Universidade de Brasília - UnB
Institute of Phisics

Siesta : A complete guide

Dr. Carlos Maciel de Oliveira Bastos

Brasília
2024

Sumário

1 Siesta Introduction **5**

2 A Pedestrian's Guide to Density Functional Theory. **8**
2.1 Electronic Structure Calculation 8
2.2 Hohenberg-Kohn Theorems and Kohn-Sham formalism 9
2.3 Exchange-Correlation Functionals 13
2.4 Van der Waals correction 16
2.5 It's your turn! 17

3 A Brief history of atomic basis set and pseudopotential **18**
3.1 Introduction to Localized Basis Sets used in SIESTA 18
3.2 Why pseudopotential? 19
3.3 How to obtain pseudopotential through ATOM code? 28
3.3.1 INPUT file 28
3.3.2 All Electron Calculation 31
3.3.3 Pseudopotential Generation Using the ATOM Code . . 32
3.3.4 Transferability Testing of Pseudopotentials Using ATOM 42
3.4 It's your turn! 46

4 Getting Started: Installing SIESTA. **48**
4.1 SIESTA's Install 48
4.2 How to execute the SIESTA and Specific execute options . . . 50
4.3 It's Your Turn! 51

5 Main Input Parameters in SIESTA **52**
5.1 Pseudopotential Format in SIESTA 52
5.2 The FDF Input File Format 53
5.3 Main Parameter Flags 55
5.3.1 Output and Input Structural Information 60
5.4 Output Files in SIESTA 62
5.5 It's your turn! 64

6 First Steps: Obtaining the Molecular Properties 65
6.1 H_2 Molecule: A Simple Example 65
6.1.1 Example FDF File for H_2 66
6.1.2 Method 1: Energy Curve Analysis 68
6.1.3 Method 2: Structural Optimization 69
6.2 The Box and Periodic Boundary Conditions 69
6.3 It's Your Turn! 71

7 A Pedestrian's Guide to Solid State 72
7.1 Crystal Structure 72
7.1.1 Lattice and Basis 72
7.1.2 Unit Cell and Symmetry 74
7.1.3 Bravais Lattice 76
7.2 Reciprocal Space 78
7.2.1 Brillouin Zone 79
7.3 Strain and Stress 80
7.3.1 Stress 80
7.3.2 Strain 81
7.3.3 Stress-Strain Relationship 82
7.4 Crystal Vibrations, Phonons, and Thermal Stability 82
7.4.1 Phonons 83
7.4.2 Thermal Properties and Phonons 83
7.5 Energy Band Theory 84
7.5.1 Formation of Energy Bands 84
7.5.2 Kronig-Penney Model 86
7.5.3 Band Gaps in Semiconductors and Insulators 86
7.5.4 Fermi Energy 87
7.6 Optical properties 89
7.6.1 The Optical Gap 90
7.6.2 Kramers-Kronig Relations 90
7.6.3 Fermi's Golden Rule in Optical Transitions 91
7.6.4 Optical Parameters 91
7.7 Exciton Theory 92
7.7.1 Formation of Excitons 92
7.7.2 Theoretical Framework 93
7.8 Group Theory in Solid-State Physics 94
7.8.1 Introduction to Group Theory 95
7.8.2 Applications of Group Theory in Solid-State Physics . 95
7.8.3 Representation Theory in Crystals 96
7.8.4 Key Results from Group Theory 97

7.8.5 Example: Application to Diamond and Zincblende Structures . . . 97
7.9 It's Your Turn! . . . 98

8 Convergence Tests in SIESTA 99
8.1 Energy Cutoff (Mesh Cutoff) . . . 99
8.2 K-Point Sampling . . . 100
8.3 Self-consistency charge density . . . 100
8.4 Fermi Smearing . . . 101
8.5 Forces in Structural Relaxation . . . 101
8.6 Main Convergence Variables in SIESTA . . . 102
8.7 Your Turn! . . . 102

9 Computing structural properties 103
9.1 Structural Relaxation in SIESTA . . . 103
9.1.1 Description of the Crystal Lattice . . . 103
9.1.2 Structural Optimization in SIESTA . . . 104
9.1.3 Example: GaAs in Zinc Blende Structure . . . 104
9.2 Calculating the Bulk Modulus Using SIESTA . . . 105
9.2.1 Example: Bulk Modulus of GaAs . . . 108
9.3 Your Turn! . . . 108

10 Obtaining Electronic Properties 109
10.1 First Brillouin Zone and k-Path Selection . . . 109
10.2 Extracting and Plotting the Band Structure . . . 110
10.3 Calculating Density of States with Siesta . . . 111
10.4 DOS Analysis with Siesta Utilities . . . 112
10.5 Visualizing the DOS . . . 112
10.6 Your Turn! . . . 113

11 D-Grimme correction for Van der Waals systems 114

12 Hybrid Functionals 115
12.1 Introduce to hybrid functional . . . 115
12.2 HSE06 hybrid functional: the study case . . . 115

13 LDA+U formalism in Siesta 116
13.1 An introduction of LDA+U Formalism . . . 116
13.2 Applying LDA+U in semiconductor material . . . 116

14 Optical and Phonons Properties 117

15 Interface Siesta and Wannier90: An introduction **118**
15.1 Siesta and Wannier90 118
15.2 Calculate the Fermi energy surface 118

16 Toolbox and post-processing **119**
16.1 Bader Charge 119

17 Complete caracterization of a Material: **120**

A Exchange-Correlation functionals in ATOM code **122**

Capítulo 1

Siesta Introduction

The **Spanish Initiative for Electronic Simulations with Thousands of Atoms (SIESTA)** is a computational software package specifically designed for simulating the electronic structure of materials at both atomic and molecular levels. Since its inception in the mid-1990s, SIESTA has developed into a leading tool in the field of computational materials science, enabling researchers to efficiently investigate systems composed of thousands of atoms.

SIESTA was created in response to the increasing demand for simulation techniques capable of addressing large-scale systems while preserving a high level of accuracy. Traditional quantum mechanical methods, such as Hartree-Fock and full Density Functional Theory (DFT), often become computationally prohibitive as system sizes increase. SIESTA overcomes this challenge by employing DFT, which simplifies the complex many-body problem by concentrating on electron density rather than wave functions.

DFT is a quantum mechanical method used to explore the electronic structure of many-body systems. The fundamental principle of DFT is that the ground-state properties of a system of electrons can be derived from the electron density, rather than from the many-body wave function. This reduction in complexity allows DFT to efficiently manage larger systems compared to wave function-based methods.

The accuracy of DFT is heavily reliant on the choice of the exchange-correlation functional, which approximates the effects of electron-electron interactions. SIESTA supports various functionals, including: LDA and GGA functionals.

SIESTA employs **numerical atomic orbitals (NAOs)** as basis functions. This selection enables a highly localized representation of electronic wave functions, which significantly reduces the computational costs associated with larger systems. The localized nature of the basis set leads to sparse

matrices that can be efficiently solved using advanced numerical techniques.

One of the most notable innovations in SIESTA is its implementation of linear-scaling algorithms. This advancement reduces computational complexity from cubic to linear with respect to the number of atoms, thereby making it feasible to conduct calculations on systems containing thousands of atoms. By capitalizing on the locality of the basis functions, SIESTA effectively handles electronic structure calculations.

In addition to static electronic structure calculations, SIESTA can perform **ab initio molecular dynamics (AIMD)** simulations. This capability allows researchers to investigate the time evolution of atomic systems under quantum mechanical forces, providing valuable insights into dynamic processes such as chemical reactions and phase transitions. AIMD simulations are instrumental in understanding material behavior under varying temperature and pressure conditions. SIESTA is designed to be user-friendly, catering to a wide range of users from novices to seasoned researchers. The software comes with "comprehensive documentation", and its modular structure facilitates easy integration with other codes and workflows, enhancing its usability across diverse research environments.

The performance of SIESTA is often assessed based on its ability to handle larger systems while maintaining accuracy. Benchmarks have shown that SIESTA can effectively simulate systems with thousands of atoms at a computational cost that scales linearly with system size. This capability is particularly beneficial for studies in nanotechnology, materials science, and biophysics, where large-scale simulations are frequently required.

SIESTA has found successful applications across various fields, including:

- **Investigating Electronic Properties of Nanostructures**: For example, studies on carbon nanotubes and quantum dots to understand their potential applications in electronics and materials science. SIESTA's capability to manage large systems allows for detailed investigations of the electronic and optical properties of these nanomaterials.

- **Studying Bulk Materials and Surfaces**: SIESTA is adept at examining the properties of bulk materials, surfaces, and interfaces, as well as analyzing defects and phase transitions in crystalline and amorphous materials. This functionality enables researchers to predict the mechanical, thermal, and electronic properties of new materials, guiding experimental efforts in material discovery.

- **Exploring Reaction Mechanisms**: SIESTA aids in understanding catalytic processes and the behavior of complex molecules, providing insights into molecular interactions and bonding that are crucial for

the design of new catalysts and the comprehension of biochemical processes.

- **Investigating Biomolecular Systems**: The software's capability to simulate large biomolecular complexes, such as proteins and nucleic acids, makes it an invaluable tool for studying biological systems at the atomic level.

In summary, SIESTA represents a significant advancement in computational materials science, enabling researchers to study large-scale systems with high accuracy and efficiency. Its implementation of DFT, combined with localized basis sets and linear-scaling algorithms, positions SIESTA as an essential tool for exploring the behavior of materials at the atomic level. As computational capabilities continue to evolve, SIESTA will remain a vital resource for scientists investigating the intricate properties of complex systems.

This materials is part of 6 tutorials where we recommend this approach,

Tutorial	**Chapters**
1 -Introduction to DFT Calculation and Pseudopotential theory	Chapter 1,2 and 3
2 - First Contact with SIESTA and simulation parameters	Chapters 4,5 and 6
3 - Periodic Systems	Chapter 7, 8, 9 and 10
4 - Improve the calculation	Chapter 11, 12 and 13
5 - Optical properties, Phonons and post-processig tools	Chapter 14, 15 and 16
6 - Complete caracterization	chapter 17

Tabela 1.1: Caption

Capítulo 2

A Pedestrian's Guide to Density Functional Theory.

2.1 Electronic Structure Calculation

Theoretical studies in condensed matter systems require an understanding of the dynamics involving many-particle interactions, such as electron-electron and electron-nucleus interactions. By accurately describing these interactions, one can derive both the structural and dynamical properties of a system, ultimately determining its electronic structure. This, in turn, enables the identification of structural properties, such as atomic arrangement, magnetic moments, phonons, and cohesive energy, among others. Additionally, the electronic excitation spectrum provides insight into dynamical properties, including molecular vibrational states and optical processes.

Theoretical methods can generally be classified into two different approaches: atomistic and effective models. In the atomistic approach, the many-body Hamiltonian, which includes fundamental interactions, is solved directly. This method requires significant computational resources and involves several approximations, typically without the need for adjustable parameters. This class of methods is also referred to as first-principles or ab initio approaches. The second approach involves constructing a suitable Hamiltonian model. By utilizing available information about the system, one can identify key aspects and develop an effective Hamiltonian to describe the relevant physics.

Here, we present a brief review of density functional theory (DFT), which is fundamentally an ab initio method. It begins with the electronic Hamiltonian, derived from the fundamental Hamiltonian using the Born–Oppenheimer approximation. This approximation allows for the separation of the total Ha-

miltonian into nuclear and electronic components, simplifying the treatment of many-body interactions.

In the Born–Oppenheimer approximation, the Hamiltonian of a system with N electrons and M nuclei can be separated into electronic (H_e) and nuclear (H_n) components:

$$H = H_e + H_n \tag{2.1}$$

here H_e represents the electronic Hamiltonian, responsible for describing the electronic interactions, while H_n governs the dynamics of the nuclei. Therefore, the electronic part can be write as,

$$H_e = \sum_i^N \frac{\hbar^2}{2m} \nabla^2_{\mathbf{r}_i} + \sum_{i<j}^N \frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|} - \sum_i^N \sum_j^P \frac{Z_j e}{|\mathbf{r}_i - \mathbf{R}_j|}. \tag{2.2}$$

2.2 Hohenberg-Kohn Theorems and Kohn-Sham formalism

In 1964, Pierre Hohenberg and Walter Kohn published two theorems that laid the foundation for modern density functional theory, now known as the Hohenberg–Kohn (HK) theorems. These theorems provided a mathematical proof demonstrating that stationary many-body systems can be fully characterized by their ground-state electron density. To begin, let us consider the Hamiltonian for a system of N electrons interacting under the Born-Oppenheimer (BO) approximation[1],

$$H_e = \sum_i^N \frac{\hbar^2}{2m} \nabla^2_{\mathbf{r}_i} + \sum_{i<j}^N \frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|} - \sum_i^N \sum_j^P \frac{Z_j e}{|\mathbf{r}_i - \mathbf{R}_j|}. \tag{2.3}$$

The last term of the equation represents the interaction between electrons and nuclei. Since this term depends on the positions of the nuclei, it is referred to as the external potential. Now, let us proceed to find the solution to the Schrödinger equation (SE),

$$\hat{H}_e \left|\psi_k\right\rangle = E_k \left|\psi_k\right\rangle . \tag{2.4}$$

For simplicity, we will consider only the non-degenerate ground state ψ_0. The Schrödinger equation depends on the external potential, the wave functions, and indirectly on the ground-state density (which is the squared modulus of the wave function). Thus, the solution to the SE can be viewed as a mapping

[1]For simplicity, we do not consider the presence of electromagnetic external fields and we consider $T = 0$. Both considerations can be extended to treat general cases.

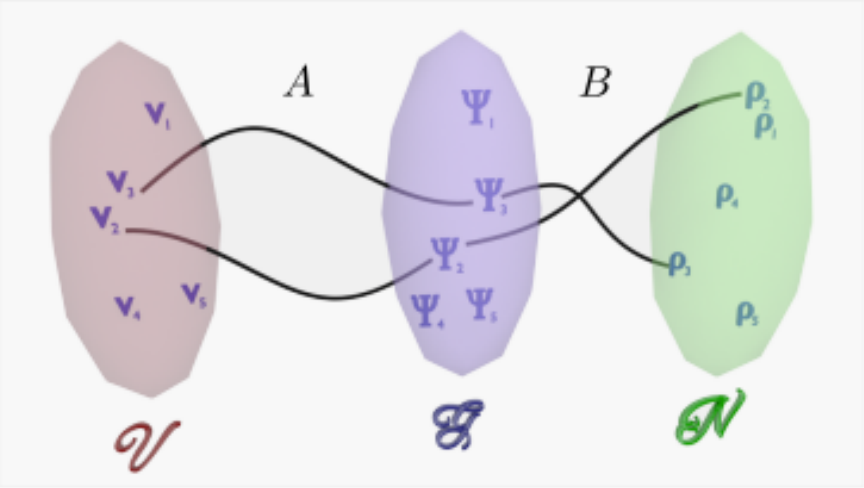

Figura 2.1: Mapping between external potential V, wave-function G and density N

between the external potential, the wave functions, and the ground-state density, i.e., Let V represent the set of external potentials v_{ext} corresponding to a non-degenerate ground state $|\psi_0\rangle$, with the condition that $v'_{\text{ext}} \neq v_{\text{ext}} + \text{const}$. Similarly, let G denote the set of ground-state wave functions $|\psi_0\rangle$ corresponding to an element of V, where $|\psi'_0\rangle \neq e^{i\theta}|\psi_0\rangle$ (i.e., the wave functions are unique up to a phase factor). Finally, let N represent the set of ground-state densities ρ_0, where $\rho_0 = |\langle\psi_0|\psi_0\rangle|^2$ and $\psi_0 \in G$.

Now, consider the mappings:

$$A : V \to G,$$
$$B : G \to N,$$

as illustrated in Fig 2.1.

Hohenberg and Kohn showed that the mapping A is unique, meaning it is impossible to obtain the same ground state $|\psi_0\rangle$ from two different external potentials. Similarly, it is impossible to obtain the same ρ_0 (the ground-state density) from two different wave functions, i.e., the mapping B is also unique. Based on these arguments, Hohenberg and Kohn formulated and mathematically proved two theorems:

$$|\psi_0\rangle = |\psi[\rho_0]\rangle \tag{2.5}$$

This theorem is a direct consequence of the correspondence between the external potential, the ground-state wave function, and the electronic density.

There is also a minimum for $E[\rho]$: if ρ is the ground-state density corresponding to v_{ext}, we have for all densities $\rho' \neq \rho_0$ the following:

$$E[\rho_0] < E[\rho'] \quad \Longleftrightarrow \quad E_0 = \min E[\rho], \quad \text{with}\, \rho \in N$$

This theorem is a consequence of the variational principle and the unique relationship between ρ and $|\psi_0\rangle$. The domain of the functional $E[\rho]$ is restricted to the densities in N, i.e., only the ground-state densities that are solutions of the Schrödinger equation (SE). Despite the remarkable advancement brought by the Hohenberg–Kohn (HK) theorems, this formalism does not have immediate practical applications because the explicit forms of certain functionals are unknown, such as the kinetic energy functional.

In 1965, Walter Kohn and Lu Sham proposed a method to determine the density for an interacting system using an effective non-interacting system, where the explicit form of the density functional is known. This exact mapping is called the Kohn–Sham formalism (KS). Consider a non-interacting system with an external potential v_s, where the Hamiltonian is given by

$$\hat{H}_s = \hat{T}_s + \hat{V}_s, \tag{2.6}$$

where $\hat{T}_s$ represents the kinetic energy. Using the HK theorems, the ground state—being the solution of the SE for this Hamiltonian—is a unique functional of the ground-state density $|\Phi[\rho]\rangle$. Since the system is non-interacting, the wave function can be constructed using a Slater determinant. The explicit forms of the Coulomb potential and kinetic energy are known, and the energy functional is given by

$$E_s[\rho] = \langle \Phi[\rho] | \hat{T}_s[\rho] | \Phi[\rho] \rangle + \int v_s(r)\rho(r)\,\mathrm{d}r. \tag{2.7}$$

It is important to note that the wave function $|\Phi[\rho]\rangle$ from the non-interacting system is not identical to the wave function of the interacting system. However, there exists an external potential v_s such that the ground-state density $\rho_0(r)$ is the same as the ground-state density of the interacting system with potential v_{ext}. While v_s and v_{ext} differ, they yield the same ground-state density. Therefore, the ground-state density of an interacting system can be represented by a non-interacting system. This representation is called the Kohn–Sham system, and its wave functions ϕ_i^{KS} are known as Kohn–Sham orbitals.

In other words, the Kohn–Sham system can be viewed as a mapping between an interacting system and a non-interacting system, both having the same ground-state density as the interacting system:

$$\rho_0(r) \equiv \rho_0^s(r) = \sum_i |\phi_i^{\mathrm{KS}}(r)|^2. \tag{2.8}$$

It is important to emphasize that while the electronic densities obtained from the non-interacting system and the Kohn–Sham system are numerically identical, they are fundamentally different. The first results directly from a non-interacting system, while the second is a non-trivial representation of the interacting system.

In order to find the v_s that reflects the nature of the interacting system, Kohn and Sham construct the wave function using the Kohn–Sham orbitals and evaluate the total energy functional. The total energy of the system is given by three terms,

$$E[\rho] = T_s[\rho] + E_H[\rho] + E_{\mathrm{xc}}[\rho]. \tag{2.9}$$

In the first term, the kinetic energy of the interacting system is split into two parts: i) the single-particle contribution, which represents the majority of the kinetic energy and has a known functional form; and ii) the contribution from the interacting particles. The latter is included in the exchange-correlation term, represented by the last term in equation (2.9).

The second and third terms originate from the Coulomb interaction, which, due to the anti-symmetric nature of the wave function (constructed using a Slater determinant), is divided into two components: the Hartree term $E_H[\rho]$ and the exchange term. The exchange term is combined with the electron correlation, which is absent in the non-interacting system. This combination forms the exchange-correlation term $E_{\mathrm{xc}}[\rho]$.

The second term, the Hartree energy, has an exact known functional form, given by

$$E_H[\rho] = \frac{1}{2} \int \mathrm{d}r' \, \frac{\rho(r')\rho(r)}{|r - r'|}. \tag{2.10}$$

Note that this integral is performed over all electrons, including the one interacting with the effective potential.

In the Kohn–Sham formalism, the exchange-correlation term $E_{\mathrm{xc}}[\rho]$ incorporates the complex many-body effects that are not accounted for by T_s and E_H. However, E_{xc} does not have an exact explicit functional form and thus requires approximations to be solved. As a consequence of these approximations, the self-interaction in the Hartree term is not fully cancelled. To correct for this, an additional potential is included in the exchange-correlation functional to cancel the self-interaction.

Therefore, the exchange-correlation energy is composed of four components: i) the exchange term, ii) the electronic correlation, iii) the correction

to the kinetic energy, and iv) the self-interaction energy correction. This can be expressed as:

$$E_{\mathrm{xc}}[\rho(r)] = E_x[\rho(r)] + E_c[\rho(r)] + (T[\rho(r)] - T_s[\rho(r)]) + E_{\mathrm{si}}[\rho(r)]. \quad (2.11)$$

This decomposition of the total energy is valid for densities that are simultaneously interacting and non-interacting, meaning the electronic density for an interacting system can be obtained using the non-interacting system. The terms T_s and E_H, which are exact functional forms of the density, can be evaluated directly. These components typically dominate over the exchange-correlation energy. Therefore, even an approximate treatment of E_{xc} allows for a sufficiently accurate description of most many-particle systems.

With the energy partition, the ground state can be obtained using the variational method, which results in the following equation:

$$\{T_s[\rho] + v_H[\rho] + v_{\mathrm{xc}}[\rho]\}\, \phi_i^{\mathrm{KS}}(r) = \epsilon_i \phi_i^{\mathrm{KS}}(r), \quad (2.12)$$

where $\phi_i^{\mathrm{KS}}(r)$ are the Kohn–Sham orbitals, $v_H[\rho]$ is the Hartree potential, $v_{\mathrm{xc}}[\rho]$ is the exchange-correlation potential, and ϵ_i is the corresponding energy eigenvalue for the i-th orbital. The equation (2.12) is called the Kohn–Sham equation and is traditionally solved using a self-consistent method. This method starts with a trial density $\rho^{(1)}(r)$, allowing the construction of a tentative potential

$$v_s^{(1)}(r) = v_H[\rho^{(1)}](r) + v_{\mathrm{ext}}[\rho^{(1)}](r) + v_{\mathrm{xc}}[\rho^{(1)}](r), \quad (2.13)$$

assuming that the explicit functional $v_{\mathrm{xc}}[\rho^{(1)}](r)$ is known. The solution of the equation (2.12) using the potential $v_s^{(1)}(r)$ then provides a set of orbitals $\phi_i^{\mathrm{KS}(2)}(r)$ that lead to an improved density

$$\rho^{(2)}(r) = \sum_i |\phi_i^{\mathrm{KS}(2)}(r)|^2. \quad (2.14)$$

The density $\rho^{(2)}$ can, in turn, be used to obtain improved potentials. This iterative procedure is repeated until convergence criteria are reached, such as when the difference between the densities obtained in two successive iterations falls below a predefined accuracy threshold. In practice, managing this self-consistent procedure can be non-trivial.

2.3 Exchange-Correlation Functionals

In the Kohn–Sham (KS) formalism, the only term that fundamentally requires approximation is the exchange-correlation functional. This term

can be analyzed through a model derived from the homogeneous electron gas (HEG). The HEG is conceptualized as an infinite system with interacting electrons uniformly distributed throughout space.

When analyzing the Coulomb interaction in the HEG, it is essential to note that the long-range nature of this interaction necessitates addressing the energy density of the HEG. This issue is resolved by incorporating a homogeneous background of positive charge, which cancels the divergence associated with the long-range interaction. Since the net charge within any finite volume in space is zero, the long-range Coulomb force effectively becomes zero, resulting in a finite energy density.

Using the HEG model, one can derive an approximation for the analytical exchange energy functional. For the correlation term, the energy functional can be obtained in two distinct regimes: high density and low density. The correlation energy functional across these two regimes is determined through interpolation.

In real systems, such as solids and molecules, the electron densities are typically inhomogeneous. However, the HEG can still be employed to estimate the exchange and correlation energies in systems with inhomogeneous electron densities. In the local density approximation (LDA), the inhomogeneous system is treated as if it locally possesses the same density as a HEG, denoted as $\rho_{\mathrm{HGE}} = \rho_{\mathrm{inhom}}$.

Other corrections included in the exchange-correlation term, such as the interacting part of the kinetic energy, are also based on the same local concept for exchange and correlation terms. As the LDA is a first-principles functional of ρ, it does not depend on any free parameters that introduce a physical scale or experimental data, and it is treated locally as an HEG. Therefore, it is expected that the LDA is particularly suitable for systems that share properties with the HEG, such as metals.

A different class of exchange-correlation functionals based on the homogeneous electron gas (HEG) is the Generalized Gradient Approximation (GGA), which incorporates information from both the electron density and its gradient. Unlike the Local Density Approximation (LDA), the construction of GGA functionals is not unique, allowing for various parameterizations, such as the Perdew–Wang (PW91) and Perdew–Burke–Ernzerhof (PBE) functionals, among others.

