

Wavelets for electronic structure calculations

T. Deutsch and L. Genovese

Laboratoire de Simulation Atomistique (LSim), SP2M, INAC, CEA-UJF, 17 Av. des Martyrs, 38054 Grenoble, France

Abstract. In 2005, the EU FP6-STREP-NEST BigDFT project funded a consortium of four laboratories, with the aim of developing a novel approach for Density Functional Theory (DFT) calculations based on Daubechies wavelets. Rather than simply building a DFT code from scratch, the objective of this three-years project was to test the potential benefit of a new formalism in the context of electronic structure calculations. Daubechies wavelets exhibit a set of properties which make them ideal for a precise and optimised DFT approach. In particular, their systematicity allows to provide a reliable basis set for high-precision results, whereas their locality (both in real and reciprocal space) is highly desired for improve the efficiency and the flexibility of the treatment. In this contribution we will provide a bird's-eye view on the computational methods in DFT, and we then focus on DFT approaches and on the way they are implemented in the BigDFT code, to explain how we can take benefit from the peculiarities of such basis set in the context of electronic structure calculations.

In the recent years, the development of efficient and reliable methods for studying matter at atomistic level has become an asset for important advancements in the context of material science. Both modern technological evolution and the need for new conception of materials and nanoscaled devices require a deep understanding of the properties of systems of many atoms from a fundamental viewpoint. To this aim, the support of computer simulation can be of great importance. Indeed, via computer simulation scientists try to model systems with many degrees of freedom by giving a set of “rules” of general validity (under some assumptions).

Once these “rules” come from first-principles laws, these simulation have the ambition to model system properties from a fundamental viewpoint. With such a tool, the properties of existing materials can be studied in deep, and new materials and molecules can be conceived, with potentially enormous scientific and technological impact. In this context, the advent of modern supercomputers represents an important resource in view of advancements in this field. In other terms, the physical properties which can be analysed via such methods are tightly connected to the computational power which can be exploited for calculation. A high-performance computing electronic structure program will make the analysis of more complex systems and environments possible, thus opening a path towards new discoveries. It is thus important to provide reliable solutions to benefit from the enhancements of computational power in order to use these tools in more challenging systems.

1. ATOMISTIC SIMULATIONS

As an overview, before focusing on more detailed descriptions, we will start this contribution by a brief presentation of the different methods which are available to simulate the structure of matter or molecules at the atomic level. There are mainly two intrinsic difficulties for numerical atomistic simulations:

This is an Open Access article distributed under the terms of the Creative Commons Attribution-Noncommercial License 3.0, which permits unrestricted use, distribution, and reproduction in any noncommercial medium, provided the original work is properly cited.

Interactions *i.e.* the way that the atoms interact is described by the quantum mechanics which is really intractable to simulate exactly except for few bodies. The only possibilities is to approximate the atomic interactions considering the electrons as the density functional theory or only a model of interaction between atoms as the empirical potentials.

Exploration of the atomic configuration space which is the key to understand the properties of the matter by means of molecular dynamics or statistical physics. This space is multidimensional, *i.e.* three times the number of atoms, has lots of energy minima which represent the stable atomic structures and needs to be integrated by means of statistical physics tools in order to have the properties depending on the temperature. We refer in this book to the chapter written by Jean-Louis Barrat on this topic.

Table 1. Three criteria to classify the different atomistic methods of simulation.

Criteria	Goals
General (Generality)	all atoms of the periodic table all kinds of bonds (transition state)
Precision	
geometry	
atomic positions	± 0.001 angström
lengths of bonds	± 0.001 angström
angles of bonds	± 1
energy of bonds	0.001 eV/atom (0.1 kJ/mol)
vibration frequencies	1 GHz
dipolar moments	0.01 Debye (1 Debye = 3.336×10^{-30} C.m)
UV and visible spectra	0.01 eV
photo-emission	0.01 eV
magnetic moments	0.01 μ_B
System size	10^4 atoms for <i>ab initio</i> methods 10^6 atoms for parametrized methods
Time scale	1 s

We will concentrate on the description of interactions between atoms and the accurate calculation of properties of the atoms. As soon as we have determined the most stable structure, we are interesting in the electronic properties as photo-emission or absorption, magnetic properties, etc. For each atomistic methods, we give three criteria (see table 1):

Generality. A method should simulate any kind of atoms of the periodic table for any kinds of bonds.

This is roughly the case for the methods which consider explicitly the electronic structure of the atoms.

Precision. The atomistic methods should have the same accuracy as the experimental results which is a strong challenge nowadays totally unrealistic. Mainly we have an accuracy of few percents about the atomic structure. For the properties sometimes, the result is considered good if we have only errors of the order of 10%.

System size and Time scale. One would like to simulate atomic systems as large as possible. Sometimes, this is no necessary depending on the properties we want to simulate. Even if the performances of computers and algorithms are increasing, only 10^2 to 10^4 atoms for *ab initio* methods and 10^6 to 10^9 atoms for parametrized methods can be simulated. To have a correct idea of the dynamics, 1s should be enough except for the aging of materials. By molecular dynamics, we simulate a time scale of the order of picosecond up to microsecond which is rather small.

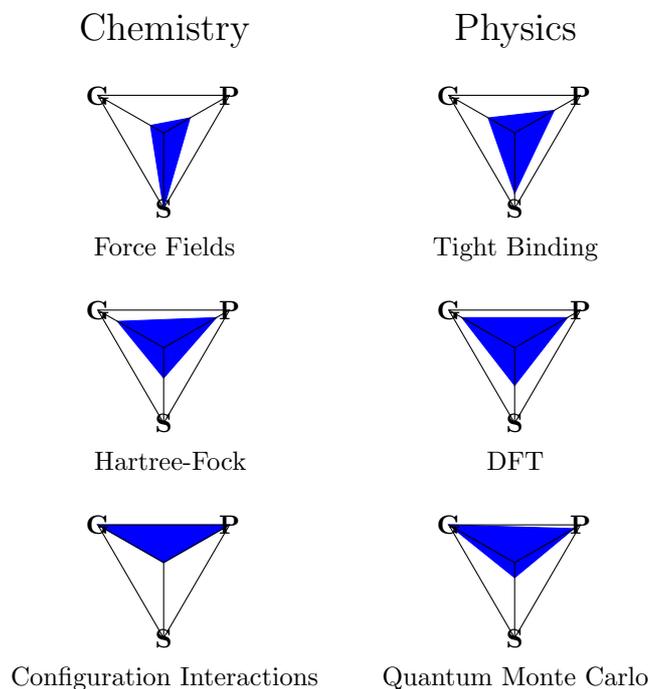


Figure 1. Criteria of choice of atomistic methods.

With these three criteria (generality, precision and system size), we will briefly describe different methods used in chemistry and physics. The idea is to give some clues in order to choose the more appropriate method for a given simulation in function of the nature and size of the system which we want to simulate and the required accuracy.

First we will consider the methods based on Quantum Mechanics: Hartree-Fock, Post Hartree-Fock as configuration Interaction, Quantum Monte-Carlo and DFT. We will add two parametrized methods as tight-binding method and force fields (empirical potentials). The figure 1 gives a panorama of the different methods used to simulate the structure at the atomic level.

1.1 Hartree-Fock and post Hartree-Fock

These methods consider the restricted solution of the full N -body Hamiltonian in a certain class of antisymmetric wavefunctions. The Hamiltonian is not approximated. The simple antisymmetric wavefunction is the antisymmetrized wavefunction of the one-particle functions of each electron in different states which do not interact. This wavefunction is obtained by a determinant of the one-particle functions called Slater determinant. The cost of calculation is bigger than with the Kohn-Sham formalism of the density function theory due to the exchange term so the size of the considered systems are smaller and the accuracy is less.

In order to have a better accuracy, there are many methods called post Hartree-Fock as coupled-cluster method, full configuration interactions which consider many Slater determinants. The cost of calculation is bigger than with Hartree-Fock. The first step is to calculate the Hamiltonian term over four one-particle functions which scales at least as N^4 where N is the number of electrons. Then these methods minimize the total energy in function of the coefficients of the Slater determinants which can

be increased exponentially as the number of electrons for the full configuration interaction method. The accuracy is excellent but the size of the system quite small (few atoms).

These methods are mainly used by chemists to study small molecules.

1.2 Quantum Monte Carlo

In the same idea of using a class of restricted antisymmetric wavefunctions, physicists have developed Quantum Monte Carlo methods. The idea of the variational Quantum Monte Carlo is to consider an antisymmetric wavefunction which is the product of a Slater determinant and a function called Jastrow factor which depends explicitly on the distance between electrons. Because the calculation of the Hamiltonian is complicated by the presence of this Jastrow factor, a Monte Carlo method is used to calculate the integrals. From this solution, the diffusive Monte Carlo refines the solution using stochastic processes. The accuracy is excellent up to one hundred electrons. But it is really difficult to calculate atomic forces and minimize the atomic position. In solid state physics, this method is the reference but the cost of calculation is huge.

1.3 Density Functional Theory in the Kohn-Sham formalism

We will later describe in more details density functional theory. The main idea is to map the many-body problem into a non-interacting particle problem in an effective potential. This is a theory which is exact. The problem is that we do not know the right shape of this effective potential. The Kohn-Sham formalism is derived from the DFT and uses approximations to model the interaction between electrons. The cost of calculation is less than the Hartree-Fock method and the accuracy is better. This is the main reason why DFT became so popular. The accuracy of the structural properties as the bond lengths are few percents (one or two). For energy determination, it depends from few percents for formation energies to more than 50% for some cohesive energies. The calculation of properties depending on the linear response are well developed and are very accurate. On contrary, the spectroscopic properties are totally wrong because this is a theory of the ground state. There exists an extension, the Time-Dependent Density Functional Theory (TD-DFT) which could calculate accurately the excited states for small molecules but the approximation of the electron-electron interaction is too rough for large systems. Another alternative is the GW method based on the many-body perturbation theory which gives better results for the calculation of photo-emission and photo-absorption spectroscopies.

1.4 Tight Binding method

Another method which considers also the electronic structure is the tight binding method. This method parametrizes the elements of the Hamiltonian in function of the distance of the atoms so the calculation is faster. The main problem is the transferability. For some systems, it is possible to have a very good accuracy due to the parametrization. If the calculation explores some new domains, then the transferability becomes critical. Millions of atoms can be considered with this method. Many electronic properties can be calculated as optics, transport or magnetic properties. In this case, the parametrization is done on the band structure in order to reproduce the electronic properties and so the structural properties need to be calculated using another set of parametrization or empirical potentials.

1.5 Force Fields (empirical potentials)

Finally, the method which does not consider the electronic structure but only the distance between atoms are the force fields. There are different classes of force fields with a different cost of calculations from the Lennard-Jones to the polarizable models. These methods are two orders of magnitude faster comparing to a DFT calculations and can handle many billions of atoms. This is possible to do very accurate structural calculations if the parametrization is good. The transferability can be a real problem for instance to calculate the migration energy of diffusion processes.

It is not possible to do all calculations with the *ab initio* methods. In function of the objective, it is better to use an appropriate method as force fields or tight binding methods.

2. METHODS FOR ELECTRONIC STRUCTURE

We will continue this contribution by a brief presentation of the Kohn-Sham formalism of Density Functional Theory. A number of good references which treats this topic exists. Here we will present some notes, with the aim of defining suitably the problem and fixing notations.

2.1 Born-Oppenheimer Hamiltonian

There is of course no question that a fundamental treatment of a system with many atoms should be performed via the laws of Quantum Mechanics. The properties of the systems are thus governed by its wavefunction, which is related to the Hamiltonian via the Schrödinger equation. It is evident that an immediate solution to this problem does not exist. For a system with N atoms and n electrons, the wavefunction has $3(N + n)$ variables, and the Hamiltonian, in atomic units, has the following form:

$$H = \sum_{i=1}^n -\frac{1}{2}\nabla_{r_i}^2 + \frac{1}{2} \sum_{i \neq j} \frac{1}{|r_i - r_j|} - \sum_{a=1}^N \sum_{i=1}^n \frac{Z_a}{|R_a - r_i|} + \sum_{a=1}^N -\frac{1}{2M_a} \nabla_{R_a}^2 + \frac{1}{2} \sum_{a \neq b} \frac{Z_a Z_b}{|R_a - R_b|}. \quad (2.1)$$

In this equation the Hamiltonian of the electrons (first two terms) is coupled with the one of the ions (last two terms) via the electromagnetic interaction (central term). In atomic units, the action is measured in units of \hbar , the mass in units of the electron mass m_e and the charge in units of the electronic charge $|e|$. For these reasons, the kinetic term which is associated to the nuclei is suppressed by the mass of the ions M_a , which is at least two thousands times heavier than the electrons. It appears thus more than justified to decouple the dynamics of the ions to the one of the electrons. In other terms, the Hamiltonian can be split in two parts:

$$H = \sum_{i=1}^n -\frac{1}{2}\nabla_{r_i}^2 + \frac{1}{2} \sum_{i \neq j} \frac{1}{|r_i - r_j|} + V_{ext}(\{r\}, \{R\}) + H_{ions}[\{R\}]. \quad (2.2)$$

The Born-Oppenheimer (BO) approximation consists in treating the dynamic of the ions classically. The wavefunction of the system will thus become associated only to the electrons (thus with $3n$ variables), with an external potential $V_{ext}(\{r\}, \{R\})$ which depend of the atomic positions $\{R\}$, which will then appear as external parameters to the quantum problem.

Even though the BO approximation effectively reduces the complexity of the description only to the electronic part, we are still really far from a formalism which is able to treat systems with many electrons. The number of variables of the wavefunction is still much too high to be handled while solving the equation explicitly. One may actually wonder whether we *really* need the complete wavefunction to extract the properties of the system we are interested to. For example, the energy of the system in a given quantum state $|\Psi\rangle$ is

$$E[\Psi] = \langle \Psi | H | \Psi \rangle, \quad (2.3)$$

which can be interpreted as a functional of the wavefunction $|\Psi\rangle$. A closer inspection reveals that the wavefunction contains too much information for calculating the energy. Since the Hamiltonian contains two-body operators (the electron-electron interaction), it is easy to show that actually the energy is a

functional of the 2-particle reduced density matrix (2-RDM) γ_2 :

$$E = \text{tr}(H\gamma_2) = E[\gamma_2], \quad (2.4)$$

where

$$\gamma_2(x_1, x_2; x'_1, x'_2) = \binom{n}{2} \int dx_3 \cdots dx_N \Psi(x_1, \dots, x_N) \Psi^*(x'_1, x'_2, x_3, \dots, x_N), \quad (2.5)$$

is a function of 12 variables. The formulation of the problem seems thus simpler in this way, but the 2-RDM cannot be a generic function. It must be chosen such that it comes from the contraction of a wavefunction as indicated in Eq. (2.5). Taking into account such a constraint (the so-called n -representability problem) is definitely a far-from-trivial task, and still keeps the formalism difficult to handle.

