[@31] European Processor Initiative. 2021. URL: `https://www.european-processor-initiative.eu/epi-epac1-0-risc-v-test-chip-samples-delivered/` (cit. on p. 3).

[@107] Oak Ridge National Laboratory. *Frontier Spec sheet*. 2019. URL: `https://www.olcf.ornl.gov/wp-content/uploads/2019/05/frontier_specsheet.pdf` (cit. on p. 35).

[@108] Anandtech. *El Capitan Supercomputer Detailed: AMD CPUs & GPUs To Drive 2 Exaflops of Compute*. 2020. URL: `https://www.anandtech.com/show/15581/el-capitan-supercomputer-detailed-amd-cpus-gpus-2-exaflops` (cit. on p. 35).

[@110] Daniël de Kok. *Intel MKL on AMD Zen*. 2020. URL: `https://danieldk.eu/Posts/2020-08-31-MKL-Zen.html` (cit. on pp. 37, 38).

[@111] ITC RWTH AACHEN. *CLAIX Hardware overview*. 2021. URL: `https://help.itc.rwth-aachen.de/service/rhr4fjjutttf/article/fbd107191cf14c4b8307f44f545cf68a/` (cit. on pp. 43, 45).

[@112] Jülich Supercomputing Centre. *Configuration Jureca DC*. 2021. URL: `https://apps.fz-juelich.de/jsc/hps/jureca/configuration.html` (cit. on pp. 43, 45, 55).

[@114] Leibnitz Rechenzentrum. *Hardware of SuperMUC-NG*. 2021. URL: `https://doku.lrz.de/display/PUBLIC/Hardware+of+SuperMUC-NG` (cit. on p. 55).

[@146] Matthias Redies. *Github - unique_eigenvectors*. 2021. URL: `https://github.com/JuDFTteam/unique_eigenvectors` (cit. on p. 101).

Defining unique eigenvectors

A

Applying a lot of changes, like the ones described in Chapter 4 and Chapter 5, to a large code base like FLEURs requires not only a lot of care, but also suitable tools to ensure that no new bugs are introduced and old ones are eliminated. In this section we present a debugging tool that was developed as part of this thesis [@146].

A lot of quantities calculated in FLEUR depend on the eigenvector matrix $\mathbf{Z}$. This matrix is not uniquely defined, but instead each eigenvector can have a phase and additionally, for a set of degenerate eigenvalues the corresponding eigenvectors span the so-called *eigenspace*. This eigenspace can be represented by infinitely many, equally correct eigenvectors.

While the physical quantities that FLEUR outputs, such as the density, are independent of the choice of this representation, this is not true for intermediate immeasurable quantities such as the wave functions. Therefore, it can be very time consuming to locate problems in the code. We might know for sure that the final quantity is wrong, but for all intermediate results this might be due to the bug we are trying to find or it might be due to a different representation of the eigenspace. In this appendix we present an algorithm to transform $\mathbf{Z}$ into a unique representation and therefore eliminate this uncertainty. While this tool was initially develop for debugging purposes, it also proves useful for other purposes such as wannier postprocessing [147], which relies on wave functions as an input. In the following we introduce an algorithm that transforms the eigenvector matrices into a unique representation.

Consider the following matrix

$$\mathbf{H} = \begin{pmatrix} 4.83 & -2.85 & -0.42 \\ -2.85 & 3.13 & 0.31 \\ -0.42 & 0.31 & 1.05 \end{pmatrix} \tag{A.1}$$

We diagonalize this hermitian matrix twice. Once by passing the upper triangle to LAPACK and once by passing the lower triangle to the same routine. Both result in the same set of eigenvalues $\{1, 1, 7\}$, but in very different eigenvector matrices:

$$\mathbf{V}_U = \begin{pmatrix} -0.58 & 0.15 & -0.80 \\ -0.80 & 0.06 & 0.60 \\ 0.14 & 0.99 & 0.09 \end{pmatrix} \qquad \mathbf{V}_L = \begin{pmatrix} 0.00 & 0.60 & -0.80 \\ -0.15 & 0.79 & 0.60 \\ 0.99 & 0.12 & 0.09 \end{pmatrix} \tag{A.2}$$

Multiplying a unitary matrix to an eigenvector submatrices of degenerate eigenvalues transforms it into a different representation of the *same* eigenspace, without effecting the orthonormality. Therefore, we are looking for two unitary matrices $\mathbf{Q}_U$ and $\mathbf{Q}_L$ that transform $\mathbf{V}_U$ and $\mathbf{V}_L$ into the same representation.