Both LDA and GGA utilize the HEG framework; however, they share a common deficiency: neither of these approximations can be accurately described in the context of a one-particle system. Specifically, for a single electron, the exchange term does not simplify to a pure Coulomb self-interaction. Consequently, in a many-particle system, the self-interaction error propagates through all particles. As suggested by Zunger and Perdew, this self-interaction

error is responsible for the inability of Density Functional Theory (DFT) to accurately predict the energy gap in semiconductors.

The primary issue with local and semi-local exchange-correlation approximations is their inability to account for non-local effects, as these approximations consider only local interactions. To address this limitation, a new class of functionals known as hybrid functionals was proposed. Hybrid functionals combine the exact exchange energy, typically evaluated using the Hartree–Fock (HF) method, with semi-local exchange energy.

The first hybrid functional, proposed by Perdew et al., is known as the PBE0 functional. This functional is formulated by mixing 75% of the PBE exchange energy with 25% of the non-local Fock exchange derived from the Hartree–Fock method. The correlation energy is taken from the PBE functional, leading to the expression:

$$E_{XC}^{\mathrm{PBE0}} = E_c^{\mathrm{PBE}} + \alpha E_x^{\mathrm{HF}} + (1-\alpha) E_x^{\mathrm{PBE}}, \tag{2.15}$$

where $\alpha = 0.25$. The mixing ratio of 0.25 for the non-local Fock exchange was determined by Perdew et al. through adiabatic perturbation theory.

Another hybrid functional, the HSE functional, was proposed by Heyd, Scuseria, and Ernzerhof. This functional is derived from PBE0 by introducing a screening function to separate the semi-local PBE exchange and the non-local Fock term into short-range (SR) and long-range (LR) exchange contributions. Heyd et al. demonstrated that the non-local Fock LR contribution cancels with a portion of the PBE LR exchange, resulting in the hybrid HSE functional defined by the equation:

$$E_{XC}^{\mathrm{HSE}} = E_c^{\mathrm{PBE}} + E_{x,\mathrm{LR}}^{\mathrm{PBE}}(\omega) + \alpha E_{x,\mathrm{SR}}^{\mathrm{HF}}(\omega) + (1-\alpha) E_{x,\mathrm{SR}}^{\mathrm{PBE}}(\omega), \tag{2.16}$$

In this equation, a new parameter ω is introduced, which measures the intensity of the Coulomb screening, indicating the extent of non-local Fock interactions. For example, if $\omega = 0\,\text{Å}^{-1}$, the SR term is equivalent to the full Fock operator, and the LR contribution becomes zero. Conversely, as ω approaches infinity, the range of the SR terms decreases, asymptotically recovering the PBE functional. Similar to the PBE0 functional, the parameter α in the HSE functional controls the proportion of PBE exchange replaced by the non-local Fock exchange and, in principle, can vary from 0 to 1.

The most widely used parametrization for the HSE functional is referred to as HSE06, which is characterized by the parameters $\alpha = 0.25$ and $\omega = 0.206\,\text{Å}^{-1}$. These values were derived from the same evaluation conducted by Perdew for the PBE0 functional and were fitted based on a large number of systems.

2.4 Van der Waals correction

Due to the relatively minor influence of Van der Waals interactions on total energy calculations, their correction is often neglected in the development of most correlation functionals. However, for layered and two-dimensional materials such as graphene and transition metal dichalcogenides (TMDs), this correction becomes fundamental. The interactions between different layers are described through Van der Waals potentials, expressed as

$$V_{dW} = V_{\text{dip-dip}}(r) + V_{\text{dip-ind}}(r) + V_{\text{London}}(r), \tag{2.17}$$

where $V_{\text{dip-dip}}(r)$, $V_{\text{dip-ind}}(r)$, and $V_{\text{London}}(r)$ represent the dipole-dipole, dipole-induced dipole, and London potentials, respectively.

The dipole-dipole term describes the electrostatic potential between permanent dipoles in a polar system. This interaction is part of the Coulomb potential in the Kohn–Sham equation and does not require further approximations. The dipole-induced dipole term involves interactions between polar and non-polar systems; similar to the dipole-dipole interaction, this potential is also incorporated into the Coulomb potential.

The London potential describes the interaction between instantaneous dipole moments, which arise from fluctuations in electron distribution. A classical expression for this potential between two heterogeneous bodies is given by

$$V_{\text{London}}(r) = -\frac{3}{2}\frac{\alpha_A \alpha_B}{R_{AB}^6}\frac{I_A I_B}{I_A + I_B}, \tag{2.18}$$

where I denotes the ionization potential. The London interaction represents a purely electronic correlation between two bodies and cannot be captured within the one-body mean-field approximation. Consequently, it is not included in Kohn–Sham calculations using conventional (semi-)local functionals and must be introduced via an additional term.

In the literature, several types of dispersion corrections have been proposed, generally classified into five categories: i) classical dispersion corrections, ii) corrections derived through perturbation theories, iii) linear-response theories, iv) Van der Waals (dispersion) functionals, and v) semiempirical dispersion-corrected functionals. In nowdays, the most common used is the semiempirical dispersion correction, which is based on the classical empirical form of the London interaction:

$$E_{\text{London, disp}} = -\sum_{i>j} C_{ij}^6 \frac{1}{R_{ij}^6} f_{\text{damp}}(R_{ij}), \tag{2.19}$$

where C_{ij}^6 is a parametrized interatomic dispersion coefficient and f_{damp} is a damping function that cuts off unnecessary short-range interactions. In

DFT-D corrections, there are three versions: DFT-D1, DFT-D2, and DFT-D3, each representing different levels of dispersion corrections.

This type of dispersion correction is efficient because it constitutes an energy correction performed a posteriori, meaning that the London dispersion energy is included after the completion of the Kohn–Sham cycles. Due to this a posteriori inclusion in the total energy, this method incurs a relatively low computational cost compared to other Van der Waals correction methods, such as the random phase approximation (RPA).

2.5 It's your turn!

1. Demonstrate the First Hohenberg-Kohn Theorem: The first Hohenberg-Kohn theorem establishes that the ground state electron density $\rho(r)$ uniquely determines the external potential $v_{ext}(r)$ (up to an additive constant). This theorem provides a foundation for density functional theory by asserting that all properties of a many-body quantum system can be derived from its ground state density.

2. Demonstrate the Second Hohenberg-Kohn Theorem: the second theorem states that the ground state energy of a system can be expressed as a functional of the electron density $\rho(\mathbf{r})$. This theorem allows us to minimize the energy functional to find the ground state density, which corresponds to the minimum energy of the system. Mathematically, it can be represented as:

$$E[\rho] \geq E_0,$$

   where E_0 is the true ground state energy, and equality holds for the true ground state density.

3. Demonstrate the Local Density Approximation (LDA) Functional: The Local Density Approximation is a popular method used in DFT where the exchange-correlation functional is approximated by considering the system to be locally homogeneous. In LDA, the exchange-correlation energy per particle $\epsilon_{\text{xc}}(\rho)$ is taken from the homogeneous electron gas model.

4. (Challenge) Demonstrate the Perdew-Burke-Ernzerhof (PBE) Functional: the PBE functional is a generalized gradient approximation that incorporates information about the density and its gradient. It aims to improve upon the LDA by accounting for inhomogeneities in the electron density.

Capítulo 3

A Brief history of atomic basis set and pseudopotential

3.1 Introduction to Localized Basis Sets used in SIESTA

The goal of the SIESTA project is to develop a computational code capable of achieving linear scaling, meaning it can efficiently handle systems with many atoms at a reasonable computational cost. A key factor in achieving this scalability is the choice of the basis set. The basis set is a collection of functions used to expand the wavefunction, where the properties of these functions are already well understood.

A widely used class of basis sets in quantum mechanical calculations is the plane-wave basis set. However, due to their extended nature, plane-wave basis sets are less suitable for linear-scaling calculations because of their high computational cost. In contrast, localized basis sets, such as Gaussian-type orbitals, truncated atomic orbitals, and wavelets, are more efficient in linear-scaling methods. SIESTA employs a localized spherical-wave basis set, which is particularly well-suited for linear scaling, aligning with the project's goal of computational efficiency.

In terms of computational efficiency, atomic orbitals allow for convergence with relatively few orbitals, making them highly desirable. However, this basis set must balance properties such as transferability and smoothness. For atomic wavefunctions, the primary flexibility is in the radial shape, and Numerical Atomic Orbitals (NAO) offer both flexibility in linear scaling and optimization capabilities that are more challenging in other basis sets, such as Gaussian-type basis sets.

The initial concept of NAOs stems from the tight-binding approach, fo-

cusing on minimal bases (single ζ). These are determined by finding the eigenfunctions of isolated atoms confined within spherical potential wells. These atoms are strictly localized, meaning the orbitals become zero beyond a certain cutoff radius r_c. An extension of this idea is the use of multiple ζ, based on the split-valence concept, but adapted to the strictly localized NAOs.

The starting point is to compute atomic orbitals, which will be used as the basis set, using the Kohn-Sham Hamiltonian for isolated pseudoatoms, with the same exchange-correlation functional and pseudopotential (as discussed in the next section). A confining potential is then added to impose boundary conditions on the basis set, truncating the function beyond a certain radius to define the basis set. In the minimal basis, SIESTA defines this as Single-Zeta (SZ).

The first step to increase the flexibility of the basis set is to add a second radial function, resulting in a Double-Zeta (DZ) basis. Several methods have been proposed for generating this second function, with the most common approach inspired by the split-valence method. This method begins with the Gaussian expansion of an atomic orbital, where the most contracted Gaussian is used to define the first orbital of the DZ. This process supplements each basis orbital with a new basis function that exactly reproduces the tail of the original orbital. This scheme can be extended to include more zetas (Triple-Zeta, etc.). Angular flexibility can be introduced by adding shells of higher angular momentum, known as polarization orbitals, which SIESTA also uses.

Another key consideration is the definition of the cutoff radii, where the basis functions are truncated. It is essential to find a balanced and systematic method for defining all the different cutoff radii. In SIESTA, the cutoff radii are determined based on a single parameter: the energy shift experienced by the orbital when confined, known as the energy shift parameter. Another important property is the consistency of the pseudopotential framework with the pseudoatomic orbitals in the core region. This is achieved by using the solution of the same pseudopotential in the free atom as the basis orbital.

3.2 Why pseudopotential?

When using atomic basis sets, it is not common to employ pseudopotentials, as the issues associated with other basis sets typically do not arise in this context. For instance, describing the wave function near the atomic nucleus with a plane-wave basis requires a high number of plane waves due to the rapid oscillation of the wave function in this region, which considerably increases the computational cost. However, pseudopotentials can enhance

computational efficiency, especially when core electrons do not significantly contribute to the calculations. Pseudopotentials yield a smoother pseudo-density on the grid, thereby improving convergence.

For periodic systems, plane waves are a natural choice due to their compatibility with translational symmetry and the satisfaction of boundary conditions. As the plane-wave (PW) method is fundamental to understanding pseudopotentials, we will provide a brief overview of this method. Although it is not utilized by SIESTA, it exemplifies the primary concepts underlying pseudopotentials.

A function of the form $e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}$ satisfies Bloch's theorem, which guarantees the preservation of translational symmetry in periodic systems. Using this form as a basis set, the wave function can be expanded as:

$$|\Psi_k\rangle = \sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}, \tag{3.1}$$

where C_{k+G} are the coefficients that describe the contribution of each plane wave to the total wave function. Additionally, the Hamiltonian is characterized by a periodic potential, such that

$$V(\mathbf{r}) = V(\mathbf{r}+\mathbf{G}), \tag{3.2}$$

where $\mathbf{G}$ is a reciprocal lattice vector. This periodicity in the potential is essential for the application of Bloch's theorem, leading to the conclusion that the eigenstates of the Hamiltonian can be expressed in the form of a Bloch function.

Therefore, the Schrödinger equation can be expressed as,

$$\left[-\frac{\hbar^2}{2m}\nabla^2 + \hat{V}(\mathbf{r})\right]\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right) = \epsilon\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right). \tag{3.3}$$

This leads to,

$$\left[-\frac{\hbar^2}{2m}\nabla^2\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right)\right] + \left[\hat{V}(\mathbf{r})\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right)\right] - \epsilon\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right) = 0. \tag{3.4}$$

Next, we can simplify this expression to,

$$\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right)\left[\frac{\hbar^2}{2m}(\mathbf{k}+\mathbf{G})^2\right] + \left[\hat{V}(\mathbf{r})\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right)\right] - \epsilon\left(\sum_G C_{k+G} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}\right) = 0. \tag{3.5}$$

Multiplying by $e^{-i(\mathbf{k}+\mathbf{G}')\cdot\mathbf{r}}$ and integrating over the unit cell volume V_c yields,

$$C_{k+G}\left[\frac{\hbar^2}{2m}(\mathbf{k}+\mathbf{G})^2 - \epsilon + \int_{V_c} \hat{V}(\mathbf{r}) e^{i(\mathbf{G}+\mathbf{G}')\cdot\mathbf{r}}\, dV_c\right] = 0. \tag{3.6}$$

Therefore, we obtain,

$$\frac{\hbar^2}{2m}(\mathbf{k}+\mathbf{G})^2 - \epsilon + \int_{V_c} \hat{V}(\mathbf{r}) e^{i(\mathbf{G}+\mathbf{G}')\cdot\mathbf{r}}\, dV_c = 0, \tag{3.7}$$

where the last term is a Fourier component of the crystal potential, denoted as $V_{G-G'}$. The eigenvalues can be obtained from the secular equation,

$$\det\left[\left(\frac{\hbar^2}{2m}(\mathbf{k}+\mathbf{G})^2 - \epsilon\right)\delta_{GG'} + \hat{V}_{G-G'}(\mathbf{r})\right] = 0. \tag{3.8}$$

In atomic units, the elements of the Hamiltonian are given by,

$$H_{GG'}(\mathbf{k}) = |\mathbf{k}+\mathbf{G}|^2\delta_{GG'} + \hat{V}_{G-G'}, \tag{3.9}$$

where $H_{GG'}(\mathbf{k})$ represents the matrix elements of the Hamiltonian in reciprocal space.

In the context of density functional theory, the Kohn-Sham partition energy can be expressed as:

$$\hat{V}_{G-G'} = \hat{V}_{el}(G-G') + \hat{V}_H(G-G') + \hat{V}_{XC}(G-G'), \tag{3.10}$$

where V_{el}, V_H, and V_{XC} represent the electron-ion interaction, the Hartree potential, and the exchange-correlation potential, respectively.

Typically, the number of plane waves is defined by the maximum kinetic energy $|\mathbf{G}|^2$, referred to as the cutoff energy. Depending on the problem, convergence can be slow due to the large number of plane waves necessary to accurately describe the wave function, primarily because the valence states may be orthogonal to the core states.

However, in 1940, Herring proposed a method that allows for a reasonable description of the physical properties using a smaller number of plane waves. This method introduced the concept of orthogonal plane waves (OPW), which are linear combinations of the core states and plane waves. The OPW functions are constructed to be orthogonal to the core states.

Practically, the OPW functions exhibit plane wave behavior at distances far from atoms but retain atomic characteristics when in proximity to the atomic cores. This method contributed to the advancement of plane wave basis sets; however, it presents significant convergence issues. Despite the construction of orthogonal states with respect to the core, the method complicates the secular equation calculations due to the complex terms involved.

In 1959, Phillips and Kleinman, along with Antoncik and Austin, utilized Herring's ideas to simplify the secular equation calculation through a method

known as the Pseudopotential Method. They considered the projection operator for core orbitals,

$$\hat{P} = \sum_c |\phi_c\rangle \langle\phi_c| \,, \tag{3.11}$$

where $|\phi_c\rangle$ represents the core orbitals. An OPW function can then be expressed as,

$$|k+G\rangle_{OPW} = (1-\hat{P})\,|k+G\rangle = |k+G\rangle - \sum_c |\phi_c\rangle \langle\phi_c|k+G\rangle \,. \tag{3.12}$$

Here, $|\phi_c\rangle$ and $|k+G\rangle$ denote the core orbital and a plane wave with wave vector $\mathbf{k}+\mathbf{G}$, respectively. Consequently, the electronic orbital can be written as a combination of the OPWs,

$$|\phi_{ki}\rangle = (1-\hat{P}) \sum_G C_{kG}\,|k+G\rangle \,. \tag{3.13}$$

This formulation ensures that $|k+G\rangle_{OPW}$ is always orthogonal to $|\phi_c\rangle$. Thus, the Schrödinger equation can be solved using a basis set of the OPWs:

$$\hat{H}\,|\phi_{ki}\rangle = \epsilon\,|\phi_{ki}\rangle \,. \tag{3.14}$$

Expanding this, we have,

$$\left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right](1-\hat{P}) \sum_G C_{kG}\,|k+G\rangle = \epsilon_k(1-\hat{P}) \sum_G C_{kG}\,|k+G\rangle \,. \tag{3.15}$$

This leads to,

$$\sum_G C_{kG}\left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right](1-\hat{P})\,|k+G\rangle = \sum_G C_{kG}\epsilon_k(1-\hat{P})\,|k+G\rangle \,. \tag{3.16}$$

Therefore, we have

$$\begin{aligned}\left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right]|k+G\rangle - \left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right]\hat{P}\,|k+G\rangle \\ = \epsilon_k\,|k+G\rangle - \epsilon_k\hat{P}\,|k+G\rangle\end{aligned} \tag{3.17}$$

Substituting the operator, we get

$$\begin{aligned}\left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right]|k+G\rangle - \left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right]\sum_c |\phi_c\rangle \langle\phi_c|k+G\rangle \\ = \epsilon_k\,|k+G\rangle - \epsilon_k \sum_c |\phi_c\rangle \langle\phi_c|k+G\rangle\end{aligned} \tag{3.18}$$

In the second term of the equation, we can identify the Schrödinger equation of the atomic orbitals. Therefore, we obtain

$$\begin{aligned}\left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right]|k+G\rangle - \sum_c \epsilon_c |\phi_c\rangle \langle\phi_c|k+G\rangle \\ = \epsilon_k |k+G\rangle - \epsilon_k \sum_c |\phi_c\rangle \langle\phi_c|k+G\rangle\end{aligned} \tag{3.19}$$

where ϵ_c represents the core state energies. Simplifying, we get

$$\begin{aligned}\left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{r})\right]|k+G\rangle + \sum_c (\epsilon_k - \epsilon_c) |\phi_c\rangle \langle\phi_c|k+G\rangle \\ = \epsilon_k |k+G\rangle\end{aligned} \tag{3.20}$$

Thus, we can write

$$\begin{aligned}\left[\hat{H} + \sum_c (\epsilon_k - \epsilon_c) |\phi_c\rangle \langle\phi_c|\right]|k+G\rangle \\ = \epsilon_k |k+G\rangle\end{aligned} \tag{3.21}$$

The second term in the equation can be interpreted as a repulsive potential derived from the sum of all core states over the plane wave,

$$\hat{V}_R = \sum_c (\epsilon_k - \epsilon_c) |\phi_c\rangle \langle\phi_c| . \tag{3.22}$$

Therefore, the states are identical to those of the original Schrödinger equation, independent of the choice of $|\phi_k\rangle$. Thus, we can choose a smooth function, referred to as a pseudofunction. The projector can be written as

$$|\phi_{ki}\rangle = (1 - \hat{P}) \left|\phi_{ki}^{PS}\right\rangle , \tag{3.23}$$

where $\left|\phi_{ki}^{PS}\right\rangle$ is called the pseudofunction. The Schrödinger equation can then be expressed as

$$\left[\hat{T} + \hat{V} + \hat{V}_R\right] \left|\phi_{ki}^{PS}\right\rangle = \epsilon_k \left|\phi_{ki}^{PS}\right\rangle , \tag{3.24}$$

with $\hat{T}$, $\hat{V}$, and $\hat{V}_R$ representing the kinetic energy, crystal potential, and repulsive potential, respectively. We can define the pseudopotential as

$$\hat{V}_{PS} = \hat{V} + \hat{V}_R, \tag{3.25}$$

leading to the equation

$$\left[\hat{T} + \hat{V}_{PS}\right] \left|\phi_{ki}^{PS}\right\rangle = \epsilon_k \left|\phi_{ki}^{PS}\right\rangle . \tag{3.26}$$

It is important to note that this equation is exact, and the eigenstates are the same as those of the original Schrödinger equation. However, they are obtained from a smooth function $\left|\phi^{PS}\right\rangle$, which depends on the appropriate choice of the pseudopotential $\hat{V}_{PS}$. Furthermore, pseudopotentials are not unique, and there are various methods for constructing them.

In the literature, there are many ways to construct pseudopotentials, which can be divided into two main categories: i) empirical pseudopotentials, defined from experimental data; and ii) ab-initio pseudopotentials, obtained from the solution of the Schrödinger equation for atomic cases.

In the second category, we highlight the methods proposed by Bachelet, Hamann, and Schlüter (BHS), as well as the Troullier-Martins method. Both belong to the class known as *Norm-Conserving Pseudopotentials*. SIESTA primarily works with norm-conserving pseudopotentials, which is why we will focus on this class. However, there are pseudopotentials that do not conserve norm, such as the method developed by Vanderbilt, which we will not discuss here.

To discuss norm-conserving pseudopotentials, we can calculate the norm of the pseudopotential, which is the difference between the wavefunctions from the original Schrödinger equation $|\phi\rangle$ and the wavefunction from the Schrödinger equation using a pseudopotential $\left|\phi^{PS}\right\rangle$. This difference is known as the *depletion charge*. Note that, while the states remain the same when using the pseudopotential method, the charge in the core regions differs due to variations in the wavefunction in this region.

Consider the two Schrödinger equations in atomic units,

$$\left(-\nabla^2 + V\right) \phi = \epsilon\phi, \tag{3.27}$$

$$\left(-\nabla^2 + V^{PS}\right) \phi^{PS} = \epsilon\phi^{PS}. \tag{3.28}$$

We can obtain the derivatives of the equations with respect to the eigenfunction,

$$\left(-\nabla^2 + V\right) \frac{\partial\phi}{\partial\epsilon} + \frac{\partial V}{\partial\epsilon}\phi = \phi + \frac{\partial\phi}{\partial\epsilon}, \tag{3.29}$$

$$\left(-\nabla^2 + V^{PS}\right) \frac{\partial\phi^{PS}}{\partial\epsilon} + \frac{\partial V^{PS}}{\partial\epsilon}\phi^{PS} = \phi^{PS} + \frac{\partial\phi^{PS}}{\partial\epsilon}. \tag{3.30}$$

Now, in the first equation, we multiply by ϕ^* and ϕ^{PS*} in their respective equations and integrate over the volume of the core region to find the charge.

After algebraic manipulation and using Green's theorem, we can subtract the two equations to verify that,

$$Z_d = \int_{\text{vol}} |\phi|^2 dv - \int_{\text{vol}} |\phi^{PS}|^2 dv. \tag{3.31}$$

This can also be expressed as,

$$Z_d = i \sum_i \int_{\text{vol}} \phi_i^{PS} \frac{\partial V_R}{\partial \epsilon_i} \phi_i^{PS} dv. \tag{3.32}$$

This result represents the difference in charge when pseudopotentials are used, indicating that the core charge is lower. This difference in charges is known as the depletion charge and represents the norm variation. In the case of norm-conserving pseudopotentials, the core charge must remain the same, i.e., $Z_d = 0$.

For the class of norm-conserving pseudopotentials, there exists a large number of methods to derive pseudopotentials. Here, we discuss the method proposed by Zunger and Cohen, which involves inverting the Schrödinger equation for a free ion ($V = 0$) including the pseudopotential. Consider the radial part of the free ion equation in atomic units,

$$\left[-\frac{1}{2} \frac{d^2}{dr^2} + \frac{l(l+1)}{2r^2} + V_{PS}^l(r) \right] |\psi_{ps,l}\rangle = \epsilon_l |\psi_{ps,l}\rangle . \tag{3.33}$$

Here, $|\psi_{ps,l}\rangle$ is the pseudofunction. Therefore, we can express the equation as,

$$-\frac{1}{2} \frac{d^2 |\psi_{ps,l}\rangle}{dr^2} + \frac{l(l+1)}{2r^2} |\psi_{ps,l}\rangle + V_{PS}^l(r) |\psi_{ps,l}\rangle = \epsilon_l |\psi_{ps,l}\rangle . \tag{3.34}$$

Rearranging this equation yields,

$$\boxed{V_{PS}^l(r) = \frac{1}{2 |\psi_{ps,l}\rangle} \frac{d^2 |\psi_{ps,l}\rangle}{dr^2} - \frac{l(l+1)}{2r^2} + \epsilon_l} \tag{3.35}$$

where $V_{PS}^l(r)$ is the pseudopotential for the l value, ϵ_l is the valence eigenvalue for the quantum number orbital l, and $|\psi_{ps,l}\rangle$ is the pseudofunction.