A big simplification to the problem of finding the ground-state energy of the system have been provided by Hohenberg and Kohn in 1964, via their famous theorem (HK):

Hohenberg–Kohn Theorem. *For a fixed number of electrons n , the charge density of the ground-state of a quantum system determines uniquely – up to an additive constant – the external potential of the electronic Hamiltonian.*

If we take into account that, of course, given *both* n and an external potential, the charge density of the ground state is determined, the HK theorem states that there is a one-to-one correspondence between the charge density, a functional of the 2-RDM¹ $\rho(\mathbf{r}) = \frac{2}{n-1} \int d\mathbf{r}_1 \gamma_2(\mathbf{r}, \mathbf{r}_1; \mathbf{r}, \mathbf{r}_1) = \rho[\gamma_2]$ and the external potential which determines the inhomogeneity of the electron gas. This implies that the ground state energy E_0 is a functional of the electronic density ρ . Such a functional reaches its minimum for the true ground state density ρ_0 :

$$E = E[\rho] = \min_{\gamma_2 \text{ s.t. } \rho[\gamma_2] = \rho} \left\{ \text{tr} \left(\left[\sum_{i=1}^n -\frac{1}{2} \nabla_{r_i}^2 + \frac{1}{2} \sum_{i \neq j} \frac{1}{|r_i - r_j|} \right] \gamma_2 \right) \right\} + \int d\mathbf{r} V_{ext}(\{\mathbf{r}\}, \{R\}) \rho(\mathbf{r}), \quad (2.6)$$

and $E[\rho_0] = E_0$, which is at the basis of the *Density Functional Theory*. We have assumed here that the system has n electrons, i.e. $\int d\mathbf{r} \rho(\mathbf{r}) = n$. Via Eq. (2.6), we can see that in the functional of the density there is a term which does not depend explicitly of the external potential, which for this reason can be considered as a *universal* functional:

$$\begin{aligned} F[\rho] &= \min_{\gamma_2 \text{ s.t. } \rho[\gamma_2] = \rho} \left\{ \text{tr} \left(\left[\sum_{i=1}^n -\frac{1}{2} \nabla_{r_i}^2 + \frac{1}{2} \sum_{i \neq j} \frac{1}{|r_i - r_j|} \right] \gamma_2 \right) \right\} = \\ &= \min_{\Psi \text{ s.t. } \rho[\Psi] = \rho} \left\{ \left\langle \Psi \left| \sum_{i=1}^n -\frac{1}{2} \nabla_{r_i}^2 + \frac{1}{2} \sum_{i \neq j} \frac{1}{|r_i - r_j|} \right| \Psi \right\rangle \right\}. \end{aligned} \quad (2.7)$$

For densities of systems with n electrons, the quantity $E[\rho] = F[\rho] + \int \rho V_{ext}$ reaches its minimum for the ground-state density ρ_0 . It is important to stress that all quantities depends of n , which is supposed fixed.

The demonstration of the HK theorem is independent of the form of the pure electronic Hamiltonian. For a n electron system which has no Coulombic interaction the HK functional (let us call it $T_s[\rho]$) has

¹ In most of the formulations the charge density is seen as a functional of the wavefunction (via e.g. the Levy's constrained search formulation). This allows to bypass the n -representability problem of the 2-RDM. Here we prefer to use this formulation to show that the 2-RDM problem actually *contains* the HK formulation. Indeed, it is easy to see that the minimum of the energy for *all* n -representable γ_2 satisfies the constraint $\rho[\gamma_2] = \rho_0$.

a pure kinetic term:

$$T_s[\rho] = \min_{\gamma_2: \rho[\gamma_2]=\rho} \left\{ \text{tr} \left(\left[\sum_{i=1}^n -\frac{1}{2} \nabla_{\mathbf{r}_i}^2 \right] \gamma_2 \right) \right\} = \min_{\Psi: \rho[\Psi]=\rho} \left\{ \left\langle \Psi \left| \sum_{i=1}^n -\frac{1}{2} \nabla_{\mathbf{r}_i}^2 \right| \Psi \right\rangle \right\}. \quad (2.8)$$

Moreover, the pure Coulombic energy of a system with density ρ is known, and can be seen as (half) the potential energy where the potential is the Hartree potential $V_H[\rho](\mathbf{r}) = \int d\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}$:

$$E_H[\rho] = \frac{1}{2} \int d\mathbf{r} d\mathbf{r}' \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} = \frac{1}{2} \int d\mathbf{r} V_H[\rho](\mathbf{r})\rho(\mathbf{r}). \quad (2.9)$$

Given these quantities, we can define the Exchange and Correlation functional $E_{xc}[\rho]$, and the associated Exchange and Correlation density *per particle* $\epsilon_{xc}(\mathbf{r})$

$$E_{xc}[\rho] = \int d\mathbf{r} \rho(\mathbf{r}) \epsilon_{xc}[\rho](\mathbf{r}) = F[\rho] - T_s[\rho] - E_H[\rho] \quad (2.10)$$

The quantity $E[\rho] = T_s[\rho] + E_H[\rho] + E_{xc}[\rho] + \int \rho V_{ext}$ should then be minimal in ρ_0 for all densities which sums up to n . This implies that $E[\rho_0 + \delta\rho] = E[\rho_0]$ for $\int d\mathbf{r} \delta\rho(\mathbf{r}) = 0$. Hence

$$\begin{aligned} 0 &= \int d\mathbf{r} \delta\rho(\mathbf{r}) \frac{\delta E}{\delta\rho(\mathbf{r})} \\ &= \int d\mathbf{r} \delta\rho(\mathbf{r}) \left\{ \frac{\delta T_s[\rho]}{\delta\rho(\mathbf{r})} + V_H[\rho](\mathbf{r}) + \frac{d}{d\rho} (\rho \epsilon_{xc}[\rho])(\mathbf{r}) + V_{ext}(\mathbf{r}) \right\}. \end{aligned} \quad (2.11)$$

It is easy to see that the above equation is the same that one would obtain by searching the ground state of the non-interacting Hamiltonian H_{KS} (so-called Kohn-Sham Hamiltonian):

$$H_{KS}[\rho] = \sum_{i=1}^n -\frac{1}{2} \nabla_{\mathbf{r}_i}^2 + V_H[\rho] + V_{xc}[\rho] + V_{ext}, \quad (2.12)$$

where we have defined the Exchange and Correlation potential

$$V_{xc}[\rho](\mathbf{r}) = \frac{d}{d\rho} (\rho \epsilon_{xc}[\rho])(\mathbf{r}) = \frac{\delta}{\delta\rho(\mathbf{r})} E_{xc}[\rho]. \quad (2.13)$$

Since the Hamiltonian is only made of one-body operators, the energy of such a system can be expressed via the eigenfunctions of H_{KS} and via the one particle reduced density matrix (1-RDM) derived from them:

$$H_{KS}[\rho] |\psi_p\rangle = \epsilon_p^{KS} |\psi_p\rangle, \quad p \in \mathbb{N} \quad (2.14)$$

so that the 1-RDM of this system is

$$\gamma_1^{KS} = \sum_p f_p |\psi_p\rangle \langle \psi_p|, \quad (2.15)$$

where the occupation numbers $0 \leq f_p \leq 1$, $\sum_p f_p = n$ guarantee the n -representability of γ_1^{KS} . The energy of the original system is thus:

$$E[\rho] = \text{tr} (H_{KS}[\rho] \gamma_1^{KS}) - E_H[\rho] + \int d\mathbf{r} \rho(\mathbf{r}) (\epsilon_{xc}[\rho](\mathbf{r}) - V_{xc}[\rho](\mathbf{r})), \quad (2.16)$$

and, of course, $\rho(\mathbf{r}) = \gamma_1^{KS}(\mathbf{r}; \mathbf{r})$.

We have followed the main steps of the demonstration of the

Kohn–Sham Theorem. *An electronic density which is associated to the ground state of an interacting electron system is also solution of a non-interacting problem submitted to a mean-field potential $V_{xc} + V_H + V_{ext}$.*

The consequences of this theorem are potentially important. If the quantity $\epsilon_{xc}[\rho]$ is known, the energy of the system can be found iteratively: for a given ρ , the eigenvalue problem of the KS Hamiltonian would provide a set of eigenfunctions $|\psi_p\rangle$, and then a new electronic density. Convergence is reached for the density ρ_0 , which minimizes $E[\rho]$. Even though ρ_0 comes from a non-interacting electron system, ρ_0 is the *exact* charge density of the interacting system.

2.2 LDA and GGA exchange correlation approximations

Clearly, the difficulty now resides in finding the correct $\epsilon_{xc}[\rho]$. Surprisingly, if we take as the XC density per electron from a homogeneous electron gas of density n , results of rather good quality can already be obtained. In this way $\epsilon_{xc}[\rho](\mathbf{r}) = \epsilon_{xc}^{\text{hom}}(\rho(\mathbf{r}))$. This is the Local Density Approximation (LDA), which can be parametrised from numerical results. Several other approximations exist, which give good results for the extraction of several properties of real materials.

A particularly used set of XC functionals is implemented as a functional of the density and of its gradient (its modulus for rotational invariance). This is the so-called Generalised Gradient Approximation (GGA):

$$\epsilon_{xc}(\mathbf{r}) = \epsilon_{xc}(\rho(\mathbf{r}), |\nabla\rho(\mathbf{r})|) . \quad (2.17)$$

In this case, the exchange correlation potential has an additional term:

$$\begin{aligned} V_{xc}(\mathbf{r}) &= \frac{\delta}{\delta\rho(\mathbf{r})} \int \rho(\mathbf{r}') \epsilon_{xc}(\rho(\mathbf{r}'), |\nabla\rho(\mathbf{r}')|) \\ &= \frac{d}{d\rho} (\rho \epsilon_{xc})(\mathbf{r}) + \int \rho(\mathbf{r}') \frac{\partial \epsilon_{xc}}{\partial |\nabla\rho|}(\mathbf{r}') \frac{\delta}{\delta\rho(\mathbf{r})} |\nabla\rho(\mathbf{r}')| d\mathbf{r}' \\ &= \epsilon_{xc}(\mathbf{r}) + \rho(\mathbf{r}) \frac{\partial \epsilon_{xc}}{\partial \rho}(\mathbf{r}) + \int \frac{\rho}{|\nabla\rho|} \frac{\partial \epsilon_{xc}}{\partial |\nabla\rho|}(\mathbf{r}') \sum_{i=x,y,z} \partial_i \rho(\mathbf{r}') \frac{\delta}{\delta\rho(\mathbf{r})} \partial_i \rho(\mathbf{r}') d\mathbf{r}' . \end{aligned} \quad (2.18)$$

The different components of the gradient of the density $\partial_i \rho(\mathbf{r})$, $i = 1, 2, 3$ can be seen here as a linear functional of the density. For example, for a finite-difference computation on a grid we have

$$\partial_i \rho(\mathbf{r}) = \sum_{\mathbf{r}'} c_{\mathbf{r},\mathbf{r}'}^i \rho(\mathbf{r}') , \quad (2.19)$$

such that $\frac{\delta}{\delta\rho(\mathbf{r})} \partial_i \rho(\mathbf{r}') = \sum_{\mathbf{r}''} c_{\mathbf{r}',\mathbf{r}''}^i \delta(\mathbf{r} - \mathbf{r}'')$. This expression can be used to calculate the last term of Eq. (2.18).

2.3 Hybrid functionals and exact exchange operator

The Kohn-Sham theorem showed us that there exists an antisymmetric wavefunction $|\Phi_0\rangle$ of a n -electron system which satisfy the following properties:

1. The density originated from $|\Phi_0\rangle$ corresponds *exactly* to the density of the original inhomogeneous electron gas:

$$\langle \Phi_0 | \mathbf{r} \rangle \langle \mathbf{r} | \Phi_0 \rangle = \rho_0(\mathbf{r}) ; \quad (2.20)$$

2. The wavefunction is the ground state of the *non interacting* Schrödinger equation:

$$H_{KS}[\rho_0] |\Psi_0\rangle = E_0^{KS}[\rho_0] |\Psi_0\rangle , \quad (2.21)$$

and $E_0[\rho_0] = E_0^{KS}[\rho_0] - E_H[\rho_0] + E_{xc}[\rho_0] - \int \rho_0 V_{xc}[\rho_0]$ is the ground-state energy of the interacting system;

3. The density ρ_0 minimizes the value of E_0 , and it is a fixed point for E_0^{KS} .

This wavefunction can be written in the basis of Slater determinants of the eigenfunctions of the one-body Hamiltonian. In this basis, it is easy to show that $E_0^{KS} = \sum_p f_p \epsilon_p^{KS}$, where f_p is the occupation number defined above. In this context, it is easy to see that for a system for which the Kohn-Sham energies have a gap between ϵ_n^{KS} and ϵ_{n+1}^{KS} , the minimum energy is attained when $f_p = \theta(n - p)$ and thus $|\Phi_0\rangle$ is made of only one Slater determinant. Otherwise, multideterminantal configurations are possible.

In this context it is interesting to calculate the contribution of the non-interacting system to the two-body electron-electron interaction. This has a formal equivalence with the Hartree-Fock exchange operator:

$$E_x^{HF} = -\frac{1}{2} \sum_{\sigma=1,2} \sum_{p,q} f_{p,\sigma} f_{q,\sigma} \int d\mathbf{r} d\mathbf{r}' \frac{\psi_{p,\sigma}(\mathbf{r}) \psi_{q,\sigma}^*(\mathbf{r}) \psi_{p,\sigma}^*(\mathbf{r}') \psi_{q,\sigma}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}, \quad (2.22)$$

where the spin quantum number σ of the non-interacting electrons has been explicited. Of course, the system of Kohn-Sham orbitals would now become interacting. This implies that an operator D_x^{HF} should be added to the Kohn-Sham Hamiltonian. The action of this operator onto a wavefunction can be calculated knowing that E_x^{HF} originates from a trace of such an operator over the KS wavefunctions $|\psi_p\rangle$:

$$E_x^{HF} = \sum_{p,\sigma} f_{p,\sigma} \langle \psi_{p,\sigma} | D_x^{HF} | \psi_{p,\sigma} \rangle ;$$

$$\langle \mathbf{r} | D_x^{HF} | \psi_{p,\sigma} \rangle = \frac{1}{f_{p,\sigma}} \frac{\delta E_x^{HF}}{\delta \psi_{p,\sigma}^*(\mathbf{r})} \quad (2.23)$$

$$= - \sum_q f_{q,\sigma} \int d\mathbf{r}' \frac{\psi_{q,\sigma}^*(\mathbf{r}') \psi_{p,\sigma}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \psi_{q,\sigma}(\mathbf{r}) ; \quad (2.24)$$

As already suggested in the seminal paper of Kohn and Sham, such construction can be used to define an alternative scheme for the Kohn-Sham procedure. By defining a hybrid Kohn-Sham – Hartree-Fock Hamiltonian

$$H_{\text{KS}^{\text{SHF}}}[\rho] = \sum_{i=1}^n -\frac{1}{2} \nabla_{\mathbf{r}_i}^2 + V_H[\rho] + V_{xc}^{\text{KS}^{\text{SHF}}}[\rho] + V_{ext} + \alpha D_x^{HF} \quad (2.25)$$

and finding its eigenvalues $\epsilon_p^{\text{KS}^{\text{SHF}}}$, the energy would become

$$E[\rho] = \sum_p f_p \epsilon_p^{\text{KS}^{\text{SHF}}} - E_H[\rho] + \alpha E_x^{HF} + E_{xc}^{\text{KS}^{\text{SHF}}}[\rho] - \int \rho V_{xc}^{\text{KS}^{\text{SHF}}}[\rho],$$

$$E_{xc}^{\text{KS}^{\text{SHF}}}[\rho] = E_{xc}[\rho] - \alpha E_x^{HF}, \quad (2.26)$$

$$V_{xc}^{\text{KS}^{\text{SHF}}}[\rho] = \frac{\delta E_{xc}^{\text{KS}^{\text{SHF}}}[\rho]}{\delta \rho}. \quad (2.27)$$

2.4 Finding the Kohn-Sham wavefunctions: Direct minimisation algorithm

We have seen that the electronic density of the system can be constructed via the KS wavefunctions $\psi_p(\mathbf{r})$, which are in turn eigenfunctions of the KS Hamiltonian, which also depends of the density. Thus, a fixed point equation has to be solved. Once the fixed point is reached, the energy of the system

can be extracted. The HK theorem guarantees us that the energy $E[\rho]$ is minimal in the ground-state density ρ_0 . The KS construction simplifies things a bit. The problem corresponds to minimize the energy of the KS Hamiltonian as if such Hamiltonian does not evolve. A new Hamiltonian can then be defined. In other terms, in the typical KS procedure the variation is performed over the wavefunctions (supposing that the occupation numbers are integers). The interesting quantity is thus

$$\frac{\delta E[\rho[\{\psi_p\}]]}{\delta \langle \psi_p |} = f_p H_{KS}[\rho] | \psi_p \rangle + \int d\mathbf{r} \frac{\delta \rho(\mathbf{r})}{\delta \langle \psi_p |} \frac{\delta E[\rho]}{\delta \rho(\mathbf{r})}, \quad (2.28)$$

As already discussed, if $\rho = \sum_p f_p |\psi_p|^2$ the last term of the rhs of this equation is zero. Things goes as if the KS Hamiltonian is fixed. The fixed-point solution ρ_0 thus minimizes both $E[\rho]$ and its KS wavefunctions minimize $E_{KS}[\rho_0]$.

