Aalto University
School of Science

Simiam Ghan

Self-consistent implementation of the Van der Waals Density Functional into the FHI-aims electronic structure code

Master´s thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology in the Degree Programme in Engineering Physics and Mathematics.

Espoo, 04.10.2011

Supervisor: Professor Martti Puska

Instructor: Dr. Sc. Ville Havu
**Aalto University**  
**School of Science**  

**ABSTRACT OF THE MASTER’S THESIS**

**Author:** Simiam Ghan

**Title:** Self-consistent implementation of the Van der Waals Density Functional into the FHI-aims electronic structure code

**Title in Finnish:** Van der Waals tiheys-funktionalaalin implementaatio FHI-aims ohjelmassa

**Degree Programme:** Degree Programme in Engineering Physics and Mathematics

**Major subject:** Engineering Physics (F3005)  
**Minor subject:** Computational Science and Engineering (F3009)

**Chair (code):** TfY-105

**Supervisor:** Professor Martti Puska  
**Instructor:** Dr. Sc. Ville Havu

**Abstract:**

Van der Waals (vdW) interactions play an important role in the bonding of sparse biological matter. However, standard Density Functional Theory (DFT), which is commonly used to simulate solid-state systems, fails to properly describe vdW interactions. This is because the common generalized-gradient or local density approximations of electron exchange and correlation fail to account for nonlocal electron correlations which are the cause of vdW forces. Schemes for restoring van der Waals interactions to the framework of DFT are currently the subject of much attention.

This work describes the implementation of the Langreth-Lundqvist Van der Waals Density Functional (vdW-DF) into the electronic structure code FHI-aims. The vdW-DF has been implemented in numerous other codes successfully and has been shown to greatly increase the accuracy of calculations with systems in which vdW interactions are prominent. Implementation of vdW-DF into FHI-aims, however, will require special considerations due to the use of all-electron density rather than pseudopotentials. Among these, a method is developed to allow accurate interpolation of all-electron density from an adaptive octree grid.

The implementation is tested in calculations of the interaction energies of noble gas dimers and the S22 set of biological complexes. The accuracy achieved is comparable to that of other implementations and is a significant improvement over standard DFT. This allows for the prospect of simulations of important biological systems with vdW-DF in FHI-aims. The computational performance of the algorithm is assessed and options for continuing development are outlined.

**Date:** 04.10.2011  
**Language:** English  
**Number of pages:** 67+2

**Keywords:** van der Waals, Density Functional Theory, nonlocal correlation, FHI-aims, Octrees
Contents

1 Introduction 1

2 Theory of Van der Waals Interactions 3
   2.1 Derivation ........................................... 4
   2.2 Calculating Dispersion .............................. 7

3 Theory of Electronic Structure 9
   3.1 The many-body problem ............................. 9
   3.2 Hartree-Fock Method ............................... 10
   3.3 Post-Hartree-Fock Methods ......................... 12
      3.3.1 Configuration Interaction ..................... 12
      3.3.2 Møller-Plesset Perturbation Theory ............. 13
      3.3.3 Coupled Clusters ............................. 14
   3.4 Density Functional Theory ......................... 16
      3.4.1 Kohn-Sham equations .......................... 16
      3.4.2 Approximating exchange and correlation .......... 18
      3.4.3 Semi-empirical van der Waals corrections ....... 20
      3.4.4 The Langreth-Lundqvist van der Waals density functional ............................................. 20

4 Self-consistent Implementation of the Van der Waals Density Functional 23
   4.1 The FHI-aims ab initio molecular simulations package .................................................. 23
   4.2 Algorithm for self-consistent potential .............. 24
      4.2.1 Numerical Integration ........................... 25
      4.2.2 Kernel Function ................................ 30
      4.2.3 Octree method for all-electron density interpolation .............. 31
   4.3 Testing and Verification ............................ 36
      4.3.1 Octree settings ............................... 36
      4.3.2 Integration grid settings ....................... 40
      4.3.3 Confirming self-consistent Energy and Forces .......... 42

5 Benchmark Calculations 44
   5.1 Noble Gas Dimers ................................ 44
   5.2 S22 Benchmark ..................................... 46
   5.3 Computational Performance ......................... 49
      5.3.1 Scaling ....................................... 49
      5.3.2 Run time ..................................... 50
      5.3.3 Memory ...................................... 51
6 Discussion

7 Summary and Conclusions

Appendices

A Derivatives of $q_0(r)$

B 3D Cubic Hermite Interpolation

References
Chapter 1

Introduction

Kohn-Sham Density Functional Theory (DFT) has enjoyed wide success in the prediction of properties of condensed matter systems. By formulating the electronic Schrödinger equation in terms of electron density, ground state properties of many-body systems can be solved at much greater speeds than with more sophisticated quantum chemical methods, with comparable accuracy in many cases. This capability has been applied extensively, for example to the study of molecules and nanoclusters on surfaces [1, 2], the interpretation of images from Atomic Force Microscopy [3] and to the structural and electronic properties of periodic solids [4, 5, 6, 7] 1. Despite success in modeling solid-state and surface systems, however, standard DFT still faces challenges with sparse matter systems such as biological molecules, layered systems and select surface systems, as well as noble gases [11, 12]. Inaccuracies occur because van der Waals (vdW) interactions, which characterize these systems, are not accounted for by the standard approximations for electron exchange and correlation. In order to accurately describe vdW interactions, corrections are needed which either introduce vdW semi-empirically or, more desirably, reformulate the entire exchange-correlation functional to better describe vdW phenomena. This is no easy task, however, as nonlocal correlation, which is responsible for vdW forces, can be very expensive to calculate. The efficient restoration of vdW to DFT has been the subject of much work over the past decade.

Recent proposals for vdW corrections in DFT have achieved considerable success in the modeling of vdW-influenced systems. Improved descriptions have been achieved for surface systems such as CO on transition metals [13], benzene on silicon [14], organic molecules on insulators [15], water bilayers on metal surfaces [16, 17] and graphene on metal substrates [18, 19, 20]. Also, vdW corrections have been shown to stabilize the structure of double-walled carbon nanotubes and layered graphene [21], as well as graphite and oxide crystals [22, 23], systems where standard DFT predicts incorrect structure. Importantly, vdW-corrected DFT offers the exciting prospect of improved ab initio simulations of biological molecules. Such systems, due to their size and complexity, have often been accessible only to less accurate semi-empirical or experimental methods [24]. VdW corrections enable correct prediction of, for example, the structure of helical polypeptides, basic constituents of proteins, at high temperatures [25]. Molec-

---

1 For general reviews see [8, 9, 10].
ular crystals of hemozoin, a product of the malaria parasite, are described correctly with vdW corrections, allowing for study of possible mechanisms for anti-malarial drugs [26]. While standard DFT often suffices to model the structure of DNA [27, 28, 29, 30], vdW corrections have been shown necessary for the study of stacked nucleobases [31, 32] and the intercalation of molecules between stacked base pairs [33, 34], a mechanism central to cancer drug design. Indeed, vdW-corrected DFT appears capable of offering valuable insights in important problems in medicine and molecular biology. If sufficiently developed, vdW-corrected DFT could also be applied to larger systems, for example in the prediction of protein folding [35] or the design of molecules which inhibit viral proteins [36]. There is clearly much potential for the application of DFT to the investigation of fundamental biological processes. Recent studies, however, have been limited to relatively small systems. Further work is needed to test and extend the use of existing vdW correction schemes - and their implementations - to progressively larger systems, with the goal of efficient performance for thousands of atoms.

This work concerns the implementation of the Langreth-Lundqvist Van der Waals Density Functional (vdW-DF) [37] into the electronic structure theory code FHIaims [38]. Though only one of many approaches to vdW, this vdW-DF is the most commonly implemented and perhaps the most thoroughly tested of existing vdW functionals. Implementation of the vdW-DF into FHI-aims, however, requires special considerations as the code utilizes all-electron density rather than the more common method of pseudopotentials. The implementation is found to greatly improve the description of benchmark vdW-dominated systems with a manageable increase in computational expense.

The work is structured as follows. Chapter 2 presents the theory of van der Waals interactions, demonstrating their nonlocal nature. Chapter 3 discusses the various ab initio methods available for capturing the vdW interaction and shows DFT to be the most promising method for large systems. Competing approaches for including vdW into DFT are then discussed. Chapter 4 presents the self-consistent implementation of the vdW-DF into FHI-aims and introduces an algorithm for interpolation of the all-electron density. Benchmark calculations are then presented in Chapter 5 for noble gas dimers and the S22 set of biological complexes. Computational performance of the algorithm is also analyzed. Options for continuing development are discussed in Chapter 6 and conclusions are drawn in Chapter 7.
Chapter 2

Theory of Van der Waals Interactions

Attempts to describe long-range interactions between individual molecules were motivated in 1873 by Johannes D. van der Waals’ equation of state,

\[(P + a \frac{n^2}{V^2})(V - nb) = nRT,\]  \hspace{1cm} (2.1)

which approximates the behavior of real gases using two parameters \(a\) and \(b\). Here \(P, V, T, n\) and \(R\) are pressure, volume, temperature, number of moles of gas and the universal gas constant. By modifying the ideal gas law, \(PV = nRT\), to account for a pairwise intermolecular attraction (through parameter \(a\)) and the finite volume occupied by gas molecules (parameter \(b\)), van der Waals provided an equation which predicts many qualitative properties of real fluids. These include an incompressible liquid phase and the existence of a critical temperature at which liquid and gas phases coexist [39, 40].

The success of the equation motivated research into the origins of these intermolecular forces which are often referred to broadly as van der Waals forces. It was known from the equation that the attractive force between two neutral molecules should be proportional to \(1/R^6\) at long range [41]. Various early considerations sought to explain this by assuming the presence of permanent electric dipoles or quadrupoles in the molecules. Keesom [42] derived a \(1/R^6\) asymptotic attraction for two dipolar molecules by averaging over their possible orientations using Boltzmann statistics. Debye [43] achieved the same asymptotic form by considering the induction of dipoles in naturally nonpolar molecules by nearby molecules with permanent dipoles. The problem with such descriptions was made clear by the arise of quantum mechanics, which demonstrated that atoms such as the noble gases had spherically symmetric charge distributions. This undermined descriptions which relied on the presence of permanent dipoles or higher-order multipoles. Even for atoms with natural quadrupoles, the quadrupole moment given by quantum mechanics was found to be much smaller in magnitude than that required by available descriptions of the forces [44, 45].

In 1930, F. London [46] offered a quantum mechanical formulation for the long-range forces between nonpolar atoms and molecules which he termed the dispersion
force. Instead of permanent dipoles, forces arose due to quantum mechanical fluctuations in the electron clouds of atoms. In this description the zero-point motion of electrons results in instantaneous dipoles which induce correlated dipoles in other atoms or molecules, illustrated in Figure 2.1. The resulting pairs of dipoles experience an attractive force of the form $1/R^6$ at large separations.

2.1 Derivation

The quantum mechanical origins of the dispersion interaction can be demonstrated for two Hydrogen atoms using perturbation theory [47] [48]. In this framework, the exact Hamiltonian of a system is approximated by an unperturbed Hamiltonian $H^0$ and a perturbation potential: $H_{\text{exact}} \cong H^0 + V$. The total energy of the system can then be calculated from the energy of the unperturbed Hamiltonian, $E^0$, and a series of corrections from the perturbation:

$$E = E^0 + E^1 + E^2 + \cdots$$

(2.2)

where with superscripts we denote the order of the corrections.

![Figure 2.1: Origin of van der Waals forces.](image)

For large separations, the wavefunctions of individual H atoms can be assumed not to overlap and the interaction between systems can be treated as a perturbation. This
arrangement leads to an unperturbed Hamiltonian of (see Figure 2.2)

\[ H^0 = -\frac{\hbar^2}{2m}(\nabla_1^2 + \nabla_2^2) - \frac{e^2}{|\mathbf{r}_1|} - \frac{e^2}{|\mathbf{r}_2|} \]  

(2.3)

and the perturbation

\[ V_{12} = \left( \frac{e^2}{R} + \frac{e^2}{|\mathbf{r}_1 - \mathbf{r}_2 - \mathbf{R}|} - \frac{e^2}{|\mathbf{r}_1 - \mathbf{R}|} - \frac{e^2}{|\mathbf{r}_2 + \mathbf{R}|} \right). \]  

(2.4)

Here Coulomb’s constant \((1/4\pi\epsilon_0)\) has been set to unity following practice in the literature \([47, 49]\). For large separation \(\mathbf{R}\), the interaction \(V_{12}\) can be given as a multipole expansion and approximated to the leading order by

\[ V_{12} = \frac{e^2}{R^3}(x_1x_2 + y_1y_2 - 2z_1z_2), \]  

(2.5)

where the cartesian coordinates \(\mathbf{r}_i = (x_i, y_i, z_i)\) of the electrons are used. Now, according to the basic expressions of perturbation theory \([47]\), the energy contribution \(\Delta E\) of the first and second-order perturbation corrections is

\[ \Delta E = \langle 0 | V_{12} | 0 \rangle + \sum_{n} \frac{|\langle 0 | V_{12} | n \rangle|^2}{E_0 - E_n}, \]  

(2.6)

where the second-order term sums over allowed energy levels \(n\). For a spherically symmetric (nonpolar) charge distribution the first order correction is zero, while for atoms or molecules with dipole or higher-order moments, this term describes orientation and induction contributions to the total energy. Inserting Equation (2.5) into (2.6) and assuming spherically-symmetric hydrogen wavefunctions, we arrive at

\[ \Delta E = -\frac{C_6}{R^6}, \]  

(2.7)

with the dispersion coefficient

\[ C_6 = -e^4 \sum_{m,n} \frac{|x_{0,m}x_{0,n} + y_{0,m}y_{0,n} - 2z_{0,m}z_{0,n}|^2}{2E_0 - E_m - E_n}. \]  

(2.8)

Here \(E_0, E_n\) are ground and excited-state energies of individual atoms, respectively. Matrix elements \(x_{0,n}\) are to be evaluated using wavefunctions of these states. F. London rewrote Equation (2.7) using dispersion f-values or oscillator strengths \(f_{lm} = (2m/\hbar^2)(E_m - E_l)|z_{nl}|^2\), defined for dipole transitions between the states \(|l\) and \(|m\). This results in the form

\[ \Delta E = \frac{3e^4\hbar^4}{2m^2R^6} \sum_{m,n} \frac{f_{0,m}f_{0,n}}{(E_m - E_0)(E_n - E_0)(E_0 - E_m) + (E_0 - E_n)}, \]  

(2.9)

which is generally valid for atoms and molecules. Now, oscillator strengths for an atom with \(n_1\) electrons obey the sum rule

\[ \sum_{m} f_{lm} = n_1. \]  

(2.10)
2.1. DERIVATION

Thus, for two atoms or molecules with dominant transitions at frequencies $\omega_1$ and $\omega_2$, we can set the corresponding dominant oscillator strengths equal to $n_1$ and $n_2$. Introducing these modifications, Equation (2.9) simplifies to the well known formula of F. London [49, 46, 44]:

$$\Delta E = -\frac{3e^4\hbar}{2m^2R^6} \frac{n_1n_1}{\omega_1\omega_2(\omega_1 + \omega_2)}. \tag{2.11}$$

Equation (2.11) can be used to estimate the dispersion interaction between two molecules using their dominant transition (absorption) frequencies $\omega_{1,2}$. Values of $\omega$ can be determined from experimental spectroscopic data. The expression is not, however, practical for accurately calculating the dispersion coefficient $C_6$.

It is possible to rewrite Equation (2.9) in terms of the dynamic dipole polarizability [47]

$$\alpha(\omega) = 2e^2 \sum_n \frac{(E_n - E_0)|z_n|^2}{(E_n - E_0)^2 - (\hbar\omega)^2}$$

$$= \frac{e^2}{m} \sum_n \frac{f_{\omega}}{\omega^2 - \omega_n^2} \tag{2.12}$$

and using the identity

$$\frac{1}{ab(a + b)} = \frac{2}{\pi} \int_0^\infty \frac{du}{(a^2 + u^2)(b^2 + u^2)}. \tag{2.13}$$

Inserting these into Equation (2.9), we arrive at an improved formula for the dispersion interaction energy, first given by Casimir and Polder [50]:

$$E_{\text{disp}} = -\frac{3\hbar}{\pi R^6} \int_0^\infty d\omega_1 |i\omega_1^2| \alpha_2(i\omega). \tag{2.14}$$

Here, the imaginary $iu$ replaces the real $\omega$ of Equation (2.12). Calculations of the dynamic polarizability $\alpha(i\omega)$ of atoms and small molecules can be performed with ab initio methods [48].