Next, we must find a pseudofunction that matches the wavefunction for distances far from the atomic nucleus, specifically for $r > r_c$, where r_c is the cutoff radius. Thus, we define,

$$|\psi_{ps,l}\rangle = \begin{cases} r^{l+t} f(r) & r < r_c \\ |\psi_l\rangle & r > r_c \end{cases} \tag{3.36}$$

where $|\psi_l\rangle = rR_l(r)$ and R_l is the radial part of the wavefunction obtained using all electrons (i.e., the wavefunction of the original Schrödinger equation). This can be calculated using DFT by solving the Kohn-Sham equations with an appropriate exchange-correlation functional.

For the case $r < r_c$, the parameter t can be set to $t = 1$, which gives the pseudopotential an exponential behavior $\sim \frac{1}{r}$, or $t = 2$, which results in an asymptotic behavior $\sim \frac{1}{r^2}$. The function $f(r)$ is chosen as,

$$f(r) = e^{\sum_i a_i r_i}. \tag{3.37}$$

Using $a_0 = 0$ eliminates the singularity, ensuring that all pseudopotentials exhibit good behavior (i.e., they are smooth and without singularities). Furthermore, at the point $r = r_c$, the all-electron wavefunction and the pseudofunction must be continuous. This continuity condition applies to both the first and second derivatives as well.

To characterize the norm-conserving class, an additional condition must be met: the integral of the pseudofunction over the volume from zero to r_c must equal the original charge. In other words, the charge in the core region must remain constant, ensuring that $Z_d = 0$.

In summary, a norm-conserving pseudopotential must possess the following characteristics,

1. The eigenvalues obtained for valence states from the pseudopotential must be identical to the original eigenvalues $\epsilon_i^{PS} = \epsilon_i$.

2. The all-electron function and the pseudofunction obtained from the pseudopotential must have the same values for $r > r_c$.

3. The charge density obtained from 0 to r_c using the pseudopotential must be identical to that obtained from the all-electron solution for $r > r_c$.

4. The logarithmic derivative of the pseudofunction must converge to that of the all-electron function for $r > r_c$.

Here, the all-electron function refers to the wavefunction obtained using all electrons, including the core electrons, without employing the pseudopotential. Property 3 ensures, through Gauss's theorem, that the electrostatic potential remains the same for both all-electron and pseudopotential cases, as the charge density is identical in both scenarios. On the other hand, property 4 guarantees that the screening properties are accurately reproduced. Additionally, properties 3 and 4 are conditions for *transferability*. Pseudopotentials that satisfy rules 1 to 4 are termed *norm-conserving with transferability*.

Now, the ionic pseudopotential $\tilde{V}_{PS,l}$ can be obtained by subtracting the contributions of the Hartree and exchange-correlation potentials due to valence electrons, as follows,

$$\tilde{V}_{PS,l}(\mathbf{r}) = V_{PS,l} - \int \frac{\rho_v(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} - V_{xc}(\rho_v), \tag{3.38}$$

where ρ_v is the valence electronic density. However, since the core and valence electronic densities exhibit significant overlap, a correction to the exchange-correlation potential is necessary. This correction is expressed as,

$$\tilde{V}_{PS,l}(\mathbf{r}) = V_{PS,l} - \int \frac{\rho_v(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} - V_{xc}(\rho_v + \rho_c), \tag{3.39}$$

which depends on the electronic density of the core. This correction is referred to as the core-correction.

Therefore, pseudopotentials are extremely useful in atomistic calculations. The primary parameter to adjust is the cutoff radius r_c, which depends on the specific problem. For example, consider the element Si:

1. If the calculation involves the crystal structure of pure Si, a wide range of values for r_c can be chosen. This helps to reduce the number of plane waves needed to describe the system.

2. Conversely, if the calculation involves more complex systems such as SiO_2, a hard pseudopotential is used, with r_c having a lower range value that approximates the real potential. This improves accuracy, but requires a higher number of plane waves.

Typically, r_c is chosen between the last node and the peak of the orbital solution of the Schrödinger equation for all electrons.

Once the pseudopotentials have been generated, it is necessary to test them. This testing involves evaluating properties such as lattice parameters, bulk modulus, electronic structure, cohesion energy, etc., and comparing them with experimental results. Generally, an error of the order of 5% for lattice parameters and 10% for bulk modulus is considered acceptable. There is no single method to generate pseudopotentials, and no pseudopotential is optimal for all problems. Each problem has a specific optimal pseudopotential.

Additionally, in self-consistent calculations using pseudopotentials (e.g., DFT calculations), other parameters may also need to be optimized. The main parameters to be optimized include cutoff energy and k-points mesh. These can be optimized by observing the convergence of total energy or lattice

parameters as a function of changing cutoff energies and k-point densities. In the next section, we describe a method to obtain the pseudopotentials using the ATOM code.

3.3 How to obtain pseudopotential through ATOM code?

In SIESTA, the pseudopotential is obtained from an external source, which is a user choice. In this section, we provide a brief presentation of the ATOM code. With the ATOM code, it is possible to:

- Perform all-electron DFT atomic calculations for arbitrary electronic configurations.
- Generate ab-initio pseudopotentials.
- Conduct atomic calculations in which the effect of the core is represented by a previously generated pseudopotential.
- Conduct a test to assess the transferability of the generated pseudopotential.

The code can be found on GitHub and is written in Fortran. Inside the code folder, there is information about the necessary libraries and instructions on how to compile the code.

3.3.1 INPUT file

The input file has a fixed-column format, meaning the code reads information by both columns and lines. This can be quite challenging, as precise alignment is essential. To assist with this, a ruler indicating the spacing is provided at the end of the file. Comments can be included using # at the beginning of a line. The following are the basic fields, represented using the notation (line,column) according to the ruler:

- (1,4): Type of calculation: **ae** for all electrons, **pt** for pseudopotential test, and **pg** for pseudopotential generation.
- (1,7): Title of the project, which may consist of more than one column (i.e., more than one word).
- (2,4): Chemical symbol.

- (2,9): Exchange-correlation functional, composed of two letters, which will be detailed in Appendix XXXX. The next letter represents the spin calculation mode, located at position (2,11):
  - **r**: Relativistic calculation.
  - **s**: Spin-polarized calculation (non-relativistic).
  - *blank*: Non-polarized calculation.

  It is possible to use the `libxc` library, which contains approximately 200 exchange-correlation functionals, if compiled together with `libGridXC`. Refer to the documentation of the ATOM code for more information.
- (3,8): Standard value of 0.0. This line is dedicated to creating atoms with fractional atomic numbers (which is not common to use).
- (4,5): Number of core orbitals.
- (4,10): Number of valence orbitals.
- (5-...,5): Principal quantum number (n) for the valence orbitals, with one line for each orbital.
- (5-...,10): Angular momentum quantum number (l) for the valence orbitals, with one line for each orbital.
- (5-...,17): Occupation of the orbital in electrons (spin up).
- (5-...,27): Occupation of the orbital in electrons (spin down).

```
#234567890123456789012345678901234567890123456789012345678901234567890      Ruler
   ae Ga Ground state all-electron
   Ga   pbr
       0.0
    6    2
    4    0       1.00       1.00
    4    1       1.00       0.00
#234567890123456789012345678901234567890123456789012345678901234567890      Ruler
```

Listing 3.1: Example of INP for all electrons

This format will be followed for both all-electron calculations and pseudopotential tests. For the pseudopotential generation mode, it is necessary to add two additional lines. We will repeat most of the above lines to create a didactic text and indicate the lines and columns[1].

For the pseudopotential generation mode, the following fields are required,

[1]If the column is not exact, the code will not produce a result and will not indicate the error. Therefore, it is important to ensure proper formatting.

- (1,4): Type of calculation **pg** for pseudopotential generation.
- (1,7): Title of the project, which may consist of more than one column, i.e., more than one word.
- (2,9): Generation scheme to calculate the pseudopotential, which can be **tm2** for Improved Troullier-Martins, **hsc** for Hamann-Schlüter-Chiang, and **ker** for Kerker. The Kerker and tm2 schemes can accommodate larger r_c due to their wavefunction matching conditions.
- (2,17): Radius at which the logarithmic derivatives are computed, for test purposes only.
- (3,4): Chemical symbol.
- (3,9): Exchange-correlation functional composed of two letters, as detailed in Appendix XXXX. The next letter represents the spin calculation mode, located at position (3,11):
  - **r**: Relativistic calculation.
  - **s**: Spin-polarized calculation (non-relativistic).
  - *blank*: Non-polarized calculation.

  It is possible to use `libxc`, which includes approximately 200 exchange-correlation functionals, if compiled together with `libGridXC`. Refer to the documentation of the ATOM code.
- (4,8): Standard value 0.0. This line is dedicated to creating atoms with fractional atomic numbers (which is not common to use).
- (5,5): Number of core orbitals.
- (5,10): Number of valence orbitals.
- (6-...,5): Principal quantum number (n) for the valence orbitals, with one line for each orbital.
- (6-...,10): Angular momentum quantum number (l) for the valence orbitals, with one line for each orbital.
- (6-...,17): Occupation of the orbital in electrons (up).
- (6-...,27): Occupation of the orbital in electrons (down).
- (last line,7): Values of radius cutoff for the orbital s.

- (last line,17): Values of radius cutoff for the orbital p.
- (last line,27): Values of radius cutoff for the orbital d.
- (last line,37): Values of radius cutoff for the orbital f.
- (last line,47): Values for non-local correction. See documentation.
- (last line,57): Values for non-local correction. See documentation.

```
#234567890123456789012345678901234567890123456789012345678901234567890123456789
   Ruler
   pg Ga pseudopotential generate
        tm2     2.00
   Ga   pbr
       0.0
    6    2
    4    0       1.00       1.00
    4    1       1.00       0.00
      1.80       2.00       0.00       0.00       0.00       0.00
#23456789012345678901234567890123456789
```

Listing 3.2: Example of INP for pseudopotential generate

3.3.2 All Electron Calculation

For the all electron calculation it's necessary only to run the executable of the ATOM code in the same folder of the INP file. Remember, the file have been following the rules described in the previous section 3.3.1 and with name INP. The INP files used are in Apenddix XXXX

The ATOM code will generate the following files

- **OUT:** Contains details about the run.
- **AECHARGE:** Contains the charge density (multiplied by $4\pi r^2$) in four columns[2]:
  1. r: radial distance.
  2. Spin "up"part of the *total* charge density (only for spin-polarized calculations).
  3. Spin "down"part of the *total* charge density (only for spin-polarized calculations).

[2]In order to obtain only the valence density, it is necessary to subtract the contributions of up and down valence parts from the total charge density, i.e., columns 2 and 3 minus column 4.

4. Total *core* charge density.

- **CHARGE:** This file is the same as AECHARGE (created for compatibility with other versions of ATOM).
- **RHO:** Similar to AECHARGE, but without the $4\pi r^2$ factor.
- **AEWFNR0...AEWFNR3:** Radial part (r) of the wavefunctions for s, p, d, and f orbitals, respectively. The file names differ by number, i.e., 0, 1, 2, and 3 for s, p, d, and f orbitals.

We analyze here two examples: Ga and As atoms, using the PBE functional with and without relativistic corrections, as well as the LDA exchange-correlation functional.

As shown in Figures 3.1 and 3.1, the two graphs at the top of the figure represent the radial part of the wave function for the s and p orbitals of the valence electrons. As discussed in Section 3.2, the wave function exhibits strong oscillations near the atomic nucleus. Additionally, we compare different exchange-correlation potentials and the effect of relativistic corrections. A slight deviation is observed between the exchange-correlation functionals, which is expected, as the exchange-correlation energy constitutes a small part of the total energy in the Kohn-Sham energy partitioning. However, this slight deviation can propagate in calculations involving many atoms and chemical bonds. Therefore, it is essential to use the same exchange-correlation functional consistently throughout all calculations to avoid error propagation and ensure consistent results.

These two figure represent the total and valence charge density (left) and the spin charge density (right). Similarly, the results show small variations across the different exchange-correlation functionals, for the same reasons mentioned earlier. All-electron calculations are crucial for determining the behavior of atomic properties and, more importantly, for defining the cutoff radius (r_c) to be used in pseudopotential generation.

3.3.3 Pseudopotential Generation Using the ATOM Code

In order to calculate the pseudopotential, we use the INP file described in section 3.3.1. For the pseudopotential calculation, it is necessary to specify the cutoff radius r_c. Typically, r_c is chosen between the last node and the peak of the all-electron wave function. When r_c is closer to the node, the calculation is more precise, but requires a larger number of plane waves in the basis set, increasing computational cost. On the other hand, selecting r_c

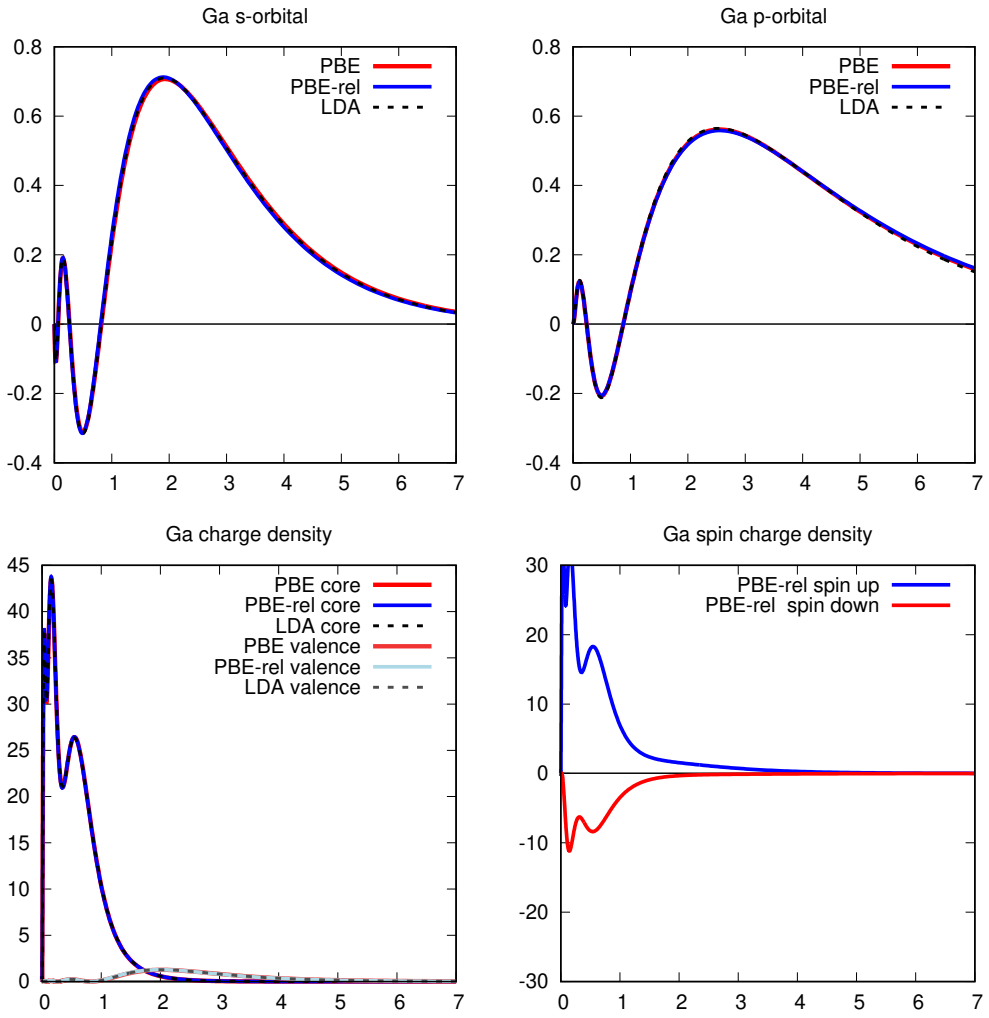

Figura 3.1: Charge density and radial part of the wave function for the Gallium atom, obtained from all-electron calculations using the ATOM code. The exchange-correlation functionals employed are LDA, PBE, and PBE with relativistic corrections. The figure illustrates the variations in charge distribution and radial wave function shape among these different approximations.

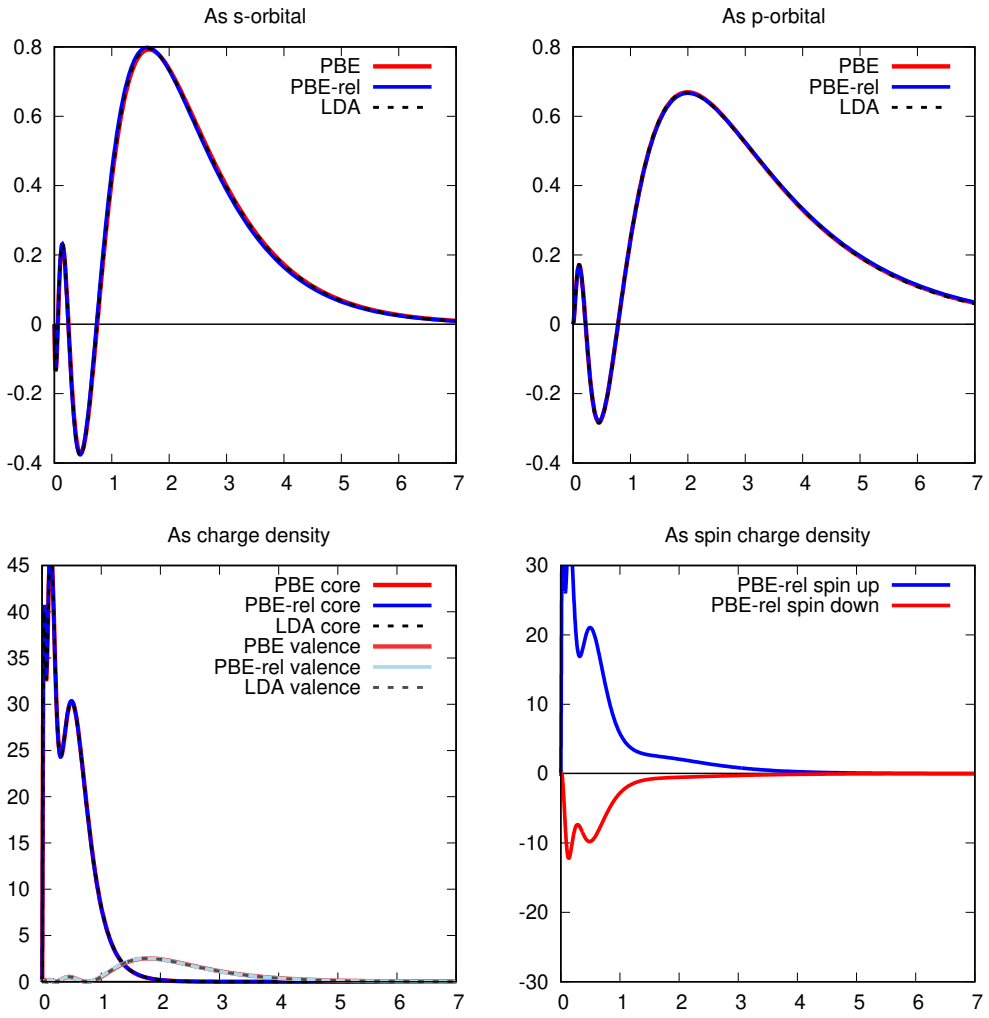

Figura 3.2: Charge density and radial part of the wave function for the Arsenic atom, obtained from all-electron calculations using the ATOM code. The exchange-correlation functionals used are LDA, PBE, and PBE with relativistic corrections. The results highlight the differences in charge distribution and radial wave function behavior across these approximations.

closer to the wave function peak reduces computational cost but results in lower precision. For more theoretical details on r_c, refer to Section 3.2.

For the pseudopotentials of Ga and As, we initially select $r_c^s = 1.6$ and $r_c^p = 2.1$ for Ga, and $r_c^s = 1.3$, $r_c^p = 1.6$ for As. Both sets of cutoff radii are chosen close to the wave function peaks.

In the same folder, run the ATOM code executable to generate the pseudopotentials. After the calculation, we obtain[3]:

- **AECHARGE, AEWFNR0...AEWFNR3, CHARGE, and OUT**: These files are the same as in the all-electron case.
- **PSCHARGE:** Contains the pseudo charge density (multiplied by $4\pi r^2$) in four columns:
  1. r
  2. Spin "up"part of the *pseudo* charge density (only for spin-polarized calculations)
  3. Spin "down"part of the *pseudo* charge density (only for spin-polarized calculations)
  4. Total *pseudo core* charge density
- **PSWFRNR0...PSWFRNR3**: Radial part of the valence pseudo wavefunctions for s, p, d, and f orbitals.
- **PSPOTR0...PSPOTR3**: Radial function of the ionic pseudopotential (unscreened) for s, p, d, and f orbitals. The last column contains the pseudo Coulomb potential, $-2Z_{ps}/r$, for the pseudo atom.
- **PSPOTQ0...PSPOTQ3**: Fourier transform of the pseudopotential $V(q)$ (in units of q^2/Z_{ps}) as a function of q (in $1/a_0$) for s, p, d, and f orbitals.
- **PSWFNQ0...PSWFNQ3**: Fourier transform of the pseudo wavefunctions as a function of q (in $1/a_0$) for s, p, d, and f orbitals.
- **VPSOUT and VPSFMT**: Formatted and unformatted files containing the pseudopotential, which can be used in SIESTA and other calculation software.

In Figures 3.3 and 3.4, we present the wavefunctions and their logarithmic derivatives for the s and p orbitals. The solid line represents the wavefunction

[3]INP files enable in Appendix XXXX.