This fact can be derived from the explicit form of KS Hamiltonian:

$$\begin{aligned} \frac{\delta E[\rho]}{\delta \rho(\mathbf{r})} &= \frac{\delta E_{KS}[\rho]}{\delta \rho(\mathbf{r})} - V_H[\rho](\mathbf{r}) + V_{xc}[\rho](\mathbf{r}) - \frac{\delta}{\delta \rho(\mathbf{r})} \int d\mathbf{r}' \rho(\mathbf{r}') V_{xc}[\rho](\mathbf{r}') \\ &= \sum_p f_p \langle \psi_p | \frac{\delta H_{KS}[\rho]}{\delta \rho(\mathbf{r})} | \psi_p \rangle - V_H[\rho](\mathbf{r}) - \rho(\mathbf{r}) \frac{dV_{xc}[\rho]}{d\rho}(\mathbf{r}). \end{aligned}$$

Let us now consider the first term:

$$\begin{aligned} \langle \psi_p | \frac{\delta H_{KS}[\rho]}{\delta \rho(\mathbf{r})} | \psi_p \rangle &= \int d\mathbf{r}' d\mathbf{r}'' \psi_p^*(\mathbf{r}') \psi_p(\mathbf{r}'') \langle \mathbf{r}' | \frac{\delta H_{KS}[\rho]}{\delta \rho(\mathbf{r})} | \mathbf{r}'' \rangle \\ &= \int d\mathbf{r}' d\mathbf{r}'' \psi_p^*(\mathbf{r}') \psi_p(\mathbf{r}'') \left[\langle \mathbf{r}' | \frac{\delta V_H[\rho]}{\delta \rho(\mathbf{r})} | \mathbf{r}'' \rangle + \langle \mathbf{r}' | \frac{\delta V_{xc}[\rho]}{\delta \rho(\mathbf{r})} | \mathbf{r}'' \rangle \right]. \end{aligned} \quad (2.29)$$

Now the results can be written in term of the Dirac distribution:

$$\langle \mathbf{r}' | \frac{\delta V_H[\rho]}{\delta \rho(\mathbf{r})} | \mathbf{r}'' \rangle = \frac{\delta(\mathbf{r}' - \mathbf{r}'')}{|\mathbf{r}' - \mathbf{r}''|}, \quad (2.30)$$

$$\langle \mathbf{r}' | \frac{\delta V_{xc}[\rho]}{\delta \rho(\mathbf{r})} | \mathbf{r}'' \rangle = \delta(\mathbf{r}' - \mathbf{r}'') \delta(\mathbf{r}' - \mathbf{r}) \frac{dV_{xc}[\rho]}{d\rho}(\mathbf{r}). \quad (2.31)$$

Hence, since the wavefunctions squares to the same ρ :

$$\begin{aligned} \sum_p f_p \langle \psi_p | \frac{\delta H_{KS}[\rho]}{\delta \rho(\mathbf{r})} | \psi_p \rangle &= \int d\mathbf{r}' \rho(\mathbf{r}') \left[\frac{1}{|\mathbf{r}' - \mathbf{r}|} + \delta(\mathbf{r}' - \mathbf{r}) \frac{dV_{xc}[\rho]}{d\rho}(\mathbf{r}) \right] \\ &= V_H[\rho](\mathbf{r}) + \rho(\mathbf{r}) \frac{dV_{xc}[\rho]}{d\rho}(\mathbf{r}), \end{aligned} \quad (2.32)$$

which implies the KS lagrangian condition $\frac{\delta E[\rho]}{\delta \rho(\mathbf{r})} = 0$.

While performing the search for the fixed point, the so-called Self Consistent Field (SCF) cycle, the wavefunctions have to be modified between one step and the other, while mantaining orthogonality. The latter can be implemented via a Lagrange multiplier Λ_{pq} , which define the Lagrangian

$$L[\{\psi_p\}] = E[\rho[\{\psi_p\}]] - \sum_{p,q} \Lambda_{pq} (\langle \psi_p | \psi_q \rangle - \delta_{pq}). \quad (2.33)$$

Imposing $\frac{\delta L[\{\psi_p\}]}{\delta \langle \psi_p |} = 0$ gives $\Lambda_{pq} = \langle \psi_q | H_{KS}[\rho] | \psi_p \rangle$. Of course, only wavefunctions which are occupied contribute to the energy. The gradient of the KS energy wrt the wavefunction is then

$$|g_p\rangle = H_{KS}[\rho] | \psi_p \rangle - \sum_q \langle \psi_q | H_{KS}[\rho] | \psi_p \rangle | \psi_q \rangle. \quad (2.34)$$

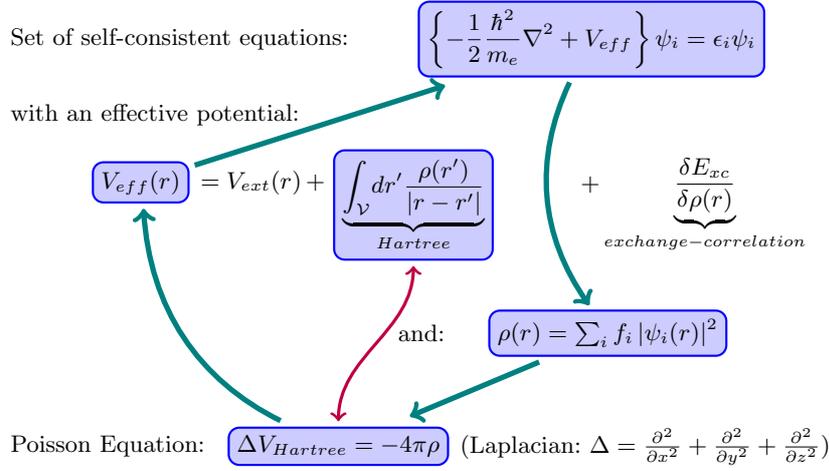


Figure 2. Self-consistent equations used in the diagonalisation scheme.

The vectors $\{|g_p\rangle\}$ provide the direction in which the energy varies the most for a given set of wavefunctions $\{\psi_p\}$. Different algorithms can then be used to find the fixed-point solution. This is the so-called Direct minimisation algorithm. In the figure 4, the flowchart of the operations are indicated in the case of a plane wave basis set. This flowchart is roughly the same as in the case of other basis sets. The main limitation part for systematic basis sets as the number of atoms increases is the orthonormalisation part which scales cubically in function of the number of atoms if the orbitals are extended over the whole system.

Beware that, since it only involves occupied wavefunctions, such algorithm is correctly defined only if *any* KS Hamiltonian of the SCF cycle exhibits an energy gap, and if the updated wavefunction at each step have components onto all the first n eigenspaces of the new Hamiltonian.

2.5 Finding the Kohn-Sham wavefunctions: Diagonalisation of the Hamiltonian

To calculate properly metallic system, the only possibility is to diagonalize the Hamiltonian at each step and populates the Kohn-Sham orbitals in function of the Kohn-Sham eigenvalues.

In the figure 2, the self-consistent equations are shown. At each iteration, the Hamiltonian needs to be diagonalised. We give more details in the plane wave section (see the figure 5) about the different operations. Iterative algorithms are used to diagonalize the Hamiltonian in the case of systematic basis sets because the number of computed orbitals are quite small (by a factor of 100) compared to the number of components. The most used iterative algorithm are conjugate gradient scheme, Davidson [34], Lanczos, RMM-DIIS (Residual Minimum Method – Direct Inversion of the Iterative Subspace used in VASP code [35]) or LOBPCG methods (Locally Optimal Block Preconditioned Conjugate Gradient [36]). Except the conjugate gradient scheme, this algorithms can be parallelized which is really important to handle systems composed of few hundred of atoms.

3. PSEUDOPOTENTIALS

The KS formalism presents thus a procedure to study the electronic properties of a system with many atoms. However, for such a system the interesting properties are determined by the valence electrons of the atoms involved. Electrons close to the nuclei have a behaviour which can be considered independent to the system under consideration. These electrons contribute to the screening of the atomic charge, but

have no significant influence on the behaviour of the peripheral electron. It may thus appear convenient to consider a system in which *only* the valence electrons appear, where the electron-ion interaction potential is substituted by another object, the *pseudopotential*, (PSP) which mode the effect of the core electron.

From a computational viewpoint, the advantage of using pseudopotential approximation is twofold: on one hand, the overall number of electrons in the system is reduced, which make lighter the computational treatment. On the other hand, the PSP operator makes the KS wavefunctions close to the position of the nuclei smoother than the ordinary ion-electron potential. This is also important from the implementation viewpoint since a smooth function is always easier to express numerically.

It can be understood easily that the PSP approximation is less severe than the XC approximation. However, the PSP operator should be defined carefully such that several conditions must be respected. Moreover, the influence of the core electrons on the nuclei must be expressed by the insertion of non-local operators, since the screening of the core electrons is different for any of the multipoles of the electron-ion potential.

4. KOHN-SHAM DFT WITH DAUBECHIES WAVELETS

In the recent years the KS formalism has been proven to be one of the most efficient and reliable first-principle methods for predicting material properties and processes which undergo a quantum mechanical behavior. The high accuracy of the results together with the relatively simple form of the most common exchange-correlation functionals make this method probably the most powerful tool for *ab-initio* simulations of the properties of matter. The computational machinery of DFT calculations has been widely developed in the last decade, giving rise to a plethora of DFT codes. The usage of DFT calculation has thus become more and more common, and its domain of application comprises solid state physics, chemistry, materials science, biology and geology.

From a computational point of view, one of the most important characteristics of a DFT code is the set of basis functions used for expressing the KS orbitals. The domain of applicability of a code is tightly connected to this choice. For example, a non-localised basis set like plane waves is highly suitable for electronic structure calculations of periodic and/or homogeneous systems like crystals or solids, while it is much less efficient in expanding localised information, which has a wider range of components in the reciprocal space. For these reasons DFT codes based on plane waves are not convenient for simulating inhomogeneous or isolated systems like molecules, due to the high memory requirements for such kind of simulations.

A remarkable difference should be also made between codes which use systematic and non-systematic basis sets. A systematic basis set allows us to calculate the exact solution of the KS equations with arbitrarily high precision as the number of basis functions is increased. In other terms, the numerical precision of the results is related to the number of basis functions used to expand the KS orbitals. With such a basis set it is thus possible to obtain results that are free of errors related to the choice of the basis, eliminating a source of uncertainty. A systematic basis set allows us thus to really calculate the solution of a particular exchange correlation functional. On the other hand, an example of a non-systematic set is provided by Gaussian type basis, for which over-completeness may be achieved before convergence. Such basis sets are more difficult to use, since the basis set must be carefully tuned by hand by the user, which will sometimes require some preliminary knowledge of the system under investigation. This is the most important weakness of this popular basis set.

Another property which has a role in the performances of a DFT code is the orthogonality of the basis set. The use of nonorthogonal basis sets requires the calculation of the overlap matrix of the basis function and performing various operations with this overlap matrix such as inverting the matrix. This makes methods based on non-orthogonal basis functions not only more complicated but also slower.

In the figure 3, we give an overview of the different possibilities to solve the Kohn-Sham equations. The choice of a basis set determines strongly the accuracy of a code and the different operations which

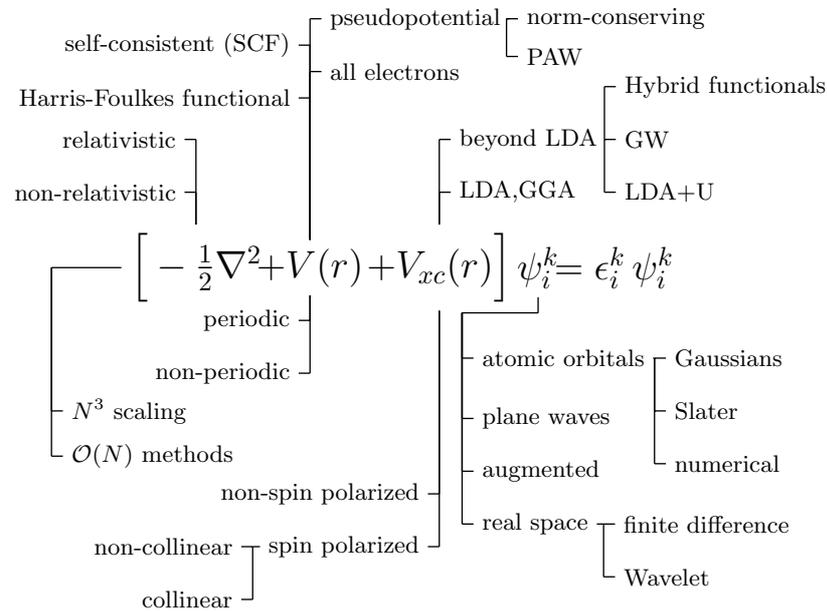


Figure 3. List of options for a DFT code.

need to be computed. The cost of each step in the self-consistent loop is not same and can differ drastically for gaussian or plane wave basis sets.

From the point of view of the developer, some formalisms are easier to program than the other ones. This is the case for plane wave or wavelet basis set in the case, for instance of calculating atomic forces. Another point is the flexibility of the possible boundary conditions (isolated or periodic systems, surfaces or wires). In the section 7, we develop this point applied to the calculation of the Hartree potential *i.e.* the Poisson solver.

We give a short list of codes which is not really exhaustive but give an idea of the diversity of proposed solution to solve the Kohn-Sham equations:

- Plane Waves
 - ABINIT — Louvain-la-Neuve — <http://www.abinit.org>
This code is available under GPL licence and has a strong community of developers and users; The forum discussions are very active and are useful to help the beginners. ABINIT can do electronic structure calculation and calculates many properties based on the linear response as well the many-body perturbation theory (GW method).
 - CPMD — Zurich, Lugano — <http://www.cpmc.org>
The code CPMD (Car-Parrinello Molecular Dynamics) is freely distributed and is one the first developed code based on plane waves and massively parallel. It is used to do geometry optimization, molecular dyanmics and can be combined with other codes in the multiscale approach of QM/MM (Quantum Mechanics, Molecular Modelling).
 - PWSCF — Italy — <http://www.pwscf.org>
The code PWSCF is distributed over the GPL license. He has also a strong community of users and has many capabilities specially to calculate electronic properties based on the linear response as ABINIT.