Equations (2.11) and (2.14) demonstrate the attraction arising between well-separated neutral, spherical charge distributions. Natural fluctuations of the electrons between energy states are assumed to occur most often at a molecule’s characteristic absorption frequency $\omega$, resulting in instantaneous dipoles.\(^1\) When two systems interact, fluctuating dipoles become correlated and reduce the total energy of the system, leading to the so-called dispersion attraction. Because the dispersion force is the result of correlations of well-separated electrons, it is highly nonlocal. As we shall see below, this property makes dispersion expensive to calculate within DFT.

While the term 'van der Waals' is used in the literature to refer to dispersion, induction and orientation effects, it will in this work refer only to London dispersion as derived above.\(^2\)

\(^1\) Only dipole fluctuations are accounted for in $C_6$; less significant higher order multipoles can be treated separately [48].

\(^2\) Casimir forces [51],[50], a type of dispersion force occurring at very long ranges (several thousands of Ångstroms), will also not be treated in this work.
2.2 Calculating Dispersion

Methods for calculating van der Waals interactions vary according to the size and structure of the system. For well-separated pairs of atoms and small molecules, Equations (2.11) and (2.14) provide, in principle, the exact dispersion energy. Exact expressions for larger systems are not available, however, and approximations must be used.

![Interaction energy vs Separation](image)

**Figure 2.3:** Lennard-Jones pair-potential, \( V(r) = \epsilon \left( \left( \frac{r_m}{r} \right)^{12} - 2 \left( \frac{r_m}{r} \right)^{6} \right) \), shown for an Argon dimer and calibrated to experimental values for the well depth \( \epsilon = 12 \text{ meV} \) and equilibrium bond length \( r_m = 3.76 \text{ Å} \) [32].

A natural approach for well-separated systems with many atoms is to sum up the van der Waals interactions over distant atom pairs. This technique is employed in simulations with semi-empirical pair-potentials. Here the \( 1/R^6 \) asymptotic form is combined with a short-range repulsive potential and calibrated to approximate the full interatomic potential [53, 54, 55]. Approximative pair-potentials, such as that shown in Figure 2.3, allow for computationally inexpensive simulations of systems over a broad range of sizes. The assumption of additivity of dispersion interactions is fundamentally incorrect, however, as correlated fluctuations between atom pairs spread to other atoms, resulting in many-body interactions [12]. This, in addition to their dependence on empirical parameters, severely limits the accuracy of pair-potential models.

A more accurate approach for macroscopic systems is given by the theory of Lifshitz [56, 57], which formulates dispersion in terms of bulk properties such as the dielectric constant, ignoring atomic structure altogether. Exact expressions for the interactions between simple systems such as parallel surfaces and atoms near surfaces can be derived, valid at separations much larger than interatomic distances. In practice, however, real systems such as large interacting proteins have complex geometries which must be approximated. Lifshitz theory can then be used to calculate dispersion...
coefficients for bodies with simplified geometries [48].

The above options for calculating van der Waals interactions show the lack of a universal method which could be used accurately regardless of systems' size and structure. In particular, the above methods are perturbative and limited therefore to application only with well-separated systems. For systems with overlapping wavefunctions, many-body interactions such as electron exchange and local electron correlation occur, and it becomes necessary to consider the dispersion interaction in the context of the full Schrödinger equation. Such 'ab initio' approaches offer high accuracy and transferability, as they are free of empirical parameters and are in principle suitable for systems of all sizes and separations. Due to their accuracy and the wide range of electronic, mechanical and thermodynamical properties which can be calculated, ab initio methods serve as the primary tools of modern computational materials physics. The expense of calculating the electronic structure of any system is significant, however, particularly if nonlocal correlation of the electrons is to be included. As we shall see below, Density Functional Theory with nonlocal correlation holds great potential for ab initio simulations of large vdW systems.
Chapter 3

Theory of Electronic Structure

3.1 The many-body problem

In the case of atoms with overlapping wavefunctions, interactions are no longer limited to the nonlocal correlation of charge distributions. Bonding becomes more complex in this regime as dispersion interactions become indistinguishable from local electron correlations, and electron exchange and other bonding phenomena take place. An exact description of interactions is provided in this case by the many-body Schrödinger equation

$$\hat{H}\Psi_i(R_1, R_2, \ldots; r_1, r_2, \ldots) = E_i\Psi_i(R_1, R_2, \ldots; r_1, r_2, \ldots),$$

(3.1)

with the full Hamiltonian

$$\hat{H} = -\frac{\hbar^2}{2m_e} \sum_i \nabla_i^2 - \frac{1}{4\pi\epsilon_0} \sum_{i,j} \frac{Z_i e^2}{|r_i - R_j|} + \frac{1}{2} \frac{1}{4\pi\epsilon_0} \sum_{i\neq j} \frac{e^2}{|r_i - r_j|}$$

$$- \sum_i \frac{\hbar^2}{2M_i} \nabla_i^2 + \frac{1}{2} \frac{1}{4\pi\epsilon_0} \sum_{i\neq j} \frac{Z_i Z_j e^2}{|R_i - R_j|}.$$  (3.2)

Here indices $i$ and $I$ refer to individual electrons and nuclei, respectively. Vectors $r_i$ are the positions of the electrons and $R_I$, $M_I$ and $Z_I$ are the position, mass and atomic number of each nuclei. We see in Equation (3.2) kinetic energy operators for electrons and nuclei as well as coulombic potentials for electron-nucleus, electron-electron and nucleus-nucleus interactions. In principle, the solution of the problem offers the exact ground and excited state energies $E_i$ of any collection of atoms, as well the many-body wavefunction $\Psi_i$. In practice, however, the large dimension of the problem (with 3N variables for N particles) makes analytical solutions impossible for all but the simplest cases, such as the hydrogen atom.

In order to enable numerical solutions of the many-body problem, approximations are necessary. Chief among these is the Born-Oppenheimer approximation [58, 9], which observes that nuclei are essentially stationary objects when compared to the velocities of electrons. Under this assumption, the many-body problem is reduced to one where nuclear coordinates $R_I$ are stationary parameters which determine an external potential
3.2. HARTREE-FOCK METHOD

\( V_{\text{ext}} \) that influences the electrons. Contributions of the nuclei to the system’s total energy can be calculated separately. The Hamiltonian for the electrons then reads

\[
\hat{H} = -\frac{\hbar^2}{2m_e} \sum_i V_i^2 - \frac{1}{4\pi\epsilon_0} \sum_{i,j} \frac{Z_i e^2}{|r_i - R_j|} + \frac{1}{2} \frac{1}{4\pi\epsilon_0} \sum_{i \neq j} \frac{e^2}{|r_i - r_j|}.
\]

(3.3)

Solutions with this simplified Hamiltonian are still computationally prohibitive, however, as the dimension of the problem is still large for most systems. Achieving solutions which are computationally feasible yet accurately represent most types of interactions is the goal of numerous approximative methods.

Here we summarize a number of the most common approaches to the many-body problem which are capable of capturing nonlocal electron correlations. The Hartree-Fock method, which neglects correlation, is presented first as it is the basis for more advanced correlated methods. While highly accurate methods are available, we shall see that most are far too expensive for simulations with large systems.

3.2 Hartree-Fock Method

The Hartree-Fock (HF) method is perhaps the most conceptually simple approach to electronic structure. By approximating the many-electron wavefunction as an antisymmetric product of one-electron wavefunctions, the difficult N-electron Schrödinger equation can be reduced to a set of N one-electron problems which is computationally easier to solve. The approach has been used widely to predict the chemical properties of atoms and molecules, but because electron correlations are lost in the approximations the method is not fit for systems with dispersion interactions. Here we outline the primary features of the HF method; complete descriptions are available elsewhere [58, 59, 9].

In the electronic Hamiltonian, electron-pair interactions

\[
g(i, j) = \frac{1}{|r_i - r_j|}
\]

(3.4)

and the resulting electron-electron interaction potential

\[
V_{ee} = \sum_{i < j}^N g(i, j)
\]

(3.5)

require the presence of multiple electron coordinates simultaneously, causing significant computational expense (here and below, Hartree atomic units \( e = h = m_e = 1 / 4\pi\epsilon_0 = 1 \) are used for simplicity). To reduce this expense, electrons can be decoupled by means of the so-called independent particle approximation. Here, an electron at \( \mathbf{r} \) is assumed to interact with other electrons through a mean field determined by their wavefunctions \( \psi_i(\mathbf{r}') \). Individual correlations between electrons present in Equation (3.5) are thus neglected, and each electron now experiences the so-called Hartree potential:
3.2. HARTREE-FOCK METHOD

\[ \hat{V}_{\text{Hartree}}(\mathbf{r}) = \sum_{l=1}^{N} \int d\mathbf{r}' |\psi_l(\mathbf{r}')|^2 \frac{1}{|\mathbf{r} - \mathbf{r}'|}. \]  

(3.6)

In the Hartree-Fock method, an independent particle formulation of the Schrödinger equation is achieved by approximating the many-electron wavefunction \( \Psi(\mathbf{r}_1, ..., \mathbf{r}_N) \) by an antisymmetric product of one-electron wavefunctions \( \psi(\mathbf{r}) \), in accordance with the Pauli exclusion principle. This can be written as a Slater determinant

\[ \Psi_{AS}(\mathbf{r}_1, ..., \mathbf{r}_N) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \psi_1(\mathbf{r}_1) & \psi_2(\mathbf{r}_1) & \cdots & \psi_N(\mathbf{r}_1) \\ \psi_1(\mathbf{r}_2) & \psi_2(\mathbf{r}_2) & \cdots & \psi_N(\mathbf{r}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_1(\mathbf{r}_N) & \psi_2(\mathbf{r}_N) & \cdots & \psi_N(\mathbf{r}_N) \end{vmatrix}. \]  

(3.7)

Now, according to the variational principle [58], the energy eigenvalue \( E \) of any trial many-electron wavefunction \( \Psi \) will be greater or equal to the actual ground state energy \( E_0 \) of a many-body system:

\[ E_0 \leq E[\Psi] = \frac{\langle \Psi | H | \Psi \rangle}{\langle \Psi | \Psi \rangle}. \]  

(3.8)

Solving a many-body Schrödinger equation for the ground state wavefunction and energy hence simply becomes a matter of minimizing the energy expectation value with respect to a trial wavefunction. Applying energy minimization conditions and using the Slater determinant (3.7) as a trial wavefunction, the many-body Schrödinger equation can be reformulated into a one-electron eigenproblem

\[ \hat{F}\psi_k(\mathbf{r}) = \epsilon_k \psi_k(\mathbf{r}) \]  

(3.9)

containing the so-called Fock operator,

\[ \hat{F}\psi_k(\mathbf{r}) = \left[ -\frac{1}{2} \nabla^2 - \sum_n \frac{Z_n}{|\mathbf{r} - \mathbf{R}_n|} \right] \psi_k(\mathbf{r}) + \sum_{l=1}^{N} \int d\mathbf{r}' |\psi_l(\mathbf{r}')|^2 \frac{1}{|\mathbf{r} - \mathbf{r}'|} \psi_k(\mathbf{r}) \]

\[ - \sum_{l=1}^{N} \int d\mathbf{r}' \psi_l^*(\mathbf{r}') \psi_k(\mathbf{r}') \psi_l(\mathbf{r}). \]  

(3.10)

For an N-electron system, the N lowest-energy wavefunctions \( \psi_k(\mathbf{r}) \) solved here form a solution for the ground-state many-electron wavefunction. We see in the Fock operator terms for electron kinetic energy and electron-nucleus interactions, followed by the Hartree term described above. The last term, dubbed an exchange term, cancels out the self-interaction present in the Hartree term for \( l = k \). A mean field has replaced individual electron correlations, allowing for a computational advantage. Equations (3.9) and (3.10), however, are now nonlinear with respect to the orbitals \( \psi_k(\mathbf{r}) \) as the Fock operator depends on these. The scheme must hence be solved self-consistently.
starting from an initial guess for the form of the wavefunctions \( \psi_k(\mathbf{r}) \). These are usually written as a linear combination of atomic orbitals,

\[
\psi_k(\mathbf{r}) = \sum_l C_{l,k} \phi_l(\mathbf{r}),
\]

(3.11)

which transforms the Hartree-Fock Equation (3.9) into an eigenproblem for the coefficients \( C_{l,k} \) and eigenvalues \( \epsilon_k \):

\[
\mathbf{F} \mathbf{C}_k = \epsilon_k \mathbf{S} \mathbf{C}_k.
\]

(3.12)

Here \( \mathbf{S} \) is the overlap matrix for the atomic orbital basis set \cite{58}, with elements defined as

\[
S_{lm} = \langle \phi_l | \phi_m \rangle = \int \phi_l(\mathbf{r}) \phi_m(\mathbf{r}) d\mathbf{r}.
\]

(3.13)

The Hartree-Fock Equations (3.9) and (3.10) scale formally as \( N^4 \) with the number of atoms and nearly linear scaling implementations are possible through further approximations \cite{60}. The use of a single Slater determinant to approximate the many-electron wavefunction allows for this advantageous reformulation of the many-body problem. For accurate results including electron correlation, however, a treatment of electron-electron interactions beyond the mean-field approximation is necessary. A number of approaches build on the HF method in this respect. These either incorporate correlations into the HF method through perturbation theory or build correlated wavefunctions using the HF wavefunction as a starting point. We briefly look at three such methods below.

### 3.3 Post-Hartree-Fock Methods

#### 3.3.1 Configuration Interaction

Electron correlations can be accounted for exactly - in principle - in the method of Configuration Interactions\(^1\). Here, the many-electron Schrödinger problem is again solved for via the variational theorem by minimizing the energy eigenvalue with respect to a trial wavefunction. Instead of a single Slater determinant, the trial wavefunction is now a linear combination of so-called configuration functions \( \Phi \),

\[
\Psi = \sum_R c_R \Phi_R + \sum_I c_I \Phi_I,
\]

(3.14)

where \( \Phi_R \) are reference functions and \( \Phi_I \) are determinants containing excitations. Often a single reference, the HF wavefunction, is used. Excited determinants are formed from the HF wavefunction by replacing single (S), double (D) or more one-electron orbitals with any of the excited or 'virtual' HF orbitals which remained unoccupied in the HF ground state. Reformulating the Schrödinger problem with this trial wavefunction and conditions for minimization of the energy yields a massive eigenproblem from which the wavefunction coefficients \( c_R \) and \( c_I \) can be solved.

The Configuration Interaction (CI) method offers the highest accuracy of all post-Hartree-Fock methods but is also the most expensive. The use of excited determinants

\(^1\)For detailed reviews, see Refs. \cite{59, 61, 62, 63}.\}
in the wavefunction allows for correlation of the electrons but heavily increases the dimension of the CI eigenproblem. Full-CI, which uses all possible excited determinants, solves the Schrödinger equation exactly within the basis set used for the electron orbitals. The number of configuration functions (CF) involved in such calculations is exorbitant, however. For example, on the order of $10^{12}$ CF’s are needed to calculate the ground state of the Ne atom with a proper basis set [48]. This number increases drastically with the number of electrons in the system, as well as with the dimension of the basis set. As a result, full-CI is in practice limited to systems with only about a dozen electrons. It is common to reduce the number of CF’s used by truncating the second sum in Equation (3.14) to include only single and double excitations (CISD). This reduces computational expense, but with a loss of size-extensivity\(^2\) and some accuracy. While the high expense of CI calculations limits their use to very small systems, results of full-CI, for example, serve as important benchmarks for testing much faster, approximative methods [12].