First we have to identify the groups of degenerate eigenvectors. For this we define a numerical cutoff Δ_ε and we regard all eigenvalues i and j as degenerate if $|\varepsilon_i - \varepsilon_j| < \Delta_\varepsilon$.

In our example the first $n_{\text{deg}} = 2$ eigenvalues are degenerate, while the last one is not. The eigenvectors are written columns-wise in the matrix. Clearly, the degenerate eigenspace in the first two columns has different representations in $\mathbf{V}_U$ and $\mathbf{V}_L$, while the eigenvector in the last column is the same.

The first step of mapping the different representations onto a common one is to pick the first n_{deg} rows in each matrix, which are linearly independent. In our example the first two rows are linearly independent and therefore the resulting submatrices are

$$\mathbf{A}_U = \begin{pmatrix} -0.58 & 0.15 \\ -0.80 & 0.06 \end{pmatrix} \qquad \mathbf{A}_L = \begin{pmatrix} 0.00 & 0.60 \\ -0.15 & 0.79 \end{pmatrix}. \tag{A.3}$$

For each of these we perform a QR-decomposition, so that $\mathbf{A} = \mathbf{QR}$, where $\mathbf{Q}$ is an unitary matrix and $\mathbf{R}$ is an upper triangular matrix. We can make this representation unique, by requiring that the diagonal elements of $\mathbf{R}$ are non-negative. The matrices $\mathbf{Q}_L$ and $\mathbf{Q}_U$ now transform the $\mathbf{A}_L$ and $\mathbf{A}_U$ matrices into the same upper triangular matrix $\mathbf{R} = \mathbf{R}_L = \mathbf{R}_U$.

$$\mathbf{A}_U = \mathbf{Q}_U\mathbf{R} \tag{A.4}$$

$$\mathbf{A}_L = \mathbf{Q}_L\mathbf{R} \tag{A.5}$$

Finally, we use the unitary matrices $\mathbf{Q}_L$ and $\mathbf{Q}_U$ to transform the submatrix of $\mathbf{V}_L$ and $\mathbf{V}_U$ into the unique representation.

$$\begin{aligned} \begin{pmatrix} -0.58 & 0.15 \\ -0.80 & 0.06 \\ 0.14 & 0.99 \end{pmatrix} \underbrace{\begin{pmatrix} -0.97 & -0.26 \\ 0.26 & -0.97 \end{pmatrix}}_{\mathbf{Q}_U} &= \left(\begin{array}{cc} 0.60 & 0.00 \\ 0.79 & 0.15 \\ \hline 0.12 & -0.99 \end{array}\right) =: \mathbf{Z}'_U \\ \begin{pmatrix} 0.00 & 0.60 \\ -0.15 & 0.79 \\ 0.99 & 0.12 \end{pmatrix} \underbrace{\begin{pmatrix} 0.00 & -1.00 \\ 1.00 & 0.00 \end{pmatrix}}_{\mathbf{Q}_L} &= \left(\begin{array}{cc} 0.60 & 0.00 \\ 0.79 & 0.15 \\ \hline 0.12 & -0.99 \end{array}\right) =: \mathbf{Z}'_L \end{aligned} \tag{A.6}$$

Notice, that in the right hand side of Eq. A.6 the 2×2 submatrix above the red line is $\mathbf{R}^\dagger$.

In order to test the accuracy of this method we calculate the difference of the resulting eigenvector representations $\|\mathbf{Z}'_U - \mathbf{Z}'_L\|_2 = 3.4 \cdot 10^{-16}$, which is very close to double precision accuracy. In Table A.1 we performed numerical tests for a 70×70 matrix with highly degenerate eigenvalues and even in this extreme case the numerical accuracy is below 10^{-13}.

Degree of degeneracy	Numerical error $\|\mathbf{Z}'_U - \mathbf{Z}'_L\|_2$
3	$3.96 \cdot 10^{-16}$
5	$1.93 \cdot 10^{-15}$
7	$5.25 \cdot 10^{-15}$
9	$8.35 \cdot 10^{-15}$
11	$1.46 \cdot 10^{-14}$
13	$4.03 \cdot 10^{-14}$
15	$1.76 \cdot 10^{-14}$

Tab. A.1.: The algorithm described in this section has been applied to a 70×70 matrix with highly degenerate eigenvalues. The first columns gives the degree of degeneracy for a set of eigenvalues and the second columns gives the difference in the eigenvectors after this algorithm has been applied to a pair of different representations of this eigenspace.