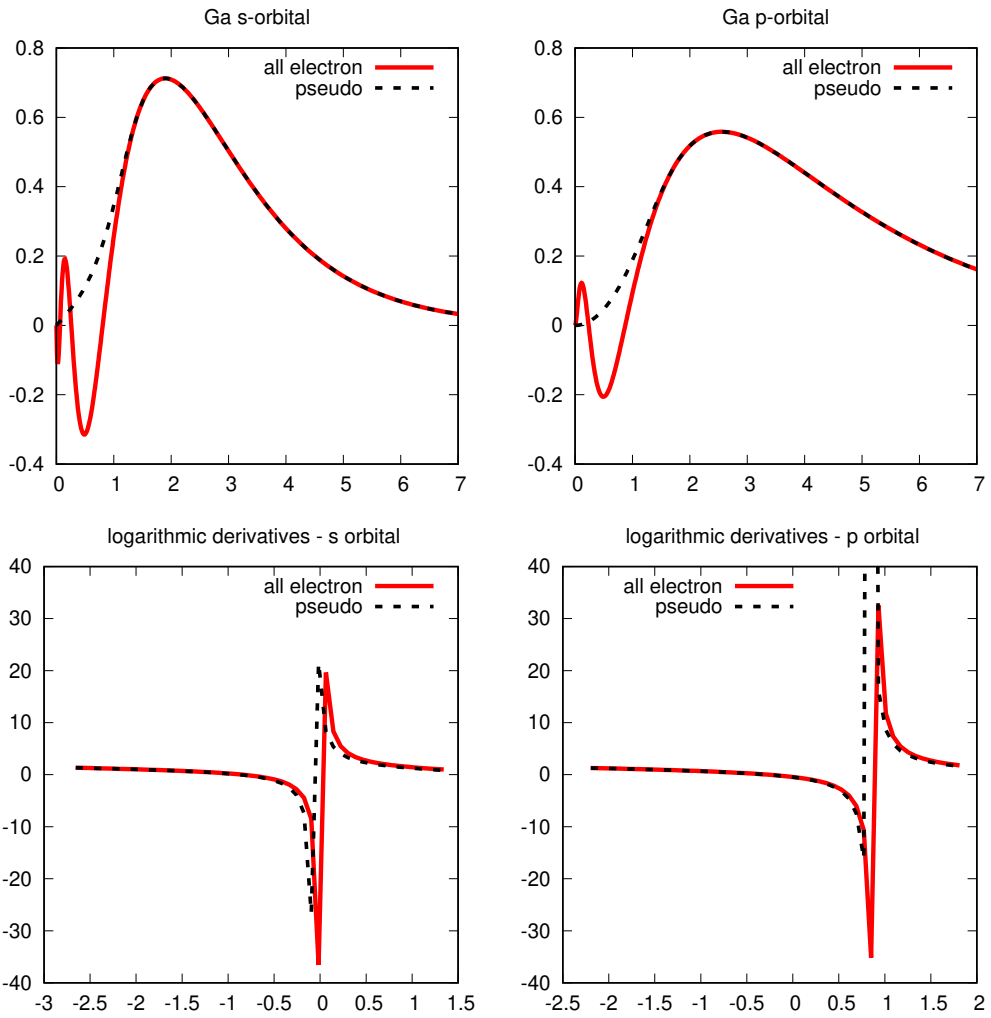
Ga s-orbital
all electron
pseudo
Ga p-orbital
all electron
pseudo
logarithmic derivatives - s orbital
all electron
pseudo
logarithmic derivatives - p orbital
all electron
pseudo

Figura 3.3:

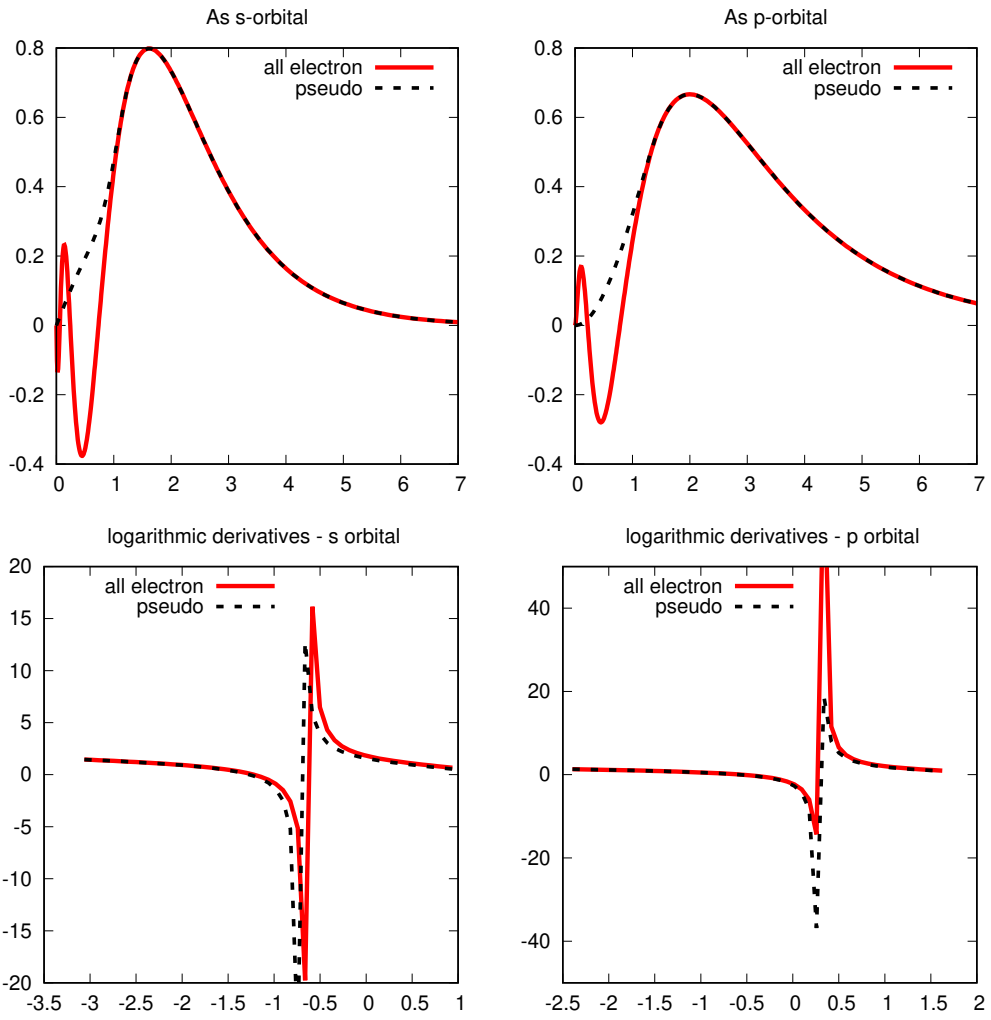
As s-orbital
all electron
pseudo
As p-orbital
all electron
pseudo
logarithmic derivatives - s orbital
all electron
pseudo
logarithmic derivatives - p orbital
all electron
pseudo

Figura 3.4:

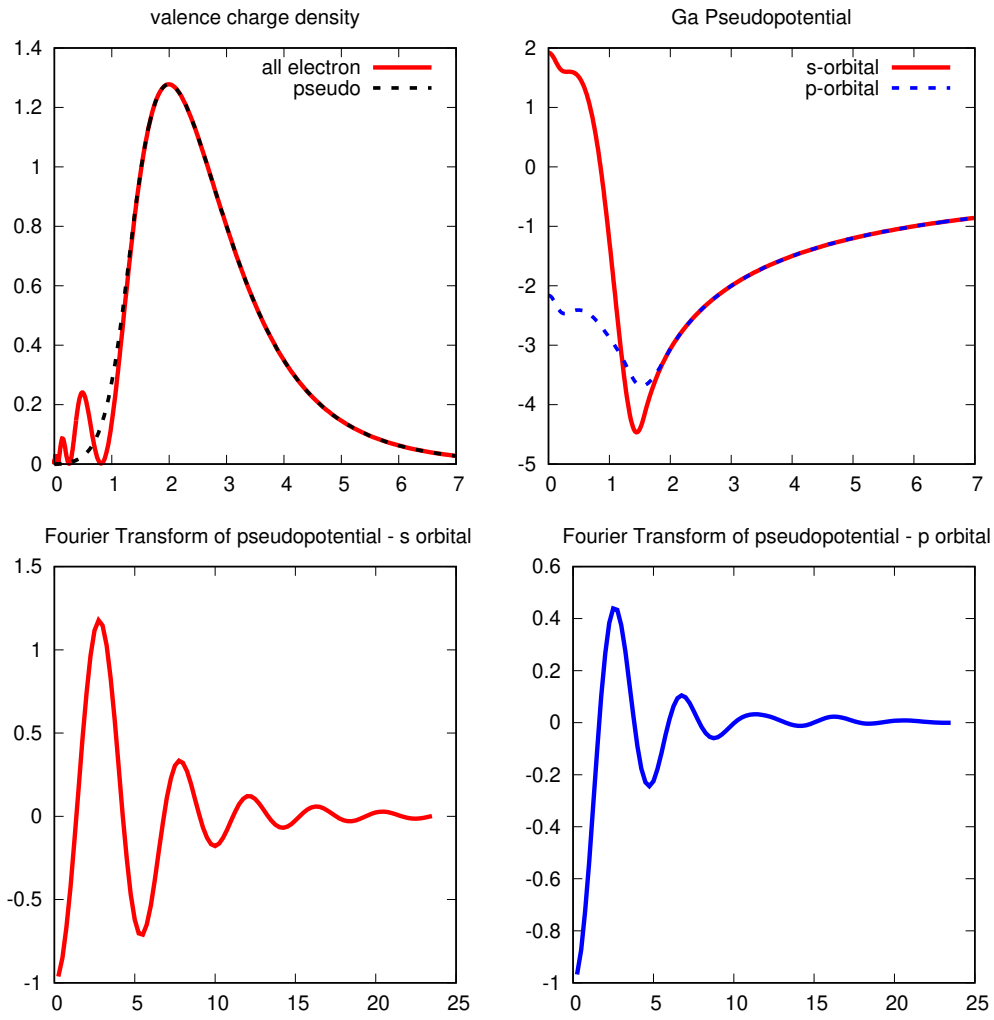
valence charge density
all electron
pseudo
Ga Pseudopotential
s-orbital
p-orbital
Fourier Transform of pseudopotential - s orbital
Fourier Transform of pseudopotential - p orbital

Figura 3.5:

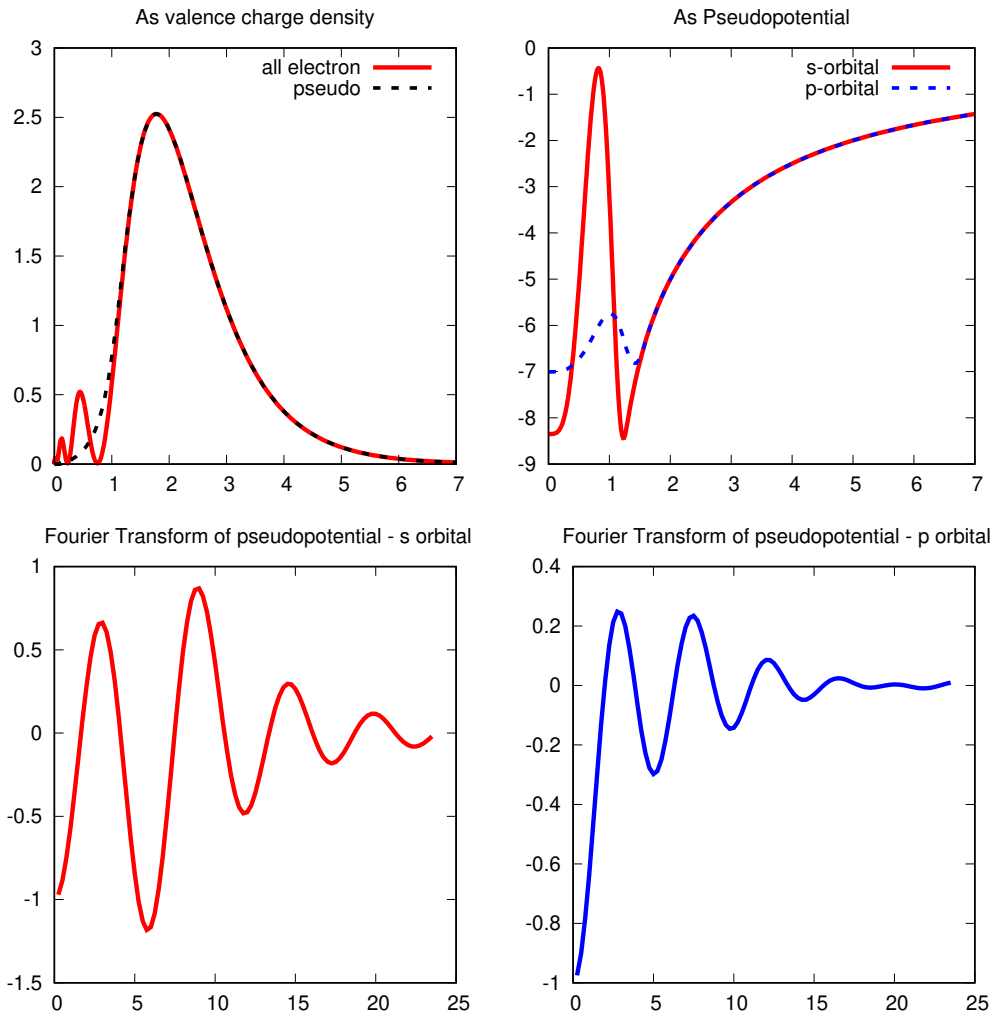
As valence charge density
all electron
pseudo
As Pseudopotential
s-orbital
p-orbital
Fourier Transform of pseudopotential - s orbital
Fourier Transform of pseudopotential - p orbital

Figura 3.6:

calculated using all-electron methods, while the dashed line corresponds to the wavefunction obtained from the pseudopotential. Notably, the dashed curve of the wavefunction is smooth and does not exhibit the oscillations characteristic of the all-electron wavefunction. Beyond the cutoff radius r_c, the pseudo and all-electron wavefunctions converge, consistent with the rules for norm-conserving pseudopotentials discussed in Section 3.2. Furthermore, the logarithmic derivatives are nearly identical; although slight deviations occur due to numerical methods and calculations, the results remain in good approximation.

Figures 3.5 and 3.6 display the valence charge densities (top left). According to the norm-conserving pseudopotential rule, the total charge from 0 to r_c must equal that of the all-electron case. It is important to note that while the total charge remains consistent, the charge densities differ due to the inherent differences between the pseudopotential and all-electron potential. This is illustrated by the distinction between the solid line (all-electron charge density) and the dashed line (pseudo charge density). The top right of the figure presents the pseudopotentials for the s (solid line) and p (dashed line) orbitals, while the bottom figures show the Fourier transform of the pseudopotential.

Another critical parameter is the cutoff radius r_c. To illustrate the effect of r_c, we generated the pseudopotential for the antimony atom (Sb), employing the PBE exchange correlation and the Troullier-Martins method. We defined three different cutoff radii: i) hard, close to the last knot of the wavefunction ($r_c = XXXX$); ii) medium, positioned midway between the hard cutoff radius and the peak of the wavefunction ($r_c = XXXX$); and iii) soft, with the cutoff radius near the peak of the wavefunction ($r_c = XXXX$). The results are displayed in Figure 3.3.3. All results adhere to the principles of norm-conserving pseudopotentials as described in Section XXXXX.

Additionally, the hard profile is recommended for systems where atoms interact with one another, especially in environments with significant overlap between wavefunctions, such as SiO_2. While a lower cutoff radius enhances the approximation of the pseudopotential to the all-electron potential, it necessitates a greater number of plane wave expansions, which increases computational costs. Conversely, a softer profile with a larger r_c reduces computational costs but is suitable for systems with similar chemical environments, such as crystalline Si.

Another important parameter is the method (or flavor) used to create the pseudopotential. In the ATOM we can choose: i) Trouiller-Martins (tm2); ii) Hamman-Schluter-Chiang and iii) Kerker (ker). In the figure 3.3.3 we compare all methods for case of Sb using PBE exchange correlation. The ker and tm2 show a good agreement among themselves, while hsc have a difference a

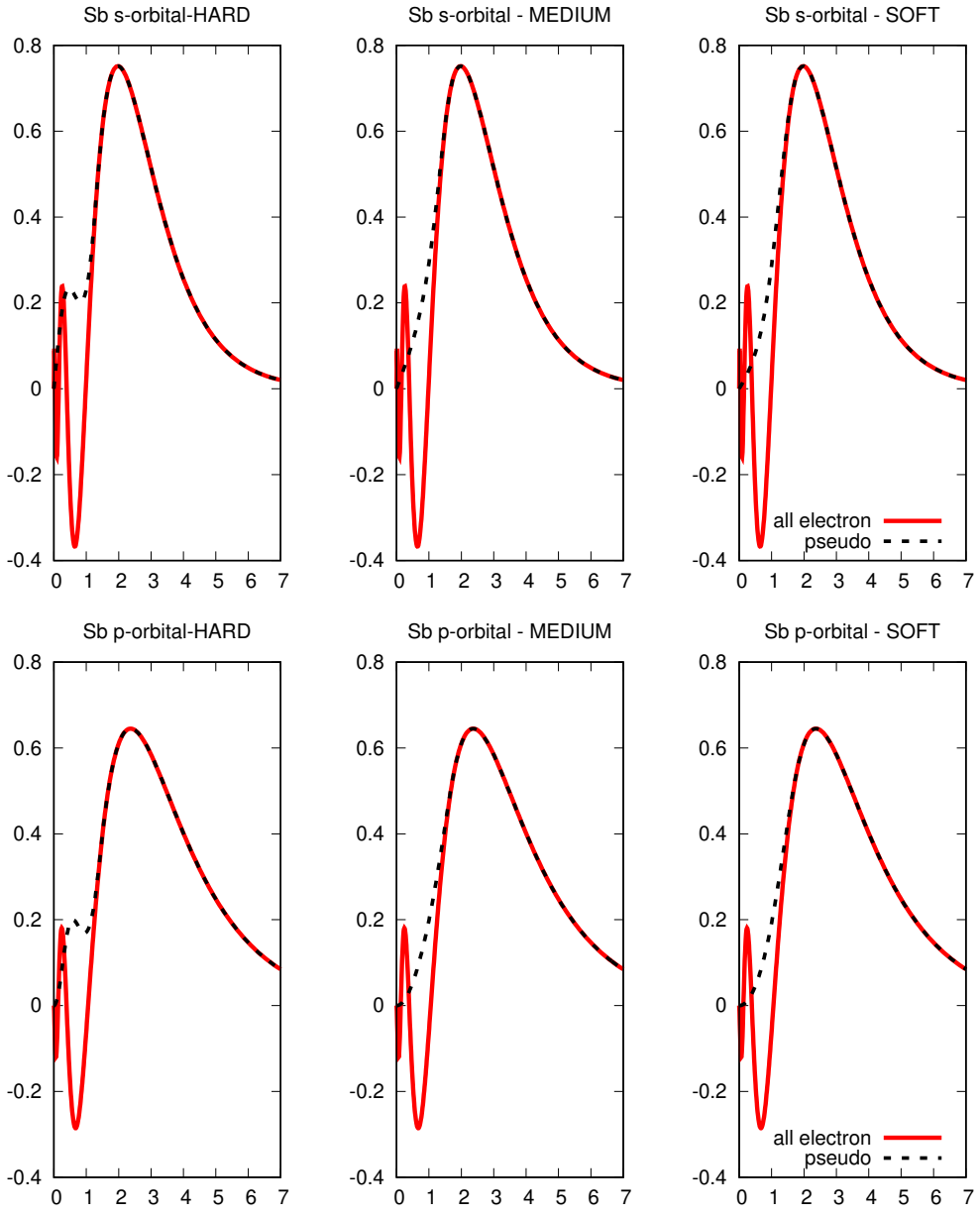
Sb s-orbital-HARD
Sb s-orbital - MEDIUM
Sb s-orbital - SOFT
Sb p-orbital-HARD
Sb p-orbital - MEDIUM
Sb p-orbital - SOFT
all electron
pseudo
all electron
pseudo

Figura 3.7:

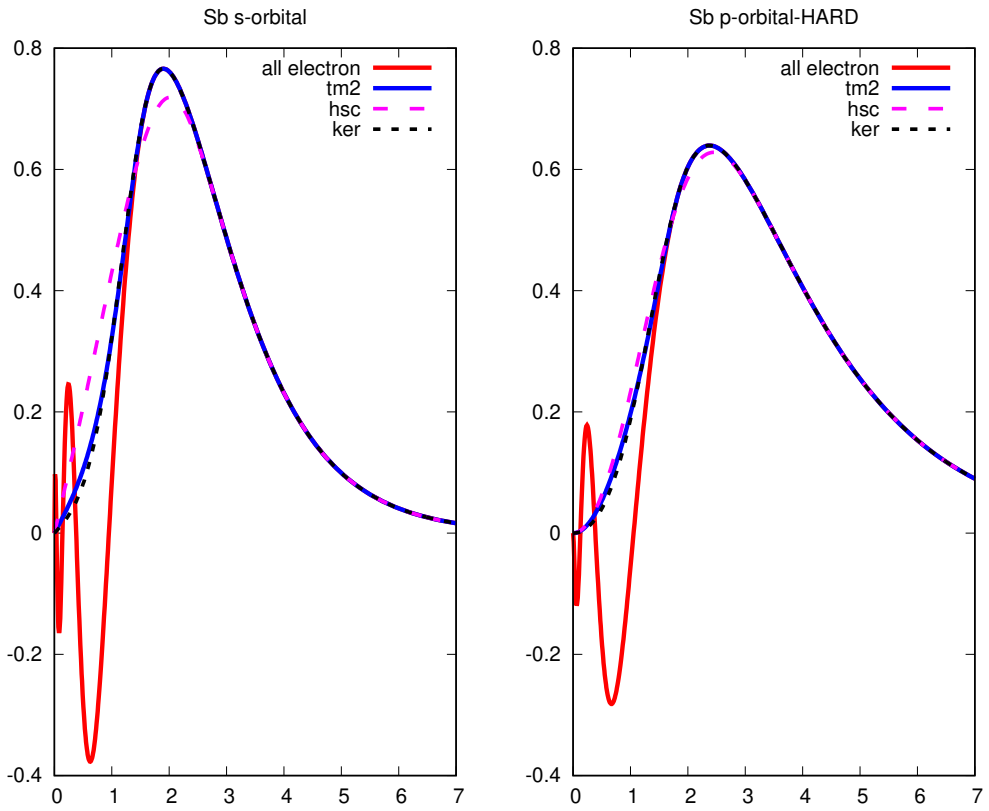

Figura 3.8:

pseudowavefunction. Also, all rules about norm-conserving pseudopotential are satisfied. Now is important to test this pseudopotentials before to use. The First test can be performed using the ATOM and will be described in the next section.

3.3.4 Transferability Testing of Pseudopotentials Using ATOM

Now that we have created our pseudopotential and examined the corresponding wavefunction, pseudopotential, and charge, it is important to note that these ”comparisons”do not substitute for a formal transferability test, which is a critical criterion for a high-quality pseudopotential. A pseu-

dopotential with good transferability will accurately reproduce all-electron properties, such as energy levels and wavefunctions, in various environments, including when forming bonds in solids and molecules. As discussed in section 3.2, norm-conserving pseudopotentials, by design, tend to exhibit good transferability. However, due to numerical approximations and the methods used to generate the pseudopotential, this transferability may not always be perfectly conserved.

To evaluate transferability, we change the valence electron configuration to occupy different orbitals. For example, in the case of arsenic (As), the valence electron configuration is $4s^2 4p^3$. The test consists of modifying this configuration, such as moving one electron from the s state to the p state, resulting in $4s^1 4p^4$, and then comparing the energy calculated from the pseudopotential with that from the all-electron method for this new configuration. Ideally, the energy difference between the two methods would be zero, but small deviations of a few mRy are acceptable due to the approximations used in the pseudopotential construction.

To carry out this calculation and comparison in `ATOM`, both the all-electron and pseudopotential energies must be evaluated for each test configuration. The number of configurations tested is determined by the user.

In `ATOM`, two files are required: the input file (INP) and the pseudopotential file in VPS format (renamed to VPSIN). The INP file contains the same input parameters as those used for the all-electron calculation, but with additional lines specifying the test configurations. In the first section of the INP file, the `all-electron` flag (ae) is used, followed by the same configurations in the second part using the `pseudopotential test` flag (pt). The input flags are further discussed in Section 3.3.1. For example, in the case of arsenic (As), the procedure would be as follows,

```
# All electrons configuration - Part I
   ae    GS- 4s2 2p3
   As    pbr
         0.0
    6     2
    4     0       1.00       1.00
    4     1       2.00       1.00
#234567890123456789012345678901234567890
   ae    - 4s2 4p2 4d1
   As    pbr
         0.0
    6     3
    4     0       1.00       1.00
    4     1       1.00       1.00
    4     2       1.00       0.00
```

```
#234567890123456789012345678901234567890
   ae    - 4s1 2p2 4d2
   As    pbr
         0.0
    6     3
    4     0       1.00       0.00
    4     1       1.00       1.00
    4     2       1.00       1.00
#234567890123456789012345678901234567890
   ae    - 4s1 2p4
   As    pbr
         0.0
    6     2
    4     0       1.00       0.00
    4     1       2.00       2.00
#234567890123456789012345678901234567890
   ae    - 4s0 2p5
   As    pbr
         0.0
    6     2
    4     0       0.00       0.00
    4     1       2.00       3.00
#234567890123456789012345678901234567890i
#
# All electrons configuration for pseudopotential - Part II
#
   pt    - GS 4s2 2p3
   As    pbr
         0.0
    6     2
    4     0       1.00       1.00
    4     1       2.00       1.00
#234567890123456789012345678901234567890
   pt    - 4s2 4p2 4d1
   As    pbr
         0.0
    6     3
    4     0       1.00       1.00
    4     1       1.00       1.00
    4     2       1.00       0.00
#234567890123456789012345678901234567890
   pt    - 4s1 2p2 4d2
   As    pbr
         0.0
    6     3
    4     0       1.00       0.00
    4     1       1.00       1.00
    4     2       1.00       1.00
#234567890123456789012345678901234567890
```

```
   pt    - 4s1 2p4
   As    pbr
         0.0
    6     2
    4     0       1.00       0.00
    4     1       2.00       2.00
#234567890123456789012345678901234567890
   pt    - 4s0 2p5
   As    pbr
         0.0
    6     2
    4     0       0.00       0.00
    4     1       2.00       3.00
#234567890123456789012345678901234567890i
```

Listing 3.3: Example of INP for pseudopotential input

In this section, we test 5 different electronic configurations. After running the `ATOM` program, the comparison files are generated. However, the key results are found in the output file (`OUT`). To extract the relevant data from the output file, you can use the following shell command:

```
# grep "\&d" OUT
```

This command will display the comparison table, as shown in the following example,

```
ATM4.2.7  17-SEP-24      GS- 4s2 2p3     &v&d
ATM4.2.7  17-SEP-24      - 4s2 4p2 4d1   &v&d
ATM4.2.7  17-SEP-24      - 4s1 2p2 4d2   &v&d
ATM4.2.7  17-SEP-24      - 4s1 2p4       &v&d
 ATM4.2.7  17-SEP-24      - 4s0 2p5      &v&d
 &d total energy differences in series
 &d           1         2         3         4         5
 &d  1    0.0000
 &d  2    0.5380    0.0000
 &d  3    2.0198    1.4818    0.0000
 &d  4    0.6893    0.1513   -1.3305    0.0000
 &d  5    1.3948    0.8568   -0.6250    0.7055    0.0000

*----- End of series ----* spdfg &d&v
 ATM4.2.7  17-SEP-24      - GS 4s2 2p3         &v&d
 ATM4.2.7  17-SEP-24      - 4s2 4p2 4d1        &v&d
 ATM4.2.7  17-SEP-24      - 4s1 2p2 4d2        &v&d
 ATM4.2.7  17-SEP-24      - 4s1 2p4            &v&d
 ATM4.2.7  17-SEP-24      - 4s0 2p5            &v&d

 &d total energy differences in series
 &d           1         2         3         4         5
 &d  1    0.0000
```

```
 &d  2    0.5344   0.0000
 &d  3    1.9936   1.4592   0.0000
 &d  4    0.6885   0.1541  -1.3051   0.0000
 &d  5    1.3909   0.8564  -0.6028   0.7023   0.0000
*----- End of series ----* spdfg &d&v
```

Listing 3.4: Example of INP for pseudopotential test results

The energies are expressed in Rydbergs (Ry). The tables present the total energy differences between electronic configurations. In the first table (upper), the energies are calculated using the all-electron method, while in the second table (lower), the energies are computed using the pseudopotential method. The test consists of comparing the energy differences between the all-electron and pseudopotential calculations. Ideally, if the pseudopotential is perfectly transferable, the difference would be zero.