### 3.3.2 Møller-Plesset Perturbation Theory

Møller-Plesset perturbation theory [64] accounts for electron correlations by treating them as a perturbation of the Hartree-Fock Hamiltonian. The method is not as accurate as CI but is considerably faster. Here, the sum of one-electron Fock operators is taken as the unperturbed Hamiltonian, with correlations represented by a perturbation potential:

$$\hat{H} = \sum_{i=1}^{N} \hat{F}(i) + \hat{V}. \quad (3.15)$$

$\hat{V}$ is hence defined as the difference between the exact (electronic) Hamiltonian and Fock operators, which is

$$\hat{V} = \sum_{i<j}^{N} g(i,j) - \sum_{j}^{N} \hat{V}_H(j), \quad (3.16)$$

or the difference between the instantaneous and mean-field (Hartree) interaction energies of Equations (3.5) and (3.6). $\hat{V}$ written in this manner is known as a fluctuation potential. Proceeding as in Section 2 with the standard expressions of perturbation theory, we can now calculate the ground-state energy of the system as a sum of the unperturbed energy and perturbative corrections:

$$E_0 = E_0^0 + E_0^1 + E_0^2 + \ldots \quad (3.17)$$

The unperturbed ground-state energy $E_0^0$ is calculated by taking the expectation value of the sum of Fock operators (now denoted by $\hat{F}$) with respect to HF wavefunctions $\Psi_0$:

$$E_0^0 = \langle \Psi_0 | \hat{F} | \Psi_0 \rangle = \sum_{i}^{N} \epsilon_i. \quad (3.18)$$

\(^2\)This is a desirable property for many-body methods whereby the energy calculated for two non-interacting systems will equal the sum of energies calculated separately for each system.
3.3. POST-HARTREE-FOCK METHODS

In this case the unperturbed energy is just a sum of HF one-electron eigenvalues. For the first-order correction we have

$$ E_0^1 = \langle \Psi_0 | \hat{V} | \Psi_0 \rangle = \langle \Psi_0 | \hat{H} - \hat{F} | \Psi_0 \rangle = E_{HF} - \sum_i^N \epsilon_i. \quad (3.19) $$

Hence, through first order, the Møller-Plesset energy is simply the Hartree-Fock energy. Indeed, in MP¹ theory, correlation contributions only occur in second and higher-order terms. The second order correction is of the form

$$ E_0^2 = \sum_{v \neq 0} \frac{|\langle \Psi_0 | \hat{V} | \Psi_v \rangle|^2}{E_0 - E_v}, \quad (3.20) $$

where the sum is over all excited versions of the wavefunction $\Psi_v$. These are formed, as in the CI method, by promoting one or more electrons in $\Psi_0$ from their ground-state spin-orbitals to excited orbitals. Due to reasons demonstrated elsewhere³, determinants with single excitations, as well as those with triple or more excitations have zero contributions to the second-order correction. Only double excitations remain, leading to the form

$$ E_0^2 = \sum_{i<j}^{N_{exc}} \sum_{a<b}^{N_e} \frac{|\langle \Psi_0 | \hat{V}_{ij} | \Psi_{ab} \rangle|^2}{\epsilon_i + \epsilon_j - \epsilon_a - \epsilon_b}. \quad (3.21) $$

Here $\Psi_{ij}^{ab}$ indicates a Slater determinant with spin-orbitals $i$ and $j$ replaced with excited states $a$ and $b$, and $N_e$ is the number of excited or 'virtual' spin-orbitals available, determined by the atomic orbital basis set. This expression can be rewritten in terms of one-electron spin-orbitals, after which the second order MP correction, dubbed MP2, can be solved.

Møller-Plesset calculations are considerably faster than CI but are much less accurate and still limited to small systems. MP2 accounts for as much as 80-90% of the correlation energy [60] and scales as $N^5$ with the number of atoms [24]. Higher order corrections have also been implemented (MP4, MP5), but scaling quickly becomes unmanageable ($N^{10}$ or worse for MP5 [66]). Also, calculations have been shown not to converge for some systems as higher-order corrections are introduced [48, 12]. For higher accuracy results, the method of Coupled Clusters, described below, is hence often preferred. Nevertheless, MP2 calculations are among the fastest of post-Hartree-Fock methods and are hence commonly used in computational chemistry for small molecules where correlation plays a significant role.

3.3.3 Coupled Clusters

The method of Coupled Clusters (CC) is able to account for correlation through a unique parametrization of the many-electron wavefunction⁴. This is given a so-called cluster expansion,

$$ \Phi_{CC}(\mathbf{r}_1, \ldots, \mathbf{r}_N) = e^T | \Psi_{HF}(\mathbf{r}_1, \ldots, \mathbf{r}_N) \rangle, \quad (3.22) $$

³Full derivations of the MP equations are available in [59, 65, 60, 64].
⁴For detailed reviews, see Refs. [61, 59, 62, 12, 48].
in which an excitation operator $\hat{T}$ transforms the ground state HF wavefunction into a linear combination of determinants containing a finite number of excitations. $\hat{T}$ is often truncated to allow only single and double excitations,

$$\hat{T} = \hat{T}_1 + \hat{T}_2$$  \hspace{1cm} (3.23)

where the singles operator, for example, is defined as

$$\hat{T}_1 = \sum_i \hat{t}_i = \sum_{i,a} t^{a}_i a_i^\dagger a_i.$$  \hspace{1cm} (3.24)

The effect of operator $\hat{t}_i$ is to delete the one-electron orbital $\psi_i$ from the HF wavefunction and replace it by an excited orbital $\psi_a$. This is accomplished through operators $a_i^\dagger$ and $a_i$ which introduce and remove orbitals, respectively. $\hat{T}_1$ thus expands the HF wavefunction in terms of all possible singly excited determinants $\Phi^{a}_{i}$, each with coefficient $t^{a}_i$. Similar expressions for $\hat{T}_2$ give an expansion in terms of doubly excited determinants $\Phi^{ab}_{ij}$.

It now remains to solve for the expansion coefficients. Inserting (3.22) into the Schrödinger equation leads to

$$\hat{H} e^{\hat{T}} |\Psi_0\rangle = E e^{\hat{T}} |\Psi_0\rangle,$$  \hspace{1cm} (3.25)

where we now denote the HF wavefunction by $\Psi_0$. Left-multiplying both sides by $e^{-\hat{T}}$ and integrating with respect to $\Psi_0$ yields an expression for energy,

$$\langle \Psi_0 | e^{-\hat{T}} \hat{H} e^{\hat{T}} |\Psi_0\rangle = E.$$  \hspace{1cm} (3.26)

Integrating instead with respect to an excited determinant (with for example two excitations) yields

$$\langle \Phi^{ab}_{ij} | e^{-\hat{T}} \hat{H} e^{\hat{T}} |\Psi_0\rangle = 0,$$  \hspace{1cm} (3.27)

where the orthonormality of $\Phi^{ab}_{ij}$ and $\Psi_0$ has been assumed. Equations (3.26) and (3.27) are the basic expressions of the Coupled Cluster method. In Equation (3.27) we have a condition for the value of coefficient $t^{ab}_ij$ of the excited determinant $\Phi^{ab}_{ij}$. A system of equations hence emerges, from which all unknown coefficients can be solved. Each Equation (3.27) contains the full operator $\hat{T}$ in the exponent, however, and as a result the equations are nonlinear with respect to the coefficients and must be solved iteratively. Total system energy is then given by Equation (3.26).

The CC method differs from other methods above in that it does not use variational minimization of energy. Computational expense hence arises from the iterative solution of algebraic expressions (3.27) rather than from a large-dimension eigenproblem such as that in CI. If excitations of all electron orbitals were allowed, the Coupled Cluster equations would solve the Schrödinger equation exactly for the given basis set and within the BO approximation [62]. This would be very expensive, however. The most common implementation, CCSD(T), which includes single (S), double (D) and a perturbative treatment of triple (T) excitations, scales typically as $N^7$ [24] with respect to the number of atoms. This is faster than CI methods and allows for comparable accuracy [12], but still limits the use of CC to systems with tens of atoms. The high accuracy of CCSD(T) has earned the designation 'Gold Standard', and the method has
often been applied to calculate benchmark values for small molecular systems. For example, the popular S22 set of interaction energies of biological duplexes [67], including vdW-bonded systems, was calculated with CCSD(T) and is used as a benchmark in this and other works concerning vdW. These benchmarks play an important role in the development of ab initio methods faster than CC.

The quantum-chemical methods described above are all able to achieve accuracies greater than the HF method for correlated systems, but, as we have seen, their scaling with system size (MP2 at $N^5$ being the fastest) makes them all intractable for simulations of molecules with more than tens of atoms. In order to calculate electron correlation for systems with hundreds or thousands of atoms, including many important biological molecules, a more promising method is the electron Density Functional Theory. Without nonlocal correlation, this scales from $N^1$ [68, 38] to $N^3$ [24] depending on the implementation and is hence readily applied to large systems. Importantly, the inclusion of nonlocal correlation into DFT has been shown possible without severe increases in computational expense. We now take a closer look at DFT and the methods which can most efficiently extend its application to vdW systems.

### 3.4 Density Functional Theory

Density Functional Theory (DFT) is the most common ab initio approach to simulations of solid-state systems. Its main feature is the reformulation of the many-body Schrödinger equation in terms of electron density $n(r)$, which is a simple scalar quantity, instead of the many-electron wavefunction containing the 3N electron coordinates. This follows from the work of Hohenberg and Kohn [69] which established that in principle any property of a system, for example the ground-state energy or position of nuclei, can be expressed as a functional of the ground state density $n_0(r)$. A practical scheme for calculating the ground state density and energy is given by Kohn and Sham [70].

#### 3.4.1 Kohn-Sham equations

Under the Kohn-Sham ansatz [9, 58, 10, 70], the ground state electron density of an interacting many-body system is assumed equal to the density of a fictitious non-interacting system. This leads to a Hartree-Fock-like approach where the many-body Schrödinger equation is again reformulated into an auxiliary set of independent-particle problems. Here, however, many-body interactions such as correlation are accounted for exactly, in principle.

Building on a proposal of Hohenberg and Kohn [69], Kohn and Sham write the energy functional of an interacting many-body system as

$$E_{KS}[n] = T_s + \int V_{ext}(r)n(r)dr + \frac{1}{2} \int \int \frac{n(r)n(r')}{|r-r'|}drdr' + E_{xc}[n],$$

(3.28)

where $n(r)$ is the density of a non-interacting system, defined in terms of one-electron
3.4. DENSITY FUNCTIONAL THEORY

spin-orbitals:

\[ n(r) = \sum_{k=1}^{N} |\psi_k(r)|^2. \] (3.29)

Here the sum is over the \( N \) spin orbitals \( \psi_k(r) \) which have the lowest eigenvalues \( \epsilon_k \) (defined below), and \( N \) is the number of electrons. Equation (3.28) contains terms for the external and Hartree energies, as well as \( T_s \), which is the kinetic energy of noninteracting electrons,

\[ T_s = \sum_{k=1}^{N} \langle \psi_k | -\frac{1}{2} \nabla^2 | \psi_k \rangle. \] (3.30)

Any many-body effects not included in these terms are grouped into the so-called exchange-correlation energy \( E_{xc} \).

Now, according to the theorem of Hohenberg and Kohn, the energy functional (3.28) will have its minimum at the ground state energy, and this will occur at the ground state density. Solving for the ground state density hence becomes a matter of minimizing Equation (3.28) with respect to density. As \( T_s \) is expressed in terms of spin-orbitals rather than density, variation of (3.28) with respect to spin-orbitals is performed, and density-dependent terms are varied with respect to density via the chain rule. Assuming real-valued spin-orbitals \( \psi_k(r) \), the minimization condition hence becomes [9]

\[ \frac{\delta E_{KS}}{\delta \psi_k(r)} = \frac{\delta T_s}{\delta \psi_k(r)} + \left[ \frac{\delta E_{ext}}{\delta n(r)} + \frac{\delta E_{hartree}}{\delta n(r)} + \frac{\delta E_{xc}}{\delta n(r)} \right] \frac{\delta n(r)}{\delta \psi_k(r)} = 0, \] (3.31)

subject to the constraints of orthonormal spin-orbitals,

\[ \langle \psi_k | \psi_l \rangle = \delta_{k,l}. \] (3.32)

Using the method of Lagrange multipliers, the above constrained minimization problem can be reformulated as a set of Schrödinger-like one-particle equations

\[ \left[ -\frac{1}{2} \nabla^2 + V_{KS}(r) \right] \psi_k = \epsilon_k \psi_k, \] (3.33)

where

\[ V_{KS}(r) = V_{ext}(r) + \int \frac{n(r')}{|r - r'|} dr' + V_{xc}[n](r). \] (3.34)

Here, the middle term is recognized as the Hartree potential \( V_{Hartree} \), and the exchange-correlation potential \( V_{xc} \) is defined as the functional derivative:

\[ V_{xc}[n](r) = \frac{\delta E_{xc}[n]}{\delta n(r)}. \] (3.35)

Given Equation (3.33), it would be convenient to rewrite the energy functional (3.28) in terms of the KS eigenvalues \( \epsilon_k \). Comparing Equations (3.34) and (3.28), we see that the sum of eigenvalues \( \epsilon_k \) includes twice the Hartree energy, as the Hartree potential in (3.34) allows for unphysical self-interactions. Also, differences in the exchange-correlation terms occur. Taking these differences into account, the total energy can now

---

5Here \( r \) will refer to both spatial and spin coordinates of electrons.
be expressed as
\[ E_{KS}[n] = \sum_k^N \varepsilon_k - \frac{1}{2} \int \frac{n(r')n(r)}{|r - r'|} dr'dr - \int V_{xc}(r)n(r)dr + E_{xc}[n]. \quad (3.36) \]

Equations (3.33), (3.34), (3.29) and (3.36) comprise the Kohn-Sham method for calculating the ground state density and energy of a many-body system. Self-consistent iterations are used to solve the ground-state density, which enters the eigenproblem (3.33) non-linearly. The ground state energy is then given by Equation (3.36) or (3.28).

It should be noted that the formulation above is exact as no approximations have yet been made. Indeed, this is a significant achievement: the computationally intractable many-body Schrödinger equation has been mapped exactly onto an auxiliary set of one-electron problems which is much easier to solve, though valid only for the ground state of the system. All necessary many-body effects have been transferred to the exchange-correlation (XC) functional. In practice, however, no exact form of this functional is known and approximations must be used. This is discussed in detail below.

### 3.4.2 Approximating exchange and correlation

The many-body interactions of \( E_{xc} \) include exchange effects such as the Pauli repulsion of like-spinned electrons (after the exclusion principle) and the correlation of electrons resulting from Coulombic repulsion between individual particles. As \( E_{xc} \) can comprise up to 100% of the atomization energy of a system [10], it is essential to develop quality approximations for accurate models. Three of the most common approximations are described below.

#### Local Density Approximation

Kohn and Sham [70] originally proposed to approximate the exchange-correlation energy \( E_{xc} \) of real systems using the known exchange-correlation energy per particle \( \epsilon_{xc}^{\text{hom}}[n] \) of a homogenous electron gas. The resulting local density approximation (LDA),
\[ E_{xc}^{\text{LDA}}[n] = \int n(r)\epsilon_{xc}^{\text{hom}}[n(r)]dr, \quad (3.37) \]

assumes a uniform density distribution and was originally expected to be useful only for systems with slowly-varying density. However, LDA has proven surprisingly accurate for a wide variety of materials, including very inhomogenous cases [9, 8]. This is in part due to a convenient cancellation of error between the exchange and correlation components of the functional. Since \( \epsilon_{xc}^{\text{hom}}[n] \) has been derived from theory, LDA is a truly ab initio functional which fulfills a number of formal conditions set upon the exact XC functional. This includes a sum rule regarding the so-called exchange-correlation hole [9, 8]. LDA is also inexpensive to calculate.

Despite achieving qualitatively good results in many cases, LDA typically overestimates bond strengths for solids and molecules, with bond lengths too short by several
percent. Atomization energies for molecules are predicted within the order of 1 eV, which is far from the desired chemical accuracy of 0.05 eV \[^8\]. Furthermore, fundamental band gaps\[^6\] are consistently underestimated to the point that some semiconductors, such as Germanium, are falsely predicted to be semimetallic \[^8\]. The generalized-gradient approach described below seeks to improve accuracy in these areas.