- VASP — Vienna — <http://cms.mpi.univie.ac.at/vasp>
This code has been tuned to be fast and robust. This code is more dedicated to the calculation of structural properties. This code is widely used and has a strong community of users.
- Gaussian
 - CP2K — <http://cp2k.berlios.de>
This code under GPL license combines a Gaussian basis set to describe the wavefunction and plane waves or wavelet to express the electronic density and calculates the Hartree potential.
 - Gaussian — <http://www.gaussian.com>
Gaussian code is a well-known commercial code created by John Pople.
 - DeMon — <http://www.demon-software.com>
DeMon was originally developed in Montreal and is freely available.
- ADF — Amsterdam —
Amsterdam Density Functional code uses Slater orbitals to express the wavefunctions. It is a commercial code with many capabilities.
- Siesta — Madrid — <http://www.uam.es/departamentos/ciencias/fismateriac/siesta>
Siesta uses a numerical basis sets to express the wavefunctions and plane wave to calculate the electronic density and the Hartree potential.
- Wien — Vienna — <http://www.wien2k.at>
This code uses a full-potential linear augmented plane wave (FPLAPW) basis set tuned to represent with few orbitals the wavefunctions in a solid.
- Real space basis set
 - ONETEP — <http://www.onetep.soton.ac.uk>
This code uses sinus cardinal which can represent exactly a plane wave basis set for a given energy cutoff. $O(N)$ approach is already implemented.
 - BigDFT — http://inac.cea.fr/L_Sim/BigDFT
This is the first code based on wavelet using pseudopotential, massively parallel. It is also integrated in the ABINIT package.
 - GPAW —
Under GPL license, GPAW uses a finite difference scheme with projected-augmented-wave (PAW) pseudopotentials.

During the last years, developers have tried to share common developments as exchange-correlation library (libXC) or input/output libraries (ETSF-IO). The idea is to reuse as much as possible already existing code in order to decrease the cost of development. The main part is the debugging and the maintenance of a code. Using library has the advantage to force the modularity of a code and concentrate the effort only to the original part.

Systematic basis sets are plane wave or wavelet have the advantage to permit an easy control over the accuracy of the calculation. We develop first the specificity of plane wave basis sets and then concentrate on wavelet basis sets.

4.1 Plane waves basis set

Plane waves are widely used as an orthogonal systematic basis set. They are well adapted for periodic systems and based on the Fast Fourier transform. The idea from Car and Parrinello is to express the operators involved in Hamiltonian in the Fourier space for the kinetic operator and in the real space for the local potential. Each time, the operator is diagonal and easy to calculate.

In the figure 4, the flowchart of operations are indicated in the case of the direct minimisation. As we mention already, the main cost becomes the orthonormalisation of wavefunctions which is cubic versus

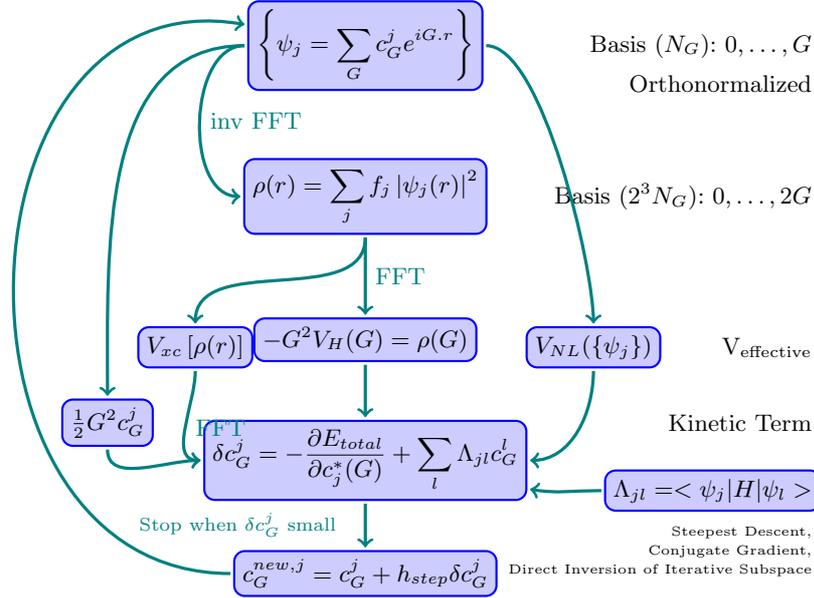


Figure 4. Direct Minimisation: Flowchart for a code based on the plane wave basis set.

the number of atoms because the number of scalar products grows quadratically and the cost of one scalar product is linear. The cost of the application of the Hamiltonian on one wavefunction is $N \log(N)$ due to the Fast Fourier transform which is almost linear. So the cost of calculating the Hamiltonian over the whole set of the wavefunctions grows quadratically.

This means that the use of plane wave basis sets for the Kohn-Sham equations are limited to few hundred of atoms.

We show in the figure 5, the flowchart of the diagonalisation scheme applied to the plane wave basis sets. The advantage is that metallic systems or systems with a small gap can be properly calculated. What we need is to have a good family of pseudopotentials and a good density mixing. If the electronic density coming from the new set of wavefunctions is used directly, the calculation does not converge which is a consequence of the non-linearity of the equations in function of the electronic density. To circumvent this problem, density mixing are used as Anderson, Broyden, DIIS mixing. The robustness of a code is mainly due to the choice of good density mixing.

4.2 Daubechies wavelets family

Daubechies wavelets [3] have virtually all the properties that one might desire for a basis set. They form a systematic orthogonal and smooth basis that is localized both in real and Fourier space and that allows for adaptivity. A DFT approach based on such functions will meet both the requirements of precision and localisation found in many applications. We will in the following describe in detail a DFT method based on a Daubechies wavelets basis set. This method is implemented in a DFT code, named BigDFT, distributed under GNU-GPL license and integrated in the ABINIT [4] software package. In the next few paragraphs we will discuss the importance of the properties of Daubechies wavelets in the context of electronic structure calculations.

A wavelet basis consists of a family of functions generated from a mother function and its translations on the points of a uniform grid of spacing h . The number of basis functions is increased by decreasing the value of h . Thanks to the systematicity of the basis, this will make the numerical

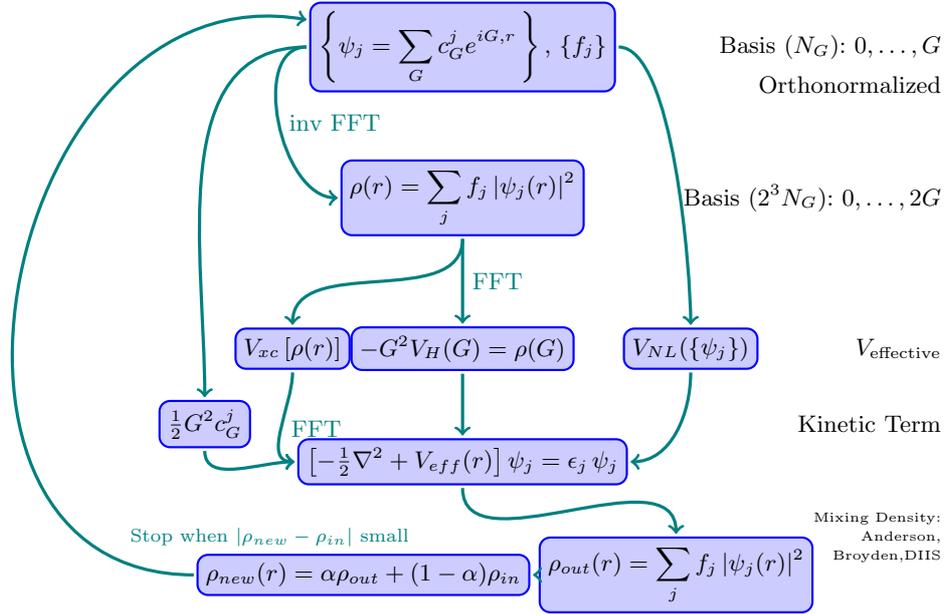


Figure 5. Diagonalisation Scheme: Flowchart.

description more precise. The degree of smoothness determines the speed with which one converges to the exact result as h is decreased. The degree of smoothness increases as one goes to higher order Daubechies wavelets. In our method we use Daubechies wavelets of order 16. This together with the fact that our method is quasi variational gives a convergence rate of h^{14} . Obtaining such a high convergence rate is essential in the context of electronic structure calculations where one needs highly accurate results for basis sets of acceptable size. The combination of adaptivity and a high order convergence rate is typically not achieved in other electronic structure programs using systematic real space methods [6]. An adaptive finite element code, using cubic polynomial shape functions [7], has a convergence rate of h^6 . Finite difference methods have sometimes low [8] h^3 or high convergence rates [9] but are not adaptive.

The most important property of these functions is that they satisfy the so-called refinement equations

$$\phi(x) = \sqrt{2} \sum_{j=1-m}^m h_j \phi(2x - j) \quad (4.1)$$

$$\psi(x) = \sqrt{2} \sum_{j=1-m}^m g_j \psi(2x - j)$$

which establishes a relation between the scaling functions on a grid with grid spacing h and another one with spacing $h/2$. h_j and $g_j = (-1)^j h_{-j+1}$ are the elements of a filter that characterizes the wavelet family, and m is the order of the scaling function-wavelet family. All the properties of these functions can be obtained from the relations (4.1). The full basis set can be obtained from all translations by a certain grid spacing h of the mother function centered at the origin. The mother function is localized, with compact support. The maximally symmetric Daubechies scaling function and wavelet of order 16 that are used in this work are shown in Fig. 6.

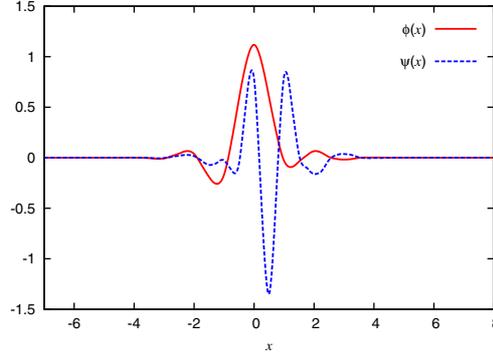


Figure 6. Daubechies scaling function ϕ and wavelet ψ of order 16. Both are different from zero only in the interval from -7 to 8 .

For a three-dimensional description, the simplest basis set is obtained by a set of products of equally spaced scaling functions on a grid of grid spacing h'

$$\phi_{i,j,k}(\mathbf{r}) = \phi(x/h' - i) \phi(y/h' - j) \phi(z/h' - k). \quad (4.2)$$

In other terms, the three-dimensional basis functions are a tensor product of one dimensional basis functions. Note that we are using a cubic grid, where the grid spacing is the same in all directions, but the following description can be straightforwardly applied to general orthorombic grids.

The basis set of Eq. (4.2) is equivalent to a mixed basis set of scaling functions on a twice coarser grid of grid spacing $h = 2h'$

$$\phi_{i,j,k}^0(\mathbf{r}) = \phi(x/h - i) \phi(y/h - j) \phi(z/h - k) \quad (4.3)$$

augmented by a set of 7 wavelets

$$\begin{aligned} \phi_{i,j,k}^1(\mathbf{r}) &= \psi(x/h - i) \phi(y/h - j) \phi(z/h - k) \\ \phi_{i,j,k}^2(\mathbf{r}) &= \phi(x/h - i) \psi(y/h - j) \phi(z/h - k) \\ \phi_{i,j,k}^3(\mathbf{r}) &= \psi(x/h - i) \psi(y/h - j) \phi(z/h - k) \\ \phi_{i,j,k}^4(\mathbf{r}) &= \phi(x/h - i) \phi(y/h - j) \psi(z/h - k) \\ \phi_{i,j,k}^5(\mathbf{r}) &= \psi(x/h - i) \phi(y/h - j) \psi(z/h - k) \\ \phi_{i,j,k}^6(\mathbf{r}) &= \phi(x/h - i) \psi(y/h - j) \psi(z/h - k) \\ \phi_{i,j,k}^7(\mathbf{r}) &= \psi(x/h - i) \psi(y/h - j) \psi(z/h - k) \end{aligned} \quad (4.4)$$

This equivalence follows from the fact that, from Eq. (7.3), every scaling function and wavelet on a coarse grid of spacing h can be expressed as a linear combination of scaling functions at the fine grid level h' and vice versa.

The points of the simulation grid fall into 3 different classes. The points which are very far from the atoms will have virtually zero charge density and thus will not carry any basis functions. The remaining grid points are either in the high resolution region which contains the chemical bonds or in the low resolution regions which contains the exponentially decaying tails of the wavefunctions. In the low resolution region one uses only one scaling function per coarse grid point, whereas in the high resolution region one uses both the scaling function and the 7 wavelets. In this region the resolution is thus doubled

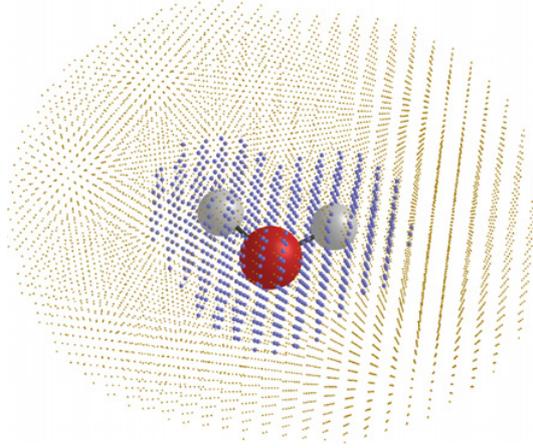


Figure 7. A 2-level adaptive grid around a H_2O molecule. The high resolution grid points carrying both scaling functions and wavelets are shown in blue (larger points), the low resolution grid points carrying only a single scaling function are shown in yellow (smaller points).

in each spatial dimension compared to the low resolution region. Fig. 7 shows the 2-level adaptive grid around a water molecule.

A wavefunction $\Psi(\mathbf{r})$ can thus be expanded in this basis:

$$\Psi(\mathbf{r}) = \sum_{i_1, i_2, i_3} s_{i_1, i_2, i_3} \phi_{i_1, i_2, i_3}^0(\mathbf{r}) + \sum_{j_1, j_2, j_3} \sum_{v=1}^7 d_{j_1, j_2, j_3}^v \phi_{j_1, j_2, j_3}^v(\mathbf{r}) \quad (4.5)$$

The sum over i_1, i_2, i_3 runs over all the grid points contained in the low resolution region and the sum over j_1, j_2, j_3 over all the points contained in the smaller high resolution region.

The decomposition of scaling function into coarser scaling functions and wavelets can be continued recursively to obtain more than 2 resolution levels. We found however that a high degree of adaptivity is not of paramount importance in pseudopotential calculations. In other terms, the pseudopotentials smooth the wavefunctions so that two levels of resolution are enough in most cases to achieve good computational accuracy. In addition, more than two resolution levels lead to more complicated algorithms such as the non-standard operator form [14] that, in turn, lead to larger prefactors.

The transformation from a pure fine scaling function representation (a basis set which contains only scaling functions centered on a finer grid of spacing h') to a mixed coarse scaling function/wavelet representation is done by the fast wavelet transformation [13] which is a convolution and scales linearly with respect to the number of basis functions being transformed.