For example, the total energy difference E_{ae}^{1-2} between configuration 1 ($4s^2\ 4p^3$) and configuration 2 ($4s^2\ 4p^2\ 4d^1$) is $E_{\text{ae}}^{1-2} = 0.5380$ Ry for the all-electron calculation (first table), while the difference calculated using the pseudopotential (second table) is $E_{\text{ps}}^{1-2} = 0.5344$ Ry. Hence, the difference between the all-electron and pseudopotential calculations is $E_{\text{ae}}^{1-2} - E_{\text{ps}}^{1-2} = 0.003$ Ry, or 3 mRy. Ideally, this difference should be as small as possible. In our case, the maximum energy difference between configurations is 22 mRy, which is considered acceptable for practical purposes. The smaller this value, the better the pseudopotential's transferability.

Beyond transferability tests and the comparison of atomic wavefunctions and charge densities, a pseudopotential is generally considered good if the physical properties it predicts align closely with experimental data. Therefore, it is crucial to test the pseudopotential by calculating physical properties such as bond lengths, lattice parameters, bulk modulus, band structures, and molecular energy levels, and comparing these results to experimental values. Once a pseudopotential passes these tests, it can be reliably used for further research and simulations.

IMPORTANT : TEST YOUR PSEUDOPOTENTIAL

3.4 It's your turn!

1. Install the ATOM code in your pc.

2. Compare the orbitals and charge density using all electron mode for each atom Li, O, Si, Cu and Pb , using the PBE exchange-correlation

3. Using all electron mode, for Pb, compare the effect of exchange-correlation in the wavefunction using: LDA (ca), PBE (pb), PBEsol (ps), BLYP (bl), Van der Waals (vw).

4. Create a pseudopotential for O atom and compare the wavefunction between all electron, Hamann-Schluter-Chiang, Kerker and Improved Troullier-Martins with relativistic PBE.

5. Create a pseudopotential for Pb considering with and without 5d10 orbital as valence electron, using relativistic PBE and Kerker method. Compare the pseudowavefunctions with all electron. Use the appropriated r_c with your choose.

6. Create a pseudopotential for Bi, using PBE with spin-polarization and Troullier-Martins method and made a transferability test of your pseudopotential. Choose your r_c

7. **Now we generated a really pseudopotential! For Si :**

   - **calculate all electron using PBE and Kerker and plot the wavefunction and determine the medium values for r_c.**
   - **Create a pseudopotential using the determines r_c, PBE and Kerker method. Compare the wavefunction and charge density.**
   - **Made a transferability test and analysis which the pseudopotential obey the transferability. If it is not satisfied, change your r_c. Recalculate the pseudopotential up to obtain the reasonable transferability in pseudopotential.**

8. Repeat the before exercise for Ga, As, Pb and O. Hint: Create a shell script to automatized the calculations.

Capítulo 4

Getting Started: Installing SIESTA.

4.1 SIESTA's Install

The installation of SIESTA can be perceived as complex by some users; however, this is a misconception. It is important to have a minimal understanding of compilers and libraries. Additionally, hardware considerations are essential to achieve maximum efficiency in your calculations. Information regarding compiler flags and the necessary libraries can be found in the SIESTA User's Guide. These libraries are crucial for optimizing the installation, especially on clusters with many processors or for simulating large systems containing thousands of atoms. In this section, we describe a non-traditional installation method using a Conda environment with binary files, eliminating the need to compile the source code. While this installation may not be fully optimized, it allows for the easy installation of the parallel version, and we can be supply our proposed.

A Conda package is a distribution format used by to manage software packages and their dependencies in a consistent and efficient manner across various environments. It simplifies the installation and management of software, especially in data science and scientific computing, by allowing users to easily create isolated environments with specific versions of packages. Each Conda package contains all the necessary files, libraries, and metadata required for installation, ensuring compatibility and reducing the risk of conflicts with other software. With its ability to handle both Python and non-Python packages, Conda streamlines workflows for users.

If you have Conda installed on your PC, the following steps are unnecessary. To proceed, we will install a minimal Conda package called Miniforge,

assuming that the operating system in use is Linux. Open a terminal and follow these steps to download and install the Miniforge package:

```
$ curl -L -O "https://github.com/conda-forge/miniforge/\
releases/latest/download/Miniforge3-$(uname)-$(uname -m).sh"
   | bash Miniforge3-$(uname)-$(uname -m).sh
```

Listing 4.1: "Conda Mini-forge install"

Accept the terms and choose your preferred installation path. Once the Conda installation is complete, open a new terminal. It is recommended to create a new environment for SIESTA. This action ensures that the installed packages will not conflict with other packages in the system, which is a significant advantage of using Conda. To create a new Conda environment for SIESTA, open a terminal,

```
$ conda create -n siesta
$ conda activate siesta
```

Listing 4.2: "Create an environment called siesta"

After creating the SIESTA environment, activate it with the second command. It is important to activate the siesta environment every time you intend to use SIESTA. Now, you can install the parallel version of the SIESTA binary using Conda. Ensure that your terminal is activated in the siesta environment, then execute the following command:

```
$ conda install -c conda-forge "siesta=*=*openmpi*"
```

Listing 4.3: "install SIESTA with openmpi"

In addition to the SIESTA installation, you may require some additional libraries or software commonly used in conjunction with it. Once the installation is complete, SIESTA should be ready for use. To run SIESTA using MPI (Message Passing Interface), we suggest using the following command,

```
$ mpirun -np <number_of_processes> siesta < calc.fdf
```

Listing 4.4: "Running SIESTA with OpenMPI"

Replace `<number_of_processes>` with the desired number of processes for your calculation. This command will allow you to efficiently execute SIESTA calculations in parallel. The calc.fdf is the input file of the SIESTA, which will be described in the next section.

4.2 How to execute the SIESTA and Specific execute options

SIESTA relies on some input files. The first is the .fdf file format, where the simulation parameters and flags are defined. This file will be discussed in detail in the next chapter. Another essential file contain the the pseudo-potential, which can be generated as described in Chapter XXXX and will also be elaborated on in the next chapter. With these two files, one can, in principle, test the SIESTA installation.SIESTA can be executed in two modes: i) serial,

```
$ siesta < calc.fdf > calc.out
```

or ii) paralell,

```
$ mpirun -np <number_of_processes> siesta < calc.fdf > calc.
   out
```

Additionally, there are alternative methods to run the program, depending on the computational resources and user preferences.

```
$ mpirun -np <number_of_processes> siesta < calc.fdf > calc.
   out
```

```
$ mpirun -np <number_of_processes> siesta calc.fdf > calc.out
```

```
$ mpirun -np <number_of_processes> siesta calc.fdf --out calc
   .out
```

The calc.out file contains the output of the calculation. The second execution method is available for SIESTA versions > 4.1, while the third option involves using specific flags. If the installation is successful, several output files will be created, and calc.out should conclude with the message "job completed."You can test your installation using the files located in Chapter-5/test_installation within the Supplementary Materials.

In versions up to SIESTA 4.1, special execution flags are acceptable. All flags must begin with either a single hyphen (-) or a double hyphen (–).

- **–help:** Provides information on how to use the software. Example: siesta –help
- **–version:** Displays version information.
  Example: 'siesta –version'
- **–out:** Specifies the output file.
  Example: 'siesta calc.fdf –out calc.out'

- **–fdf:** Specifies any FDF flag directly in the ‘.fdf‘ file.
  Example: ‘siesta calc.fdf –fdf ”Mesh.Cutoff 600 Ry”‘

These flags can be useful for temporary adjustments or for testing cases where changing a single parameter is needed without rewriting the input .fdf file. You can test these flags using the files provided in Chapter-5/test_install.

4.3 It’s Your Turn!

1. Install SIESTA using a Conda environment, as described in Section 4.1.
2. Build the serial version of SIESTA by compiling from the source code.
   Tip: You can refer to the official manual at (https://docs.siesta-project.org/projects/siesta/en/latest/installation/build-manually.html) for detailed instructions. Ensure that all dependency packages are resolved.
3. (Challenge) Compile the parallel version of SIESTA.

Capítulo 5

Main Input Parameters in SIESTA

To run a SIESTA simulation, at least two files are required. The first is the input file in `.fdf` format, and the second is the pseudopotential file. The formats of both files will be described in detail in the following sections.

5.1 Pseudopotential Format in SIESTA

As discussed in the previous chapter, SIESTA uses norm-conserving pseudopotentials. Unlike plane-wave codes, in SIESTA, pseudopotentials are used to create an atomic basis set, where the core radial wavefunction is replaced by the pseudowavefunction. Therefore, pseudopotentials are crucial for SIESTA to determine the basis set and must be supplied by the user. In Section XXXX, we learned how to create a norm-conserving pseudopotential. Here, we will discuss how to provide this pseudopotential to SIESTA.

SIESTA can use different files, one or more for each unique chemical species, depending on the user's setup. For example, in a graphene system with two carbon atoms, both can share the same pseudopotential file, or each carbon atom can have its own pseudopotential. We will delve into these details in the subsequent sections.

SIESTA accepts the following formats for pseudopotential files:

- `.vps` (unformatted)
- `.psf` (ASCII)
- `.psml` (PSML format)

By default, files are searched in the current directory, unless a path is specified in the `%ChemicalSpeciesLabel` flag. The pseudopotential file must have the same name as specified in the `%ChemicalSpeciesLabel`. SIESTA follows these rules when selecting pseudopotential files:

- If there are multiple pseudopotentials with the same name (e.g., `C.vps`, `C.psf`), and the extension is not specified, SIESTA will use them in this order of precedence: `.vps`, `.psf`, `.psml`.
- If an absolute path is provided (e.g., `/home/pseudopotentials/C.psf`), the search will only be conducted at that path.

Note that the `ATOM` program can only generate `.vps` pseudopotentials. For `.psf` and `.psml` formats, third-party pseudopotential software is required, which is beyond the scope of this text but can be found on platforms like GitHub. Pseudopotentials can also be sourced from databases such as the Pseudo-Dojo project, but it is essential to test these pseudopotentials to ensure high-quality calculations. For more information on the PSML format, refer to the SIESTA manual.

SIESTA also provides a script called `Gen-basis` in the `Util` folder of the compiled source version. This script allows users to generate basis sets for later use. The folder includes a tutorial on how to use the script, though it is limited to single atomic species. Here, we focus on the automatic basis generation by SIESTA using the provided pseudopotentials.

5.2 The FDF Input File Format

The Flexible Data Format (FDF) was developed by the SIESTA group to define the parameters of simulations and their physical units. This format allows parameters to be specified in *any order* and supports *various physical units*. In FDF, data labels are followed by their corresponding values, and default values are used for any parameters not explicitly specified. The main characteristics of the FDF format are as follows:

- The # character is used to comment out a line.
- FDF labels are case-insensitive.
  Example: `Mesh.Cutoff` is equivalent to `mesh.cutOff`
- Characters such as `-`, `_`, and `.` are ignored.
  Example: `Mesh.Cutoff` is the same as `Mesh_Cut-off`

- Logical values can be specified as `T`, `true`, `.true.`, `yes`, `F`, `false`, `.false.`, or `no`. If a variable is specified without a value, it is considered `true` by default.
- Real values require a decimal point to distinguish them from integers.
- Values representing physical quantities must be followed by their units. See Appendix XXX for supported units. Example:

```
LatticeConstant 10 Ang
```

- Complex data structures are represented using blocks.
  Example:

```
%block ChemicalSpeciesLabel
  1   1  H
%endblock ChemicalSpeciesLabel
```

- Flags can be included from another FDF file. Example:

```
%include cutoff.fdf
```

- If a label is misspelled, refer to the `fdf.log` file, which contains all parameters used by SIESTA in a given run.

Below is an example of an FDF file for simulating an H_2 molecule, where the flag for mesh cutoff energy is read from the `cutoff.fdf` file.

```
#### SYSTEM
SystemLabel  H2
SystemName   H2 Molecule

###### ATOMS
NumberOfSpecies 1
NumberOfAtoms 2

%block ChemicalSpeciesLabel
  1   1  H
%endblock ChemicalSpeciesLabel

###### Basis Sets
PAO.BasisSize DZP
PAO.EnergyShift 0.01 Ry

### Structure
```

```
LatticeConstant 10 Ang
AtomicCoordinatesFormat Ang

%block LatticeVectors
1.00            0.00    0.00
0.00            1.00    0.00
0.00            0.00    1.00
%endblock LatticeVectors

%block AtomicCoordinatesAndAtomicSpecies
   5.00  5.00   5.25 1
   5.00  5.00   4.75 1
%endblock AtomicCoordinatesAndAtomicSpecies

###### Exchange-Correlation Functional
XC.Functional LDA
XC.Authors PW92

###### Spin Polarization
Spin non-polarized

###### Self-consistent-field
MaxSCFIterations  300

###### Real-Space and Mesh
%include cutoff.fdf
```

Do not worry about understanding all the parameters now; we will describe the main parameters in the next sections.

5.3 Main Parameter Flags

SIESTA requires only two parameters to run:
(i) `%block ChemicalSpeciesLabel`,
where the atomic species are specified, and
(ii) `%block AtomicCoordinatesAndAtomicSpecies`,
where the atomic positions are defined.
All other parameters can use their default values, although this is generally not recommended. It is important to control various parameters to ensure accurate and optimized simulations. In this section, we describe the most important parameters to consider. For clarity, we will use the following format to describe each parameter:

- **Parameter** <*format*> Default value: description

Determining the System and Basis Set

- **SystemLabel** < *string*> siesta : A single word to represent the system, used in output files. The maximum length is 20 characters.
- **SystemName** < *string*> (no default value) : A brief description of the system, with a maximum of 150 characters.
- **NumberOfSpecies** < *integer*> : The number of lines in the `ChemicalSpeciesLabel` block. This indicates the number of different species or atoms that require different pseudopotentials or basis sets. Although not required, this parameter serves as an important check on the number of atom types used.
- **NumberOfAtoms** < *integer*> : The number of lines in the `AtomicCoordinatesAndAtomicSpecies` block. This represents the total number of atoms in the simulation. Although not required, this parameter serves as a useful check on the total number of atoms.
- **%block ChemicalSpeciesLabel** < *block*> No default value: This block specifies the different chemical species or different pseudopotentials used in the simulation. SIESTA distinguishes atoms by atomic number, for example:

```
%block ChemicalSpeciesLabel
  1   1  H
  2   6  C_bond   LDA/C.lda.vps
  3   6  C
%endblock ChemicalSpeciesLabel
```

  The first column is a unique number identifying the chemical species. This number will be used to build the structure by associating it with the atomic positions. The second column specifies the atomic number, e.g., carbon has atomic number 6. The third column defines a label for the atom, which can be used to identify the set of atoms. In the fourth column, the pseudopotential can be specified. If left blank, the pseudopotential is assumed to have the same name as the label (e.g., `C_bond.vps`). Additionally, the pseudopotential can be located in a specific folder, as shown in the example. For ghost atoms, a negative atomic number should be used.

- **PAO.BasisSize** < *string*> DZP : Defines the size of the basis set, as discussed in Section XXXX. Possible values include:

  - SZ (Single Zeta - minimum basis)
  - DZ (Double Zeta)
  - SZP (Single Zeta Polarized)
  - DZP (Double Zeta Polarized)

- **PAO.EnergyShift** < *float*> 0.01 Ry : Controls the orbital cutoff radii, as discussed in Section XXX.

Atomic Structure

The main flags to define the atomic structure are: `LatticeConstant`, `LatticeParameters`, `LatticeVectors`, and `SuperCell`. If none of these flags are set, SIESTA will construct a cubic cell using the atoms specified in the `%block ChemicalSpeciesLabel` and their positions from the `%block AtomicCoordinatesAndAtomicSpecies`, effectively modeling an atomic cluster.

- **LatticeConstant** < *float*> 1 Å : Specifies the lattice constant. Other units such as bohr can be used by appending the unit to the number.
- **%block LatticeVectors** < *block*> None : Specifies the lattice vectors in units of the lattice constant. Each vector should be on a separate line, as shown below:

```
%block LatticeVectors
  1.0   0.0   0.0
  0.0   1.0   0.0
  0.0   0.0   1.0
%endblock LatticeVectors
```

- **%block AtomicCoordinatesAndAtomicSpecies** < *block*> None : Specifies the position of each atom in the structure. The first three columns represent the x, y, and z coordinates of the atom according to the format chosen in `AtomicCoordinatesFormat`, i.e., projected in Cartesian axes or in lattice vectors with their respective units. The fourth column is the atom identifier, which corresponds to the label specified in `ChemicalSpeciesLabel`. This label determines which atom occupies the specified position. For example:

```
%block AtomicCoordinatesAndAtomicSpecies
    5.00 5.00 6.00 1
    5.00 5.00 4.00 1
    3.50 3.75 5.25 2
%endblock AtomicCoordinatesAndAtomicSpecies
```

This example represents two identical atoms with label 1 at positions (5, 5, 6) and (5, 5, 4), respectively, and one atom with label 2 at position (3.50, 3.75, 5.25).

- **%block SuperCell** < *block*> None : Specifies a 3×3 matrix that defines the supercell in terms of the unit cell. Any value greater than 1 in the matrix expands the unit cell in the corresponding direction. For example, to expand the unit cell by 2 units in both the x and y directions:

```
%block SuperCell
  2.0   0.0   0.0
  0.0   2.0   0.0
  0.0   0.0   1.0
%endblock SuperCell
```

  Important,this will create 4 unit cells. Use with caution as the system size can grow quickly. Warnings:

  - Symmetry from the original unit cell might be lost, impacting calculation efficiency.
  - The `NumberOfAtoms` flag refers only to the unit cell.
  - All other input parameters apply to the supercell, including `Kgrid.MonkhorstPack`.

  Tip: Use visualization software to verify the correct creation of the supercell.

- **AtomicCoordinatesFormat** < *string*> Bohr : Specifies the format for atomic positions (unit and scale). Options include:

  - **Bohr**: Cartesian coordinates in Bohr units.
  - **Ang**: Cartesian coordinates in Angstroms.
  - **LatticeConstant**: Cartesian coordinates in units of the lattice constant.
  - **Fractional**: Atomic positions given as projections on the lattice vectors.

Convergence and Self-Consistent Cycle

- **%block Kgrid.MonkhorstPack** < *block*> Γ : Specifies the k-point grid for the self-consistent cycle using the Monkhorst-Pack scheme. It

is defined using a 3×4 matrix , where the first three columns represent the discretization grid and the last column indicates the displacement of the origin, depending on symmetry. For a $4 \times 4 \times 1$ grid with an origin displacement of 0.5:

```
%block Kgrid.MonkhorstPack
    4   0   0   0.5
    0   4   0   0.5
    0   0   1   0.5
%endblock Kgrid.MonkhorstPack
```

SIESTA will optimize the grid based on symmetry and warn if the displacement is non-optimal. An alternative, simpler k-point specification is described in the following parameter.

- **Kgrid.MonkhorstPack** [**M N P**] < *list*> None : Specifies a simplified k-point grid using only diagonal elements, i.e., discretization in the M $\hat{x}$, N $\hat{y}$, and P $\hat{z}$ directions without displacement.

- **XC.Functional** < *string*> LDA : Defines the exchange-correlation functional class. Options are:

  - LDA (Local Density Approximation)
  - GGA (Generalized Gradient Approximation)
  - VDW (van der Waals)

- **XC.Authors** < *string*> PZ : Specifies the formulation of the exchange-correlation functional (e.g., PBE). The formulation must match the functional class defined by `XC.Functional`. See Appendix XXX or the SIESTA manual for available options.

- **Spin** < *string*> non-polarized : Defines spin treatment in the calculation. Options are:

  - **non-polarized**: Treats spin with degeneracy.
  - **polarized**: Considers collinear spin with 2 components.
  - **non-collinear**: Considers non-collinear spin with 4 components.
  - **spin-orbit**: Includes spin-orbit coupling (requires a relativistic pseudopotential).

- **Mesh.Cutoff** < *float*> 300 Ry : Some integrals are more easily computed in reciprocal space, which can be evaluated using Fourier transforms. The `Mesh.Cutoff` parameter specifies the cutoff for the kinetic energy of plane waves used in constructing the 3D integration mesh. Setting this parameter determines the precision of the mesh and affects the accuracy of the integrals. Proper convergence testing for `Mesh.Cutoff` is essential to ensure reliable results while balancing computational cost.

- **MaxSCFIterations** < *integer*> 100 : The maximum number of SCF iterations allowed for convergence. For molecular dynamics, this applies to each time step.

5.3.1 Output and Input Structural Information

- **DM.UseSaveDM** < *logical*> true : Use a density matrix stored in `System.DM` calculated in a previous run and use this matrix to start the calculation.

- **Write.DM** < *logical*> true : Save the density matrix in a file. This file can be used to continue the calculation or for post-processing purposes.

Note: These flags are considered some of the most relevant for basic SIESTA calculations. Naturally, there are many additional flags that control various approximations and numerical details, which can be found in the SIESTA manual. Main flags specific to different types of calculations, such as relaxation, electronic structure, and optical properties, will be analyzed in their respective chapters. To conclude this chapter, we present an example of an FDF file containing the primary flags for a H_2 molecule calculation.

```
#### SYSTEM
SystemLabel  H2
SystemName   H2 Molecule

###### ATOMS
NumberOfSpecies 1
NumberOfAtoms   2

%block ChemicalSpeciesLabel
  1   1  H
%endblock ChemicalSpeciesLabel

###### Basis Sets
 PAO.BasisSize      SZP
 PAO.EnergyShift    0.01 Ry
```

```
### Structure
LatticeConstant          10 Ang
AtomicCoordinatesFormat  Ang

%block LatticeVectors
1.00            0.00    0.00
0.00            1.00    0.00
0.00            0.00    1.00
%endblock LatticeVectors

%block AtomicCoordinatesAndAtomicSpecies
   5.00   5.00    6.00 1
   5.00   5.00    4.00 1
%endblock AtomicCoordinatesAndAtomicSpecies

% block SuperCell
  1.0 0.0 0.0
  0.0 1.0 0.0
  0.0 0.0 1.0
% endblock SuperCell

###### K-Point Sampling
% block Kgrid.MonkhorstPack
  1.0 0.0 0.0 0
  0.0 1.0 0.0 0
  0.0 0.0 1.0 0
% endblock Kgrid.MonkhorstPack

###### Exchange-Correlation Functional
XC.Functional LDA
XC.Authors    PW92

###### Spin Polarization
Spin non-polarized

###### Self-consistent-field
MaxSCFIterations  300

###### Real-Space and Planewaves
Mesh.Cutoff 300 Ry

#### OUTPUT FILES
DM.UseSaveDM true
Write.DM     true
```

5.4 Output Files in SIESTA

SIESTA produces various output files that are essential for analyzing the results and understanding the properties of the simulated system. Below, we provide an overview of the key output files and their purposes:

- **Standard Output (stdout)**:
  - **Format**: Plain text printed in the terminal or redirected to a file.
  - **Contents**: This output shows real-time information on the simulation's progress, including initial settings, SCF cycle convergence, warnings, and final results. Monitoring this output ensures that the simulation runs correctly and helps identify potential issues.
- **SystemLabel.ion.xml**:
  - **Format**: XML
  - **Contents**: Contains ion configuration and structural data. This file is used for restarting or continuing simulations and for post-processing.
- **SystemLabel.DM**:
  - **Format**: Binary
  - **Contents**: The stored density matrix from the calculation. If the `Write.DM` flag is set to true, this file is generated and can be reused with `DM.UseSaveDM` for subsequent calculations.
- **SystemLabel.WFSX and SystemLabel.PDOS**:
  - **SystemLabel.WFSX**: Stores the wavefunctions for post-processing to extract electronic properties or visualize molecular orbitals.
  - **SystemLabel.PDOS**: Provides the projected density of states (PDOS), detailing the contributions of different atoms or orbitals.
- **SystemLabel.XV**:
  - **Format**: Plain text
  - **Contents**: Includes the final atomic coordinates and cell vectors after relaxation or molecular dynamics. Useful for verifying the relaxed structure of the system.
- **SystemLabel.FA**:

  - **Description**: Contains the forces acting on the atoms at the last step of the calculation, helping to confirm convergence in geometry optimization.

- **SystemLabel.CHR**:
  - **Format**: Binary
  - **Contents**: Represents the total charge density of the system, used for further analysis such as plotting electron density maps.

- **SystemLabel.VH**:
  - **Format**: Binary
  - **Contents**: Stores the Hartree potential on the grid for potential energy analysis and visualization.

- **SystemLabel.bands**:
  - **Contents**: Provides band structure data if a band structure calculation is performed. This file is typically processed for visualizing the electronic band structure.

- **SystemLabel.EIG**:
  - **Format**: Plain text
  - **Contents**: Lists eigenvalues for each k-point, crucial for electronic structure analysis.