**Generalized Gradient Approximation**

The Generalized Gradient Approximation (GGA) is an attempt to preserve the benefits of the LDA yet improve accuracy by accounting for non-uniformities in a system’s charge density. Nonuniformities enter the equation in the form of the gradient of density,

\[
E_{xc}^{GGA}[n] = \int F(n(r), \nabla n(r))dr. \tag{3.38}
\]

Various versions of the functional \(F(n, \nabla n)\) are available in the literature \[^10\]. A commonly used form of the GGA is that of Perdew, Burke, and Ernzerhof (PBE) \[^9\]. While it also significantly underestimates band gaps \[^8\], it predicts, for example, atomization energies within the order of 0.3 eV compared to experimental values \[^8\] - an improvement over LDA. RevPBE \[^7\]1, a revised version of PBE, provides improved accuracy for atomization energies and is also commonly used. Errors in both LDA and GGA largely originate from the Hartree self-interaction error which is not compensated for by their exchange portions.

**Hybrid density functionals**

In an effort to improve the accuracy of band structure calculations, hybrid functionals have been developed which replace a fraction of the LDA or GGA exchange energy with the exchange energy taken from Hartree-Fock theory. The underlying observation is that band gaps evaluated with the pure density functionals LDA and GGA are consistently too small, while band gaps produced by the Hartree-Fock method are consistently too large \[^8, 9\]. The PBE0 \[^72\], HSE03 \[^73\] and HSE06 \[^74\] hybrid functionals are outcomes of this effort, and each shows improved results for band structure and lattice parameter calculations \[^75, 76\]. Improvements are partially due to the cancellation of the Hartree self-interaction error by the Hartree-Fock exchange. Including exact Hartree-Fock exchange is expensive, however, and screening methods are necessary to allow for simulations with large molecules and solids.

A common feature among the functionals described above is their local or semi-local treatment of electron correlation. Electron correlation is present only in very short-range form, and the neglect of nonlocal correlation means that the dispersion interaction is not properly described. A number of approaches are available for incorporating dispersion forces into DFT calculations. It is possible to introduce them as a semi-empirical correction to the ground state energy after the KS problem has been

---

\[^6\]Band structure can be estimated in DFT using the KS eigenvalues \(\epsilon_k\). These are not physical energy levels, however.
solved, or reformulate the exchange-correlation functional to include nonlocal correlation in the self-consistent scheme. Møller-Plesset perturbation theory can also be applied to the KS equations, similar to with the Hartree-Fock method. Next we look at two promising methods for vdW forces in DFT.

3.4.3 Semi-empirical van der Waals corrections

Perhaps the simplest option for introducing vdW forces into DFT is through semi-empirical pair potentials of the form $C_6/R^6$. These introduce a long-range asymptotic attraction between neutral atoms which is not present in the standard exchange-correlation functionals. In the method of Grimme [77], a dispersion correction is calculated as an add-on to the converged Kohn-Sham ground state energy:

$$E_{\text{disp}} = -\sum_{i<j}^{N_{\text{atoms}}} f_{\text{disp}}(R_{ij}) \frac{C_6^{ij}}{R_{ij}^6}. \quad (3.39)$$

Here the sum is over all atom pairs. To avoid the singularity at zero separation, a damping function $f_{\text{disp}}(R_{ij})$ is employed. The interatomic dispersion coefficients $C_6^{ij}$, discussed in Section 2, are determined empirically.

In order to reduce dependence on empirical parameters, it is possible to formulate the dispersion coefficients $C_6^{ij}$ in Equation (3.39) as functionals of density [78]. Tkatchenko and Scheffler (TS) [79] follow this approach and offer a $C_6/R^6$ correction which is dependent on reference values of homonuclear dispersion coefficients only ($C_6^{ij}$ for $i$ and $j$ of same species only). The dispersion coefficients between nuclei of different species and between molecules are then calculated from the electron density. This method has been shown to accurately predict dispersion coefficients and provides overall high accuracy as coefficients are updated with the density rather than held rigid. In a study of helical polypeptide molecules, the TS method was shown to greatly improve the description of molecule structure at high temperatures compared to standard PBE [25]. The method also is relatively cheap, scaling nearly the same as PBE. The speed and accuracy of this semi-empirical method make it a promising candidate for further simulations with large biological systems [26].

3.4.4 The Langreth-Lundqvist van der Waals density functional

While semi-empirical corrections have the advantage of speed, they do not enter the Kohn-Sham framework self-consistently and hence have no direct influence on the electron density\(^7\). In order to solve for density which includes nonlocal correlation, modification of the exchange-correlation functional itself is required.

A number of efforts over the past decade have sought to formulate an accurate and efficient nonlocal correlation functional. For the full XC functional, these typically use a GGA functional for exchange and LDA for correlation. Nonlocal correlation is given

\(^{7}\)A self-consistent implementation of the TS correction is being considered [79].
3.4. DENSITY FUNCTIONAL THEORY

Figure 3.1: Kernel function of the LLvdW density functional, \( \varphi(D, \Delta) \), as a function of the effective distance \( D \) for three values of the asymmetry parameter \( \Delta \).

separately:

\[
E_{xc} \left[ n \right] = E_{x}^{GGA} + E_{c}^{LDA} + E_{c}^{nl},
\]

(3.40)

Perhaps the most successful approach so far has been that of Langreth and Lundqvist et al. [37, 80, 33], which has seen a wide range of applications to small biological and surface systems [31, 33, 15, 23]. Starting from the adiabatic-connection theorem\(^9\), a compact expression for the nonlocal correlation energy is obtained [37]:

\[
E_{c}^{nl} \left[ n \right] = \frac{1}{2} \int \int n(\mathbf{r}) \varphi(\mathbf{r}, \mathbf{r}') n(\mathbf{r}') d\mathbf{r} d\mathbf{r}'.
\]

(3.41)

We see here a truly nonlocal interaction between regions of density at \( \mathbf{r} \) and \( \mathbf{r}' \), with the strength of interaction determined by a kernel function \( \varphi(\mathbf{r}, \mathbf{r}') = \varphi(q_{0}(\mathbf{r}), q_{0}(\mathbf{r}')) \). This is a universal function, suitable for all geometries, which depends on density and its gradient through the wavenumber \( q_{0}(\mathbf{r}) = q_{0}(n(\mathbf{r}), \nabla n(\mathbf{r})) \), defined in Section 4.2.2. Using the (dimensionless) effective distance

\[
D = \frac{q_{0}(\mathbf{r}') + q_{0}(\mathbf{r})}{2} |\mathbf{r} - \mathbf{r}'|,
\]

(3.42)

and an asymmetry parameter

\[
\Delta = \frac{q_{0}(\mathbf{r}) - q_{0}(\mathbf{r}')} {q_{0}(\mathbf{r}) + q_{0}(\mathbf{r}')}.
\]

(3.43)

the kernel can be presented in a scalable form, shown in Figure 3.1. We see that for long and intermediate effective distances the kernel is attractive, while at short range

---

\(^8\)This functional is hereafter referred to as LLvdW, vdW-DF or ‘the vdW density functional.’

\(^9\)A common starting point for many functionals; see usage for example in Refs. [48, 33, 9, 12].
there is repulsion. The area under the black curve in Figure 3.1 is zero meaning that $E_R^{vdW}$ is zero for systems of uniform density. Also, the kernel is designed to yield the correct $1/R^6$ asymptote for interactions of atoms and molecules.

Other vdW functionals are available which have the same form as Equation (3.41) but with different kernel functions. Voorhis et al. [82] allow for adjustment of parameters in their kernel according to the exchange functional used. Also kernel adjustments are possible to ensure proper calculation of $C_6$ dispersion coefficients. This approach is similar in cost [81] to the LLvdW but has seen comparably little application. Lee et al. [83] have slightly modified the LLvdW functional by changing one parameter in the kernel and using a different exchange functional (PW86 instead of revPBE). Indeed, the choice of exchange functional for vdW calculations has been its own topic of debate [84, 85]. These comparatively recent approaches have not yet been tested to the extent of the original LLvdW functional, however.

The LLvdW functional has received a number of successful implementations into mainstream electronic structure programs [37, 80, 83, 21, 85]. The implementation of Gulans et al. [85] into SIESTA\footnote{A linear-scaling DFT program using pseudopotentials and an atomic orbital basis set.} [86, 68] stands out as it achieves in principle linear scaling with respect to system size. While so far the functional has been applied only to relatively small systems such as stacked nucleobases [32] and carbon nanotubes [21], its accuracy, freedom from empirical parameters and the possibility for efficient implementation make it a leading candidate for restoring vdW interactions to DFT programs. Also, it can easily be modified to become the newer but less tested version of Lee et al. [83] if necessary. We now describe the implementation of this functional into the electronic structure theory code FHI-aims.
Chapter 4

Self-consistent Implementation of the Van der Waals Density Functional

4.1 The FHI-aims ab initio molecular simulations package

FHI-aims (Fritz-Haber Institute ab initio molecular simulations) \cite{fridriksson2003,fridriksson2004} is a package for $O(N)$-like scaling ab initio molecular dynamics using all-electron density rather than the more common method of pseudopotentials. Density is stored on atom-centered spherical grids rather than the Cartesian grids used in many pseudopotential-based codes. Atom-centered numerical functions are used for the atomic orbital basis set, and the package is capable of either serial or massively-parallel simulations of up to thousands of atoms, with clusters and periodic systems on equal footing.

While standard (LDA,GGA) and hybrid functionals are available with the package, the LLvdW density functional has been previously implemented only in post-processing form using a Monte-Carlo integration scheme \cite{wolff2007,voigt2010}. This allows for calculation of an energy correction, but not self-consistently and without a corresponding correction to atomic forces. The code also contains the TS vdW correction scheme. A self-consistent implementation of the vdW density functional would be desirable as it would be expected to provide greater accuracy and reliability than post-processing methods and hence allow for their verification. Also, self-consistency would allow for calculation of interatomic forces with the Hellmann-Feynman theorem\footnote{This allows interatomic forces to be expressed in terms of the self-consistent density \cite{hellmann1936}. Because the basis functions in FHI-aims depend on the atomic positions, Pullay forces must also be considered \cite{pullay1989}.}. This would allow for efficient relaxation of geometries.

The exchange-correlation functional enters FHI-aims\footnote{See for example Equation (29) in the FHI-aims reference \cite{fridriksson2003}.} through the Kohn-Sham Hamiltonian of Equation (3.34). This contains the XC potential $V_{xc}[\rho](\mathbf{r})$, which is defined
in Equation (3.35) as the functional derivative of $E_{xc}$:

$$V_{xc}[n](\mathbf{r}) = \frac{\delta E_{xc}[n]}{\delta n} = \frac{\partial e_{xc}}{\partial n} - \nabla \left( \frac{\partial e_{xc}}{\partial [\nabla n]^2} 2\nabla n \right).$$  \hfill (4.1)

Here, $E_{xc}$ is written in terms of the energy density $e_{xc}[n, |\nabla n|] = e_{xc}[n, |\nabla n|]n(\mathbf{r})$, where $e_{xc}[n, |\nabla n|]$ is the exchange-correlation energy per electron:

$$E_{xc} = \int e_{xc}[n, |\nabla n|]d\mathbf{r}. \hfill (4.2)$$

In FHI-aims, the outer gradient operator of Equation (4.1) is avoided using integration by parts when constructing the Hamiltonian matrix\(^3\). Thus, calculating $V_{xc}[n](\mathbf{r})$ is simply a matter of determining the derivatives $\frac{\partial e_{xc}}{\partial n}$ and $\frac{\partial e_{xc}}{\partial [\nabla n]^2}$. For GGA and LDA functionals, these derivatives are already in place in the code. For the LLvdW functional, we have from Equation (3.40) that

$$e_{xc} = e_x^{GGA} + e_c^{LDA} + e_{xc}^{nl}. \hfill (4.3)$$

Hence it will yet be necessary to implement expressions for $e_{xc}^{nl}$ and its derivatives. These can then be added to the GGA and LDA derivatives to achieve the full XC potential. Total XC energy is then given by Equation (4.2).

In this work the vdW-DF will be implemented with both PBE and revPBE exchange functionals. The resulting combinations will be named pbe-vdw and revpbe-vdw, respectively.

### 4.2 Algorithm for self-consistent potential

A number of implementations of the van der Waals density functional have been undertaken, both in self-consistent and post-processed form. Approaches differ primarily in the method used to calculate the expensive double integral,

$$E_{xc}^{nl}[n] = \frac{1}{2} \int \int n(\mathbf{r}) \varphi(\mathbf{r}, \mathbf{r}') n(\mathbf{r}') d\mathbf{r} d\mathbf{r}', \hfill (4.4)$$

and, for self-consistent implementations, in the formulation of the potential $V_{xc}^{nl}$. Originally, the double integral was evaluated directly in real space \([37, 90, 33, 83, 80]\) in a manner which scales as $N^2$ with the number of atoms - too slow for many large molecules. A self-consistent reciprocal-space implementation which uses Fast Fourier Transforms has achieved $N\log(N)$ scaling \([21]\). However, this is best suited for periodic systems - or nonperiodic systems with extensive simulation boxes - and for use with pseudopotentials. Thonhauser et al. \([80]\) have implemented the LLvdW functional self-consistently by deriving an expression for the potential $V_{xc}^{nl}$. The implementation gave results very similar to post-processed results and enabled calculation of interatomic forces. The computational complexity, again $N^2$, was relatively high, however. The self-consistent formulation chosen for this work is that of Gulans et al. \([89]\), which...

\(^3\)See Ref. [38] Equation (30).
achieves linear scaling through a simpler expression for the potential as well as use of an adaptive integration grid scheme.

The potential formulation of Gulans et al. [85] can be summarized as follows. From Equation (4.4) we have that the nonlocal correlation energy per electron is \(^4\)
\[
\epsilon_{\text{nl}}^c(r) = \frac{1}{2} \int \varphi(r, r') n(r') dr'.
\]  
(4.5)

Now, following Dion et al. [37] the kernel function \(\varphi(r, r')\) can be written in terms of the scaling variables \(D\) and \(\Delta\) from Equations (3.42) and (3.43): \(\varphi(r, r') = \varphi(D, \Delta)\). The functional derivatives \(\frac{\partial e_{\text{nl}}^c}{\partial n}\) and \(\frac{\partial e_{\text{nl}}^c}{\partial (|\nabla n|^2)}\) from Equation (4.1) can then be calculated in terms of \(D\) and \(\Delta\) using the chain rule. After some straightforward but tedious steps, we obtain \(^5\)
\[
\frac{\partial e_{\text{nl}}^c}{\partial n(r)} = 2e_{\text{nl}}^c + \frac{1}{2} n(r) \frac{\partial q_0(r)}{\partial n(r)}
\]
\[
\times \int n(r') \left( \frac{\partial \varphi}{\partial D} |r - r'| + \frac{\partial \varphi}{\partial \Delta} \left( \frac{4q_0(r')}{(q_0(r) + q_0(r'))^2} \right) \right) dr'.
\]  
(4.6)

and
\[
\frac{\partial e_{\text{nl}}^c}{\partial (|\nabla n|^2)} = \frac{1}{2} n(r) \frac{\partial q_0(r)}{\partial (|\nabla n|^2)}
\]
\[
\times \int n(r') \left( \frac{\partial \varphi}{\partial D} |r - r'| + \frac{\partial \varphi}{\partial \Delta} \left( \frac{4q_0(r')}{(q_0(r) + q_0(r'))^2} \right) \right) dr'.
\]  
(4.7)

Equations (4.5), (4.6) and (4.7) together with (4.1) form the basic expressions of our implementation of the LLvdW density functional. They contain only single integrals over \(r\) (indeed, (4.6) and (4.7) share the same integral) and can be evaluated in spherical coordinates. Together, the equations yield the potential formula of Gulans et al. [85], with a minor departure being that the potential is expressed here in terms of \(\frac{\partial e_{\text{nl}}^c}{\partial n}\) rather than \(\frac{\partial e_{\text{nl}}^c}{\partial (|\nabla n|^2)}\), following the practice in FHI-aims. The equations require as input density and its gradient at any point \(r'\), as well as values of the kernel function \(\varphi(D, \Delta)\) and \(q_0(r)\) and their functional derivatives. Also, a numerical integration scheme is needed. We now discuss these aspects in detail.