The wavefunctions are stored in a compressed form where only the nonzero scaling function and wavelets coefficients are stored. The basis set being orthogonal, several operations such as scalar products among different orbitals and between orbitals and the projectors of the non-local pseudopotential can directly be done in this compressed form. In the following sections we will illustrate the main operations which must be performed in the context of a DFT calculation.

5. OVERVIEW OF THE METHOD

The KS wavefunctions $|\Psi_i\rangle$ are eigenfunctions of the KS Hamiltonian, with pseudopotential V_{psp} :

$$\left(-\frac{1}{2} \nabla^2 + V_{\text{KS}}[\rho] \right) |\Psi_i\rangle = \epsilon_i |\Psi_i\rangle. \quad (5.1)$$

The KS potential

$$V_{\text{KS}}[\rho] = V_H[\rho] + V_{\text{xc}}[\rho] + V_{\text{ext}}, \quad (5.2)$$

contains the Hartree potential, solution of the Poisson's equation $\nabla^2 V_H = -4\pi\rho$, the exchange-correlation potential V_{xc} and the external ionic potential V_{ext} acting on the electrons. The method we illustrate in this paper is conceived for isolated systems, namely free boundary conditions.

In our method, we choose the pseudopotential term $V_{\text{ext}} = \sum_{a=1}^N V_{\text{psp}}^{(a)}(\mathbf{r} - \mathbf{R}_a)$ to be of the form of norm-conserving GTH-HGH pseudopotentials [15–17], which have a local and a nonlocal term, $V_{\text{psp}} = V_{\text{local}} + V_{\text{nonlocal}}$. For each of the ions these potentials have this form:

$$V_{\text{local}}(\mathbf{r}) = -\frac{Z_{\text{ion}}}{r} \operatorname{erf}\left(\frac{r}{\sqrt{2}r_{\text{loc}}}\right) + \exp\left[-\frac{1}{2}\left(\frac{r}{r_{\text{loc}}}\right)^2\right] \times \\ \times \left[C_1 + C_2 \left(\frac{r}{r_{\text{loc}}}\right)^2 + C_3 \left(\frac{r}{r_{\text{loc}}}\right)^4 + C_4 \left(\frac{r}{r_{\text{loc}}}\right)^6 \right] \quad (5.3)$$

$$V_{\text{nonlocal}} = \sum_{\ell} \sum_{i,j=1}^3 h_{ij}^{(\ell)} |p_i^{(\ell)}\rangle \langle p_j^{(\ell)}| \quad (5.4)$$

$$\langle \mathbf{r} | p_i^{(\ell)} \rangle = \frac{\sqrt{2} r^{\ell+2(i-1)} \exp\left[-\frac{1}{2}\left(\frac{r}{r_{\ell}}\right)^2\right]}{r_{\ell}^{\ell+(4i-1)/2} \sqrt{\Gamma\left(\ell + \frac{4i-1}{2}\right)}} \sum_{m=-\ell}^{+\ell} Y_{\ell m}(\theta, \phi),$$

where $Y_{\ell m}$ are the spherical harmonics, and r_{loc} , r_{ℓ} are, respectively, the localization radius of the local pseudopotential term and of each projector.

The analytic form of the pseudopotentials together with the fact that their expression in real space can be written in terms of a linear combination of tensor products of one dimensional functions is of great utility in our method.

Each term in the Hamiltonian is implemented differently, and will be illustrated in the following sections. After the application of the Hamiltonian, the KS wavefunctions are updated via a direct minimisation scheme [18], which in its actual implementation is fast and reliable for non-zero gap systems, namely insulators. Since we are using direct minimisation algorithm, at present we have concentrated on systems with a gap, however we see no reason why the method can not be extended to metallic systems.

6. TREATMENT OF KINETIC ENERGY

The matrix elements of the kinetic energy operator among the basis functions of our mixed representation (i.e. scaling functions with scaling functions, scaling function with wavelets and wavelets with wavelets) can be calculated analytically [19]. For simplicity, let us illustrate the application of the kinetic energy operator onto a wavefunction Ψ that is only expressed in terms of scaling functions.

$$\Psi(x, y, z) = \sum_{i_1, i_2, i_3} s_{i_1, i_2, i_3} \phi(x/h - i_1) \phi(y/h - i_2) \phi(z/h - i_3)$$

The result of the application of the kinetic energy operator on this wavefunction, projected to the original scaling function space, has the expansion coefficients

$$\hat{s}_{i_1, i_2, i_3} = -\frac{1}{2h^3} \int \phi(x/h - i_1) \phi(y/h - i_2) \phi(z/h - i_3) \times \\ \times \nabla^2 \Psi(x, y, z) dx dy dz.$$

Analytically the coefficients s_{i_1, i_2, i_3} and \hat{s}_{i_1, i_2, i_3} are related by a convolution

$$\hat{s}_{i_1, i_2, i_3} = \frac{1}{2} \sum_{j_1, j_2, j_3} K_{i_1 - j_1, i_2 - j_2, i_3 - j_3} s_{j_1, j_2, j_3} \quad (6.1)$$

where

$$K_{i_1, i_2, i_3} = T_{i_1} T_{i_2} T_{i_3}, \quad (6.2)$$

where the coefficients T_i can be calculated analytically via an eigenvalue equation:

$$\begin{aligned} T_i &= \int \phi(x) \frac{\partial^2}{\partial x^2} \phi(x - i) dx \\ &= \sum_{v, \mu} 2h_v h_\mu \int \phi(2x - v) \frac{\partial^2}{\partial x^2} \phi(2x - 2i - \mu) dx \\ &= \sum_{v, \mu} 2h_v h_\mu 2^{2-1} \int \phi(y - v) \frac{\partial^2}{\partial y^2} \phi(y - 2i - \mu) dy \\ &= \sum_{v, \mu} h_v h_\mu 2^2 \int \phi(y) \frac{\partial^2}{\partial y^2} \phi(y - 2i - \mu + v) dy \\ &= \sum_{v, \mu} h_v h_\mu 2^2 T_{2i - v + \mu} \end{aligned}$$

Using the refinement equation (7.3), the values of the T_i can be calculated analytically, from a suitable eigenvector of a matrix derived from the wavelet filters [19]. For this reason the expression of the kinetic energy operator is *exact* in a given Daubechies basis.

Since the 3-dimensional kinetic energy filter K_{i_1, i_2, i_3} is a product of three one-dimensional filters (Eq. (6.2)) the convolution in Eq. (6.1) can be evaluated with $3N_1 N_2 N_3 L$ operations for a three-dimensional grid of $N_1 N_2 N_3$ grid points. L is the length of the one-dimensional filter which is 29 for our Daubechies family. The kinetic energy can thus be evaluated with linear scaling with respect to the number of nonvanishing expansion coefficients of the wavefunction. This statement remains true for a mixed scaling function-wavelet basis where we have both nonvanishing s and d coefficients and for the case where the low and high resolution regions cover only parts of the cube of $N_1 N_2 N_3$ grid points.

The Daubechies wavefunctions of degree 16 have an approximation error of h^8 , i.e. the difference between the exact wavefunction and its representation in a finite basis set (Eq. (4.5)) is decreasing as h^8 . The error of the kinetic energy in a variational scheme decreases then as $h^{2 \cdot 8 - 2} = h^{14}$ [20]. As we will see the kinetic energy is limiting the convergence rate in our scheme and the overall convergence rate is thus h^{14} . Figure 8 shows this asymptotic convergence rate.

6.1 Treatment of local potential energy

In spite of the striking advantages of Daubechies wavelets the initial exploration of this basis set [21] did not lead to any algorithm that would be useful for practical electronic structure calculations. This was due to the fact that an accurate evaluation of the local potential energy is difficult in a Daubechies wavelet basis.

By definition, the local potential $V(\mathbf{r})$ can be easily known on the nodes of the uniform grid of the simulation box. Approximating a potential energy matrix element $V_{i, j, k; i', j', k'}$

$$V_{i, j, k; i', j', k'} = \int d\mathbf{r} \phi_{i', j', k'}(\mathbf{r}) V(\mathbf{r}) \phi_{i, j, k}(\mathbf{r})$$

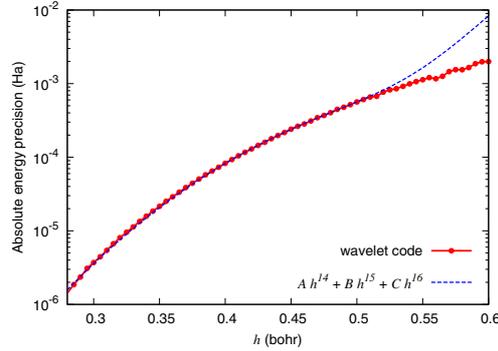


Figure 8. Convergence rate $\mathcal{O}(h^{14})$ of the wavelet code for a test run on a carbon atom. For this run the interpolation parameters are found to be, within 2% accuracy: $A = 344$, $B = -1239$, $C = 1139$. Using lower powers of h for the fit does not give accurate agreement. Other test systems gave comparable convergence rates.

by

$$V_{i,j,k;i',j',k'} \approx \sum_{l,m,n} \phi_{i',j',k'}(\mathbf{r}_{l,m,n}) V(\mathbf{r}_{l,m,n}) \phi_{i,j,k}(\mathbf{r}_{l,m,n})$$

gives an extremely slow convergence rate with respect to the number of grid points used to approximate the integral because a single scaling function is not very smooth, i.e. it has a rather low number of continuous derivatives. A. Neelov and S. Goedecker [23] have shown that one should not try to approximate a single matrix element as accurately as possible but that one should try instead to approximate directly the expectation value of the local potential. The reason for this strategy is that the wavefunction expressed in the Daubechies basis is smoother than a single Daubechies basis function. A single Daubechies scaling function of order 16 (i.e. the corresponding wavelet has 8 vanishing moments) has only 2 continuous derivatives. More precisely its index of Hölder continuity is about 2.7 and the Sobolev space regularity with respect to $p = 2$ is about 2.91 [22]. A single Daubechies scaling function of order 16 has only 4 continuous derivatives. By suitable linear combinations of Daubechies 16 one can however exactly represent polynomials up to degree 7, i.e. functions that have 7 non-vanishing continuous derivatives. The discontinuities get thus canceled by taking suitable linear combinations. Since we use pseudopotentials, our exact wavefunctions are analytic and can locally be represented by a Taylor series. We are thus approximating functions that are approximately polynomials of order 7 and the discontinuities nearly cancel.

Instead of calculating the exact matrix elements we therefore use matrix elements with respect to a smoothed version $\tilde{\phi}$ of the Daubechies scaling functions.

$$\begin{aligned} V_{i,j,k;i',j',k'} &\approx \sum_{l,m,n} \tilde{\phi}_{i',j',k'}(\mathbf{r}_{l,m,n}) V(\mathbf{r}_{l,m,n}) \tilde{\phi}_{i,j,k}(\mathbf{r}_{l,m,n}) = \\ &\sum_{l,m,n} \tilde{\phi}_{0,0,0}(\mathbf{r}_{l-i',m-j',n-k'}) V(\mathbf{r}_{l,m,n}) \tilde{\phi}_{0,0,0}(\mathbf{r}_{l-i,m-j,n-k}) \end{aligned} \quad (6.3)$$

where the smoothed wave function is defined by

$$\tilde{\phi}_{0,0,0}(\mathbf{r}_{l,m,n}) = \omega_l \omega_m \omega_n$$

and ω_l is the “magic filter”. The relation between the true functional values, i.e. the scaling function, and ω is shown in figure 9. Even though Eq. (6.3) is not a particularly good approximation for a single matrix

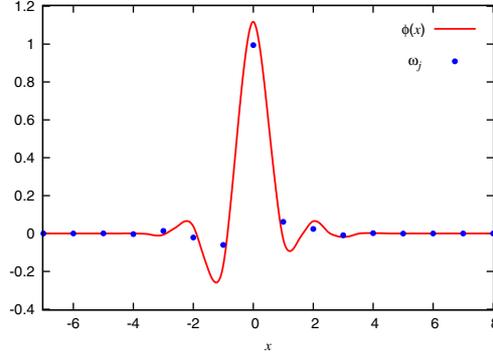


Figure 9. The magic filter ω_j for the least asymmetric Daubechies-16 basis.

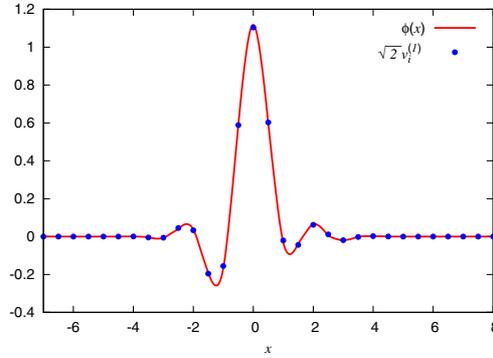


Figure 10. The fine scale magic filter $v_j^{(1)}$ (combination of a wavelet transform and the magic filter in figure 9) for the least asymmetric Daubechies-16 basis, scaled by $\sqrt{2}$ for comparison with the scaling function. The values of the filter on the graph are almost undistinguishable from the values of the scaling function. However, there is a slight difference which is important for the correct asymptotic convergence at small values of grid spacing h .

element it gives an excellent approximation for the expectation values of the local potential energy

$$\int dx \int dy \int dz \Psi(x, y, z) V(x, y, z) \Psi(x, y, z)$$

and also for matrix elements between different wavefunctions

$$\int dx \int dy \int dz \Psi_i(x, y, z) V(x, y, z) \Psi_j(x, y, z)$$

in case they are needed. Because of this remarkable achievement of the filter ω we call it the magic filter.

In practice we do not explicitly calculate any matrix elements but we apply only filters to the wavefunction expansion coefficients as will be shown in the following. This is mathematically equivalent but numerically much more efficient.

Since the operations with the local potential V are performed in the computational box on the double resolution grid with grid spacing $h' = h/2$, we must perform a wavelet transformation before applying the magic filters. These two operations can be combined in one, giving rise to modified magic filters both for scaling functions and wavelets on the original grid of spacing h . These modified magic filters can be obtained from the original ones using the refinement relations and they are shown in Figures 10 and 11. Following the same guidelines as the kinetic energy filters, the smoothed real space values $\tilde{\Psi}_{i,j,k}$ of a wavefunction Ψ are calculated by performing a product of three one-dimensional convolutions with

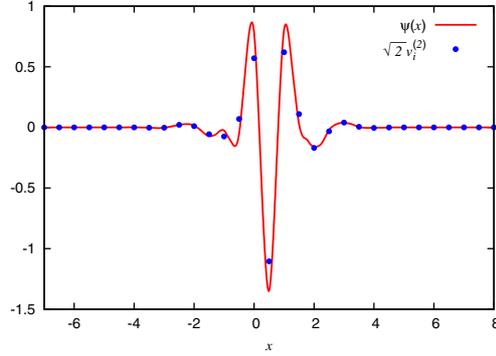


Figure 11. The fine scale magic filter $v_i^{(2)}$ (combination of a wavelet transform and the magic filter in figure 9) for the least asymmetric Daubechies-16 *wavelet*, scaled by $\sqrt{2}$ for comparison with the wavelet itself.

the magic filters along the x , y and z directions. For the scaling function part of the wavefunction the corresponding formula is

$$\tilde{\Psi}_{i_1, i_2, i_3} = \sum_{j_1, j_2, j_3} s_{j_1, j_2, j_3} v_{i_1-2j_1}^{(1)} v_{i_2-2j_2}^{(1)} v_{i_3-2j_3}^{(1)}$$

where $v_i^{(1)}$ is the filter that maps a scaling function on a double resolution grid. Similar convolutions are needed for the wavelet part. The calculation is thus similar to the treatment of the Laplacian in the kinetic energy.