Lastly, we would like to point out that in the case of a non-degenerate eigenvalue the phase is uniquely determined by performing this algorithm for the $n_{\text{deg}} = 1$ special case. We thank Christoph Friedrich for the fruitful discussion on this topic.

B

Subtracting the local exchange

In the LAPW basis the Hamiltonian inside the muffin-tins is setup as [46]

$$H^{\text{MT}}_{\hat{G}'\hat{G}}(\hat{k}) = \langle \varphi_{\hat{k}\hat{G}'} | \mathcal{H}_{\text{MT}} | \varphi_{\hat{k}\hat{G}} \rangle \,. \tag{B.1}$$

This can be split into two parts: the spherical Hamiltonian $\mathcal{H}_{\text{sph}}$ and the non-spherical contributions to the potential $\mathcal{V}_{\text{nonsph}}$

$$H^{\text{MT}}_{\hat{G}'\hat{G}}(\hat{k}) = \underbrace{\langle \varphi_{\hat{k}\hat{G}'} | \mathcal{H}_{\text{sph}} | \varphi_{\hat{k}\hat{G}} \rangle}_{H^{\text{sph}}_{\hat{G}'\hat{G}}(\hat{k})} + \langle \varphi_{\hat{k}\hat{G}'} | \mathcal{V}_{\text{nonsph}} | \varphi_{\hat{k}\hat{G}} \rangle \,. \tag{B.2}$$

If we pull out the α and β coefficients, then $H^{\text{sph}}_{\hat{G}'\hat{G}}(\hat{k})$ breaks down into four types of integrals which are straight forward to evaluate, since u and $\dot{u}$ are designed to diagonalize the radial Schrödinger equation and its derivative:

$$\int_{\text{MT}a} u_l(\|\hat{r}_a\|) Y_{lm}(\hat{e}_a)\, H^{\text{sph}}_{\hat{G}'\hat{G}}(\hat{k})\, u_{l'}(\|\hat{r}_a\|) Y_{l'm'}(\hat{e}_a) = \delta_{ll'}\delta_{mm'} E^a_l \tag{B.3}$$

$$\int_{\text{MT}a} u_l(\|\hat{r}_a\|) Y_{lm}(\hat{e}_a)\, H^{\text{sph}}_{\hat{G}'\hat{G}}(\hat{k})\, \dot{u}_{l'}(\|\hat{r}_a\|) Y_{l'm'}(\hat{e}_a) = \delta_{ll'}\delta_{mm'} \tag{B.4}$$

$$\int_{\text{MT}a} \dot{u}_l(\|\hat{r}_a\|) Y_{lm}(\hat{e}_a)\, H^{\text{sph}}_{\hat{G}'\hat{G}}(\hat{k})\, u_{l'}(\|\hat{r}_a\|) Y_{l'm'}(\hat{e}_a) = 0 \tag{B.5}$$

$$\begin{aligned}&\int_{\text{MT}a} \dot{u}_l(\|\hat{r}_a\|) Y_{lm}(\hat{e}_a)\, H^{\text{sph}}_{\hat{G}'\hat{G}}(\hat{k})\, \dot{u}_{l'}(\|\hat{r}_a\|) Y_{l'm'}(\hat{e}_a) = \\ &\delta_{ll'}\delta_{mm'} E^a_l \int_{\text{MT}a} \dot{u}_l(\|\hat{r}_a\|) Y_{lm}(\hat{e}_a)\, \dot{u}_{l'}(\|\hat{r}_a\|) Y_{l'm'}(\hat{e}_a).\end{aligned} \tag{B.6}$$

Here however, the radial Schrödinger equation used to construct u and $\dot{u}$ relies on the local exchange-correlation potential and does not contain any information about the non-local potential

$$\mathcal{H}^{\text{sph}} = \mathcal{T} + \mathcal{V}^{\text{loc}}_{xc} + \mathcal{V}_{\text{eff}}. \tag{B.7}$$

However, the hybrid exchange-correlation potential has the form

$$\mathcal{V}^{\text{hyb}}_{xc} = \alpha \mathcal{V}^{\text{exact}}_{x} + (1-\alpha)\mathcal{V}^{\text{loc}}_{x} + \mathcal{V}^{\text{loc}}_{c}. \tag{B.8}$$

Here $\mathcal{V}_x^{\mathrm{loc}}$ and $\mathcal{V}_c^{\mathrm{loc}}$ are the exchange and correlation parts of the local exchange-correlation potential. In the case of PBE0 $\alpha = {}^1\!/_4$ and the local exchange-correlation potential is PBE [3].