4.2.1 Numerical Integration

Integration in FHI-aims

In FHI-aims, calculation of the Hamiltonian matrix elements,
\[
H_{ij} = \int \psi_i(r) \hat{H}_{KS} \psi_j(r) dr,
\]  
(4.8)

\(^4\)For simplicity we write \(\epsilon_{\text{nl}}^c(r)\) here without showing the dependence on functions \([n, |\nabla n|]\).

\(^5\)For a full derivation, see Gulans et al. [85, 91, 92].
as well as the total exchange-correlation energy,
\[ E_{xc}[n] = \int \epsilon_{xc}(\mathbf{r}) n(\mathbf{r}) d\mathbf{r}, \]
(4.9)
is performed using numerical integration over atom-centered spherical grids. The use
of all-electron density results in peaks of density near atomic nuclei, making uniform
cartesian grids - commonly used in pseudopotential codes - unsuitable for this purpose.
Atom-centered grids are constructed using a set of so-called concentric Lebedev grids
distributed over a range of radii (shown as the black grid in Figure 4.1). The size of
the radial grid (i.e. number of shells) and the number of points on the shells can be
selected by the user through options for so-called light or tight grid settings [93]. Each
grid point stores values of the density and gradient, as well as other central variables.

For systems with multiple atoms, the atom-centered grids are allowed to overlap.
The extended integral of Equation (4.8), for example, is then calculated by dividing it
into atom-centered regions using a partitioning scheme:
\[ \int \psi_i(\mathbf{r}) \hat{H}_{KS} \psi_j(\mathbf{r}) d\mathbf{r} = \sum_{at=1}^{N_{at}} \int p_{at}(\mathbf{r}) \psi_i(\mathbf{r}) \hat{H}_{KS} \psi_j(\mathbf{r}) d\mathbf{r}. \]
(4.10)
Here, \( p_{at}(\mathbf{r}) \) are atom-centered partition functions whose sum at any point is unity.

Integration of the atom-centered integrals such as those in Equation (4.10) is per-
fomed in spherical coordinates using a combined radial and angular numerical quadra-
ture. According to the Lebedev quadrature [94], a surface integral over a unit sphere
can be approximated as
\[ \int_{0}^{2\pi} \int_{0}^{\pi} f(\theta, \phi) \sin(\theta) d\theta d\phi \approx 4\pi \sum_{t=1}^{N_{ang}} f(\theta_t, \phi_t) w_t, \]
(4.11)
where \( w_t \) and \( (\theta_t, \phi_t) \) are weights and points of a Lebedev grid. These are defined such
that spherical harmonic functions are integrated exactly up to a certain order. For
example the Lebedev grid of index (order) 7 has 86 grid points, and grids with up to
5810 points are available. The radial distribution of shells is determined by Delley et
al. [95] who give a radial quadrature of the form\(^6\)
\[ \int_{0}^{R} g(x) dx \approx \sum_{s=1}^{N_{rad}} g(x_s) w_s, \]
(4.12)
with expressions for the weights \( w_s \) and abscissas \( x_s \).

Combining the numerical quadrature and partitioning schemes, integrals such as
Equation (4.10) can be evaluated as a simple sum over grid points,
\[ \int \psi_i(\mathbf{r}) \hat{H}_{KS} \psi_j(\mathbf{r}) d\mathbf{r} \approx 4\pi \sum_{at=1}^{N_{at}} \sum_{rad=1}^{N_{rad}} \sum_{s=1}^{N_{ang}} \sum_{t=1}^{N_{ang}} p_{at}(\mathbf{r}) w_{rad}(s) w_{ang}(t) \psi_i(\mathbf{r}) \cdot [\hat{H}_{KS} \psi_j(\mathbf{r})], \]
(4.13)
\(^6\)See Equation (20) of Ref. [38].
where \( r = r(\alpha t, s, t) \).

\[ E_{nl}^c[n] = \frac{1}{2} \int \int n(r) \varphi(r, r') n(r') dr dr', \quad (4.14) \]

in FHI-aims will require two numerical quadratures, one for each integral. Writing this as

\[ E_{nl}^c[n] = \int n(r) \epsilon_{nl}^c(r) dr \quad (4.15) \]

with the energy per electron

\[ \epsilon_{nl}^c(r) = \frac{1}{2} \int \varphi(r, r') n(r') dr', \quad (4.16) \]

we see that \( E_{nl}^c \) consists of an outer integral, which can be evaluated on the FHI-aims integration grid, and a separate inner integral for \( \epsilon_{nl}^c(r) \). In previous implementations [80, 90], \( \epsilon_{nl}^c(r) \) was also evaluated by integrating over the primary grid. However, the codes in question were pseudopotential-based and used uniform Cartesian grids, which gave like accuracy for \( \epsilon_{nl}^c(r) \) regardless of location \( r \). This is not the case with the spherical grid(s) of FHI-aims, as the arrangement of grid points around a point \( r \) will vary with location. For consistent accuracy, following Gulans et al. [85], the integral in \( \epsilon_{nl}^c(r) \) will be evaluated using its own spherical grid centered at each point \( r \). This ‘grid for each grid point’ approach is shown in Figure 4.1.

The quadrature scheme for calculating \( \epsilon_{nl}^c(r) \) will be similar to that used on the main FHI-aims grid. The integral is divided into radial and angular components, and

\[ \varphi(r, r') n(r') dr' \]
the Lebedev quadrature is again used for the angular component. The radial integral is now performed with the Gauss-Chebyschev quadrature of the second kind\(^7\), which consists of the following approximation:

\[
\int_{-1}^{1} g(x) \sqrt{1 - x^2} dx \approx \sum_{i=1}^{n} w_i g(x_i) \tag{4.17}
\]

with the weights

\[
w_i = \frac{\pi}{n+1} \sin^2 \left( \frac{i}{n+1} \pi \right) \tag{4.18}
\]

and abscissas

\[
x_i = \cos \left( \frac{i}{n+1} \pi \right). \tag{4.19}
\]

Here \(n\) is a chosen number of points.

Equation (4.16) contains a radial integral over the interval \(r \in [0, \infty]\). However, the integral achieves the form of Equation (4.17) with integration limits \([-1,1]\) after the substitution

\[
r = r_m \sqrt{1 + x \over 1 - x}. \tag{4.20}
\]

Here \(r_m\) is a scaling parameter used to adapt the radial integration grid to the behaviour of the kernel function. Setting \(r_m = 1/q_0(r)\), a majority (about 70\%) of grid points are located within the region \((D < 2\) in Fig. 3.1\) which has the greatest contributions to \(\epsilon_c^{nl}(r)\) [85].

Writing Equation (4.16) in spherical coordinates \((r = r(r, \theta, \phi))\) and inserting Equation (4.20), we obtain an expression containing the radial quadrature and, finally, the combined radial and Lebedev quadratures:

\[
\epsilon_c^{nl}(r') = \frac{1}{2} \int_{0}^{2\pi} \int_{0}^{\pi} \int_{-1}^{1} \varphi(r', r)n(r)r^2 \sin(\theta) d\phi d\theta dr
\]

\[
= \frac{1}{2} \int_{0}^{2\pi} \int_{0}^{\pi} \int_{-1}^{1+1} \varphi(r', r)n(r)r^3_{m} \left(1 - x^2 \right)^{3/2} \sin(\theta) d\phi d\theta dx
\]

\[
\approx \frac{1}{2} \int_{0}^{2\pi} \int_{0}^{\pi} \sum_{i=1}^{N_{rad}} \sum_{j=1}^{N_{ang}} \omega_{rad}(i) \varphi(r', r)n(r)r^3_{m} \left(1 - x^2 \right)^{3/2} \sin(\theta) d\phi d\theta
\]

\[
\approx 2\pi \sum_{i=1}^{N_{rad}} \sum_{j=1}^{N_{ang}} \omega_{rad}(i) \omega_{ang}(j) \varphi(r', r)n(r)r^3_{m} \left(1 - x^2 \right)^{3/2}, \tag{4.21}
\]

where \(r = r(r(x_i), \theta_j, \phi_j)\). For convenience we have interchanged the variables \((r \rightarrow r', r' \rightarrow r)\).

Similarly, the integral in derivatives \(\frac{\partial \epsilon_c^{nl}}{\partial n}\) and \(\frac{\partial \epsilon_c^{nl}}{\partial (\nabla n)^2}\) from Equations (4.6) and (4.7) becomes

---

\(^7\) See Equation 25.4.40 from Ref. [96]
\[ I(r') = \int n(r) \left( \frac{\partial \varphi}{\partial D} |r' - r| + \frac{\partial \varphi}{\partial \Delta} \left( \frac{4q_0(r)}{(q_0(r') + q_0(r))^2} \right) \right) \, dr \]

\[ \approx 4\pi \sum_{i=1}^{N_{\text{rad}}} \sum_{j=1}^{N_{\text{ang}}} w_{\text{rad}}(i) w_{\text{ang}}(j) \times \]

\[ n(r) \left( \frac{\partial \varphi}{\partial D} |r' - r| + \frac{\partial \varphi}{\partial \Delta} \left( \frac{4q_0(r)}{(q_0(r') + q_0(r))^2} \right) \right) \frac{r^3_m}{(1 - x_i)^n}, \]

with \( r = r(x_i, \theta_j, \phi_j) \). This can be evaluated on the same grid as \( \epsilon_{c}^{nl}(r') \).

Equations (4.21) and (4.22) are easily implemented into FHI-aims using the existing Lebedev grids and weights \( w_{\text{ang}}(j) \). Radial abscissas and weights \( w_{\text{rad}}(i) \) are calculated from Equations (4.18) and (4.19).

### Adaptive Grid Algorithm

Some considerations are necessary in choosing the number of grid points \( N_{\text{rad}} \) and \( N_{\text{ang}} \) used to calculate Equations (4.21) and (4.22). The total expense of the integrals depends linearly on these settings.

Already, the decision to use a secondary grid to calculate \( \epsilon_{c}^{nl}(r) \) leads in principle to faster performance than previous implementations which used a single Cartesian grid. These required a sum over all grid points in order to calculate each \( \epsilon_{c}^{nl}(r) \). Formally, \( \epsilon_{c}^{nl}(r) \) must itself be calculated for all grid points, leading such a scheme to scale as \( N^2 \) with the number of atoms (or size of grid). By introducing a secondary grid for \( \epsilon_{c}^{nl}(r) \), the size of which is determined independently, the cost of each \( \epsilon_{c}^{nl}(r) \) is in principle fixed and scaling is linear with respect to \( N_{\text{atoms}} \). This is one significant advantage of the 'grid for each grid point’ approach presented in Gulans et al. [85].

Additional speed can be acquired by adapting the size of the secondary grid through \( N_{\text{rad}} \) and \( N_{\text{ang}} \). In regions of low density \( n(r) \), the contribution of \( \epsilon_{c}^{nl}(r) \) to \( E_{c}^{nl} \) is minimal. This is apparent from Equation (4.15). For low values of \( n(r) \) the accuracy needed for \( \epsilon_{c}^{nl}(r) \) is hence less than for regions of high density. This motivates adaptation of the grid size based on the magnitude of \( n(r) \). Following Gulans et al. [85], we select the number of radial grid points as

\[ N_{\text{rad}} = N_{\text{max}} + n_f \ln(n(r)) \]  

(4.23)

where \( n_f = 4 \) and \( N_{\text{max}} = 40 \). \( N_{\text{rad}} \) is however set to a minimum of 10 or a maximum of 40 points in the code if Equation (4.23) returns a result outside this range. The effect here is that \( N_{\text{rad}} \) varies according to density between values of 10 and 40 points. As will be demonstrated below, this practice significantly reduces expense while maintaining accuracy close to that achieved by a fixed value of \( N_{\text{rad}} \). A similar adaptation of the angular grid was not implemented as significant time savings were not anticipated. Selection of a converged value of \( N_{\text{ang}} \) is also described below.
4.2. ALGORITHM FOR SELF-CONSISTENT POTENTIAL

In summary, Equations (4.21) and (4.22) establish an efficient integration procedure required to solve the nonlocal correlation potential $V^n_c(r)$ on all points $r$ of the FHI-aims grid. The equations require as input values of $n(r')$, $q_0(r')$ and the kernel and its derivatives at points $r'$, which do not correspond with the FHI-aims grid points $r$. This will present some challenges, as detailed below.

4.2.2 Kernel Function

The kernel function $\varphi(r, r')$ determines the magnitude of interaction between two regions of density. This is originally defined in Ref. [37] as

$$\varphi(r, r') = \frac{2}{\pi^2} \int_0^\infty \int_0^\infty b^2 W(a, b) T(\nu(a), \nu(b), \nu'(a), \nu'(b)) a^2 da db$$  \hspace{1cm} (4.24)

where

$$T(w, x, y, z) = \frac{1}{2} \left[ \frac{1}{w + x} + \frac{1}{y + z} \right] \left[ \frac{1}{(w + y) (x + z)} + \frac{1}{(w + z)(y + x)} \right]$$ \hspace{1cm} (4.25)

and

$$W(a, b) = 2[(3 - a^2)b \cos b \sin a + (3 - b^2)a \cos a \sin b + (a^2 + b^2 - 3) \sin a \sin b - 3ab \cos a \cos b]/a^3 b^3.$$ \hspace{1cm} (4.26)

Here, $\nu(y) = y^2/2h(y/d)$ and $\nu'(y) = y^2/2h(y/d')$ with $d = |r - r'|q_0(r)$ and $d' = |r - r'|q_0(r')$. Also, $h(y) = 1 - e^{-\frac{4\pi}{y^2}}$. Function $q_0(r)$ is a characteristic wavenumber defined in terms of the gradient-corrected LDA energy per electron [37],

$$q_0(r) = \frac{4\pi}{3} \left[ \epsilon_{xc}^{LDA} - \epsilon_{x}^{LDA} \left( \frac{Z_{ab}}{2|n|} \right) \right],$$ \hspace{1cm} (4.27)

where $k_F^2 = 3\pi^2 n$ and $Z_{ab} = -0.8491$. Expressions for energies $\epsilon_{xc}^{LDA}$ and $\epsilon_{x}^{LDA}$ are given in Appendix A.

The kernel written as Equation (4.24) is expensive to evaluate due to its heavy double integral. Following previous implementations [37, 80, 85], values of $\varphi(r, r')$ can instead be evaluated and stored on a table in terms of the scaling variables $D$ and $\Delta$ (Eqs. (3.42) and (3.43)). Values of $\varphi(D, \Delta)$ can then be calculated from the table using bivariate cubic spline interpolation [85], which is much faster than using Equation (4.24) directly. Following Gulans et al. [85], derivatives $\frac{\partial \varphi}{\partial D}$ and $\frac{\partial \varphi}{\partial \Delta}$ are also readily evaluated from the same table.

The derivatives $\frac{\partial q_0(r)}{\partial n(r)}$ and $\frac{\partial q_0(r)}{\partial \nabla n(r)^2}$, also required for the potential $V^n_c(r)$, can be evaluated directly from Equation (4.27); expressions used in the code are given in Appendix A.

We see from Equation (4.27) that the kernel requires $q_0(r')$ as input, which is dependent on $n(r')$ and $\nabla n(r')$. These variables are defined only on the FHI-aims grid points $r$, and some scheme is hence necessary to solve for them at general points $r'$. Due to the spherical nature of the FHI-aims grid, direct interpolation from, for example, neighboring grid points is impractical. Various alternatives are considered below.
4.2. ALGORITHM FOR SELF-CONSISTENT POTENTIAL

4.2.3 Octree method for all-electron density interpolation

Multipole expansion of density

The most direct method to solve for the electron density within FHI-aims would be to simply sum up Kohn-Sham eigenfunctions according to the definition of the density,

$$ n(r) = \sum_{k=1}^{N} |\psi_k(r)|^2; \quad (4.28) $$

where $N$ is again the number of electrons. This approach was not used for the vdW-DF as it was assumed to be too computationally demanding. An alternative is to use an existing multipole expansion which is used to determine the Hartree potential [38, 93]. The multipole expansion is constructed as follows.