Once we have calculated $\tilde{\Psi}_{i,j,k}$ the approximate expectation value ϵ_V of the local potential V for a wavefunction Ψ is obtained by simple summation on the double resolution real space grid:

$$\epsilon_V = \sum_{j_1, j_2, j_3} \tilde{\Psi}_{j_1, j_2, j_3} V_{j_1, j_2, j_3} \tilde{\Psi}_{j_1, j_2, j_3}$$

The evaluation of the local potential energy ϵ_V converges with a convergence rate of h^{16} to the exact value where h is the grid spacing. Therefore, the potential energy has a convergence rate two powers of h faster than the rate for the kinetic energy.

6.2 Treatment of the non-local pseudopotential

The energy contributions from the non-local pseudopotential have for each angular momentum l the form

$$\sum_{i,j} \langle \Psi | p_i \rangle h_{ij} \langle p_j | \Psi \rangle$$

where $|p_i\rangle$ is a pseudopotential projector. Once applying the Hamiltonian operator, the application of one projector on the wavefunctions requires the calculation of

$$|\Psi\rangle \rightarrow |\Psi\rangle + \sum_{i,j} |p_i\rangle h_{ij} \langle p_j | \Psi \rangle .$$

If we use for the projectors the representation of Eq. (4.5) (i.e. the same as for the wavefunctions) both operations are trivial to perform. Because of the orthogonality of the basis set we just have to calculate scalar products among the coefficient vectors and to update the wavefunctions. The scaling function and

wavelet expansion coefficients for the projectors are given by [13]

$$\int p(\mathbf{r}) \phi_{i_1, i_2, i_3}(\mathbf{r}) d\mathbf{r}, \quad \int p(\mathbf{r}) \psi_{i_1, i_2, i_3}^v(\mathbf{r}) d\mathbf{r}. \quad (6.4)$$

where we used the notation (4.3), (4.4).

The GTH-HGH pseudopotentials [15, 16] have projectors which are written in terms of gaussians times polynomials. This form of projectors is particularly convenient to be expanded in the Daubechies basis. In other terms, since the general form of the projector is

$$\langle \mathbf{r} | p \rangle = e^{-c r^2} x^{\ell_x} y^{\ell_y} z^{\ell_z},$$

the 3-dimensional integrals can be calculated easily since they can be factorized into a product of 3 one-dimensional integrals.

$$\int \langle \mathbf{r} | p \rangle \phi_{i_1, i_2, i_3}(\mathbf{r}) d\mathbf{r} = W_{i_1}(c, \ell_x) W_{i_2}(c, \ell_y) W_{i_3}(c, \ell_z), \quad (6.5)$$

$$W_j(c, \ell) = \int_{-\infty}^{+\infty} e^{-c t^2} t^\ell \phi(t/h - j) dt \quad (6.6)$$

The one-dimensional integrals are calculated in the following way. We first calculate the scaling function expansion coefficients for scaling functions on a one-dimensional grid that is 16 times denser. The integration on this dense grid is done by the well-known quadrature introduced in [27], that coincides with the magic filter [23]. This integration scheme based on the magic filter has a convergence rate of h^{16} and we gain therefore a factor of 16^{16} in accuracy by going to a denser grid. This means that the expansion coefficients are for reasonable grid spacings h accurate to machine precision. After having obtained the expansion coefficients with respect to the fine scaling functions we obtain the expansion coefficients with respect to the scaling functions and wavelets on the required resolution level by one-dimensional fast wavelet transformations. No accuracy is lost in the wavelet transforms and our representation of the projectors is therefore typically accurate to nearly machine precision. In order to treat with the same advantages other pseudopotentials which are not given under the form of gaussians it would be necessary to approximate them by a small number of gaussians.

6.3 XC functionals and implementation of GGA's

To calculate the exchange correlation energy per particle $\epsilon^{xc}[\rho](\mathbf{r})$ and the associated XC potential $V_{xc}(\mathbf{r})$ it is important to have a real-space representation of the density. The magic filter procedure described in Sec. 6.1 can be used also to express the real-point values of the charge density.

$$\rho(\mathbf{r}) = \sum_i n_{\text{occ}}^{(i)} |\tilde{\Psi}_i(\mathbf{r})|^2, \quad (6.7)$$

Evidently, any real-space based implementation of the XC functionals fits well with this density representation. In our program we use the XC functionals as implemented in `libXC` exchange-correlation library.

A traditional finite difference scheme of fourth order is used on the double resolution grid to calculate the gradient of the charge density

$$\hat{\partial}_w \rho(\mathbf{r}_{i_1, i_2, i_3}) = \sum_{j_1, j_2, j_3} c_{i_1, i_2, i_3; j_1, j_2, j_3}^{(w)} \rho_{j_1, j_2, j_3}, \quad (6.8)$$

where $w = x, y, z$. For grid points close to the boundary of the computational volume the above formula requires grid points outside the volume. For free boundary conditions the values of the charge density outside the computational volume in a given direction are taken to be equal to the value at the border of the grid.

As described in Sec. 2.2, the relation between the gradient and the density must be taken into account when calculating V_{xc} in the standard White-Bird approach [26], where the density gradient is considered as an explicit functional of the density. There the XC potential can be split in two terms:

$$V_{xc}(\mathbf{r}_{i_1, i_2, i_3}) = V_{xc}^o(\mathbf{r}) + V_{xc}^c(\mathbf{r}), \quad (6.9)$$

where

$$\begin{aligned} V_{xc}^o(\mathbf{r}_{i_1, i_2, i_3}) &= \epsilon_{xc}(\mathbf{r}) + \rho(\mathbf{r}) \frac{\partial \epsilon_{xc}}{\partial \rho}(\mathbf{r}), \\ V_{xc}^c(\mathbf{r}_{i_1, i_2, i_3}) &= \sum_{j_1, j_2, j_3} \frac{\rho}{|\nabla \rho|} \frac{\partial \epsilon_{xc}}{\partial |\nabla \rho|}(\mathbf{r}_{j_1, j_2, j_3}) \sum_{w=x, y, z} \partial_w \rho(\mathbf{r}_{j_1, j_2, j_3}) c_{j_1, j_2, j_3; i_1, i_2, i_3}^{(w)}, \end{aligned} \quad (6.10)$$

where the ‘‘ordinary’’ part V_{xc}^o is present in the same form of LDA functionals, while the White-Bird ‘‘correction’’ term V_{xc}^c appears only when the XC energy depends explicitly on $|\nabla \rho|$. The $c^{(w)}$ are the coefficients of the finite difference formula used to calculate the gradient of the charge density (6.8).

The evaluation of the XC terms and also, when needed, the calculation of the gradient of the charge density, may easily be performed together with the Poisson solver used to evaluate the Hartree potential. This allows us to save computational time.

7. CALCULATION OF HARTREE POTENTIAL

Electrostatic potentials play a fundamental role in nearly any field of physics and chemistry. Having efficient algorithms to find the electrostatic potential V arising from a charge distribution ρ or, in other words, to solve the Poisson’s equation

$$\nabla^2 V = -4\pi\rho, \quad (7.1)$$

is therefore essential. The large variety of situations in which this equation can be found lead us to face this problem with different choices of the boundary conditions (BC). The long-range behavior of the inverse Laplacian operator make this problem to be strongly dependent on the BC of the system.

The most immediate approach to the Poisson equation can be achieved for periodic BC, where a traditional reciprocal space treatment is both rapid and simple, since the Laplacian matrix is diagonal in a plane wave representation. If the density ρ is originally given in real space, a first Fast Fourier Transformation (FFT) is used to transform the real space data in reciprocal space. The Poisson equation is then solved in reciprocal space and finally the result is transformed back into real space by a second FFT. Because of the FFT’s, the overall computational scaling is ($\mathcal{O}(N \log N)$) with respect to the number of grid points N .

The situation is different if one considers the same problem for different BC, like for example free (isolated) BC. In this case the solution of Poisson’s equation can formally be obtained from a three-dimensional integral:

$$V(\mathbf{r}) = \int d\mathbf{r}' G(|\mathbf{r} - \mathbf{r}'|) \rho(\mathbf{r}'), \quad (7.2)$$

where $G(r) = 1/r$ is the Green function of the Laplacian operator in the unconstrained \mathbb{R}^3 space. The long range nature of the kernel operator G does not allow us to mimic free BC with a very large periodic volume. Consequently, the description of non-periodic systems with a periodic formalism always introduces long-range interactions between super-cells that falsify the results. Due to the simplicity of the plane wave methods, various attempts have been made to generalize the reciprocal space approach to free BC [39–41]. All of them use a FFT at some point, and have thus a $\mathcal{O}(N \log N)$ scaling. These methods have some restrictions and cannot be used blindly. For example, the method by

Füsti-Molnar and Pulay is efficient only for spherical geometries, and the method by Martina and Tuckerman requires artificially large simulation boxes that are expensive numerically. Nonetheless, the usefulness of reciprocal space methods has been demonstrated for a variety of applications, and plane-wave based codes are widely used in the chemical physics community.

Another choice of the BC that is of great interest is for systems that are periodically replicated in two dimensions but with finite extent in the third, namely surface systems. The surface-specific experimental techniques developed in recent years produced important results [42], that can benefit from theoretical prediction and analysis. The development of efficient techniques for systems with such boundary conditions thus became of great importance. Explicit Poisson solvers have been developed in this framework [43–45], with a reciprocal space based treatment. Essentially, these Poisson solvers are built following a suitable generalization for surfaces BC of the same methods that were developed for isolated systems. As for the free BC case, screening functions are present to subtract the artificial interaction between the super-cells in the non-periodic direction. Therefore, they exhibit the same kind of intrinsic limitations, as for example a good accuracy only in the bulk of the computational region, with the consequent need for artificially large simulation boxes which may increase the computational overhead.

Electrostatic potentials can either be calculated by solving the differential Poisson equation or by solving the equivalent integral equation eq. (7.2). The methods that solve the differential equation are iterative and they require various tuning. A good representative of these methods is the multigrid approach [46]. Several different steps such as *smoothing*, *restriction* and *prolongation* are needed in this approach. Each of these steps has to be tuned to optimize speed and accuracy. Approaches based on the integral equation are in contrast straightforward and do not require such tuning.

In the following, we will describe two Poisson solvers compatible with free and surfaces boundary conditions respectively. Contrary to Poisson solvers based on reciprocal space treatment, the fundamental operations of this Poisson solver are based on a mixed reciprocal-real space representation of the charge density. This allows us to naturally satisfy the boundary conditions in the different directions. Screening functions or other approximations are thus not needed.

7.1 Interpolating scaling functions

Interpolating scaling functions (ISF) [48] arise in the framework of wavelet theory [3, 13]. They are one-dimensional functions, and their three main properties are:

- The full basis set can be obtained from all the translations by a certain grid spacing h of the mother function ϕ centered at the origin.
- They satisfy the refinement relation

$$\phi(x) = \sum_{j=-m}^m h_j \phi(2x - j) \quad (7.3)$$

where the h_j 's are the elements of a filter that characterizes the wavelet family, and m is the order of the scaling function. Eq. (7.3) establishes a relation between the scaling functions on a grid with grid spacing h and another one with spacing $h/2$.

- The mother function ϕ is symmetric, with compact support from $-m$ to m . It is equal to one at the origin and to zero at all other integer points (in grid spacing units). The expansion coefficients of any function in this basis are just the values of the function on the grid.
- Given a function in the ISF basis

$$f(x) = \sum_j f_j \phi\left(\frac{x}{h} - j\right) \quad (7.4)$$

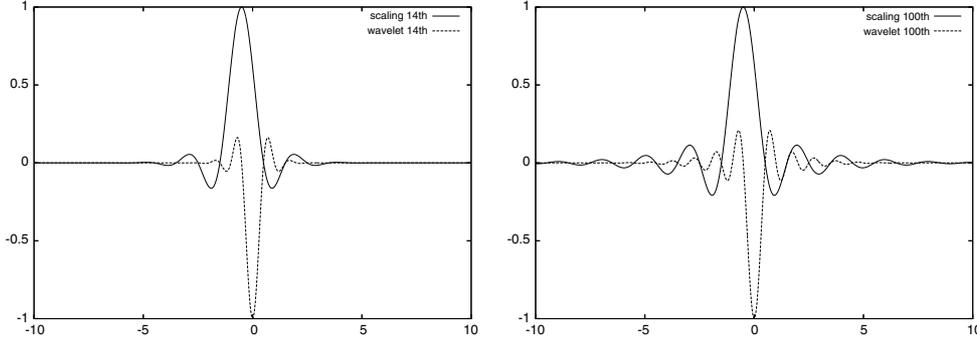


Figure 12. Plots of interpolating scaling functions and wavelets of 14-th and 100-th order.

the first m discrete and continuous moments are identical for a m -th order interpolating wavelet family, i.e.

$$h^\ell \sum_j j^\ell f_j = \int dx x^\ell f(x), \quad (7.5)$$

if $\ell < m$. This follows from the fact, proven in reference [47] that the first m moments of the scaling function obey the formula:

$$M_l = \int \phi(x) x^l dx = \delta_l, \quad l = 0, \dots, m-1 \quad (7.6)$$

Shift the integration variable, we have

$$\begin{aligned} \int \phi(x-j) x^l dx &= \int \phi(t) (t+j)^l dt = \\ &= \int \phi(t) \sum_{p=0}^l C_l^p t^p j^{l-p} dt = j^l \end{aligned}$$

Since the various multipoles of the charge distribution determine the major features of the potential the above equalities tell us that a scaling function representation gives the most faithful mapping between a continuous and discretized charge distribution for electrostatic problems. Fig. 12 shows an 14-th order and 100-th order interpolating scaling function.

7.2 Poisson solver for Free BC

Continuous charge distributions are represented in numerical work typically by their values $\rho_{i,j,k}$ on a grid. It follows from the above described properties of interpolating scaling functions that the corresponding continuous charge density is given by

$$\rho(\mathbf{r}) = \sum_{i_1, i_2, i_3} \rho_{i_1, i_2, i_3} \phi(x-i_1) \phi(y-i_2) \phi(z-i_3) \quad (7.7)$$

Denoting the potential on the grid point $\mathbf{r}_{j_1, j_2, j_3} = (x_{j_1}, y_{j_2}, z_{j_3})$ by $V_{j_1, j_2, j_3} = V(\mathbf{r}_{j_1, j_2, j_3})$ we have

$$V_{j_1, j_2, j_3} = \sum_{i_1, i_2, i_3} \rho_{i_1, i_2, i_3} \int d\mathbf{r}' \frac{\phi_{i_1}(x') \phi_{i_2}(y') \phi_{i_3}(z')}{|\mathbf{r}_{j_1, j_2, j_3} - \mathbf{r}'|}. \quad (7.8)$$

The above integral defines the discrete kernel

$$K(i_1, j_1; i_2, j_2; i_3, j_3) = \int d\mathbf{r}' \phi_{i_1}(x') \phi_{i_2}(y') \phi_{i_3}(z') \frac{1}{|\mathbf{r}_{j_1, j_2, j_3} - \mathbf{r}'|}. \quad (7.9)$$

Since the problem is invariant under combined translations of both the source point (i_1, i_2, i_3) and the observation point (j_1, j_2, j_3) the kernel depends only on the difference of the indices

$$K(i_1, j_1; i_2, j_2; i_3, j_3) = K(i_1 - j_1, i_2 - j_2, i_3 - j_3) \quad (7.10)$$

and the potential V_{j_1, j_2, j_3} can be obtained from the charge density ρ_{i_1, i_2, i_3} by the following 3-dimensional convolution:

$$V_{j_1, j_2, j_3} = \sum_{i_1, i_2, i_3} K(i_1 - j_1, i_2 - j_2, i_3 - j_3) \rho_{i_1, i_2, i_3}. \quad (7.11)$$

Once the Kernel is available in Fourier space, this convolution can be evaluated with two FFTs at a cost of $O(N \log N)$ operations where $N = n_1 n_2 n_3$ is the number of 3-dimensional grid points. Since all the quantities in the above equation are real, real-to-complex FFT's can be used to reduce the number of operations compared to the case where one would use ordinary complex-complex FFT's. Obtaining the Kernel in Fourier space from the Kernel $K(j_1, j_2, j_3)$ in real space requires another FFT.