Therefore, in the case of an hybrid exchange-correlation functional we need to subtract part of the local exchange from the Hamiltonian and add the corresponding non-local exchange

$$\begin{aligned} H^{\mathrm{MT}}_{\hat{G}'\hat{G}}(\hat{k}) = {} & \underbrace{\langle \varphi_{\hat{k}\hat{G}'} | \mathcal{H}_{\mathrm{sph}} | \varphi_{\hat{k}\hat{G}} \rangle}_{H^{\mathrm{sph}}_{\hat{G}'\hat{G}}(\hat{k})} + \langle \varphi_{\hat{k}\hat{G}'} | \mathcal{V}_{\mathrm{nonsph}} | \varphi_{\hat{k}\hat{G}} \rangle \\ & + \alpha \left[\langle \varphi_{\hat{k}\hat{G}'} | \mathcal{V}_x^{\mathrm{exact}} | \varphi_{\hat{k}\hat{G}} \rangle - \langle \varphi_{\hat{k}\hat{G}'} | \mathcal{V}_x^{\mathrm{loc}} | \varphi_{\hat{k}\hat{G}} \rangle \right]. \end{aligned} \tag{B.9}$$

In FLEUR's legacy implementation of the hybrid exchange-correlation functionals this term was explicitly calculated and subtracted from the Hamiltonian. However, as you can see in Fig. B.1 this routine has a very poor parallel performance. Therefore, we now calculate this term as a spherical special case using the routine for the non-spherical potential. This routine has extensive performance tuning work done to it and therefore has a much better parallel performance.

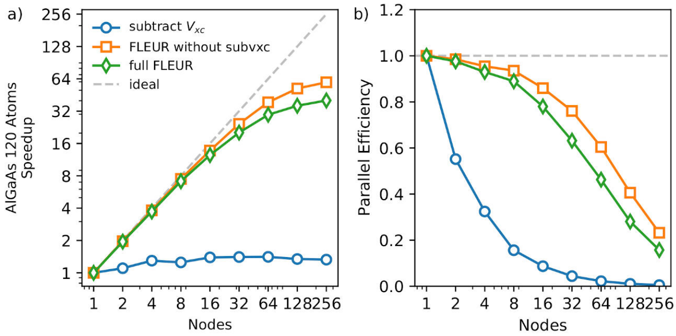

Fig. B.1.: In a) the speedup for three code parts is shown. In blue the speedup of the explicit subtraction of E_x^{loc} is shown and in green and orange FLEURs overall speedup with and without this routine is shown. In b) the corresponding parallel efficiencies are shown.

Since these changes were implemented after some of the measurement shown in Chapter 5 were taken, we manually subtracted the runtime of this routine in all measurement which still contained it. The new way of subtracting the local exchange has no visible performance impact. I would like to thank Dr. Daniel Wortmann for implementing this improved way to subtract the local exchange.

C

Using fast fourier transforms to evaluate the mixed-product basis

One of the most time consuming parts of a calculation with a hybrid exchange-correlation functional is the calculation of the mixed-product basis $\left\langle \phi^{\sigma}_{n\hat{k}} \middle| \phi^{\sigma}_{n''\hat{k}-\hat{q}} M_{\hat{q},I} \right\rangle$, specifically in the interstitial region. Here the MPB is the real-space product of two plane waves and a step function(cf. Eq. 3.22). The wave functions are stored in their reciprocal representation and so is the MPB. An obvious way to calculate the MPB in the interstitial is to evaluate the convolution explicitly as

$$\sum_{\hat{G}} z^{*\sigma}_{\hat{G}\hat{k}n} \, z^{*\sigma}_{(\hat{G}+\hat{G}')(\hat{k}+\hat{q})n'} \, . \tag{C.1}$$