Spherical free-atom contributions to the all-electron density are subtracted, yielding a difference density\(^8\)

$$ \delta n(r) = n(r) - \sum_{at} n_{at}^{free}(|r - R_{at}|). \quad (4.29) $$

The difference density is then written in terms of atom-centered multipole components

$$ \delta n_{at,lm}(r) = \int_{r=|r - R_{at}|} p_{at}(r) \cdot \delta n(r) \cdot Y_{lm}(\Omega_{at}) d\Omega_{at}, \quad (4.30) $$

where the highest order of the expansion, $l_{max}$, is specified by the user. Here the partition functions $p_{at}(r)$ are the same as described above for integration. All electron density can now be calculated at any point $r$ from the spherical free-atom and multipole components of $\delta n(r)$:

$$ n_{MP}(r) = \sum_{at} n_{at}^{free}(|r - R_{at}|) + \sum_{at,lm} \delta n_{at,lm}(|r - R_{at}|)Y_{lm}(\Omega_{at}). \quad (4.31) $$

The above procedure would allow for solution of density and, through similar considerations, its gradient, at any point up to an adjustable cutoff radius. However, when implemented in the integral for nonlocal correlation, the method proved too heavy for practical calculations. This is demonstrated in Section 4.3.1. Density must be solved at a very large number of points: for example the Ar atom with FHI-aims light grid settings has approx. $1 \times 10^3$ points while the secondary grid has approx. $4 \times 10^3$ points, leading to $\sim 4 \times 10^7$ evaluations of density in total. Interpolation becomes an attractive option as its expense per point would be considerably less than the summation procedures described here.

Interpolation of all-electron density

Solving density and its gradient through interpolation is desirable due to speed considerations. However, in the spherical geometry of the FHI-aims grid there is no straightforward rule for interpolating between grid points where density values are known.

---

\(^8\) See similar expressions (31) to (36) in Ref. [38].
Gulans et al. [85] were able to interpolate from the primary grid of SIESTA as it is Cartesian, for which rules are readily available. This motivates the option of constructing a Cartesian grid in FHI-aims for the purpose of interpolation. Values of density and its gradient could be solved for the nodes using the multipole routines described above, and values of density at given points could then be interpolated from the grid\footnote{Other options certainly exist, such as evaluating the free atom density as in Equation (4.31) and interpolating $\delta n(r)$ from a grid. After brief trials, however, such approaches did not appear as promising as simply using a single grid to interpolate the full density.}. 

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{density_comparision.png}
\caption{Comparison of radial all-electron and pseudo-densities.}
\end{figure}

The use of all-electron density presents a problem in that uniform Cartesian grids, as mentioned earlier, poorly represent density close to atom centers. The reason is made clear in Figure 4.2 where we see that for the spherically-symmetric Argon atom, for example, density peaks sharply in the core region. In comparison, the density used in a pseudopotential program (SIESTA) [86] remains relatively smooth in the core region, lending itself readily to approximation on a uniform grid. As a solution we propose constructing an adaptive Cartesian grid for which interpolation is uniformly accurate at all locations. Such an algorithm will need to build a dense grid in the core region, while less grid points are necessary for regions of less rapidly-varying density.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{octree.png}
\caption{A simple octree.}
\end{figure}
where a master cube is divided recursively into eight children. While this data structure
has had applications in computer graphics and, for example, the representation of
many-electron wavefunctions [97], it has not to our knowledge been applied previously
to the electron density. Each node (child) will store values of the density and gradient
each of its eight vertices, as solved accurately through summation of multipoles.
Density and its gradient can then be solved within a cube using 3D cubic Hermite
spline interpolation. This involves representing density within a node by a polynomial
\( f(x, y, z) \), whose coefficients are solved from the density and gradient values stored
on the node vertices. The derivatives of the polynomial in turn yield an interpolated
gradient. The interpolation polynomial and method is described in detail in Appendix
B.

Construction of the octree grid proceeds as follows. First a master cube is defined
such that it surrounds the system, with at least 6 Bohr of space between atoms and
cube faces. The density and gradient are evaluated at the cube vertices and stored. The
master cube is subsequently divided into eight children, and density and its gradient
are calculated at their vertices also. The process is then repeated until all cubes in the
system are equal to or below a maximum permissible size (1 Bohr by default). At this
point all cubes are of the same size and the grid is uniform, such as in Figure 4.4a. A
decision must then be made whether to divide the nodes further.

As the tree is to be used for interpolation of density, the accuracy of interpolation is a
logical criteria for further division of the nodes. Originally, the accuracy of interpolation
was tested by comparing interpolated density at, for example, the cube center to the
‘exact’ multipole value. If error was above a certain threshold, the node was divided.
A grid generated in this ‘interpolation-based’ manner is shown in Figure 4.5 for an Ar
dimer. Upon close inspection of the core region, we see a number of ‘bubbles’ or large
undivided nodes. These are artifacts of the division condition - in these nodes the center
point was interpolated accurately, likely due to the convenient position of the nodes
along the diagonal from the origin. However, while the center point was interpolated
accurately, interpolation in other regions of the node is not likely to be as accurate.
Nearby nodes, we see, have been divided many times further. The presence of these
oversize nodes in the final grid is hence undesirable and considered a likely source of
error.
Figure 4.5: 'Interpolation-based' octree for an Ar dimer, with one atom at origin, containing approx. 500K nodes. Holes are visible in the core region.

Figure 4.6: Illustration of ‘charge-based’ octree division condition: Charge inside a node is calculated from density at 8 vertices (a), 8 vertices and center point (b) and through integration of the interpolation polynomial (c). The node is divided if the result of either (b) or (c) differs from that of (a) by more than a set threshold ε.

Numerous other division conditions were tested, each based on the accuracy of interpolation at multiple points within the node. Each method lead to similar undesirable artifacts, however. The condition finally chosen is illustrated in Figure 4.6. Charge within the node is calculated three different ways: from density values on the 8 vertex points (Q1), from the weighted average of density on the vertices and at the cube center as given by multipoles (Q2), and finally by integrating the interpolation polynomial within the cube\textsuperscript{10} (Q3). If either charge Q2 or Q3 differs from Q1 by more than a set threshold, designated ε, the node is divided. A tree generated in this ‘charge-based’ manner is shown in Figure 4.7. We see an improved behavior when compared to the ‘interpolation-based’ tree. The grid density is more smoothly distributed in the angular direction and the number of oversize nodes has been greatly reduced.

\textsuperscript{10} described in Appendix B.
4.2 ALGORITHM FOR SELF-CONSISTENT POTENTIAL

Figure 4.7: 'Charge-based' octree for an Ar dimer, with one atom at origin. The core region shows greater uniformity than with the interpolation-based tree of Figure 4.5. Shown with approx. 500K nodes ($\epsilon=10^{-6}$).

Figure 4.8: Charge density interpolated from a 'charge-based' octree is compared to multipole values, for Ar atom. Density is evaluated at multiple points on 60 radial shells (described in text). Maximum absolute and relative error of interpolated density is shown for each radius.

The accuracy of interpolation from a (charge-based) octree is demonstrated in Figure 4.8 for an Argon atom. For 60 values of radius $r$, density was evaluated at 86 points on a Lebedev grid. This allowed for testing interpolation in many directions from the origin. The density curve at the top of Figure 4.8 shows general correspondence between interpolated and multipole values. The center plot shows the highest absolute error to occur among the 86 directions, at each radius. The lower plot shows the corresponding maximum relative error. We see that absolute error tends to increase in the core region although relative error here is low. Relative error increases with radius, but remains well below 1% for most radii. The octree used here contains approx. 276K nodes ($\epsilon=10^{-6}$) with 13 levels of divisions, meaning the smallest nodes are approximately 0.001 Å wide. This corresponds to the tree in Figure 4.7 but for one atom. As
we shall see in following sections, the accuracy of interpolation achieved by this setup is sufficient for calculations of nonlocal correlation.

The octrees offer a solution for representing all-electron density on an adaptive grid. The number of nodes on the grid is easily controlled by a single parameter $\epsilon$, which can be set according to the level of accuracy needed. Interpolation of density and its gradient is continuous and smooth within the nodes, but discontinuous across boundaries of nodes of different size.\(^{11}\) In practice, however, this discontinuity appeared to have little effect on accuracy within the integrals of the vdW scheme. The octree is also readily implemented into FHI-aims: it is deallocated and regenerated after each density update, and interpolation from the tree is readily performed within the integrals of the vdW scheme. The octree scheme allows for significant time savings over the rival method of multipoles and is also available for applications other than vdW. The computational performance of the tree, as well as selection of the parameter $\epsilon$ for vdW calculations will be discussed below.

## 4.3 Testing and Verification

The above integration and octree interpolation schemes contain numerical parameters which are selected below. All tests shown are performed with the pbe-vdw functional.

### 4.3.1 Octree settings

The octree parameter $\epsilon$ was calibrated to achieve converged results for the interaction energy of two systems: Xe$_2$ and the stacked Benzene dimer. As seen in Figure 4.9, adjustment of $\epsilon$ affects the density of the grid in atom core regions. The total potential energy curve (PEC) of Xe$_2$ is shown in Figure 4.10, including nonlocal correlation calculated\(^{12}\) with the value $\epsilon=10^{-6}$. The PEC has the anticipated smooth form with a clear binding minimum. Figure 4.11 shows a close-up of the energy minimum area of the PEC, as calculated with three values of $\epsilon$ and using multipoles. The curves calculated with octrees are clearly non-smooth, however they do converge to the accurate multipole curve. We see that the curve of $\epsilon=10^{-6}$ differs from the multipole curve by less than 0.1 meV, which is our desired level of accuracy. This corresponds to a tree with approximately 1 million nodes. A curve with considerably less nodes ($\epsilon=2.5\times10^{-6}$, 649K nodes) also achieves similar accuracy.

\(^{11}\)Continuity of density over node boundaries can be achieved by storing interpolated values on the nodes instead of multipole values. This method also leads to trees with undesirable artifacts, however.

\(^{12}\)All calculations also used the default integration settings described in Section 4.3.2.
4.3. TESTING AND VERIFICATION

![Graph of interaction energy vs. separation for Xe₂](image)

**Figure 4.9**: Octrees for Xe₂ with approximately (a) 433K and (b) 1 million nodes ($\epsilon=5\cdot10^{-6}$ and $\epsilon=1\cdot10^{-6}$, respectively). Shown at 4.36 Å separation, cross-section through z=0.

**Figure 4.10**: Total potential energy curve of Xe₂, calculated with octree ($\epsilon=10^{-6}$).
Figure 4.11: Close-up of Xe$_2$ energy curve for three different octrees and multipoles. Tree parameters are $\epsilon=10^{-6}$ (1 mil. nodes), $\epsilon=2.5\cdot10^{-6}$ (649K nodes) and $\epsilon=5\cdot10^{-6}$ (433K nodes).

Figure 4.12: Octree for stacked Benzene dimer with 1.48 million nodes ($\epsilon=10^{-6}$), shown as a cross-section through one of the two Benzene molecules. Ball-and-stick model of Benzene shown for comparison.

The octree of a stacked Benzene dimer is shown in Figure 4.12 as a cross-section through one of the Benzene molecules. The form of the molecule (shown also as a
ball-and-stick model) is seen clearly in the octree. The total potential energy curve calculated with this tree is shown in Figure 4.13 and has again a smooth form with a bond length of approx. 3.84 Å. Figure 4.14 shows a close-up of the total energy curves, as calculated with multipoles and two different octrees. Total energies, not interaction energies, are shown as the differences between octree and multipole curves was more pronounced for the former. Interaction energy requires subtraction of the individual molecule energies from the total energy, which in this case reduced the visible error of the octree curves. We see in Figure 4.14 that the value $\epsilon=10^{-6}$ leads to a curve which stays within 0.1 meV of the multipole curve. This requires a tree with approximately 1.48 million nodes.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{fig4.13.png}
\caption{Interaction energy curve for stacked Benzene dimer, calculated with octree ($\epsilon=10^{-6}$).}
\end{figure}
Figure 4.14: Close-up of stacked Benzene dimer total energy curves, for two octrees and multipoles. Tree parameters are $\epsilon=10^{-6}$ (1.48 mil. nodes) and $\epsilon=5\cdot10^{-6}$ (487K nodes).

The value $\epsilon=10^{-6}$ is hence chosen as a default setting as it leads to interaction and total energy values which are within 0.1 meV of the values calculated with multipoles. We see in the case of Xe$_2$, however, that a smaller tree may also suffice for accurate calculations. The parameter $\epsilon$ can hence be further optimized by users according to their accuracy or memory usage requirements.

The run times for the above systems demonstrate the advantage of the octree method over multipoles. For Xe$_2$, typical run time (serial runs on linux workstations) for a single scf step was $\sim$320 seconds for multipoles and $\sim$180 seconds using the octree with $\epsilon=10^{-6}$. For the Benzene dimer, a single scf step takes $\sim$18000 seconds with multipoles and only $\sim$1000 seconds with the octree ($\epsilon=10^{-6}$). We see that as system size grows, use of multipoles becomes impractically slow while use of octrees is up to 18 times faster. Further discussion of run times is provided in Section 5.3.

4.3.2 Integration grid settings

The number of radial and angular grid points to use for the inner integrals was determined by studying the convergence of $E^i_{nl}$ or a system’s interaction energy with respect to the number of grid points. This is illustrated in Figures 4.15 and 4.16 and described below.
Figure 4.15: Nonlocal correlation energy vs. number of radial grid points, $n_{rad}$, for Argon atom, after 1 scf (self-consistent-field) step. Here $i_{leb}=7$, $\epsilon=10^{-6}$.

Using the Lebedev grid $i_{leb}=7$, with 86 grid points, the nonlocal correlation energy of an Argon atom was calculated for 40 values of the parameter $n_{rad}$. We see in Figure 4.15 that convergence (within 0.1 meV accuracy) is achieved roughly beyond $n_{rad}=20$. This suggests that use of the adaptive radial grid approach [85], which uses values between $n_{rad}=10$ and $n_{rad}=40$, is justified. Table 4.1 compares the adaptive radial grid to results run with a fixed value of $n_{rad}=40$. We see that $E_{nl}$ values after 1 scf step\textsuperscript{13} are within 0.1 meV of each other for an Argon atom and within 7 meV for a Benzene dimer, while run time\textsuperscript{14} is about 30% less for the adaptive grid. The interaction energies of the Benzene dimer are however expected to be more closely converged for the adaptive grid as the subtraction of free-molecule total energy cancels out some error, as described earlier. The adaptive grid method is thoroughly tested in Ref. [85] and hence we adopt it here as a default in the interest of reducing run times. Users may however set $n_{rad}$ to fixed values also to confirm converged results.

Table 4.1: Fixed versus adaptive radial grid: run times and $E_{nl}^{\text{cl}}$ results compared for Ar atom and Benzene dimer, after 1 scf step.

<table>
<thead>
<tr>
<th></th>
<th>Time 1 scf</th>
<th>$E_{nl}^{\text{cl}}$ [eV]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ar atom</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fixed ($n_{rad}=40$)</td>
<td>85</td>
<td>5.0567</td>
</tr>
<tr>
<td>adaptive</td>
<td>60</td>
<td>5.0566</td>
</tr>
<tr>
<td><strong>Benzene dimer</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fixed ($n_{rad}=40$)</td>
<td>1640</td>
<td>20.7265</td>
</tr>
<tr>
<td>adaptive</td>
<td>1170</td>
<td>20.7329</td>
</tr>
</tbody>
</table>

\textsuperscript{13}While the scf cycle is not converged, we assume that energies given by the adaptive and fixed radial grids can be compared for numerical purposes already after 1 scf step.