- **SystemLabel.DOS**:
  - **Contents**: Contains the density of states (DOS), providing insight into the distribution of electronic states and identifying band gaps.

- **Log File (siesta.log)**:
  - **Contents**: Lists all parameters read from the input file, showing any default values explicitly. Useful for verifying parameter settings.

- **SystemLabel.MD (For Molecular Dynamics)**:
  - **Contents**: Records the evolution of atomic positions, velocities, and other properties during molecular dynamics simulations. Essential for analyzing system behavior over time.

Understanding and utilizing these output files is crucial for a comprehensive analysis of SIESTA simulations, aiding in the interpretation of structural, electronic, and energetic properties. These files will be described in detail in the corresponding chapters dedicated to each specific type of calculation.

5.5 It's your turn!

1. Create a fdf file for a diamond structure with carbon.
2. Create a fdf file for graphene.
3. See in the SIESTA manual other flags and make a list with more interesting ones that are not present here. List at least 10 flags.

Capítulo 6

First Steps: Obtaining the Molecular Properties

6.1 H_2 Molecule: A Simple Example

In this section, we discuss how to calculate molecular properties using SIESTA. Initially, we will not focus on convergence or optimization of the calculation parameters. Our goal is simply to run SIESTA and calculate:

- (Pseudo) Total Energy
- Bond Length of the Molecule

Further exploration of molecular property calculations will be covered in subsequent sections.

For the H_2 molecule, we perform a basic calculation using the LDA exchange-correlation functional as parameterized by Perdew and Wang (PW92), without spin polarization. The key input parameters are:

- `PAO.BasisSize`: DZP
- `MeshCutoff`: 300 Ry
- `XC.Functional`: LDA
- `XC.Authors`: PW92
- `SpinPolarized`: `False`

First, as SIESTA employs periodic boundary conditions, which repeat the simulation cell indefinitely, it is necessary to create a sufficiently large

simulation box to isolate the molecule and prevent interaction with its periodic images. In this example, we use a cubic box of 10 Å per side, defined in the lattice vectors. An in-depth discussion on box size will follow in later sections.

6.1.1 Example FDF File for H_2

Below is an example FDF file for the H_2 molecule:

```
#### SYSTEM
SystemLabel  H2
SystemName   H2 Molecule

###### ATOMS
NumberOfSpecies 1
NumberOfAtoms 2

%block ChemicalSpeciesLabel
  1   1  H
%endblock ChemicalSpeciesLabel

###### Basis Sets
PAO.BasisSize DZP
PAO.EnergyShift 0.01 Ry

### Structure
LatticeConstant 10 Ang
AtomicCoordinatesFormat Ang

%block LatticeVectors
1.00  0.00  0.00
0.00  1.00  0.00
0.00  0.00  1.00
%endblock LatticeVectors

%block AtomicCoordinatesAndAtomicSpecies
   5.00  5.00   4.75 1
   5.00  5.00   5.25 1
%endblock AtomicCoordinatesAndAtomicSpecies

###### Exchange-Correlation Functional
XC.Functional LDA
XC.Authors PW92

###### Spin Polarization
Spin non-polarized

###### Real-Space
```

```
MeshCutoff 300 Ry
```

Listing 6.1: Example FDF file for H_2

The FDF file defines the simulation parameters. The `SYSTEM` section specifies the system's label and name. The `ATOMS` section defines the species and number of atoms, while the `ChemicalSpeciesLabel` block maps atomic numbers to species. The `Basis Sets` section configures the basis set size and energy shift. The `STRUCTURE` section sets the lattice constant and atomic coordinates format, with the lattice vectors defining a cubic box. The `EXCHANGE-CORRELATION` section specifies the functional and its parameters. The `SPIN` and `REAL-SPACE` sections configure spin polarization and mesh cutoff, respectively. Full descriptions of these flags are provided in section XXXX.

Figura 6.1: Representation of the H_2 molecule aligned along the z-axis.

In the atomic positions specified in the FDF file, the initial distance between hydrogen atoms is 0.5 Å along the z-axis, as shown in Figure 6.1. To run SIESTA, ensure the pseudopotential file (e.g., `H.vps`) is in the same directory as the FDF file. The execution command depends on your specific installation, MPI setup, etc., as discussed previously. Successful completion will be indicated by the message `"Job Completed"`. Check the output file for any warnings or errors. The total energy can be found in the output file under:

```
siesta:  Final energy (eV): Total
```

To quickly extract the total energy, use the following terminal command:

```
$ grep "Total =" output.out
```

Listing 6.2: Command to extract total energy

For this setup, with an interatomic distance of 0.5 Å, the total energy is approximately −27.66 eV. Note that this value may vary slightly depending on the pseudopotential used, compilation flags, etc.

6.1.2 Method 1: Energy Curve Analysis

To find the bond length corresponding to the lowest energy, we plot the total energy as a function of interatomic distance. Vary the distance between 0.5 Å and 4.0 Å in steps of 0.1 Å and calculate the energy for each configuration. The plot of total energy versus distance is shown in Figure 6.2. The minimum energy, corresponding to the bond length, is found to be approximately 0.80 Å, close to the experimental value of 0.74 Å.

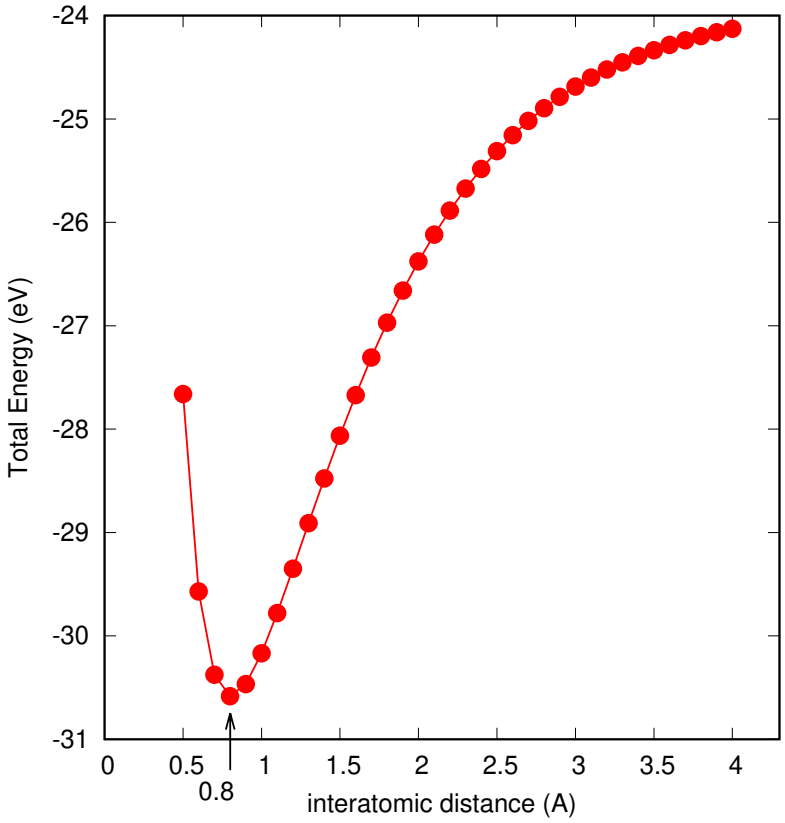

Figura 6.2: Total energy of the H_2 molecule as a function of interatomic distance.

6.1.3 Method 2: Structural Optimization

To automate the bond length determination, use SIESTA's structural optimization algorithm by adding the following flags to the FDF file:

```
###### Relaxation Structure
MD.Steps 50
MD.TypeOfRun CG
MD.MaxForceTol 0.001 eV/Ang
```

Listing 6.3: Flags for structure optimization

These flags specify the maximum number of optimization steps, the conjugate gradient method, and a force tolerance for convergence. After running the simulation, check the `H2.STRUCT_OUT` file for optimized atomic positions. The bond length, determined by the difference in z-coordinates, is approximately 0.804 Å, consistent with the result from Method 1.

6.2 The Box and Periodic Boundary Conditions

As mentioned earlier, SIESTA's use of periodic boundary conditions necessitates careful consideration of the simulation box size to prevent molecule-image interactions. Figure 6.3 illustrates the concept of periodic boundaries.

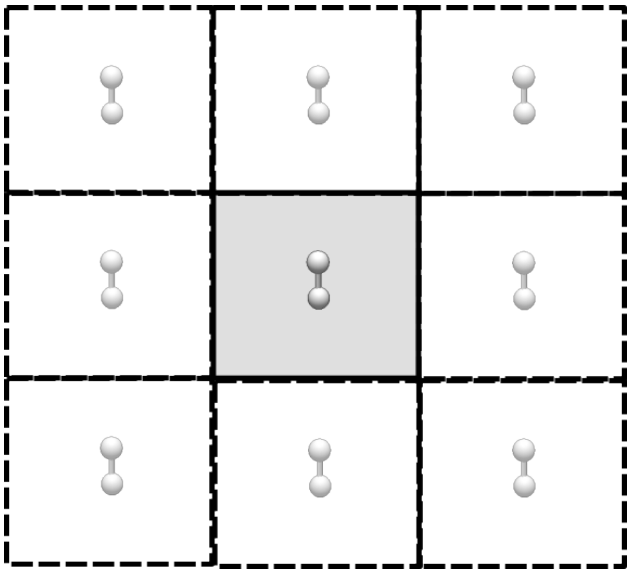

Figura 6.3: Illustration of a periodic system and its repeating cells.

To determine the appropriate box size, simulate the H_2 molecule with varying box sizes and compare the total energy. As shown in Figure 6.4, the energy converges to a constant value when the box is sufficiently large (typically above 6 Å for H_2).

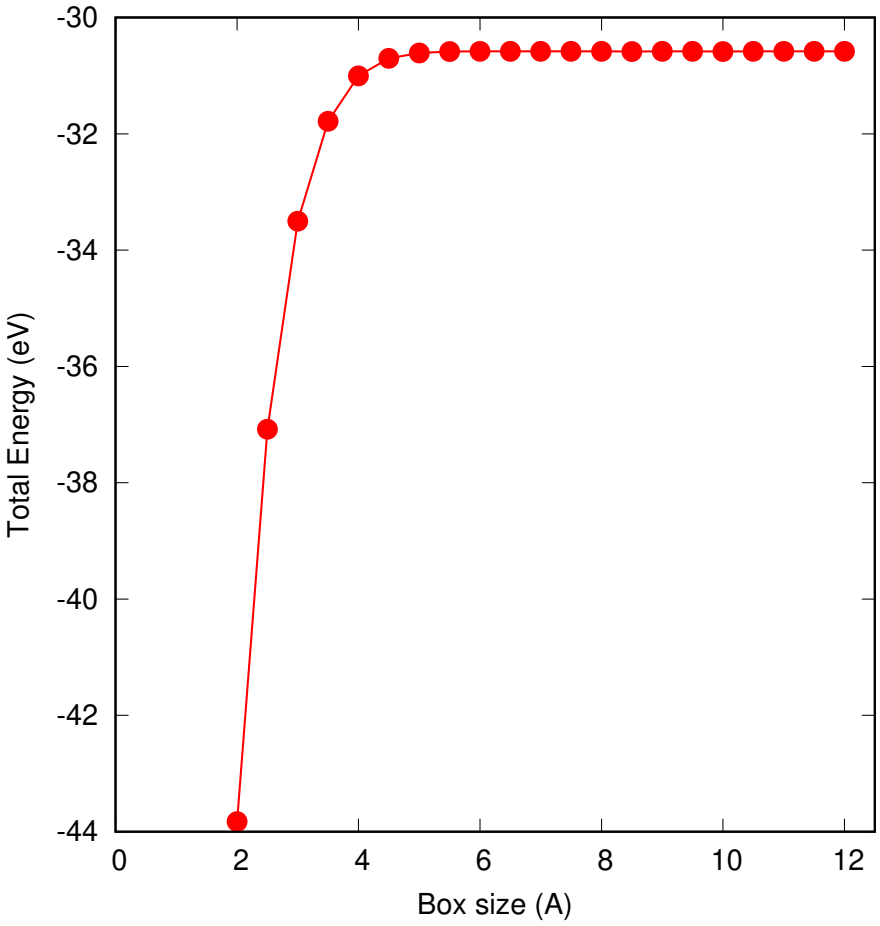

Figura 6.4: Total energy of the H_2 molecule as a function of box size.

6.3 It's Your Turn!

1. Simulate the CO molecule using the GGA exchange-correlation functional with PBE formulation and non-polarized spin.

   - Find the bond length by plotting the total energy as a function of atomic distance.
   - Calculate this distance using SIESTA's structure optimization algorithm.
   - Compare your results with experimental values.
   - Estimate the dissociation energy from the total energy curve and compare with experimental data.

2. Simulate the $\mathsf{H_2O}$ molecule and determine the $\mathsf{H-O}$ bond length and the $\mathsf{H-O-H}$ angle using the structure optimization algorithm. Compare with experimental results. (Use the exchange-correlation functional and parameters you deem appropriate.)

3. Determine the minimum box size required to simulate a $\mathsf{CH_4}$ molecule. (Use the exchange-correlation functional and parameters you deem appropriate.)

Capítulo 7

A Pedestrian's Guide to Solid State

7.1 Crystal Structure

The arrangement of atoms in a crystalline solid forms a periodic lattice structure, which is central to the study of solid-state physics. A crystal lattice is defined as a repeating array of points in space, where each point represents the position of an atom or a group of atoms. This periodicity underpins many of the unique properties of crystalline materials.

7.1.1 Lattice and Basis

The concepts of **lattice** and **basis** are foundational in describing crystal structures. A lattice provides the geometric framework, while the basis defines the atomic arrangement within this framework.

Lattice

A lattice is a three-dimensional periodic array of points in space, representing the translational symmetry of a crystal. Each lattice point corresponds to an identical environment in the crystal structure. The lattice is mathematically described by three primitive vectors, $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$, which define the unit cell. Any lattice point $\mathbf{R}$ can be expressed as:

$$\mathbf{R} = n_1\mathbf{a}_1 + n_2\mathbf{a}_2 + n_3\mathbf{a}_3,$$

where $n_1, n_2, n_3 \in Z$ are integers.

Dimensionality of the Lattice: The lattice can exist in various dimensions:

- **1D Lattice:** Points are arranged periodically along a single direction (e.g., a linear chain of atoms).
- **2D Lattice:** Points repeat periodically in two directions, forming planar structures (e.g., graphene or a square lattice).
- **3D Lattice:** Points repeat periodically in three dimensions, creating three-dimensional crystals (e.g., cubic or hexagonal lattices).

Example: Simple Cubic Lattice In a simple cubic lattice:

$$\mathbf{a}_1 = a\hat{x}, \quad \mathbf{a}_2 = a\hat{y}, \quad \mathbf{a}_3 = a\hat{z},$$

where a is the lattice constant. The lattice points form a regular grid of cubes with sides of length a.

Basis

The basis defines the atomic or molecular arrangement associated with each lattice point. By combining the lattice with the basis, the complete crystal structure is formed.

Atomic Basis: Each atom in the basis is positioned relative to a lattice point using fractional coordinates:

$$\mathbf{r}_{\text{basis}} = x\mathbf{a}_1 + y\mathbf{a}_2 + z\mathbf{a}_3,$$

where x, y, z are fractional values between 0 and 1.

Examples of Basis:

- **Single-Atom Basis:** In a simple cubic structure, each lattice point is associated with a single atom.
- **Multi-Atom Basis:** In more complex structures, such as NaCl (rock salt), the basis includes multiple atoms. For NaCl:
  - Na atom: $(0, 0, 0)$
  - Cl atom: $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$

Crystal Structure

The combination of lattice and basis creates the crystal structure. Mathematically, the positions of all atoms in a crystal can be written as:

$$\mathbf{R}_{\text{atom}} = \mathbf{R}_{\text{lattice}} + \mathbf{r}_{\text{basis}}.$$

The lattice captures the translational symmetry, while the basis accounts for the specific atomic or molecular arrangement. Together, they provide a complete description of the crystal structure, enabling the study of its electronic, optical, and mechanical properties.

7.1.2 Unit Cell and Symmetry

The unit cell and symmetry are fundamental concepts in the study of crystalline materials. They provide the foundation for understanding the periodic arrangement of atoms and the inherent properties of crystals.

Unit Cell

The **unit cell** is the smallest repeating structural unit of a crystal lattice that, through translational symmetry, generates the entire crystal structure. It is defined by the lattice vectors $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$.

Types of Unit Cells:

- **Primitive Unit Cell:** Contains exactly one lattice point. It is the simplest unit cell and fully captures the symmetry of the lattice.
- **Conventional Unit Cell:** A larger unit cell, often used to highlight the symmetry of the crystal system. It may contain multiple lattice points.

Volume of the Unit Cell: The volume of a unit cell in three dimensions is given by:

$$V_{\text{cell}} = |\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)|.$$

Fractional Coordinates: Atomic positions within the unit cell are typically described using fractional coordinates (x, y, z), which relate to the lattice vectors as:

$$\mathbf{r} = x\mathbf{a}_1 + y\mathbf{a}_2 + z\mathbf{a}_3, \quad 0 \leq x, y, z < 1.$$

Example: Simple Cubic Unit Cell For a simple cubic lattice, the unit cell is a cube with:

$$\mathbf{a}_1 = a\hat{x}, \quad \mathbf{a}_2 = a\hat{y}, \quad \mathbf{a}_3 = a\hat{z},$$

where a is the lattice constant. Each unit cell contains one lattice point and occupies a volume of a^3.

Symmetry in Crystals

Symmetry plays a critical role in determining the physical properties of crystals. It reflects the invariance of the crystal structure under certain operations.

Symmetry Operations:

- **Translational Symmetry:** The crystal structure remains unchanged when translated by a lattice vector:

$$\mathbf{R} = n_1\mathbf{a}_1 + n_2\mathbf{a}_2 + n_3\mathbf{a}_3, \quad n_1, n_2, n_3 \in Z.$$

- **Rotational Symmetry:** Invariance under rotation about an axis. Crystals exhibit specific rotational symmetries, such as 2-fold, 3-fold, 4-fold, and 6-fold axes.
- **Mirror Symmetry:** The structure remains unchanged when reflected across a mirror plane.
- **Inversion Symmetry:** The structure is unchanged under inversion through a point.
- **Glide Planes and Screw Axes:** These combine translation with reflection or rotation, respectively.

Space Groups: The combination of translational symmetry with other symmetry operations leads to 230 distinct space groups in three dimensions. These groups classify the symmetry properties of all possible crystalline structures.

Example: Symmetry in Cubic Crystals

Cubic crystals, such as those in the face-centered cubic (FCC) structure, exhibit:

- **4-fold Rotational Symmetry:** Along the cube's axes.
- **Mirror Planes:** Perpendicular to the cube faces.
- **Inversion Symmetry:** About the cube's center.

Impact of Symmetry: Symmetry affects various properties of crystals, such as:

- **Electronic Properties:** Determines band degeneracies and selection rules.
- **Optical Properties:** Governs phenomena like birefringence and optical activity.
- **Mechanical Properties:** Influences anisotropy in elasticity and thermal expansion.

7.1.3 Bravais Lattice

The **Bravais lattice** is a fundamental concept in crystallography, representing the distinct ways in which infinite arrays of points can be arranged periodically in space, such that the environment around each point is identical. These arrangements are critical for understanding crystal structures and their symmetry.

A Bravais lattice is defined as a set of discrete points generated by integer linear combinations of three independent primitive vectors, $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$:

$$\mathbf{R} = n_1\mathbf{a}_1 + n_2\mathbf{a}_2 + n_3\mathbf{a}_3,$$

where $n_1, n_2, n_3 \in Z$ (integers). These vectors define the periodicity and orientation of the lattice.

In three dimensions, there are **14 distinct Bravais lattices**, categorized into seven crystal systems based on their symmetry properties and unit cell geometry. These are:

1. **Cubic System**:
    - Simple Cubic (SC)

- Body-Centered Cubic (BCC)
- Face-Centered Cubic (FCC)

2. **Tetragonal System**:
   - Simple Tetragonal
   - Body-Centered Tetragonal
3. **Orthorhombic System**:
   - Simple Orthorhombic
   - Base-Centered Orthorhombic
   - Body-Centered Orthorhombic
   - Face-Centered Orthorhombic
4. **Hexagonal System**:
   - Simple Hexagonal
5. **Trigonal (or Rhombohedral) System**:
   - Simple Trigonal
6. **Monoclinic System**:
   - Simple Monoclinic
   - Base-Centered Monoclinic
7. **Triclinic System**:
   - Simple Triclinic

Each Bravais lattice is characterized by specific lattice parameters:

- **Edge Lengths:** a, b, c, corresponding to the magnitudes of the lattice vectors $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$.
- **Angles:** α, β, γ, the angles between the lattice vectors:

$$\alpha = \angle(\mathbf{a}_2, \mathbf{a}_3), \quad \beta = \angle(\mathbf{a}_3, \mathbf{a}_1), \quad \gamma = \angle(\mathbf{a}_1, \mathbf{a}_2).$$

The following table summarizes the relationship between crystal systems, lattice parameters, and Bravais lattices:

The 14 Bravais lattices are essential for:

Crystal System	Lattice Parameters	Bravais Lattices
Cubic	$a = b = c$, $\alpha = \beta = \gamma = 90°$	SC, BCC, FCC
Tetragonal	$a = b \neq c$, $\alpha = \beta = \gamma = 90°$	Simple, Body-Centered
Orthorhombic	$a \neq b \neq c$, $\alpha = \beta = \gamma = 90°$	Simple, Base-Centered, Body-Centered,
Hexagonal	$a = b \neq c$, $\alpha = \beta = 90°$, $\gamma = 120°$	Simple
Trigonal	$a = b = c$, $\alpha = \beta = \gamma \neq 90°$	Simple
Monoclinic	$a \neq b \neq c$, $\alpha = \gamma = 90° \neq \beta$	Simple, Base-Centered
Triclinic	$a \neq b \neq c$, $\alpha \neq \beta \neq \gamma \neq 90°$	Simple

Tabela 7.1: Crystal systems and their corresponding Bravais lattices.

- **Classifying Crystals:** They form the basis for describing all crystal structures.
- **Symmetry Analysis:** Bravais lattices encapsulate the symmetry operations permissible within a crystal system.
- **Material Properties:** Understanding the lattice type aids in predicting and explaining electronic, optical, and mechanical properties.

7.2 Reciprocal Space

Reciprocal space, also known as momentum space, is the Fourier transform counterpart of real space in crystallography. It provides a convenient framework for describing wave-like properties such as electron wavefunctions, phonons, and diffraction patterns. A reciprocal lattice is constructed from the real-space lattice vectors, defined as:

$$\mathbf{b}_1 = 2\pi \frac{\mathbf{a}_2 \times \mathbf{a}_3}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)}, \quad \mathbf{b}_2 = 2\pi \frac{\mathbf{a}_3 \times \mathbf{a}_1}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)}, \quad \mathbf{b}_3 = 2\pi \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)},$$

where $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ are the primitive vectors of the real-space lattice. The reciprocal lattice vectors $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$ form the basis of reciprocal space.

In reciprocal space, each point represents a plane wave characterized by a wavevector $\mathbf{k}$. Reciprocal lattice vectors, $\mathbf{G} = n_1\mathbf{b}_1 + n_2\mathbf{b}_2 + n_3\mathbf{b}_3$ ($n_1, n_2, n_3 \in Z$), correspond to the periodicities of the real-space lattice. The condition for constructive interference in diffraction, known as Bragg's law, can be expressed compactly in reciprocal space:

$$\mathbf{k}' - \mathbf{k} = \mathbf{G},$$

where $\mathbf{k}$ and $\mathbf{k}'$ are the incident and scattered wavevectors, and $\mathbf{G}$ is a reciprocal lattice vector.

7.2.1 Brillouin Zone

The Brillouin zone is the Wigner-Seitz cell of the reciprocal lattice. It is defined as the region of reciprocal space that contains all wavevectors $\mathbf{k}$ closer to a given reciprocal lattice point than to any other. This region is fundamental for analyzing periodic structures, as it represents the unique, irreducible part of the reciprocal lattice. To construct the Brillouin zone:

1. Identify the reciprocal lattice points.
2. Draw perpendicular bisectors (or planes in 3D) between a chosen reciprocal lattice point and its neighbors.
3. The enclosed region around the chosen point forms the Brillouin zone.

For example:

- In one dimension, the Brillouin zone is a line segment between $-\pi/a$ and π/a, where a is the lattice constant.
- In two dimensions, it is a hexagon for a hexagonal lattice.
- In three dimensions, it forms more complex polyhedra, such as cubes, dodecahedra, or truncated octahedra, depending on the crystal symmetry.