This approach has a few advantages: using symmetries of the system, we can reduce the number of terms calculated to a minimum. Additionally, while the algorithm has a computational complexity of $\mathcal{O}({N_{\text{bas}}}^2)$, its prefactor is quite small. A key disadvantage is, that the use of symmetries causes an unpredictable memory access pattern and therefore a lot of cache misses. Furthermore, this code has a lot of branches, which makes it difficult to achieve a good performance on GPUs. Alternatively, we can evaluate this term by employing the convolution theorem

$$\mathcal{F}[f * g] = \mathcal{F}[f] \cdot \mathcal{F}[g], \tag{C.2}$$

where $\mathcal{F}$ is the fourier transform operator and $*$ indicates a convolution. We can rewrite

$$\sum_{\hat{G}} z^{*\sigma}_{\hat{G}\hat{k}n} \, z^{*\sigma}_{(\hat{G}+\hat{G}')(\hat{k}+\hat{q})n'} \tag{C.3}$$

$$= \mathcal{F}^{-1}\left(\mathcal{F}\left(\sum_{\hat{G}} z^{*\sigma}_{\hat{G}\hat{k}n} \, z^{*\sigma}_{(\hat{G}+\hat{G}')(\hat{k}+\hat{q})n'} \right)\right) \tag{C.4}$$

$$= \mathcal{F}^{-1}\left(\mathcal{F}\left(\sum_{\hat{G}} z^{*\sigma}_{\hat{G}\hat{k}n} \right) \cdot \mathcal{F}\left(z^{*\sigma}_{(\hat{G}+\hat{G}')(\hat{k}+\hat{q})n'} \right)\right) , \tag{C.5}$$

where $\mathcal{F}^{-1}$ is the inverse Fourier transform. By using this expression we can evaluate the MPB in the interstitial with a computational complexity of $\mathcal{O}(N_{\text{bas}} \log(N_{\text{bas}}))$ [148]. While the constant prefactor is larger compared to the direct evaluation, using Fourier transformations allows rely on highly optimized libraries. In Fig. C.1 the runtime of these two algorithms is compared for a number of systems and the cross over point for these two algorithms seems to be roughly between 7 and 24 atoms. Since the runtime of hybrid functional calculations with less than fifty atoms is not dominated by this code part, the Fourier transformation is used to evaluate the MPB in the interstitial regardless of the system size.

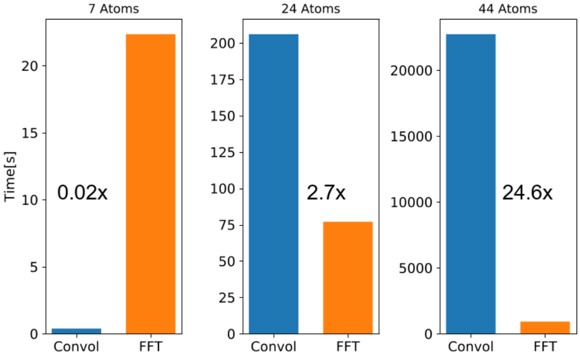

Fig. C.1.: Three systems were calculated using both the direct evaluation of the convolution and the FFT algorithm. While the constant prefactor of the direct evaluation is lower than that of the FFT algorithm, the FFT algorithm is faster for large systems, due to its lower runtime complexity. The speedup of the FFT algorithm over the explicit evaluation is indicated for each system.

Acknowledgement

I would like to thank numerous people for supporting me throughout my work on this thesis.

First and foremost, I would like to thank my supervisor Prof. Stefan Blügel for the opportunity to work on such an interesting topic in such a nice environment.

I would like to express my gratitude to Prof. Matthias Müller for reviewing this thesis and giving me access to the vast knowledge of his institute.

I am grateful to Prof. Uwe Klemradt for agreeing to serve as chair of my PhD committee and to Prof. Dante Kennes for serving as the examiner on this committee.

I would also like to thank Dr. Daniel Wortmann, Dr. Christian Terboven and Dr. Gregor Michalicek for their supervision and a lot of fruitful discussion and everyone else at the IAS-1 for making it a joy to work there.

I am grateful to Dr. Uliana Alekseeva, Dr. Juba Bouaziz, Alexander Neukirchen, Jenessa Lancaster, Dr. Gregor Michalicek and Dr. Daniel Wortmann for kindly proofreading this mauscript.

I would like to thank my family for their moral and financial support throughout my education and last, but not least I would like to thank my wife, Luotong, for everything.

Eidesstattliche Erklärung

Ich, Matthias Redies erklärt hiermit, dass diese Dissertation und die darin dargelegten Inhalte die eigenen sind und selbstständig, als Ergebnis der eigenen originären Forschung, generiert wurden.