\textsuperscript{14}Single-processor runs, described in Section 5.3.
Selection of the index of the Lebedev grid to be used poses some challenge. In Figure 4.16 we see that convergence is reached only at $i_{leb}=12$ or higher for the $E_{c^{nl}}$ of an Argon atom. Significant fluctuations in the nonlocal correlation energy are seen before this point, with a range of about 5 meV. The $i_{leb}=12$ and 13 grids were not functional in FHI-aims and are automatically replaced by $i_{leb}=14$ grids, which have 302 points. In comparison, the $i_{leb}=7$ grid has 86 points. The nearly 4 times difference in the number of points would translate into nearly 4 times greater run times. This motivates the use of a value of $i_{leb}$ which is below 12, for the sake of time savings, even if it does not appear converged. While $i_{leb}=7$ does not appear converged in Figure 4.16 for $E_{c^{nl}}$ alone (nor did it in plots of total energy), interaction energies of an Ar dimer differed less than 0.01 meV when $i_{leb}=7$ was increased to $i_{leb}=12$. As described above, this may be due to the cancellation of error when free atom energies are subtracted from the dimer total energy. In any case, the choice of $i_{leb}=7$ was not expected to lead to significant loss of accuracy for interaction energies of noble gases or S22 complexes. As seen in Section 5.2, S22 complexes were calculated with good accuracy using $i_{leb}=7$. We therefore select $i_{leb}=7$ as a default, but Figure 4.16 should remind users that this parameter may require adjustment to guarantee converged total energies.

### 4.3.3 Confirming self-consistent Energy and Forces

After selecting default octree and integration grid settings, it was necessary to confirm that the self-consistent atomic forces (as calculated by the Hellmann-Feynman theorem)

---

15Also the $i_{leb}=6$ was replaced by the $i_{leb}=7$ grid.
were being properly updated with the new nonlocal correlation component. To test that forces were consistent with the interaction energy, the interaction energy curve of an Argon dimer was calculated (using default grid settings) and plotted with the force experienced by an atom in the dimer. This is shown in Figure 4.17. We see that the force correctly changes signs (from repulsive to attractive) as atomic separation approaches the energy minimum at approximately 3.65 Å. Also a quick check using finite differences confirms that the forces are indeed consistent with the self-consistent energy curve.

![Figure 4.17: Self-consistent interaction energy and force curve for Ar dimer (pbe-vdw). Point of zero force is correctly aligned with energy minimum.](image-url)
Chapter 5

Benchmark Calculations

5.1 Noble Gas Dimers

The default settings described above were applied to calculate the interaction energy curves of three noble gas dimers using the new pbe-vdw and revpbe-vdw functionals. In addition, results using the standard PBE and revPBE functionals were calculated for comparison. ‘Tight’ settings were used for the FHI-aims atom-centered integration grids. The potential energy curves are compared, where possible, to experimental values for the bond length and energy. The results are shown in Figures 5.1 through 5.3.

![Interaction energy curves](image)

**Figure 5.1:** Interaction energy curves of Ar dimer vs. experimental result [52, 80].
As expected, we see significant improvement in the form of all three PECs. The PBE and revPBE PECs are weakly binding or nonbinding, respectively, and both pbe-vdw and revpbe-vdw show clear binding minima. Results are summarized in Table 5.1. While the bond energies are overestimated by as much as a factor of two, bond lengths for pbe-vdw are quite accurate. The results are similar to those achieved by other implementations of the vdw-DF [80].
5.2 S22 Benchmark

A set of 22 biologically-relevant, small molecular complexes, known as 'S22', has been calculated using the highly-accurate coupled cluster method CCSD(T) [67]. These results have become a common benchmark for new DFT methods, particularly for vdW functional implementations [85]. Listed in Table 5.2, complexes 1-7 are hydrogen bonded, 8-15 contain predominant dispersion interactions and remaining complexes are of mixed bond character. Each system is a duplex, or dimer of molecules such as that shown in Figure 5.4, which can easily be separated into two free molecules. The interaction energies are calculated as the difference between the total energy of the duplexes and the total energies of the free molecules.

![Adenine-thymine complex](image)

**Table 5.1:** Interaction energy and bond length results for noble gases. Experimental bond length values from [52] and energy values from [80] (no value available for Xe₂ bond energy).

<table>
<thead>
<tr>
<th>System</th>
<th>Bond length [Å]</th>
<th>Interaction energy [meV]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>expt.</td>
<td>pbe-vdw</td>
</tr>
<tr>
<td>Ar₂</td>
<td>3.76</td>
<td>3.65</td>
</tr>
<tr>
<td>Kr₂</td>
<td>4.01</td>
<td>4.01</td>
</tr>
<tr>
<td>Xe₂</td>
<td>4.36</td>
<td>4.36</td>
</tr>
</tbody>
</table>

Here, relaxed geometries of the complexes, provided by the reference paper [67], were used as inputs for calculations of interaction energy with FHI-aims. No further relaxations were performed, rather the interaction energy was calculated for each duplex frozen in its original geometry. Calculations were performed using standard PBE and the new functionals pbe-vdw and revpbe-vdw. Again, 'Tight' settings were used for the FHI-aims atom-centered grids. The results of this set-up are shown in Table 5.2 and in Figure 5.5.
Table 5.2: Interaction Energies (meV) of S22 complexes (negative results nonbinding).

<table>
<thead>
<tr>
<th>Complex</th>
<th>pbe</th>
<th>revpbe-vdw</th>
<th>pbe-vdw</th>
<th>CCSD(T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (NH₃)₂</td>
<td>124</td>
<td>107</td>
<td>164</td>
<td>137</td>
</tr>
<tr>
<td>2 (H₂O)₂</td>
<td>218</td>
<td>176</td>
<td>243</td>
<td>218</td>
</tr>
<tr>
<td>3 Formic acid dimer</td>
<td>798</td>
<td>666</td>
<td>845</td>
<td>807</td>
</tr>
<tr>
<td>4 Formamide dimer</td>
<td>646</td>
<td>564</td>
<td>720</td>
<td>692</td>
</tr>
<tr>
<td>5 Uracil dimer</td>
<td>811</td>
<td>753</td>
<td>921</td>
<td>895</td>
</tr>
<tr>
<td>6 2-pyridoxine-2-aminopyridine</td>
<td>673</td>
<td>624</td>
<td>790</td>
<td>725</td>
</tr>
<tr>
<td>7 Adenine-thymine</td>
<td>629</td>
<td>594</td>
<td>770</td>
<td>710</td>
</tr>
<tr>
<td>8 (CH₄)₂</td>
<td>15</td>
<td>46</td>
<td>107</td>
<td>65</td>
</tr>
<tr>
<td>9 (C₂H₄)₂</td>
<td>4</td>
<td>48</td>
<td>106</td>
<td>65</td>
</tr>
<tr>
<td>10 Benzene-CH₄</td>
<td>-79</td>
<td>76</td>
<td>201</td>
<td>118</td>
</tr>
<tr>
<td>11 Benzene dimer (slip-parallel)</td>
<td>-26</td>
<td>124</td>
<td>263</td>
<td>192</td>
</tr>
<tr>
<td>12 Pyrazine dimer</td>
<td>124</td>
<td>351</td>
<td>550</td>
<td>439</td>
</tr>
<tr>
<td>13 Uracil dimer (stack)</td>
<td>-92</td>
<td>118</td>
<td>294</td>
<td>226</td>
</tr>
<tr>
<td>14 Indole-benzene (stack)</td>
<td>67</td>
<td>364</td>
<td>636</td>
<td>530</td>
</tr>
<tr>
<td>15 Adenine-thymine (stack)</td>
<td>51</td>
<td>64</td>
<td>97</td>
<td>66</td>
</tr>
<tr>
<td>16 Ethene-ethine</td>
<td>93</td>
<td>114</td>
<td>180</td>
<td>142</td>
</tr>
<tr>
<td>17 Benzene-H₂O</td>
<td>43</td>
<td>80</td>
<td>140</td>
<td>102</td>
</tr>
<tr>
<td>18 Benzene-NH₃</td>
<td>125</td>
<td>143</td>
<td>222</td>
<td>193</td>
</tr>
<tr>
<td>19 Benzene-CHN</td>
<td>8</td>
<td>87</td>
<td>169</td>
<td>119</td>
</tr>
<tr>
<td>20 Benzene dimer (T-shape)</td>
<td>93</td>
<td>181</td>
<td>295</td>
<td>248</td>
</tr>
<tr>
<td>21 Indole-benzene (T-shape)</td>
<td>172</td>
<td>234</td>
<td>361</td>
<td>306</td>
</tr>
</tbody>
</table>

Figure 5.5: Pbe-vdw and revpbe-vdw binding energy results for S22 set, compared to CCSD(T) reference.
We see in Table 5.2 that both pbe-vdw and revpbe-vdw greatly improve over PBE results. In particular, complexes 10, 12 and 14, which were predicted to be nonbinding by PBE are now bound properly. The adenine-thymine stack (system 15) was predicted by PBE to be weakly bound with an energy of 93 meV, and both revpbe-vdw (364 meV) and pbe-vdw (636 meV) results are much closer to the accurate CCSD(T) value of 530 meV. From Figure 5.5 we see that in general pbe-vdw and revpbe-vdw tend to over and underestimate bond strength, respectively. While the error compared to CCSD(T) results is significant (average error of 51 meV for pbe-vdw and 67 meV for revpbe-vdw), the improvement over standard PBE is as expected and the results are again similar to those achieved in for example the implementation of Gulans et al. [85]. The results therefore indicate that the Langreth-Lundqvist vdW density functional has been implemented correctly. While the results above were calculated in a self-consistent manner, it is also possible to calculate the nonlocal correlation energy non-self-consistently, for example after a converged PBE calculation. This requires the expensive vdW functional be evaluated only once instead of in each SCF iteration, leading to significant time savings. Thonhauser et al. [80] demonstrate that interaction energies calculated with vdW-DF in both self-consistent and non-self-consistent (non-SC) form differ insignificantly from each other, justifying the use of a non-SC implementation due to its increased speed. In Figure 5.6 we show the results of non-SC calculations with FHI-aims compared to the self-consistent calculations above. Here, PBE was used to calculate the converged self-consistent density of the systems, and the vdW-DF was engaged lastly to calculate the nonlocal correlation energy. As seen in Figure 5.6, the two methods of calculation produce nearly identical results, with an average difference of about 2 meV for the S22 set. This suggests that also the current implementation in FHI-aims could be used accurately in the faster non-SC form.

**Figure 5.6:** Non-self-consistent compared to self-consistent results for S22 set with pbe-vdw.
5.3 Computational Performance

The vdW-DF implementation performed with the anticipated accuracy in the cases of noble gas dimers and the S22 set. We now turn to the computational performance of the implementation. Many important biological systems are much larger than those present in the S22 set, and in the interests of further development it is necessary to understand both the time and memory usage of this implementation.

5.3.1 Scaling

The scaling of the vdW scheme with respect to the number of atoms is expected to be nearly linear, as the integration scheme itself is linear as demonstrated in section 4.2.1. The need to search the octree before performing interpolation adds some expense to the inner integral, however. The time to search the octree depends on the number of levels present, which increases by one, roughly speaking, each time a system’s length (or largest dimension) doubles. Assuming that a system’s length is proportional to the number of atoms $N$, we have that $N_{\text{levels}} \sim \log_2(N)$. The full scaling could then be approximated as $N \log(N)$, where $N$ is the number of atoms.

![Graph showing time 1 scf with pbe-vdw](image)

**Figure 5.7:** Time for 1 scf step with pbe-vdw, showing roughly $N \log(N)$ scaling with number of atoms ($a$ and $b$ are fitting parameters). The system is a (fictitious) linear string of $n$ carbon atoms with 2 Å spacing. Run on an Intel Core 2 Duo E6600 processor (2.4 Ghz, 2 Gb RAM), with FHI-aims 'Tight' settings.

In Figure 5.7 we see the time for one self-consistent iteration (pbe-vdw) of a fictitious chain of carbon atoms, where atoms are separated by 2 Ångstroms. A near-linear scaling with respect to the number of atoms is observed, which is consistent with the
5.3. COMPUTATIONAL PERFORMANCE

Reported $O(N)$-like scaling of FHI-aims. Also, the presence of an additional logarithmic term seems likely. For comparison, an $N \log(N)$ curve is fitted to the data with the least-squares method, with the starting point (parameter $a$) fixed. The fit appears to reasonably approximate the run times. Tests with more atoms in the chain were not possible on the computer used due to memory limitations.

5.3.2 Run time

The current implementation of vdW-DF into FHI-aims is significantly slower than a similar implementation by Gulans et al. [85] in the pseudopotential code SIESTA. This is primarily because in SIESTA density can be interpolated directly from the cartesian integration grid, which is much faster than the octree interpolation scheme in FHI-aims. A direct comparison of run times of the two implementations is not performed here, but we analyze the run times of the FHI-aims implementation.

All runs for the S22 set were performed on Linux workstations with varying processor speed. As such the run times cannot be usefully compared to each other. With this arrangement, the total run times for pbe-vdw ranged from 45 minutes for the water dimer (system 2 in Table 5.2) to 10.5 hrs for the Adenine-Thymine complex (system 7). These systems also required the smallest and largest octrees, as discussed below. In contrast, PBE runs took between 30 seconds and 35 minutes. All systems reached convergence after 14 scf iterations or less for all functionals.

To better understand the time usage of the implementation, four systems were run on the same workstation with the following specs: Intel Core 2 Duo E6600 (2.4 Ghz, 2 Gb RAM). Equilibrium geometries were used and the water dimer and adenine-thymine complex are the same as systems 2 and 7 from the S22 set. We see the total run times and time for single scf steps in Table 5.3 for PBE and pbe-vdw. The pbe-vdw functional results in a large increase in run times when compared to PBE. Run times for revpbe-vdw were similar to pbe-vdw and are not shown.

**Table 5.3:** Total run time and time for 1 scf step for select systems: PBE and pbe-vdw cases. Run on an Intel Core 2 Duo E6600 processor (2.4 Ghz, 2 Gb RAM), with FHI-aims 'Tight' settings.

<table>
<thead>
<tr>
<th>System</th>
<th>Total run time [s]</th>
<th>Run time 1 scf [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>pbe</td>
<td>pbe-vdw</td>
</tr>
<tr>
<td>Xe$_2$</td>
<td>10</td>
<td>2030</td>
</tr>
<tr>
<td>(H$_2$O)$_2$</td>
<td>48</td>
<td>3016</td>
</tr>
<tr>
<td>Benzene dimer</td>
<td>681</td>
<td>14700</td>
</tr>
<tr>
<td>Adenine-Thymine</td>
<td>1548</td>
<td>26978</td>
</tr>
</tbody>
</table>

We see from Table 5.3 that the pbe-vdw functional is some 20 times more expensive than PBE for the larger systems. In comparison, the SIESTA implementation itself is up to 10 times more expensive than PBE, depending on the size of the system [91].

An analysis of the time usage in a single scf step for pbe-vdw is given in Table
5.4. The dominant components of the scf step are the generation of the octree and the integration of the Hamiltonian matrix elements, Equation (4.8). This integration is responsible for the largest increase in scf step time as it requires calculation of the nonlocal correlation potential, Equations (4.5) through (4.7). These terms require interpolation of density from the octree many times in their inner integrals, and we see in Table 5.4 that time spent interpolating from the octree is indeed the dominant component of the overall integration time. For the Benzene dimer, for example, the total run time for the scf step was 1172 seconds. 899 seconds were spent integrating the Hamiltonian matrix and of this 660 seconds were spent interpolating from the octree. The remaining time in the integration is due to other components in the $E_{\text{el}}$ integral, such as evaluation of the kernel. Clearly the integrals of the vdW-DF are highly demanding; the corresponding time for 1 scf step using PBE was only 52 seconds. Possibilities for speeding up the integrals are discussed in Section 6.

The time needed to generate the octree is below 20% of the total scf time and is much less than time spent interpolating from the tree. Nevertheless, in an absolute sense the tree generation time is significant for large systems (300 second for Adenine-Thymine, for example) and future efforts could also be made to speed up this process.

Table 5.4: Breakdown of time for 1 scf step with pbe-vdw. Dominant components are the tree build time and the time for integrating the Hamiltonian matrix elements, $t_{\text{int}}$. Interpolation from the octree, $t_{\text{oct}}$, is the largest component of $t_{\text{int}}$. Run on an Intel Core 2 Duo E6600 processor (2.4 Ghz, 2 Gb RAM), with FHI-aims 'Tight' settings.