It remains now to calculate the values of all the elements of the kernel $K(k_1, k_2, k_3)$. Solving a 3-dimensional integral for each element would be much too costly and we use therefore a separable approximation of $1/r$ in terms of Gaussians [12, 49],

$$\frac{1}{r} \simeq \sum_k \omega_k e^{-p_k r^2}. \quad (7.12)$$

In this way all the complicated 3-dimensional integrals become products of simple 1-dimensional integrals. Using 89 Gaussian functions with the coefficients ω_k and p_k suitably chosen, we can approximate $\frac{1}{r}$ with an error less than 10^{-8} in the interval $[10^{-9}, 1]$. If we are interested in a wider range, e.g. a variable R going from zero to L , we can use $r = \frac{R}{L}$:

$$\frac{L}{R} = \sum_k \omega_k e^{-\frac{p_k}{L^2} R^2}, \quad (7.13)$$

$$\frac{1}{R} = \frac{1}{L} \sum_k \omega_k e^{-P_k R^2}, \quad (7.14)$$

$$P_k = \frac{p_k}{L^2}. \quad (7.15)$$

With this approximation, we have that

$$K_{j_1, j_2, j_3} = \sum_{k=1}^{89} \omega_k K_{j_1}(p_k) K_{j_2}(p_k) K_{j_3}(p_k), \quad (7.16)$$

where

$$K_j(p_k) = \int \varphi_j(x) e^{-p_k x^2} dx \quad (7.17)$$

$$= \int \varphi_0(x) e^{-p_k(x-j)^2} dx. \quad (7.18)$$

So we only need to evaluate $89 \times \max(\{n_1, n_2, n_3\})$ integrals of the type

$$K_j(p) = \int \varphi_0(x) e^{-p(x-j)^2} dx, \quad (7.19)$$

for some value of p chosen between $3 \cdot 10^{-5}$ and $3 \cdot 10^{16}$.

The accuracy in calculating the integrals can be further improved by using the refinement relation for interpolating scaling functions (7.3).

From (7.19), we can evaluate $K_i(4p)$ as:

$$K_i(4p) = \int \varphi(x) e^{-4p(x-i)^2} dx \quad (7.20)$$

$$= \frac{1}{2} \int \varphi(x/2) e^{-p(x-2i)^2} dx \quad (7.21)$$

$$= \frac{1}{2} \sum_j h_j \int \varphi_j(x) e^{-p(x-2i)^2} dx \quad (7.22)$$

$$= \frac{1}{2} \sum_j h_j K_{2i-j}(p). \quad (7.23)$$

The best accuracy in evaluating numerically the integral is attained for $p < 1$. For a fixed value of p given by Eq. (7.12), the relation (7.23) is iterated $n = \lceil \log_4(p) \rceil$ times starting with $p_0 = \frac{p}{4^n}$. So the numerical calculation of the integrals $K_i(p)$ is performed as follows: for each p , we compute the number n of required recursions levels and calculate the integral $K_i(p_0)$. The value of n is chosen such that $p_0 \simeq 1$ so we have a gaussian functions not too sharp. The evaluation of the interpolating scaling functions is fast on a uniform grid of points so we perform a simple summation over all the grid points.

7.3 Poisson solver for Surface Boundary conditions

Consider a three-dimensional domain, periodic (with period L_x and L_y) in x and y directions, and non-periodic in z . Without loss of generality, a function f that lives in such a domain can be expanded as

$$f(x, y, z) = \sum_{p_x, p_y} e^{-2\pi i (\frac{p_x}{L_x} x + \frac{p_y}{L_y} y)} f_{p_x, p_y}(z). \quad (7.24)$$

We indicate with $f_{p_x, p_y}(z)$ the one-dimensional function associated to the vector $\vec{p} = (p_x/L_x, p_y/L_y)$ in the reciprocal space of the two dimensional surface. Following these conventions, the Poisson equation (7.1) become a relation between the reciprocal space components of V and ρ :

$$V_{p_x, p_y}(z) = -4\pi \int_{-\infty}^{+\infty} dz' G(2\pi |\vec{p}|; z - z') \rho_{p_x, p_y}(z), \quad (7.25)$$

where $|\vec{p}|^2 = (p_x/L_x)^2 + (p_y/L_y)^2$, and $G(\mu; z)$ is the Green function of the one-dimensional Helmholtz equation:

$$(\partial_z^2 - \mu^2) G(\mu; z) = \delta(z). \quad (7.26)$$

The free BC on the z direction fix the form of the Green function:

$$G(\mu; z) = \begin{cases} -\frac{1}{2\mu} e^{-\mu|z|} & \mu > 0, \\ \frac{1}{2}|z| & \mu = 0. \end{cases} \quad (7.27)$$

In numerical calculations continuous charge distributions are typically represented by their values on a grid. The mixed representation of the charge density given above immediately suggests to use a

plane wave expansion in the periodic directions, which may be easily treated with conventional FFT techniques. For the non-periodic direction z we will use interpolating scaling functions representation. The corresponding continuous charge distribution is thus given by

$$\rho(x, y, z) = \sum_{p_x=-\frac{N_x}{2}}^{\frac{N_x}{2}} \sum_{p_y=-\frac{N_y}{2}}^{\frac{N_y}{2}} \sum_{j_z=0}^{N_z} \rho_{p_x, p_y, j_z} \exp\left(-2\pi i\left(\frac{p_x}{L_x}x + \frac{p_y}{L_y}y\right)\right) \phi\left(\frac{z}{h} - j_z\right), \quad (7.28)$$

where h is the grid spacing in the z direction, and $\phi(j) = \delta_{j,0}$, $j \in \mathbb{Z}$.

Combining eq. (7.25) with 7.28, the discretized Poisson problem thus becomes

$$V_{p_x, p_y, j_z} = -4\pi h \sum_{j'_z} K(2\pi |\vec{p}|; j_z - j'_z) \rho_{p_x, p_y, j'_z}, \quad (7.29)$$

where the quantity (kernel)

$$K(\mu; j) = \int_{-\infty}^{+\infty} du G(\mu; h(j-u))\phi(u) \quad (7.30)$$

is defined via an integral in the dimensionless variable u . Due to the symmetry of ϕ , the kernel is symmetric in the non-periodic direction $K(\mu; j_z) = K(\mu; -j_z)$. The integral bounds can be restricted from $-m$ to m , thanks to the compact support of ϕ .

Once we have calculated the kernel, which will be described below, our numerical procedure is the following. We perform a two-dimensional FFT on our real space charge density to obtain the Fourier coefficients ρ_{p_x, p_y, j'_z} for all the periodic planes. Then we have to solve eq. (7.29). Since this equation is a convolution it can be calculated by zero-padded FFT's. Finally the potential is transformed back from the mixed representation to real space to obtain the potential on the grid by another two-dimensional FFT. Due to the FFT's, the total computational cost is $\mathcal{O}(N \log N)$. Since all quantities are real, the amount of memory and the number of operations for the FFT can be reduced by using real-to-complex FFT's instead of complex-complex FFT's.

It remains now to calculate the values of the kernel function $K(\mu; j)$. The core of the calculation is represented by the function

$$\tilde{K}(\lambda; j) = \begin{cases} \int du e^{-\lambda|u-j|}\phi(u) & \lambda > 0, \\ \int du |u-j|\phi(u) & \lambda = 0. \end{cases} \quad (7.31)$$

The kernel has the properties $K(\mu; j) = -\tilde{K}(\mu h; j)/(2\mu)$ for $\mu > 0$ and $K(0; j) = \tilde{K}(0; j)/2$. A simple numerical integration with the trapezoidal rule is inefficient since $G(\mu; z)$ is not smooth in $z = 0$ while the scaling function varies significantly around the integer points. Thanks to the compact support of the scaling function, this problem can be circumvented with a simple and efficient recursive algorithm. We define two functions $\tilde{K}^{(+)}$ and $\tilde{K}^{(-)}$ such that $\tilde{K}(\lambda; j) = \tilde{K}^{(+)}(\lambda; j) + \tilde{K}^{(-)}(\lambda; j)$, where we have, for $\lambda > 0$

$$\tilde{K}^{(+)}(\lambda; j) = \int_{-\infty}^j du e^{\lambda(u-j)}\phi(u), \quad (7.32)$$

$$\tilde{K}^{(-)}(\lambda; j) = \int_j^{+\infty} du e^{-\lambda(u-j)}\phi(u), \quad (7.33)$$

while with $\lambda = 0$

$$\tilde{K}^{(\pm)}(0; j) = \pm j Z_0^{(\pm)}(j) \mp Z_1^{(\pm)}(j), \quad (7.34)$$

$$Z_\ell^{(+)}(j) = \int_{-\infty}^j du u^\ell \phi(u), \quad (7.35)$$

$$Z_\ell^{(-)}(j) = \int_j^{+\infty} dx u^\ell \phi(u), \quad \ell = 0, 1. \quad (7.36)$$

These objects satisfy recursion relations:

$$\begin{aligned} \tilde{K}^{(\pm)}(\lambda; j+1) &= e^{\mp\lambda} \left[\tilde{K}^{(\pm)}(\lambda; j) \pm e^{\mp\lambda j} D_\lambda^{(\pm)}(j) \right], \\ Z_\ell^{(\pm)}(j+1) &= Z_\ell^{(\pm)}(j) \pm C_\ell(j), \quad \ell = 0, 1, \end{aligned} \quad (7.37)$$

where

$$D_\lambda^{(\pm)}(j) = \int_j^{j+1} du e^{\pm\lambda u} \phi(u), \quad (7.38)$$

$$C_\ell(j) = \int_j^{j+1} du u^\ell \phi(u), \quad \ell = 0, 1. \quad (7.39)$$

From eq. (7.32 – 7.39), and the properties

$$\begin{aligned} \tilde{K}(\lambda; j) &= \tilde{K}(\lambda; -j), \quad \tilde{K}^{(+)}(\lambda; 0) = \tilde{K}^{(-)}(\lambda; 0), \\ Z_1^{(+)}(0) &= Z_1^{(-)}(0), \quad Z_0^{(+)}(0) = Z_0^{(-)}(0) = \frac{1}{2}, \end{aligned} \quad (7.40)$$

the function $\tilde{K}(\lambda; j)$ can be calculated recursively for each $j \in \mathbb{N}$, by knowing $\tilde{K}^{(+)}(\lambda; 0)$ and $Z_1^{(+)}(0)$, then evaluating $D_\lambda^{(\pm)}(j)$ and $C_\ell(j)$ for each value of j . The integrals involved can be calculated with high accuracy with a simple higher-order polynomial quadrature. They are integral of smooth, well-defined quantities, since the interpolating scaling function goes to zero at their bounds. Moreover, for values of j lying outside the support of ϕ we can benefit of a functional relation for calculating the values of the kernel. The support of a m -th order scaling function goes from $-m$ to m , then we have $\forall p > 0$

$$\begin{aligned} K(\mu; m+p) &= e^{-\mu p} K(\mu; m), \quad \mu > 0, \\ K(0; m+p) &= K(0; m) + p Z_0^{(+)}(m). \end{aligned} \quad (7.41)$$

To summarize, we have found an efficient method for evaluating equation (7.30) for $j = 0, \dots, N_z$ and a fixed μ . Instead of calculating $N_z + 1$ integrals of range $2m$, we can obtain the same result by calculating 2 integrals of range m and $4m$ integrals of range 1, with the help of relation (7.41). This will also increase accuracy, since the integrands are always smooth functions, which would not be the case with a naive approach.

The accuracy in calculating the integrals can be further improved by using the refinement relation (7.3) for interpolating scaling functions. For positive λ we have

$$\begin{aligned} \tilde{K}(2\lambda; i) &= \int du e^{-2\lambda|u-i|} \phi(u) \\ &= \frac{1}{2} \int du e^{-\lambda|u-2i|} \phi(u/2) \\ &= \frac{1}{2} \sum_j h_j \int du e^{-\lambda|u-2i|} \phi(u-j) \\ &= \frac{1}{2} \sum_j h_j \tilde{K}(\lambda; 2i-j). \end{aligned} \quad (7.42)$$

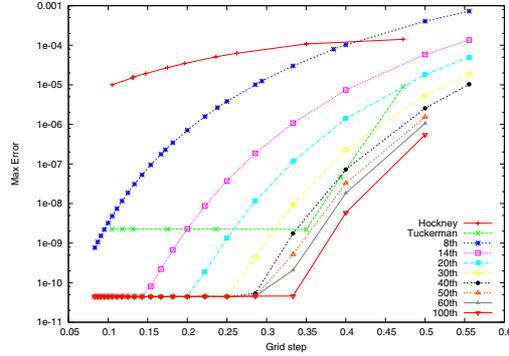


Figure 13. Accuracy comparison between our method with interpolating scaling functions of different orders and the Hockney or Martyna-Tuckerman method as implemented in CPMD. The accuracy of our method is finally limited by the accuracy of the expansion of Eq. (7.12) with 89 terms.

This relation is useful to improve the accuracy in evaluating the kernel for high λ . Since in this case the exponential function is very sharp, it is better to calculate the kernel for lower λ and an enlarged domain and then apply relation (7.42) as many times as needed. The relation (7.41) allows us to enlarge the domain with no additional computational cost. With the help of the above described properties the computational time for evaluating the kernel in Fourier space can be considerably optimized, becoming roughly half of the time needed for its application on a real space density.

7.4 Numerical results and comparison with other methods

These Poisson solvers have a convergence rate of h^m , where m is the order of the interpolating scaling functions used to express the Poisson kernel. Since we use interpolating scaling functions of order 16 the convergence rate of the electrostatic potential is faster than the rate for the kinetic energy. All these Poisson Solvers have one thing in common, they perform explicitly the convolution of the density with the Green's functions of the Poisson's equation. The necessary convolutions are done by a traditional zero-padded FFT procedure which leads to an $\mathcal{O}(N \log N)$ operation count with respect to the number of grid points N . The accuracy of the potential is uniform over the whole volume and one can thus use the smallest possible volume compatible with the requirement that the tails of the wavefunctions have decayed to very small values at the surface of this volume. The fraction of the computational time needed for the solution of the Poisson's equation decreases with increasing system size and is roughly 1% for large systems, see section 12. Moreover, the explicit Green's function treatment of the Poisson's solver allows us to treat isolated systems with a net charge directly without the insertion of compensating charges.