The Brillouin zone contains high-symmetry points and paths (e.g., Γ, X, M, R) that are crucial for understanding electronic band structures. These points are used as reference points for calculations and visualization of energy bands in reciprocal space. For periodic structures, $f(\mathbf{r})$ represents a periodic potential or electron density. By applying the Fourier transform, these periodic functions are expressed as a sum of plane waves:

$$f(\mathbf{r}) = \sum_{\mathbf{G}} f_{\mathbf{G}} e^{i\mathbf{G}\cdot\mathbf{r}},$$

where $\mathbf{G}$ are reciprocal lattice vectors, and $f_{\mathbf{G}}$ are Fourier coefficients. The Brillouin zone plays a central role in solid-state physics:

- **Electronic Properties:** The wavevector $\mathbf{k}$ within the Brillouin zone describes the behavior of electron states in a crystal. The band structure is computed over the Brillouin zone, capturing essential information about conduction, valence, and energy gaps.
- **Phonons:** The Brillouin zone is used to describe phonon dispersion, crucial for thermal and vibrational properties.

- **Optical and Transport Properties:** Many optical and transport phenomena, such as the density of states and effective mass of electrons, depend on the Brillouin zone geometry and band structure.
- **Symmetry Analysis:** The symmetry of the Brillouin zone corresponds to the underlying symmetry of the real-space crystal lattice, simplifying calculations and analysis.

The Brillouin zone, as a unique representation of reciprocal space, provides a framework for understanding the wave-like behavior of particles in periodic systems. Combined with Fourier analysis, it enables the description of complex physical phenomena, bridging real-space lattice properties with reciprocal-space representations.

7.3 Strain and Stress

Strain and stress are fundamental concepts in the study of materials science and solid mechanics. They describe how materials deform and respond to external forces, respectively. Understanding these properties is essential for characterizing the mechanical behavior of materials, from macroscopic scales down to the atomic level.

7.3.1 Stress

Stress (σ) is a measure of the internal force per unit area within a material that arises in response to external loading. It is a tensor quantity that describes both the magnitude and direction of forces acting on specific planes within the material.

Mathematically, stress is defined as:

$$\sigma = \frac{F}{A},$$

where:

- F is the applied force (in Newtons),
- A is the cross-sectional area over which the force is applied (in square meters).

Stress is expressed in Pascals (Pa), where $1\text{Pa} = 1\text{N}\,\text{m}^{-2}$. Stress can be categorized based on the nature of the applied forces,

- **Normal Stress:** Occurs when the force is perpendicular to the surface. It includes:
  - *Tensile Stress:* Pulls the material apart, increasing its length.
  - *Compressive Stress:* Pushes the material together, reducing its length.
- **Shear Stress:** Occurs when the force is parallel to the surface, causing layers of the material to slide relative to each other.

Stress in three dimensions is described by the stress tensor:

$$\sigma_{ij} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix},$$

where the diagonal components $(\sigma_{xx}, \sigma_{yy}, \sigma_{zz})$ represent normal stresses, and the off-diagonal components $(\sigma_{xy}, \sigma_{xz}, \sigma_{yz})$ represent shear stresses.

7.3.2 Strain

Strain (ε) quantifies the deformation of a material in response to stress. It is a dimensionless quantity that measures the relative change in shape or size of a material. Strain is defined as:

$$\varepsilon = \frac{\Delta L}{L_0},$$

where:

- ΔL is the change in length of the material,
- L_0 is the original length of the material.

Strain is unitless, as it represents a ratio of lengths. Similar to stress, strain can be classified based on the mode of deformation:

- **Normal Strain:** Results from changes in length or volume, caused by tensile or compressive forces.
- **Shear Strain:** Results from angular deformation due to shear forces. It is measured as the change in angle between two originally perpendicular lines.

In three dimensions, strain is described by the strain tensor:

$$\varepsilon_{ij} = \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{xy} & \varepsilon_{xz} \\ \varepsilon_{yx} & \varepsilon_{yy} & \varepsilon_{yz} \\ \varepsilon_{zx} & \varepsilon_{zy} & \varepsilon_{zz} \end{bmatrix}.$$

7.3.3 Stress-Strain Relationship

The relationship between stress and strain is governed by the material's mechanical properties and is characterized by Hooke's law for elastic deformation:

$$\sigma = E\varepsilon,$$

where E is the Young's modulus (or elastic modulus), a measure of the stiffness of the material.

- **Elastic Deformation:** The material returns to its original shape after the removal of stress. Hooke's law applies in this regime.
- **Plastic Deformation:** The material undergoes permanent deformation after exceeding the elastic limit or yield stress.

The stress-strain curve provides a graphical representation of these regimes, highlighting critical points such as the yield strength, ultimate tensile strength, and fracture point.

Another properties is the Poisson's ratio (ν), that describes the relationship between lateral and longitudinal strains in a material subjected to uniaxial stress:

$$\nu = -\frac{\varepsilon_{\text{lateral}}}{\varepsilon_{\text{longitudinal}}}.$$

Typical values of Poisson's ratio range between 0 and 0.5 for most materials.

Stress and strain are foundational concepts for characterizing the mechanical properties of materials. By understanding their relationships and behavior under various conditions, scientists and engineers can design and optimize materials and structures for specific applications.

7.4 Crystal Vibrations, Phonons, and Thermal Stability

The vibrational dynamics of atoms in a crystal lattice play a critical role in determining the physical properties of materials. These vibrations are quantized into quasiparticles called phonons, which are essential for understanding various thermal, optical, and mechanical properties of solids.

Atoms in a crystal lattice are bound together by interatomic forces, and at any temperature above absolute zero, they undergo oscillatory motion about their equilibrium positions. These vibrations can be described using classical mechanics in terms of coupled harmonic oscillators or quantum mechanically in terms of normal modes of vibration.

The vibrational modes of a crystal depend on the symmetry and geometry of the lattice. Each mode is characterized by its wavevector $\mathbf{q}$, which lies in the reciprocal space, and its frequency $\omega(\mathbf{q})$.

7.4.1 Phonons

Phonons are the quantized normal modes of vibration in a crystal lattice, analogous to photons in electromagnetic waves. They are bosonic quasiparticles that play a key role in thermal and electrical transport in materials. Phonons can be categorized into two main types based on the nature of atomic displacements:

- **Acoustic Phonons:** Correspond to vibrations where atoms in the lattice move in phase, producing sound waves in the material. The frequency of these phonons approaches zero as $\mathbf{q} \to 0$.
- **Optical Phonons:** Arise when atoms in the basis of the unit cell move out of phase with each other. These phonons typically have higher frequencies and can interact with electromagnetic radiation, leading to phenomena such as infrared absorption and Raman scattering.

The relationship between the frequency ω and wavevector $\mathbf{q}$ of phonons is described by the phonon dispersion relation. For a crystal with N atoms in its basis, there are $3N$ vibrational modes per wavevector:

- 3 acoustic modes (1 longitudinal and 2 transverse),
- $3N - 3$ optical modes.

The dispersion relation provides insight into the dynamical stability of the crystal. A negative frequency (imaginary ω) indicates dynamical instability and suggests a structural phase transition.

7.4.2 Thermal Properties and Phonons

Phonons are the primary carriers of thermal energy in non-metallic solids and govern the material's thermal properties. Their contributions can be understood in terms of:

- **Specific Heat:** At low temperatures, the specific heat of a crystal follows the Debye T^3 law due to acoustic phonons, while at high temperatures, it saturates at the Dulong-Petit limit.

- **Thermal Conductivity:** Phonons transport heat in a material, and their mean free path is affected by scattering processes such as phonon-phonon interactions, boundary scattering, and impurity scattering.

The thermal stability of a crystal is determined by the ability of the lattice to maintain its structure under thermal agitation. This depends on several factors:

- **Anharmonicity:** While the harmonic approximation provides a basic understanding of lattice vibrations, anharmonic terms in the potential energy become significant at higher temperatures. These lead to thermal expansion and phonon-phonon scattering.
- **Melting Temperature:** The melting point of a crystal is related to the strength of interatomic bonds. Stronger bonds result in higher thermal stability.
- **Debye Temperature:** The Debye temperature is a measure of the highest vibrational frequency in a material. It provides a benchmark for estimating the thermal behavior of the crystal. Higher Debye temperatures correspond to greater thermal stability.

The study of crystal vibrations and phonons is fundamental to understanding the thermal, optical, and mechanical properties of materials. Phonons not only govern heat transport and lattice dynamics but also play a crucial role in determining the thermal stability and structural integrity of crystals. Advanced experimental and theoretical techniques continue to provide insights into the complex interplay between lattice vibrations and material properties.

7.5 Energy Band Theory

Energy band theory is the foundation of our understanding of the electronic properties of solids. It describes how electrons in a periodic crystal lattice experience the lattice potential, resulting in the formation of allowed and forbidden energy regions. This theory distinguishes materials as conductors, semiconductors, or insulators and is essential for understanding solid-state devices.

7.5.1 Formation of Energy Bands

In isolated atoms, electrons occupy discrete energy levels. When atoms are brought together to form a solid, the overlap of atomic orbitals due to

proximity leads to the splitting of these discrete levels into energy bands. The degree of overlap depends on the atomic separation and the type of orbitals involved, formation the bands, that can be classified as,

- **Valence Band:** Composed of the highest energy levels that are fully occupied at absolute zero temperature. It is responsible for chemical bonding in solids.
- **Conduction Band:** Formed by higher energy states that electrons can occupy when excited. Electrons in this band are free to move, contributing to electrical conductivity.

The energy between the conduction band and the valence band can be present a forbbiden energy region, called **band gap** (E_g), i.e,

$$E_g = E^{min}_{\text{conduction}} - E^{max}_{\text{valence}}.$$

The band gap determines the material's electrical properties as,

- **Insulators:** Large E_g , electrons require significant energy to transition to the conduction band.
- **Semiconductors:** Moderate, enabling thermal or optical excitation of electrons.
- **Metals:** Overlapping valence and conduction bands, resulting in no band gap.

Bloch's theorem is a cornerstone of band theory, stating that the wavefunction of an electron in a periodic potential is of the form:

$$\psi_{n\mathbf{k}}(\mathbf{r}) = u_{n\mathbf{k}}(\mathbf{r})e^{i\mathbf{k}\cdot\mathbf{r}},$$

where $u_{n\mathbf{k}}(\mathbf{r})$ is a function with the periodicity of the lattice. The wavevector $\mathbf{k}$ represents the crystal momentum and determines the phase of the electron wavefunction. The relationship between energy (E) and wavevector ($\mathbf{k}$) is called the **dispersion relation**. In solids, the band structure is a plot of $E(\mathbf{k})$ within the first Brillouin zone. This structure determines the material's electronic properties, including:

- **Effective Mass:** Near the band edges, the curvature of the bands is related to the effective mass of the charge carriers.
- **Density of States:** The number of available electronic states per unit energy, critical for understanding thermal and transport properties.

7.5.2 Kronig-Penney Model

The Kronig-Penney model provides a simplified approach to understanding band formation by considering a one-dimensional periodic potential. The potential is defined as:

$$V(x) = \begin{cases} -V_0, & 0 \leq x \leq b, \\ 0, & b < x \leq a, \end{cases}$$

where V_0 is the depth of the potential, b is the width of the potential well, and a is the lattice constant. Solving the Schrödinger equation for this periodic potential reveals that:

- Certain energy values are allowed, forming **energy bands**.
- Other energy ranges are forbidden, corresponding to **band gaps**.

These results demonstrate how the periodic potential creates regions where electron wavefunctions interfere constructively (forming bands) or destructively (creating gaps). This demonstrate and solution is a exercise of this chapter!

7.5.3 Band Gaps in Semiconductors and Insulators

The band gap of semiconductors determines their optical and electronic properties. Direct and indirect band gaps are distinguished as:

- **Direct Band Gap:** The minimum energy transition occurs at the same **k**-point.
- **Indirect Band Gap:** The minimum energy transition involves a change in **k**, requiring phonon interaction.

For example:

- Silicon ($E_g \approx 1.1\,\text{eV}$) has an indirect band gap.
- Gallium Arsenide ($E_g \approx 1.4\,\text{eV}$) has a direct band gap, making it suitable for optoelectronic devices.

Energy band theory is crucial for understanding and engineering material properties for specific applications:

- **Electronics:** Design of transistors, diodes, and integrated circuits.

- **Optoelectronics:** LEDs and lasers rely on direct band gap semiconductors.
- **Photovoltaics:** Solar cells require semiconductors with appropriate band gaps for efficient light absorption.
- **Thermoelectrics:** Optimizing band structure to improve electrical conductivity while minimizing thermal conductivity.

Recent advances incorporate these effects, leading to more accurate predictions of electronic and optical properties. Energy band theory provides a comprehensive understanding of the electronic behavior in solids, enables the design of advanced materials for electronics, photonics, and energy applications.

7.5.4 Fermi Energy

The concepts of Fermi energy and the Fermi surface are central to understanding the electronic properties of materials. They describe the behavior of electrons in solids, particularly in metals and semiconductors, and are foundational for explaining phenomena such as electrical conductivity, thermal properties, and electronic band structures.

The probability of an electron occupying an energy state at a finite temperature is given by the Fermi-Dirac distribution:

$$f(E) = \frac{1}{1 + e^{\frac{E - E_F}{k_B T}}},$$

where:

- E is the energy of the state,
- k_B is Boltzmann's constant,
- T is the absolute temperature.

As T increases, the sharpness of the occupancy cutoff at E_F diminishes, leading to a small fraction of electrons being thermally excited above the Fermi energy.

The Fermi energy (E_F) is defined as the energy level at which the probability of an electron state being occupied is 50% at absolute zero temperature ($T = 0\,\mathrm{K}$). It serves as a reference point for the distribution of electrons in a solid and is given by:

$$E_F = \frac{\hbar^2}{2m} \left(3\pi^2 n\right)^{\frac{2}{3}},$$

where:

- $\hbar$ is the reduced Planck constant,
- m is the mass of the electron,
- n is the number density of electrons.

At $T = 0\,\mathrm{K}$, all electronic states with energy less than E_F are completely filled, while those with energy greater than E_F are empty. The Fermi energy thus delineates the boundary of occupied and unoccupied states in the electronic structure.

The value of E_F depends on the electronic density and the material's band structure. In metals, E_F lies within a partially filled conduction band, leading to high electrical conductivity. In semiconductors and insulators, E_F typically lies within the band gap, and its position relative to the conduction and valence bands determines the material's electronic behavior.

The Fermi surface is the surface in reciprocal space (momentum space) that separates occupied electronic states from unoccupied states at $T = 0\,\mathrm{K}$. It is defined by the condition:

$$E(\mathbf{k}) = E_F,$$

where $E(\mathbf{k})$ is the energy dispersion relation, and $\mathbf{k}$ is the wavevector. The Fermi surface provides critical insights into the electronic structure of metals and governs various physical properties:

- **Electrical Conductivity:** The shape and size of the Fermi surface determine the electron velocities and scattering rates, which influence conductivity.
- **Thermal Properties:** The density of states near E_F affects the electronic contribution to heat capacity and thermal conductivity.
- **Magnetoresistance and Quantum Oscillations:** The topology of the Fermi surface is closely related to phenomena like the de Haas-van Alphen effect and the Shubnikov-de Haas effect.

The Fermi surface's geometry depends on the symmetry and periodicity of the crystal lattice:

- In simple metals, the Fermi surface is nearly spherical due to free-electron-like behavior.

- In more complex materials, such as transition metals or cuprates, the Fermi surface can exhibit intricate shapes influenced by the band structure and interactions.

The concepts of Fermi energy and Fermi surface are pivotal for understanding various phenomena in solid-state physics:

- **Superconductivity:** The pairing of electrons (Cooper pairs) occurs near the Fermi surface.
- **Semiconductors:** The position of the Fermi energy relative to the band edges determines carrier concentration and type (electrons or holes).
- **Thermoelectrics:** Optimizing the Fermi level enhances the Seebeck coefficient and thermoelectric efficiency.
- **Topological Materials:** The topology of the Fermi surface is crucial for identifying surface states and Dirac/Weyl fermions.

The Fermi energy and Fermi surface are fundamental to describing the electronic properties of materials. While the Fermi energy provides a measure of the electron population at $T = 0\,\mathrm{K}$, the Fermi surface represents the boundary in reciprocal space that governs many physical phenomena. Understanding these concepts is essential for the design and application of materials in electronics, thermoelectrics, and quantum devices.

7.6 Optical properties

The optical properties of solid-state materials are critical for understanding their interaction with electromagnetic radiation. These properties reveal key insights into the electronic structure, interband transitions, and excitations in solids. This section explores the principles governing optical phenomena, including absorption, reflection, refraction, and emission, and delves into key theoretical frameworks such as the Kramers-Kronig relations, Fermi's Golden Rule, and the concept of the optical gap.

When electromagnetic radiation interacts with a solid, its electronic and vibrational states respond to the oscillating electric field. This interaction results in various optical phenomena:

- **Absorption:** Energy from the light is absorbed by the material, causing electronic transitions between energy levels or bands.

- **Reflection:** A portion of the incident light is reflected at the surface due to refractive index mismatch.
- **Refraction:** Light changes direction as it propagates through a material with a different refractive index.
- **Transmission:** Light passes through the material, with partial absorption and phase changes.

The optical properties are described by the complex dielectric function:

$$\epsilon(\omega) = \epsilon_1(\omega) + i\epsilon_2(\omega),$$

where $\epsilon_1(\omega)$ represents the real part (related to the refractive index and dispersion), and $\epsilon_2(\omega)$ is the imaginary part (associated with absorption).

7.6.1 The Optical Gap

The optical gap is a fundamental property of semiconductors and insulators, representing the minimum energy required to excite an electron from the valence band to the conduction band via photon absorption. Unlike the electronic band gap, which refers to the energy difference between the conduction and valence band edges, the optical gap accounts for optical transitions that are subject to selection rules.

- **Direct Optical Gap:** In materials with a direct band gap, the valence band maximum (VBM) and conduction band minimum (CBM) occur at the same wavevector ($\mathbf{k}$). Photon absorption directly excites electrons without requiring phonon assistance.
- **Indirect Optical Gap:** In materials with an indirect band gap, the VBM and CBM occur at different wavevectors. Optical transitions require phonon involvement to conserve momentum.

The distinction between direct and indirect optical gaps has significant implications for the material's optical absorption efficiency and its suitability for optoelectronic applications.

7.6.2 Kramers-Kronig Relations

The Kramers-Kronig relations connect the real and imaginary components of the dielectric function, ensuring consistency with the causality principle. These relations are expressed as:

$$\epsilon_1(\omega) = 1 + \frac{2}{\pi}\mathcal{P}\int_0^\infty \frac{\omega'\epsilon_2(\omega')}{\omega'^2 - \omega^2}\, d\omega',$$

$$\epsilon_2(\omega) = -\frac{2\omega}{\pi}\mathcal{P}\int_0^\infty \frac{\epsilon_1(\omega') - 1}{\omega'^2 - \omega^2}\, d\omega',$$

where $\mathcal{P}$ denotes the Cauchy principal value of the integral, therefore,

- The Kramers-Kronig relations enable the determination of the real part of the dielectric function from experimental measurements of the imaginary part, and vice versa.
- They are fundamental for analyzing optical spectra and extracting material parameters such as the refractive index and extinction coefficient.

7.6.3 Fermi's Golden Rule in Optical Transitions

Fermi's Golden Rule provides the transition rate between quantum states induced by an external perturbation, such as an electromagnetic wave. For optical transitions, it is expressed as:

$$W_{i\to f} = \frac{2\pi}{\hbar}\left|\langle\psi_f|\hat{H}_{\text{int}}|\psi_i\rangle\right|^2 \delta(E_f - E_i - \hbar\omega),$$

where:

- $W_{i\to f}$ is the transition probability per unit time,
- $|\psi_i\rangle$ and $|\psi_f\rangle$ are the initial and final quantum states,
- $\hat{H}_{\text{int}}$ is the interaction Hamiltonian, often $-e\mathbf{E}\cdot\mathbf{r}$,
- $\delta(E_f - E_i - \hbar\omega)$ ensures energy conservation.

Selection rules, derived from symmetry and conservation laws, dictate the allowed transitions, determining which optical transitions contribute to absorption.

7.6.4 Optical Parameters

The optical response of a material is quantified using parameters derived from the dielectric function:

- **Refractive Index (n):** Describes the material's ability to bend light, related to the real part of $\epsilon(\omega)$.
- **Extinction Coefficient (k):** Measures light absorption, related to the imaginary part of $\epsilon(\omega)$.

- **Absorption Coefficient (α):** Determines the rate at which light intensity decreases as it propagates through the material:

$$\alpha(\omega) = \frac{4\pi k}{\lambda}.$$

- **Reflectivity (R):** Fraction of light reflected at the surface:

$$R = \left| \frac{n - 1 + ik}{n + 1 + ik} \right|^2.$$

The study of optical properties bridges fundamental physics and practical applications. The optical gap serves as a critical parameter for understanding light-matter interactions, while theoretical frameworks like the Kramers-Kronig relations and Fermi's Golden Rule provide tools for analyzing these interactions. Together, these principles underpin the design and optimization of materials for optical and optoelectronic technologies.

7.7 Exciton Theory

Excitons are fundamental quasiparticles in solid-state physics, representing bound states of an electron and a hole created during the absorption of light in semiconductors, insulators, or molecular systems. Their study provides insights into the optical and electronic properties of materials, bridging the gap between single-particle electronic structure and collective excitations. This section presents a detailed overview of exciton theory, covering their formation, classification, theoretical description, and applications.

7.7.1 Formation of Excitons

When a photon of energy greater than or equal to the electronic band gap (E_g) is absorbed by a material, it excites an electron from the valence band to the conduction band, leaving behind a positively charged hole in the valence band. The electron and hole, due to their opposite charges, experience a Coulomb attraction that can bind them together, forming an exciton.

The energy of the exciton is slightly less than the energy of the free electron-hole pair due to this binding energy (E_b). The exciton binding energy is a critical parameter, determining its stability and existence in various materials. Excitons can be broadly classified based on their spatial extent, binding energy, and origin,

- **Frenkel Excitons**: Frenkel excitons are tightly bound electron-hole pairs where the electron and hole are localized on the same atom or molecule. These excitons typically have binding energies in the range of 0.1–1 eV and are observed in organic crystals, molecular solids, and insulators with narrow bands.
- **Wannier-Mott Excitons** Wannier-Mott excitons are weakly bound and delocalized over several lattice sites. They are typically found in inorganic semiconductors with high dielectric constants and small effective masses. Their binding energies are much smaller, typically in the range of a few meV to tens of meV. The large spatial extent of these excitons makes them sensitive to lattice properties and external perturbations.
- **Charge-Transfer Excitons** Charge-transfer excitons occur in systems where the electron and hole reside on different molecules or atoms, leading to spatial separation. These excitons are significant in organic photovoltaics and molecular systems.
- **Rydberg Excitons** Rydberg excitons are highly excited states of Wannier-Mott excitons, characterized by large spatial extents and a hydrogen-like energy level structure. These excitons are observed in materials like cuprous oxide ($\mathsf{Cu_2O}$).

7.7.2 Theoretical Framework

The behavior of Wannier-Mott excitons can be described using the effective mass approximation. In this model, the exciton is treated as a hydrogen-like system, where the electron and hole interact via Coulomb attraction. The Hamiltonian for the exciton is given by:

$$H = -\frac{\hbar^2}{2m_e^*}\nabla_e^2 - \frac{\hbar^2}{2m_h^*}\nabla_h^2 - \frac{e^2}{\epsilon_r|\mathbf{r}_e - \mathbf{r}_h|},$$

where m_e^* and m_h^* are the effective masses of the electron and hole, respectively, and ϵ_r is the relative dielectric constant of the material. The solutions to this equation yield a series of bound states with energies:

$$E_n = -\frac{\mu e^4}{2\hbar^2\epsilon_r^2 n^2},$$

where $\mu = \frac{m_e^* m_h^*}{m_e^* + m_h^*}$ is the reduced effective mass, and n is the principal quantum number.

The exciton wavefunction reflects the relative motion of the electron and hole. For Wannier-Mott excitons, it is delocalized and can be expressed in terms of spherical harmonics and radial functions similar to hydrogenic orbitals:

$$\psi_{n\ell m}(\mathbf{r}) = R_{n\ell}(r)Y_{\ell m}(\theta, \phi),$$

where $R_{n\ell}(r)$ is the radial part, and $Y_{\ell m}(\theta, \phi)$ are the spherical harmonics.

Excitons play a central role in determining the optical response of materials. The optical absorption spectrum near the band edge often exhibits peaks corresponding to excitonic transitions, which lie below the band gap due to exciton binding energy. These features are particularly prominent in materials with strong excitonic effects, such as transition metal dichalcogenides (TMDs). Excitons have wide-ranging implications in technology and fundamental science:

- **Optoelectronics:** Understanding excitonic effects is essential for designing efficient light-emitting diodes (LEDs), laser diodes, and solar cells.
- **Excitonic Insulators:** In some materials, excitons can condense into a macroscopic quantum state, leading to the excitonic insulator phase.
- **Quantum Technologies:** Rydberg excitons offer platforms for exploring quantum coherence and entanglement.
- **Energy Harvesting:** Charge-transfer excitons are critical in organic photovoltaics and artificial photosynthesis.

Excitons are fundamental to the optical and electronic properties of materials, bridging single-particle and collective excitation phenomena. Their study provides insights into light-matter interaction, phase transitions, and new material functionalities, making exciton theory a cornerstone of modern solid-state physics and materials science.