<table>
<thead>
<tr>
<th>System</th>
<th>Time 1 scf [s]</th>
<th>Tree build time [s]</th>
<th>$t_{\text{int}}$ [s]</th>
<th>$t_{\text{oct}}$ [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Xe}_2$</td>
<td>186</td>
<td>16</td>
<td>169</td>
<td>130</td>
</tr>
<tr>
<td>$(\text{H}_2\text{O})_2$</td>
<td>218</td>
<td>16</td>
<td>200</td>
<td>148</td>
</tr>
<tr>
<td>Benzene dimer</td>
<td>1172</td>
<td>240</td>
<td>899</td>
<td>660</td>
</tr>
<tr>
<td>Adenine-Thymine</td>
<td>1654</td>
<td>300</td>
<td>1293</td>
<td>959</td>
</tr>
</tbody>
</table>

5.3.3 Memory

The octree consists of cubic nodes which each store four double precision variables (density and three gradient components) at eight vertices, as well as a pointer to child nodes. A double precision variable requires 8 bytes of memory, meaning 256 bytes of memory per node are required, plus memory for pointers. Table 5.5 shows that the octrees are in some cases quite massive, certainly the largest data structures in this vdW-DF implementation. Adenine-Thymine (system 7) had the largest octree of the S22 systems, and the FHI-aims run with it required just over 1 GB of RAM. In comparison, the same run with PBE required approximately 540 MB of memory. The memory demand of pbe-vdw runs with large systems now approaches the capacity of workstations having for example only 2 GB of Memory. Reduction of memory usage will be necessary for runs with larger systems. Options for this are discussed below.

---

1It should be noted that interpolation is also the most expensive component of the vdW-DF in SIESTA. The faster method of interpolation used there largely accounts for the greater overall speed of the implementation.

2Tree sizes reported in Table 5.5 include all nodes in the octrees, while the tree sizes given in Figures 4.10 through 4.14 include only lowest-level nodes (i.e. those from which interpolation takes place).
5.3. COMPUTATIONAL PERFORMANCE

Table 5.5: Octree size for selected systems ($\epsilon=10^{-6}$), and approximate total memory usage of corresponding FHI-aims runs with pbe-vdw.

<table>
<thead>
<tr>
<th>System</th>
<th>Tree Size [nodes]</th>
<th>Total RAM used [MB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xe$_2$</td>
<td>$1.2\cdot10^6$</td>
<td>440</td>
</tr>
<tr>
<td>(H$_2$O)$_2$</td>
<td>$4.2\cdot10^5$</td>
<td>240</td>
</tr>
<tr>
<td>Benzene dimer</td>
<td>$1.7\cdot10^6$</td>
<td>760</td>
</tr>
<tr>
<td>Adenine-Thymine</td>
<td>$2.5\cdot10^6$</td>
<td>1080</td>
</tr>
</tbody>
</table>
Chapter 6

Discussion

The current implementation of vdW-DF faced challenges posed by the use of all-electron density in FHI-aims. It was necessary to develop a new method for representing electron density, and the octree method proposed indeed constitutes a major part of this work. While the accuracy achieved by the code for noble gas and S22 systems met expectations, improvements are needed in the computational performance. Run times and memory usage are much higher than for standard PBE, and a reduction in both is necessary for practical runs with systems larger than those in S22. We discuss some development options below.

A number of options exist for increasing the speed of the vdW-DF calculations. The largest increase in run times with vdW-DF is due to the nonlocal correlation integrals, Equations (4.5) through (4.7). To speed up the integrals, routines for all components in the integrands (kernel function, $q_0$ and density interpolation) were in-lined. Even so, however, interpolation of density from the octree takes much time. The interpolation routine proceeds by first searching the octree for the proper node. This involves a 3-dimensional binary search, which takes longer if there are more levels or generations of nodes. The most expensive regions to interpolate are hence the core regions of atoms: for example the Xe$_2$ dimer in Figure 4.7 contains 13 levels of division in the core region. On the other hand, solving for density using multipoles may be considerably cheaper in the core region. Here, due to the shape of the partition function, contributions to the density from neighboring atoms are relatively small and can be neglected, speeding up the multipole summation, Equation (4.31). This suggests the possibility of increasing speed by solving density in the core regions from single atoms' multipoles instead of interpolation. It would not even be necessary to generate the octree in the core region, and this would greatly reduce the total number of nodes. Another possibility for increased speed would be to use lower-order interpolation, for example trilinear. The interpolation polynomial would be quicker to evaluate; however, there would likely be an adverse effect on accuracy.

Speed can also be increased by reducing the size of integration grids throughout the code. Grids for the vdW-DF integrals can be reduced in size depending on the system, but care should be taken to ensure converged results. Finally, the FHI-aims integration grids could be reduced in size from the Tight settings used above. Even
if Tight settings are used for reasons other than vdW-DF, it may suffice to calculate vdW-DF on a lighter grid. In other words the vdW-DF likely does not need to be evaluated on each point of the FHI-aims Tight grid. Further exploration would be needed to determine the optimal size of primary grid for vdW-DF.

The increased memory usage of vdW-DF is due primarily to the size of the octree. The nodes store double precision variables, each with a size of 8 bytes. Changing the data type to, for example, single precision (4 bytes per variable) is one option for reducing the memory consumption of the tree. However, as the rest of FHI-aims uses double precision variables, a change in data type would be necessary after interpolation of single precision density. This added expense would increase run times. The size of the octrees could also be reduced by neglecting the core region, as mentioned above.

Due to improvements necessary to the octree method, the current implementation is not a final product. Rather, the current code serves as a proof-of-concept for the octree method, which to the knowledge of the author has not been previously applied to the representation of electron density. Even in its current form, however, the code is an accurate implementation of the vdW functional capable of simulations of systems with many tens of atoms. A final version of the implementation would require at least two additional features. The current implementation is intended for use with non-periodic geometries. Calculations with periodic geometries would require some further considerations to ensure that the octree contains the entire master unit cell. This was not performed as a part of this work. Furthermore, the current implementation is intended for serial calculations. Parallel calculations are possible with the code though it is not optimized for this purpose: the full octree is generated for each cpu rather than being partitioned, for example. Optimized parallel capability is desired for future versions.
Chapter 7

Summary and Conclusions

In summary, we have implemented the Langreth-Lundqvist vdW functional [37, 85] in self-consistent form into the all-electron model FHI-aims. The use of all-electron density required development of a method to allow accurate interpolation of density. The octree data structure was used to generate adaptive grids which are more dense in regions of rapidly varying density, allowing for accurate interpolation of all-electron density and its gradient. The integration grids of the nonlocal correlation functional, as well as its kernel function, were implemented as in a previous work by Gulans et al. [85].

Converged octree and integration settings were determined for the vdW-DF by analyzing the interaction energy curves of noble gas dimers and the Benzene dimer. The implementation was tested in calculations of the interaction energies of noble gas dimers and the S22 set of biological complexes. Results using both pbe-vdw and revpbe-vdw functionals showed the expected improvement over standard PBE and revPBE, and the accuracy achieved was similar to that of other implementations of vdW-DF [80, 85]. The method shows reasonable computational performance and is capable of serial runs of non-periodic systems with many tens of atoms.

Aside from enabling calculations with vdW-DF in FHI-aims, the development of an octree method for representing all-electron density may be the most significant contribution of this work. While the performance of the octrees could yet be optimized to reduce run times and memory usage, the current implementation demonstrates the potential of the octree method for use with all-electron density. It is anticipated that octrees may also be useful for applications other than vdW-DF, either in the representation of electron density or other variables.

The current implementation is a first step in allowing for studies of dispersion-bonded systems with vdW-DF in FHI-aims. Through comparison with results from pseudopotential-based codes, the current all-electron implementation also offers the rare prospect of studying the influence of core electrons on dispersion interactions. While some further development is needed, the prospects for simulations of important biological systems with vdW-DF in FHI-aims are very good.
Appendix A

Derivatives of \( q_0(\mathbf{r}) \)

Derivatives \( \frac{\partial q_0}{\partial n(\mathbf{r})} \) and \( \frac{\partial q_0}{\partial |\nabla n(\mathbf{r})|^2} \) can be obtained by opening the expression for \( q_0(\mathbf{r}) \) as given in Equation (4.27):

\[
q_0(\mathbf{r}) = -\frac{4\pi}{3} \left[ \epsilon_{xc}^{LDA} - \epsilon_x^{LDA} \left( \frac{Z_{ab}}{\nabla n} \left( \frac{\nabla n}{2k_F n} \right)^2 \right) \right], \tag{A.1}
\]

where \( k_F^3 = 3\pi^2 n \). Expressions for the energy per electron \( \epsilon_{xc}^{LDA} \) are well established:

\[
\epsilon_{xc}^{LDA} = \epsilon_c^{LDA} + \epsilon_x^{LDA}, \tag{A.2}
\]

with the exchange portion [37, 98]

\[
\epsilon_x^{LDA} = -\frac{3k_F}{4\pi} \tag{A.3}
\]

and correlation portion [99]

\[
\epsilon_c^{LDA}(\mathbf{r}) = -2A(1 - \alpha_1 r_s) \ln \left[ 1 + \frac{1}{g(r_s)} \right]. \tag{A.4}
\]

Here,

\[
g(r_s) = 2A(\beta_1 r_s^{1/2} + \beta_2 r_s) + \beta_3 r_s^{3/2} + \beta_4 r_s^2 \tag{A.5}
\]

and \( r_s \) is a so-called density parameter [99],

\[
r_s = \left( \frac{3}{4\pi n(\mathbf{r})} \right)^{1/3}. \tag{A.6}
\]

Following Gulans et al. [85], the parameters used for \( \epsilon_{xc}^{LDA} \) are taken from Perdew and Wang [99]: \( A = c_0 = 0.031091, \quad \alpha_1 = 0.21370, \quad \beta_1 = \frac{1}{2c_0} \exp(-c_1/2c_0), \quad \beta_2 = 2A\beta_1^2, \quad \beta_3 = 1.6382, \quad \beta_4 = 0.49294 \) and \( c_1 = 0.04644 \).

Inserting the above into Equation (A.1) we have for the wavenumber

\[
q_0(\mathbf{r}) = k_F - \frac{4\pi}{3} \epsilon_c^{LDA} - k_F Z_{ab} \left( \frac{\nabla n(\mathbf{r})}{6k_F n(\mathbf{r})} \right)^2. \tag{A.7}
\]
Derivatives with respect to density $n(\mathbf{r})$ and the modulus squared $|\nabla n(\mathbf{r})|^2$ are now obtained directly:

$$\frac{\partial q_0(\mathbf{r})}{\partial n(\mathbf{r})} = \frac{k_F}{3n(\mathbf{r})} - \frac{4\pi}{3} \frac{\partial \epsilon_{c,LDA}}{\partial n(\mathbf{r})} + \frac{7k_F}{3n(\mathbf{r})} \left( Z_{ab} \left( \frac{\nabla n(\mathbf{r})}{6k_F n(\mathbf{r})} \right)^2 \right), \quad \text{(A.8)}$$

where

$$\frac{\partial \epsilon_{c,LDA}}{\partial n(\mathbf{r})} = \frac{\partial \epsilon_{c,LDA}}{\partial r_s} \frac{\partial r_s}{\partial n(\mathbf{r})} = -\frac{1}{3n(\mathbf{r})} r_s \left[ \frac{\partial \epsilon_{c,LDA}}{\partial r_s} \right], \quad \text{(A.9)}$$

and

$$\frac{\partial \epsilon_{c,LDA}}{\partial r_s} = -2A\alpha_1 \ln \left[ 1 + \frac{1}{g(r_s)} \right] + 2A(1 + \alpha_1 r_s) \frac{2A(\beta_1 0.5 r_s^{-1/2} + \beta_2 + \beta_3 1.5 r_s^{1/2} + \beta_4 2 r_s)}{g(r_s)(1 + g(r_s))}. \quad \text{(A.10)}$$

Finally,

$$\frac{\partial q_0(\mathbf{r})}{\partial (|\nabla n(\mathbf{r})|^2)} = -\frac{k_F Z_{ab}}{(6k_F n(\mathbf{r}))^2}. \quad \text{(A.11)}$$
Appendix B

3D Cubic Hermite Interpolation

A 3-dimensional cubic Hermite spline is used to interpolate values of density and gradient from the octree nodes. The polynomial in question uses the strict tricubic polynomial,

\[ f(x, y, z) = \sum_{i=0}^{3} \sum_{j=0}^{3} \sum_{k=0}^{3} a_{ijk} x^i y^j z^k, \]  

(B.1)

as a starting point. As this contains 64 unknowns and only 32 known boundary conditions are stored on the tree (density and three gradient components for each cube vertex), cross terms \( x^i y^j z^k \) are excluded which are nonlinear in more than one variable.

Rather than solving for 32 coefficients using a system of equations and the available boundary conditions, a more elegant solution for the polynomial is achieved by considering second derivatives. In the one dimensional case, for example, the cubic polynomial can be expressed in terms of its second derivatives \( y'' \) on each node \( x_i \) as \( [100] \)

\[ y = Ay_0 + By_1 + Cy''_0 + Dy''_1 \]  

(B.2)

where

\[ A = \frac{x_1 - x}{x_1 - x_0} \]  

(B.3)

\[ B = \frac{x - x_j}{x_1 - x_0} \]  

(B.4)

and

\[ C = \frac{1}{6} (A^3 - A)(x_1 - x_0)^2 \]  

(B.5)

\[ D = \frac{1}{6} (B^3 - B)(x_1 - x_0)^2. \]  

(B.6)

Equation (B.2) is equivalent to writing the standard form

\[ y = \ddot{A}x^3 + \ddot{B}x^2 + \ddot{C}x + \ddot{D} \]  

(B.7)

in terms of its own second derivatives (for example \( y''_0 = 6\ddot{A}x_0 + 2\ddot{B} \)). Generalizing Equation (B.2), we can write a polynomial in 3 dimensions which reduces to the form of (B.2) on each axis and which fulfills the 32 available boundary conditions:
\[ f(x, y, z) = (A_x A_y A_z f_{000} + B_x A_y A_z f_{010} + A_x A_y B_z f_{001} \\
+ B_x B_y A_z f_{110} + B_x A_y B_z f_{101} + A_x B_y B_z f_{111} \\
+ C_x A_y A_z f_{000} + D_x A_y A_z f_{010} + C_x A_y B_z f_{001} \\
+ D_x B_y A_z f_{110} + D_x A_y B_z f_{101} + C_y B_y B_z f_{111} \\
+ A_x C_y A_z f_{000} + B_x C_y A_z f_{010} + A_x D_y A_z f_{001} \\
+ B_x D_y A_z f_{110} + B_x C_y B_z f_{011} + B_x D_y B_z f_{111} \\
+ A_x A_y C_z f_{000} + B_x A_y C_z f_{010} + A_x B_y C_z f_{001} \\
+ B_x B_y C_z f_{110} + B_x A_y D_z f_{011} + A_x B_y D_z f_{011} + B_x B_y D_z f_{111}) \]

(B.8)

Equation (B.8) contains 24 second derivatives \( (f_{000}, ..., f_{111}) \) which are readily solved in the one-dimensional case from Equation (B.7) using the boundary conditions. Subindices mark the 8 cube vertices. Derivatives \( df/dx, df/dy, df/dz \) of Equation (B.8) are easily obtained, leading to separate polynomials for interpolation of the density gradient components. For the purpose of calculating the charge inside of an octree cube, Equation (B.8) is also easily integrated across the cube intervals \([x_0, x_1], [y_0, y_1], [z_0, z_1]\). Equation (B.8) and its derivatives were used in the work of Gulans et al. [85] for density interpolation. Further details of cubic spline interpolation can be found elsewhere [100].
References


REFERENCES


REFERENCES


[42] W.H. Keesom, Proc. Amsterdam 18, 636 (1915); 23, 939 (1920); 24, 162 (1921); Phys. Z. 22, 129 (1921).


REFERENCES


[68] SIESTA (Spanish Initiative for Electronic Simulations with Thousands of Atoms) homepage: http://www.icmab.es/siesta/


[91] A. Gulans, private communication.


[96] M. Abramowitz and I. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. (Online: http://people.math.sfu.ca/~cbm/aandms/page\_889.htm)