Hiermit erkläre ich an Eides statt

1. Diese Arbeit wurde vollständig oder grösstenteils in der Phase als Doktorand dieser Fakultät und Universität angefertigt;
2. Sofern irgendein Bestandteil dieser Dissertation zuvor für einen akademischen Abschluss oder eine andere Qualifikation an dieser oder einer anderen Institution verwendet wurde, wurde dies klar angezeigt;
3. Wenn immer andere eigene- oder Veröffentlichungen Dritter herangezogen wurden, wurden diese klar benannt;
4. Wenn aus anderen eigenen- oder Veröffentlichungen Dritter zitiert wurde, wurde stets die Quelle hierfür angegeben. Diese Dissertation ist vollständig meine eigene Arbeit, mit der Ausnahme solcher Zitate;
5. Alle wesentlichen Quellen von Unterstützung wurden benannt;
6. Wenn immer ein Teil dieser Dissertation auf der Zusammenarbeit mit anderen basiert, wurde von mir klar gekennzeichnet, was von anderen und was von mir selbst erarbeitet wurde;
7. Kein Teil dieser Arbeit wurde vor deren Einreichung veröffentlicht.

Aachen, July 19, 2022

Matthias Redies

Band / Volume 246
Tailoring neuroelectronic interfaces via combinations of oxides and molecular layers
X. Yuan (2021), 113 pp
ISBN: 978-3-95806-572-7

Band / Volume 247
Stoichiometric control and magnetoelectric coupling in artificial multiferroic heterostructures
P. Schöffmann (2021), vii, 176 pp
ISBN: 978-3-95806-575-8

Band / Volume 248
A Unified Framework for Functional Renormalisation Group Calculations and its Application to Three Dimensional Hubbard Models
J. Ehrlich (2021), xvi, 213 pp
ISBN: 978-3-95806-582-6

Band / Volume 249
Photoemission electron microscopy of magneto-ionic effects in $La_{0.7}Sr_{0.3}MnO_3$
M. Wilhelm (2021), 134 pp
ISBN: 978-3-95806-592-5

Band / Volume 250
Development of a Multiplexer System and Measurement of the Neutron Yield for a Low-Energy Accelerator-Driven Neutron Source
M. Rimmler (2021), v, 200 pp
ISBN: 978-3-95806-600-7

Band / Volume 251
Resolving interface effects in voltage controlled magnetic heterostructures using advanced neutron scattering and electron microscopy methods
T. Bhatnagar-Schöffmann (2021), ix, 171 pp
ISBN: 978-3-95806-604-5

Band / Volume 252
Strain development of *Gluconobacter oxydans* and *Pseudomonas putida* for production of the sweetener 5-ketofructose
K. Wohlers (2022), VI, 118 pp
ISBN: 978-3-95806-612-0

Band / Volume 253
Topological magnonic properties of two-dimensional magnetic materials
L. Zhang (2022), xx, 154 pp
ISBN: 978-3-95806-621-2

Band / Volume 254
Role of secondary metabolites in antiphage defense in *Streptomyces*
Aël Hardy (2022), IV, 193 pp
ISBN: 978-3-95806-633-5

Band / Volume 255
Neutron Scattering
Lectures of the JCNS Laboratory Course held at Forschungszentrum Jülich and at the Heinz-Maier-Leibnitz Zentrum Garching
edited by T. Brückel, S. Förster, M. Kruteva, M. Zobel, and R. Zorn (2022), ca. 300 pp
ISBN: 978-3-95806-634-2

Band / Volume 256
Magnetoelectric Interactions in Multiferroic Thin-film Heterosystems and Nanostructures
H. Gökdemir (2022), x, 140 pp
ISBN: 978-3-95806-635-9

Band / Volume 257
High-Performance Computing Approach to Hybrid Functionals in the All-Electron DFT Code FLEUR
M. Redies (2022), xi, 109 pp
ISBN: 978-3-95806-639-7

Weitere ***Schriften des Verlags im Forschungszentrum Jülich*** unter
http://wwwzb1.fz-juelich.de/verlagextern1/index.asp

Schlüsseltechnologien / Key Technologies
Band / Volume 257
ISBN 978-3-95806-639-7

Mitglied der Helmholtz-Gemeinschaft

JÜLICH
Forschungszentrum