7.8 Group Theory in Solid-State Physics

Group theory is a powerful mathematical framework used to study symmetry in various fields of physics, including solid-state physics. It provides a systematic way to classify and analyze the symmetry properties of crystals and their physical phenomena.

7.8.1 Introduction to Group Theory

Group theory is the study of groups, which are sets of elements combined with a binary operation satisfying four fundamental properties: closure, associativity, identity, and inverses. In the context of solid-state physics, these elements typically represent symmetry operations such as rotations, reflections, and translations.

Symmetry Operations and Groups

Symmetry operations in a crystal are transformations that leave the crystal invariant. These include:

- **Rotations:** About an axis through specific angles.
- **Reflections:** Across a mirror plane.
- **Inversions:** Through a center of symmetry.
- **Translations:** By a lattice vector in the crystal.
- **Glide Reflections and Screw Rotations:** Combined operations involving translations and reflections or rotations, respectively.

Groups classify these operations into:

- **Point Groups:** Symmetries that leave at least one point fixed (e.g., rotations and reflections).
- **Space Groups:** Include translational symmetry along with point group operations, describing the full symmetry of a crystal lattice.

7.8.2 Applications of Group Theory in Solid-State Physics

Group theory is extensively used to understand and predict physical phenomena in crystals. Below are key applications:

Classification of Crystals

Using group theory, crystals are classified into **230 space groups** in three dimensions. Each space group captures the complete symmetry of a crystal lattice, influencing its physical properties, such as optical and electronic behavior.

Electronic Band Structure

Group theory simplifies the analysis of the electronic band structure by classifying electronic states using the symmetry of the crystal:

- **Irreducible Representations (IRs):** The symmetry-adapted wavefunctions of electrons are labeled using IRs of the crystal's symmetry group.
- **Selection Rules:** Determine allowed electronic transitions based on symmetry considerations, which are critical for interpreting optical spectra.

For example, at high-symmetry points in the Brillouin zone (e.g., Γ, X, M), group theory predicts degeneracies in electronic energy levels.

Vibrational Modes (Phonons)

The vibrational modes of a crystal are governed by its symmetry. Group theory helps:

- Classify phonon modes as **Raman-active**, **infrared-active**, or **silent**.
- Predict degeneracies of vibrational modes at specific symmetry points in the Brillouin zone.

For instance, in a cubic crystal, the symmetry operations of the point group can be used to classify the vibrational modes into acoustic and optical branches.

Magnetic Properties and Spin Structures

For magnetic crystals, group theory extends to include time-reversal symmetry, leading to the concept of **magnetic point groups** and **magnetic space groups**. These are crucial for understanding spin arrangements and magnetic anisotropies.

7.8.3 Representation Theory in Crystals

Representation theory is a core part of group theory, where groups are represented as matrices. In solid-state physics:

- **Symmetry-adapted Basis Functions:** Wavefunctions or atomic orbitals are expressed in terms of basis functions that transform according to the IRs of the group.
- **Character Tables:** Provide information about how different functions (e.g., atomic orbitals, vibrational modes) transform under symmetry operations.

For example, the character table for the cubic group O_h is used to classify the symmetry of orbitals (s, p, d) in a cubic crystal field.

7.8.4 Key Results from Group Theory

- **Crystal Field Splitting:** In complex crystals, group theory predicts how degenerate energy levels (e.g., d-orbitals) split under the influence of the crystal field.
- **Band Degeneracies:** High-symmetry points in the Brillouin zone often exhibit degeneracies in electronic bands, dictated by group theory.
- **Symmetry-imposed Constraints:** Many physical properties, such as piezoelectricity or ferroelectricity, are forbidden in crystals lacking specific symmetries.

7.8.5 Example: Application to Diamond and Zincblende Structures

The diamond structure (e.g., Si, C) and zincblende structure (e.g., GaAs) belong to the $F\bar{4}3m$ space group. Group theory predicts:

- Allowed electronic transitions (e.g., direct band gap in GaAs).
- Raman-active modes (T_2) in both structures.
- Symmetry-forbidden piezoelectric effect in diamond due to inversion symmetry.

Group theory is an indispensable tool in solid-state physics, offering profound insights into the symmetry and physical properties of crystals. By leveraging its principles, physicists can predict and interpret a wide range of phenomena, from electronic band structures to vibrational modes and magnetic properties.

7.9 It's Your Turn!

1. Utilize the Kronig-Penney model to solve the Schrödinger equation for a one-dimensional periodic potential. Derive the transcendental equation that relates the wavevector k and energy E. Ensure the derivation captures the essential assumptions and boundary conditions of the model.
2. Numerically solve the transcendental equation to compute the allowed energy bands. Plot the energy bands as a function of the wavevector k. Restrict the plot to the first Brillouin zone to highlight the periodicity of the reciprocal space. **Hint:** Implement this step using programming tools such as Python, Mathematica, or MATLAB. Ensure proper convergence of numerical solutions for accurate representation of the band structure.
3. Investigate the effect of varying the barrier width on the band structure. Specifically, compute and plot the energy gap as a function of the barrier width. Analyze how the changes in barrier width influence the size of the band gap, providing physical insight into the tunability of electronic properties in periodic potentials.

Capítulo 8

Convergence Tests in SIESTA

In Density Functional Theory (DFT) calculations performed with SIESTA (Spanish Initiative for Electronic Simulations with Thousands of Atoms), convergence tests are crucial to ensure the reliability and accuracy of the results. These tests are used to determine the optimal values of computational parameters such as the energy cutoff, k-point grid, and convergence thresholds for energy, charge density, and forces. Proper convergence ensures that the obtained properties of the system are not artifacts due to computational approximations, and that the simulations are both accurate and efficient.

The following sections discuss the key convergence tests used in SIESTA simulations.

8.1 Energy Cutoff (Mesh Cutoff)

The energy cutoff, specified by the `MeshCutoff` parameter in the input fdf file, controls the number of plane waves included in the calculation of the electronic wavefunctions. A higher cutoff results in a more accurate representation of the wavefunctions, but also increases the computational cost. In SIESTA, the energy cutoff corresponds to the resolution of the grid used for solving the Poisson equation and for the calculation of the charge density.

Test: To perform a convergence test with respect to the mesh cutoff, a series of calculations should be run with increasing values of the

cutoff. The total energy of the system should be monitored during these calculations. The system is considered to be converged when the energy difference between two successive values of the cutoff becomes negligible. The mesh cutoff is typically varied from around 200 Ry (for light elements) to 400 Ry or higher for heavy elements or more demanding calculations.

8.2 K-Point Sampling

The k-point grid defines the sampling of the Brillouin zone in periodic systems, and is crucial for obtaining accurate electronic structure calculations. The density of k-points affects the precision of calculated quantities such as the electronic energy bands, charge density, and forces. The more k-points used, the more accurate the calculation, but at the cost of increased computational effort.

Test: To test convergence with respect to the k-point grid, calculations should be performed with increasing k-point densities (e.g., $2 \times 2 \times 2$, $4 \times 4 \times 4$, etc.). The convergence criterion is typically based on the total energy, force, or charge density. A common practice is to monitor the energy difference between successive grids. The calculation is considered converged when the change in total energy is smaller than a specified threshold (e.g., 10^{-4} eV) as the grid is refined. For most systems, the k-point grid should be dense enough to sample the Brillouin zone well without excessive computational cost.

8.3 Self-consistency charge density

The self-consistency of the charge density is another critical aspect of DFT calculations. SIESTA uses a self-consistent field (SCF) procedure to solve for the charge density and electrostatic potential. The SCF loop iterates until the charge density converges to a solution that minimizes the total energy. The convergence of the charge density can be monitored by tracking the change in the total energy, the charge density itself, or the forces on atoms between iterations.

Test: The convergence of the SCF loop is evaluated by examining the total energy difference between successive iterations. The self-consistency criterion can be written as:

$$\frac{|E^i_{\text{total}} - E^{i-1}_{\text{total}}|}{E^i_{\text{total}}} < \epsilon_{\text{SCF}}$$

where E^i_{total} is the total energy at the i-th iteration, and ϵ_{SCF} is a small threshold, typically 10^{-4} eV. It is important to ensure that the SCF procedure is converged before interpreting the results.

8.4 Fermi Smearing

SIESTA allows for the use of smearing techniques, such as Gaussian smearing, to treat the occupation of states near the Fermi level. This is especially important when studying metallic systems or systems at finite temperatures. The smearing width affects the broadening of the Fermi surface and should be chosen such that it does not artificially affect the electronic properties of the system.

Test: Perform calculations with different smearing widths (e.g., 0.1 eV, 0.2 eV) and monitor the total energy and electronic properties. The smearing width should be chosen to be sufficiently small to avoid broadening effects that would alter the system's electronic behavior. The system is considered converged when changes in the energy gap, electronic structure, and total energy are minimal as the smearing parameter is adjusted.

8.5 Forces in Structural Relaxation

In structural relaxation or molecular dynamics simulations, the convergence of atomic forces is crucial to ensure that the system reaches a stable configuration. The forces on atoms are minimized as part of the relaxation procedure, and the calculation is considered converged when these forces become sufficiently small.

Test: The forces on the atoms should be monitored throughout the relaxation process. The calculation is considered converged when the maximum force on any atom falls below a specified threshold, often 10^{-2} eV/Å. This ensures that the system has reached a local minimum of the potential energy surface and that the forces are small enough for the system to be considered stable.

8.6 Main Convergence Variables in SIESTA

In SIESTA, several parameters in the input file control the convergence behavior. These include:

- `Mesh.Cutoff`: Sets the energy cutoff for the plane wave expansion of the wavefunctions.
- `kgrid.MonkhorstPack`: Defines the k-point mesh used for sampling the Brillouin zone.
- `DM.Tolerance`: Convergence criterion for the charge density in the self-consistent loop.
- `MD.MaxForceTol`: Convergence criterion for atomic forces during structural relaxation.
- `ElectronicTemperature`: Controls the smearing of the electronic states near the Fermi level.

To ensure accurate and reliable results, it is important to perform a systematic convergence study by gradually refining the computational parameters, starting with a coarse grid and lower cutoff and progressively increasing them until convergence criteria are satisfied.

Convergence tests in SIESTA are essential to ensure that the results of DFT simulations are accurate and not influenced by arbitrary computational choices. A well-converged simulation gives confidence that the calculated electronic, structural, and thermodynamic properties are reflective of the true behavior of the system. Convergence tests allow for an optimal balance between computational efficiency and accuracy, and they are indispensable for obtaining reliable insights into the material properties under study.

8.7 Your Turn!

1. Perform a convergence test for aluminum (Al) in the bulk FCC structure.
2. Perform a convergence test for gallium nitride (GaN) in the bulk wurtzite phase. **Hint**: Refer to the database or the Master's Dissertation by Carlos Maciel O. Bastos for information on the wurtzite structure of GaN.

Capítulo 9

Computing structural properties

9.1 Structural Relaxation in SIESTA

Structural relaxation is a fundamental process in computational material science used to determine the equilibrium configuration of a crystal system. It involves minimizing the total energy of a structure by adjusting atomic positions, lattice parameters, or both, under the influence of quantum mechanical forces. This chapter provides a comprehensive guide to performing structural relaxation using SIESTA. We focusing on the zinc blende structure of GaAs as a practical example.

9.1.1 Description of the Crystal Lattice

The *zinc blende* structure, also known as the cubic sphalerite structure, is a derivative of the face-centered cubic (FCC) lattice. It is characterized by two interpenetrating FCC sublattices displaced by one-quarter of the unit cell along the body diagonal.

- **Lattice Constants:** GaAs has a lattice constant of approximately 5.653 Å.
- **Basis Atoms:** Each unit cell contains two atoms:
  - Gallium (Ga) at $(0, 0, 0)$,
  - Arsenic (As) at $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4})$.

The zinc blende structure possesses high symmetry, belonging to the space group $F\bar{4}3m$ (No. 216). This symmetry simplifies the calculations by reducing the number of independent variables.

Exploiting crystal symmetry can significantly reduce computational effort:

1. **Space Group Operations:** The symmetry operations of $F\bar{4}3m$ ensure that all equivalent atomic positions are updated simultaneously during relaxation.
2. **Reduction in Degrees of Freedom:** For a perfect zinc blende structure, only the lattice constant is variable during lattice relaxation. Atomic relaxation is constrained by symmetry.
3. **Boundary Conditions:** Periodic boundary conditions (PBC) are used to mimic an infinite crystal.

By adhering to these symmetries, we reduce computational overhead and improve the convergence of the relaxation process.

9.1.2 Structural Optimization in SIESTA

The SIESTA package provides robust tools for structural optimization. The relaxation process involves iterative adjustments to atomic coordinates and lattice vectors to achieve force and stress tolerances within defined thresholds.

To perform structural relaxation in SIESTA, the following key parameters must be set in the input file:

- `MD.TypeOfRun CG`: Specifies relaxation mode.
- `MD.VariableCell true`: cell is treated as a dynamical variable, allowing its size and shape to change in response to applied stresses or external conditions.
- `MD.MaxForceTol`: Sets the force tolerance for convergence, typically $0.01\,\mathrm{eV\,Å^{-1}}$.
- `MD.Steps`: Defines the maximum number of relaxation steps.

9.1.3 Example: GaAs in Zinc Blende Structure

The starting point for relaxation is the zinc blende structure with a lattice constant of 7.0 Å. The input file should define the lattice vectors

and fractional atomic coordinates:

```
LatticeConstant 7.0 Ang
AtomicCoordinatesFormat Fractional

%block LatticeVectors
     0.0    0.5    0.5
     0.5    0.0    0.5
     0.5    0.5    0.0
%endblock LatticeVectors



%block AtomicCoordinatesAndAtomicSpecies
0.000   0.000   0.000   1
0.250   0.250   0.250   2
%endblock AtomicCoordinatesAndAtomicSpecies
```

After running the relaxation calculation, the output provides the optimized structure. Key results to analyze include:

- **Final Lattice Constant:** Compare the relaxed lattice constant with experimental values.
- **Residual Forces:** Ensure all forces are below the defined threshold.
- **Energy Convergence:** Verify the total energy is minimized.

The relaxation of crystal structures is a crucial step in computational materials science. Through the example of GaAs, this chapter demonstrated the importance of understanding crystal lattices, symmetry, and optimization strategies. Mastering these techniques with tools like SIESTA enables the exploration of material properties with precision and efficiency.

9.2 Calculating the Bulk Modulus Using SIESTA

The bulk modulus (B) is a fundamental mechanical property that quantifies a material's resistance to uniform compression. It is derived from

the relationship between the pressure applied to a material and the resulting change in its volume. In computational materials science, the bulk modulus can be calculated by performing a series of energy-volume calculations and fitting the results to an appropriate equation of state (EOS). This section outlines the steps for calculating the bulk modulus using SIESTA.

The bulk modulus is defined as:

$$B = -V \left(\frac{\partial P}{\partial V} \right),$$

where:

- V is the volume of the system,
- P is the pressure,
- $\partial P/\partial V$ is the derivative of pressure with respect to volume.

In practice, the bulk modulus is often obtained by fitting the energy-volume data to an equation of state.

To calculate the bulk modulus in SIESTA, the following steps are performed:

Step 1: Relaxation of the Structure

Before performing the energy-volume calculations, the structure must be relaxed to ensure it is in its equilibrium configuration. This involves:

1. Relaxing the atomic positions until the forces are below a specified threshold.
2. Relaxing the lattice parameters to minimize the stress tensor.

Key input parameters for relaxation include:

```
MD.TypeOfRun = CG
MD.MaxForceTol = 0.01 eV/Ang
```

Step 2: Volume Perturbation

After determining the equilibrium lattice constants (a_0), the next step is to perturb the volume by scaling the lattice vectors. Typical perturbations include small increments (e.g., $\pm 5\%$, $\pm 10\%$) around the equilibrium volume.

For a cubic lattice, the volume scaling can be achieved by scaling the lattice constant a as follows:

$$a' = a_0 \times \lambda,$$

where λ is the scaling factor (e.g., 0.95, 1.0, 1.05).

In the SIESTA input file, modify the lattice vectors using the scaled values:

```
%block LatticeVectors
a'   0.0   0.0
0.0   a'   0.0
0.0   0.0   a'
%endblock LatticeVectors
```

Step 3: Energy Calculation for Each Volume

For each scaled lattice parameter, perform a total energy calculation while keeping the atomic positions fixed.

Record the calculated total energy (E) and the corresponding volume (V) for each scaling factor.

Step 4: Fitting the Energy-Volume Data

Fit the energy-volume data to an appropriate equation of state. Common EOS models include:

- **Murnaghan Equation of State:**

$$E(V) = E_0 + \frac{BV}{B'}\left[\frac{(V_0/V)^{B'}}{B'-1} + 1\right],$$

  where E_0, V_0, B, and B' are fitting parameters.

- **Birch-Murnaghan Equation of State:**

$$E(V) = E_0 + \frac{9V_0B}{16}\left\{\left[\left(\frac{V_0}{V}\right)^{2/3} - 1\right]^3 B' + \left[\left(\frac{V_0}{V}\right)^{2/3} - 1\right]^2 \left[6 - 4\left(\frac{V_0}{V}\right)^{2/3}\right]\right\}.$$

Using the fitted parameters, the bulk modulus B can be extracted directly.

9.2.1 Example: Bulk Modulus of GaAs

1. Relax the GaAs structure with an initial lattice.
2. Scale the lattice constant by differents factors (max 10%)
3. Calculate the total energy for each scaled structure.
4. Fit the energy-volume data to the EOS.

For GaAs, the bulk modulus is expected to be approximately 75.6 GPa.

The bulk modulus provides insights into the mechanical properties of a material. A higher bulk modulus indicates greater resistance to compression. The calculated value can be compared with experimental data to validate the computational model.

The calculation of the bulk modulus using SIESTA involves a systematic approach of energy-volume analysis and fitting to an equation of state. By following these steps, researchers can accurately predict the mechanical properties of materials and explore the effects of structural perturbations.

9.3 Your Turn!

1. Obtain the lattice parameters of the relaxed structures for the following systems and compare them with experimental results:
   - Silicon (Si) in the diamond structure.
   - Zinc Oxide (ZnO) in the wurtzite structure.
   - Aluminum Antimonide (AlSb) in the zinc blende structure.
   - Calcium Titanate ($CaTiO_3$) in the perovskite structure.
   - Carbon (C) in the graphene structure.
   - Molybdenum Disulfide (MoS_2) in the H2 phase.
2. Calculate the bulk modulus of calcium titanate ($CaTiO_3$) and compare it with the experimental value.

Capítulo 10

Obtaining Electronic Properties

Band structure calculations are a cornerstone of solid-state physics, providing insights into the electronic properties of materials. They describe the relationship between energy levels and crystal momentum in the first Brillouin zone (FBZ). In this chapter, we will detail the steps required to calculate the band structure of graphene using the Siesta code, including the choice of k-point paths, relevant flags, and output analysis.

10.1 First Brillouin Zone and k-Path Selection

The FBZ represents the fundamental region of wavevectors in a periodic crystal lattice. For band structure calculations, it is critical to sample high-symmetry points along a predefined k-path. For graphene, which has a honeycomb lattice with a hexagonal FBZ, the standard high-symmetry k-points are:

- $\Gamma = (0, 0, 0)$: The center of the FBZ.
- $K = \left(\frac{1}{3}, \frac{1}{3}, 0\right)$: A corner of the hexagonal FBZ.
- $M = \left(\frac{1}{2}, 0, 0\right)$: The midpoint of the FBZ edge.

The standard path in the FBZ is: $\Gamma \rightarrow M \rightarrow K \rightarrow \Gamma$. This path captures the most critical electronic transitions.

To calculate the band structure, we start with a self-consistent field (SCF) calculation to obtain the converged charge density, followed by a non-self-consistent field (NSCF) calculation along the chosen k-path.

Once the SCF calculation is complete, create a k-path file for the FBZ path:

```
BandLinesScale      ReciprocalLatticeVectors
Bandskpoints        true

%block kgrid_BandLines
  3   30
  0.0   0.0   0.0   # Gamma
  0.5   0.0   0.0   # M
  0.33333   0.33333   0.0   # K
  0.0   0.0   0.0   # Gamma
%endblock kgrid_BandLines
```

The 'BandLinesScale' keyword ensures the k-path is interpreted in units of reciprocal lattice vectors.

10.2 Extracting and Plotting the Band Structure

After the NSCF run, the file NAME_SYSTEM.bands contains the band energies along the k-path. Use the utility 'gnubands' provided with Siesta to extract the band structure:

```
gnubands < SYSTEM_NAME.bands > bands.dat
```

The output file 'bands.dat' can be plotted using Gnuplot:

```
gnuplot
plot "bands.dat" using 1:2 with lines title "Graphene Band Structure"
```

Graphene exhibits a linear dispersion relationship near the K-point, characteristic of Dirac cones. This behavior is a hallmark of graphene's unique electronic properties, such as high electron mobility and zero effective mass at the Dirac point.

This scetion provided a step-by-step guide to calculating the band structure of graphene using Siesta. The methods outlined here can be adapted for other materials by defining the appropriate lattice, k-path, and input parameters. Graphene serves as an excellent example due to its simple structure and fascinating electronic properties.

10.3 Calculating Density of States with Siesta

The Density of States (DOS) is a fundamental property in solid-state physics that describes the number of electronic states available at a given energy. It provides insights into the electronic structure of materials, such as band gaps, van Hove singularities, and contributions from different atomic orbitals. In this section, we will describe how to calculate the DOS using Siesta, with graphene as an example.

The DOS, $g(E)$, is defined as:

$$g(E) = \sum_i \delta(E - E_i)$$

where E_i are the eigenvalues of the system, and δ is the Dirac delta function. In practice, the delta function is approximated using a Gaussian or Lorentzian broadening for numerical calculations. The projected DOS (PDOS) further decomposes the DOS into contributions from specific atoms, orbitals, or spins.

To calculate the DOS, a self-consistent field (SCF) calculation must be performed first, followed by a non-self-consistent field (NSCF) calculation for energy levels. For the NSCF calculation, specify a dense k-point grid to sample the Brillouin zone effectively:

```
%block PDOS.kgrid_Monkhorst_Pack
   10   0  0  0.0
    0  10  0  0.0
    0   0  1  0.0
%endblock PDOS.kgrid_Monkhorst_Pack

%block ProjectedDensityOfStates
-20.0 100.0 0.2 2000 eV
```

```
%endblock ProjectedDensityOfStates
```

A `PDOS.kgrid_Monkhorst_Pack` define the density grid, similar to `kgrid_Monkhorst_Pack` but is only used in DOS calcukation. Density grid ensures better resolution in the DOS calculation, particularly for materials with complex band structures. Also, we can define the `ProjectedDensityOfStates` with the first and second column is the minimum and maximum range of energy. The third column is the sigma of gaussian smearing. Four and five column is respectively the number of points and unit of energy.

10.4 DOS Analysis with Siesta Utilities

After running the NSCF calculation, extract the DOS using the 'fmpdos' utility provided with Siesta:

```
fmpdos
```

We must be include the information about NAME_SYSTEM.PDOS, the orbitals to separate the dos.

10.5 Visualizing the DOS

The DOS can be visualized using Gnuplot or Python. For Gnuplot:

```
gnuplot
plot "2s.dat" using 1:2 with lines title "2s partial DOS"
```

Graphene exhibits a characteristic DOS with a linear dependence near the Fermi level, reflecting its Dirac cone structure. The DOS at the Fermi level is zero, consistent with its semimetallic behavior. Peaks in the DOS correspond to van Hove singularities at the M-point of the Brillouin zone. Graphene serves as an illustrative example, showcasing its unique DOS features and potential for tuning through external parameters. These methods can be generalized to other materials to study their electronic properties.

10.6 Your Turn!

1. Obtain the band structure and Density of States (DoS) for the following systems:
   - Silicon (Si) in the diamond structure.
   - Zinc Oxide (ZnO) in the wurtzite structure.
   - Aluminum Antimonide (AlSb) in the zinc blende structure.
   - Calcium Titanate ($CaTiO_3$) in the perovskite structure.
   - Carbon (C) in the graphene structure.
   - Molybdenum Disulfide (MoS_2) in the H2 phase.

   Hint: Use the optimized structures from the previous chapter. Utilize the `sstranslate.x` tool (provided with the materials) to extract the structural parameters.

Capítulo 11

D-Grimme correction for Van der Waals systems

Capítulo 12

Hybrid Functionals

12.1 Introduce to hybrid functional

12.2 HSE06 hybrid functional: the study case

Capítulo 13

LDA+U formalism in Siesta

13.1 An introduction of LDA+U Formalism

13.2 Applying LDA+U in semiconductor material

Capítulo 14

Optical and Phonons Properties

Capítulo 15

Interface Siesta and Wannier90: An introduction

15.1 Siesta and Wannier90

15.2 Calculate the Fermi energy surface

Capítulo 16

Toolbox and post-processing

16.1 Bader Charge

Capítulo 17

Complete caracterization of a Material:

Referências Bibliográficas

Apêndice A

Exchange-Correlation functionals in ATOM code