7.4.1 Free BC

For Free BC, we have compared our method with the plane wave methods by Hockney [39] and Martyna and Tuckerman [40] as implemented in the CPMD electronic structure program [37]. As expected Hockney's method does not allow to attain high accuracy. The method by Martyna and Tuckerman has a rapid exponential convergence rate which is characteristic for plane wave methods. Our new method has an algebraic convergence rate of h^m with respect to the grid spacing h . By choosing very high order interpolating scaling functions we can get arbitrarily high convergence rates. Since convolutions are performed with FFT techniques the numerical effort does not increase as the order m is increased. The accuracy shown in Figure 13 for the Martyna and Tuckerman method is the accuracy in the central part of the cube that has 1/8 of the total volume of the computational cell. Outside this volume errors blow

Table 2. The elapsed time in seconds required on a Cray XT3 (based on AMD Opteron processors) to solve Poisson’s equation on a 128^3 grid as a function of the number of processors. Since Poisson’s equation is typically solved many times, the time for setting up the Kernel is not included. Including the set up time of the Kernel increases the total timing by about 50 percent, since one additional FFT is needed.

1	2	4	8	16	32	64
.92	.55	.27	.16	.11	.08	.09

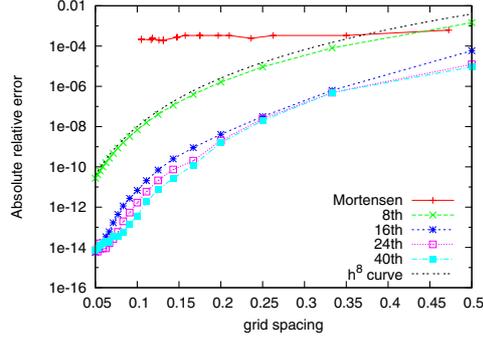


Figure 14. Accuracy comparison (Max. difference between the computed and the analytical result) between our method with scaling functions of different orders and the Mortensen solver for surface systems as implemented in CPMD. The results of the Hockney method are not shown since they are much less precise. The h^8 curve is plotted to show the algebraic decrease of the precision with respect to the grid space h . The accuracy is finally limited by the evaluation of the integral (7.31), which is computed with nearly machine precision.

up. So the main disadvantage of this method is that a very large computational volume is needed in order to obtain accurate results in a sufficiently large target volume. For this reason the less accurate Hockney method is generally preferred in the CPMD program [29].

A strictly localized charge distribution, i.e. a charge distribution that is exactly zero outside a finite volume, can not be represented by a finite number of plane waves. This is an inherent contradiction in all the plane wave methods for the solution of Poisson’s equation under free boundary conditions. For the test shown in Figure 13 we used a Gaussian charge distribution whose potential can be calculated analytically. The Gaussian was embedded in a computational cell that was so large that the tails of the Gaussian were cut off at an amplitude of less than $1.e-16$. A Gaussian can well be represented by a relatively small number of plane waves and so the above described problem is not important. For other localized charge distributions that are less smooth a finite Fourier representation is worse and leads to a spilling of the charge density out of the original localization volume. This will lead to inaccuracies in the potential.

Table 2 shows the required CPU time for a 128^3 problem as a function of the number of processors on a Cray parallel computer. The parallel version is based on a parallel 3-dimensional FFT.

7.4.2 Surfaces BC

Our method was compared with the reciprocal space methods by Hockney [45] and Mortensen [44] (which is a suitable generalization for 2D slab geometries of the method described in [40]) as implemented in the CPMD electronic structure program [37]. The accuracy tests shown in Figure 14 are performed with an analytical charge distribution that is the Laplacian of $V(x, y, z) = \exp(\cos(\frac{2\pi}{L_x}x) + \cos(\frac{2\pi}{L_y}y)) \exp(-\frac{z^2}{50L_z^2} - \tan(\frac{\pi}{L_z}z)^2)$. Its behavior along the xy surface is fully periodic, with all the reciprocal space components taken into account. The $\exp(-\tan(\frac{\pi}{L_z}z)^2)$ guarantees a localized behavior in the non-periodic direction with the potential going explicitly to zero at the borders. This makes also this function suitable for comparison with reciprocal space based approach. The Gaussian factor is

Table 3. Evaluation error of the Hartree energy (Ha) for different values of the size L of the nonperiodic direction, for a system with an electrostatic density which is localized in the nonperiodic direction with characteristic length L_0 . The density of this system is identical to the one used for the accuracy tests of figure 14, with $2L_0 = L_z$ (see text). The accuracy of the Mortensen approach with $L = 2L_0$ is of the same order of the accuracy obtained by our approach with $L = L_0$, which means that to obtain the same precision with Mortensen method the size of the system must be roughly doubled.

L_0/L	0.5	0.6	0.7	0.8	0.9	1
$m=14$	$1 \cdot 10^{-12}$	$7 \cdot 10^{-12}$	$4 \cdot 10^{-6}$	$2 \cdot 10^{-3}$	$3 \cdot 10^{-2}$	$2 \cdot 10^{-1}$
Mortensen	0.2	1.3	3.7	6.0	6.8	6.2

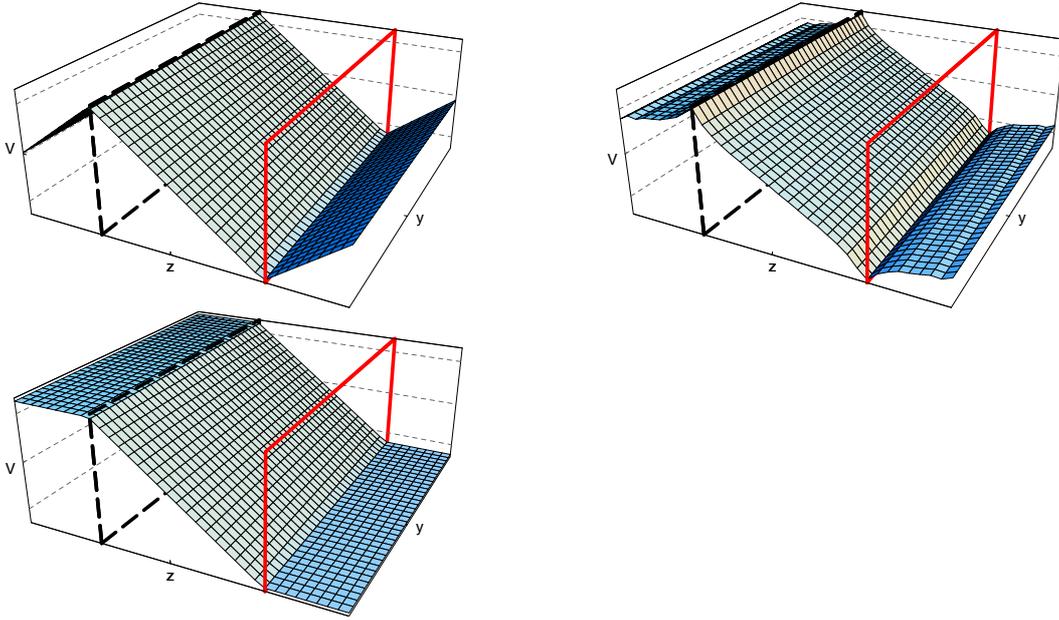


Figure 15. Electrostatic potential V for a system with two periodic planes charged with opposite sign (plane capacitor), oriented along the z direction, calculated by different Poisson solvers. The values of V are taken in the middle of the x (periodic) direction. The position of the positive (black, dashed line) and the negative (red, solid) charged plane is schematically shown in the figure. The represented solution are, from top to bottom, the results from the Mortensen, the Hockney and our Poisson solver.

added to suppress high frequency components. Tests with other analytical functions gave comparable accuracies. The reciprocal space Poisson solvers turn out to be much less precise than our approach, which explicitly preserves the BC along each direction. Moreover, the accuracy shown for the Mortensen approach is calculated only for planes that lies in the bulk of the non-periodic direction (30% of the total volume). Outside of this region, errors in the potential blow up. Table 3 shows the behaviour of the errors in computing the Hartree energy following the size of the system in the nonperiodic direction. To obtain the same accuracy of our approach with the Mortensen method we need a system which is roughly twice larger, which will imply that a very large computational volume is needed to obtain accurate results in a sufficiently large domain of the non-periodic direction.

To show that our method genuinely preserves the boundary conditions appropriate for surfaces we calculated the electrostatic potential for a plane capacitor. For this system only the zero-th Fourier components in the plane are non-vanishing. Figure 15 shows the results either in the Mortensen/Hockney reciprocal space methods or with our approach. For the plane capacitor, the screening function

Table 4. The elapsed time in seconds required on a Cray XT3 (based on AMD Opteron processors) to solve Poisson’s equation with surface BC on a 128^3 grid as a function of the number of processors. The time for setting up the Kernel (around 50% of the total time) is not included. For a large number of processors, the communication time needed to gather the complete potential to all the processors becomes dominant.

1	2	4	8	16	32	64	128
.43	.26	.16	.10	.07	.05	.04	.03

used in the Mortensen approach vanishes, and the solution is equal to what we would have obtained with a fully periodic boundary conditions. To obtain the good “zero electric field” behavior in the borders that we obtain directly with our method one would have to postprocess the solution obtained from the Mortensen method, by adding to the potential a suitable linear function along the non-periodic direction. This is legitimate since a linear function is annihilated by the Laplace operator and the modified potential is thus also a valid solution of the Poisson equation just with different boundary conditions. The Hockney method presents a better qualitative behavior, though the results are not accurate. Only with our approach we get both accurate and physically sound results.

Table 4 shows the required CPU time for solving the Poisson equation on a grid of 128^3 grid points as a function of the number of processors on a Cray XT3 parallel computer. The parallel version is based on a parallel 3-dimensional FFT, where the input/output is properly distributed/gathered to all the processors. The FFT’s are performed using a modified version of the algorithm described in ref.[50] that gives high performances on a wide range of computers.

To summarize, we have presented a method that allows us to obtain accurate potentials arising from charge distributions on surfaces with a $O(N \log N)$ scaling in a mathematically clean way. This method preserves explicitly the required boundary conditions, and can easily be used for applications inside electronic structure codes where the charge density is either given in reciprocal or in real space. The potential applications of these Poisson solver are of great interest in the electronic structure calculations community.

7.5 Exact Exchange operator with ISF Poisson Solver

An example of the applications of the above described Poisson solvers may be found in the calculation of the Exact exchange operator of Eq. (2.22). One may write this operator in the following way:

$$E_x^{HF} = -\frac{1}{2} \sum_{\sigma=1,2} \sum_{p,q} \int d\mathbf{r} \rho_{p,q,\sigma}(\mathbf{r}) V_{p,q,\sigma}(\mathbf{r}), \quad (7.43)$$

$$\rho_{p,q,\sigma}(\mathbf{r}) = \psi_{p,\sigma}(\mathbf{r}) \psi_{q,\sigma}^*(\mathbf{r}), \quad (7.44)$$

$$V_{p,q,\sigma}(\mathbf{r}) = \int d\mathbf{r}' \frac{\rho_{q,p,\sigma}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}, \quad (7.45)$$

$$D_x^{HF} |\psi_{p\sigma}\rangle = - \sum_q f_{q,\sigma} V_{q,p,\sigma} |\psi_{q,\sigma}\rangle. \quad (7.46)$$

The features of the Poisson Solver makes the calculation of the objects above convenient for several reasons. First of all, the ISF Poisson solver technology implements the correct BC automatically, with optimal efficiency and accuracy. Moreover, the advantage of using a basis set which is independent from the atomic position make simpler the calculation of the atomic forces, since no Pulay terms have to be inserted as corrections. Different parallelisation schemes of the Poisson solver application can be implemented, thanks to the flexibility of the real-space implementation of the ISF basis.

8. CALCULATION OF FORCES

Atomic forces can be calculated with the same method used for the application of the Hamiltonian onto a wavefunction. Since the scaling function/wavelet basis is not moving together with atoms, we have no Pulay forces [30] and atomic forces can be evaluated directly through the Feynman-Hellmann theorem. Except for the force arising from the trivial ion-ion interaction, which for the i -th atom is

$$\mathbf{F}_i^{(\text{ionic})} = \sum_{j \neq i} \frac{Z_i Z_j}{R_{ij}^3} (\mathbf{R}_i - \mathbf{R}_j), \quad (8.1)$$

the energy terms which depend explicitly on the atom positions are related to the pseudopotentials. As shown in the previous sections, the GTH-HGH pseudopotentials we are using are based on separable functions [15, 16], and can be splitted into a local and a non-local contribution.

For an atom i placed at position \mathbf{R}_i , the contribution to the energy that comes from the local part of the pseudopotential is

$$E_{\text{local}}(\mathbf{R}_i) = \int d\mathbf{r} V_{\text{local}}(|\mathbf{r} - \mathbf{R}_i|) \rho(\mathbf{r}). \quad (8.2)$$

Where the local pseudopotential can be split into long and a short-ranged terms $V_{\text{local}}(\lambda) = V_L(\lambda) + V_S(\lambda)$, and

$$\begin{aligned} V_L(\lambda) &= -\frac{Z_i}{\lambda} \operatorname{erf}\left(\frac{\lambda}{\sqrt{2}r_\ell}\right), \\ V_S(\lambda) &= \exp\left(-\frac{\lambda^2}{2r_\ell^2}\right) \left[C_1 + C_2 \left(\frac{\lambda}{r_\ell}\right)^2 + C_3 \left(\frac{\lambda}{r_\ell}\right)^4 + C_4 \left(\frac{\lambda}{r_\ell}\right)^6 \right], \end{aligned} \quad (8.3)$$

where the C_i and r_ℓ are the pseudopotential parameters, depending on the atom of atomic number Z_i under consideration. The energy contribution $E_{\text{local}}(\mathbf{R}_i)$ can be rewritten in an equivalent form. It is straightforward to verify that

$$E_{\text{local}}(\mathbf{R}_i) = \int d\mathbf{r} \rho_L(|\mathbf{r} - \mathbf{R}_i|) V_H(\mathbf{r}) + \int d\mathbf{r} V_S(|\mathbf{r} - \mathbf{R}_i|) \rho(\mathbf{r}), \quad (8.4)$$

where V_H is the Hartree potential, and ρ_L is such that $\nabla_{\mathbf{r}}^2 V_L(|\mathbf{r} - \mathbf{R}_i|) = -4\pi\rho_L(|\mathbf{r} - \mathbf{R}_i|)$. This analytical transformation remains also valid in our procedure for solving the discretized Poisson's equation. From equation (8.4) we can calculate

$$\rho_L(\lambda) = -\frac{1}{(2\pi)^{3/2}} \frac{Z_i}{r_\ell^3} e^{-\frac{\lambda^2}{2r_\ell^2}}, \quad (8.5)$$

which is a localized (thus short-ranged) function. The forces coming from the local pseudopotential are thus

$$\begin{aligned} \mathbf{F}_i^{(\text{local})} &= -\frac{\partial E_\ell(\mathbf{R}_i)}{\partial \mathbf{R}_i} \\ &= \frac{1}{r_\ell} \int d\mathbf{r} \frac{\mathbf{r} - \mathbf{R}_i}{|\mathbf{r} - \mathbf{R}_i|} \left[\rho'_L(|\mathbf{r} - \mathbf{R}_i|) V_H(\mathbf{r}) + V'_S(|\mathbf{r} - \mathbf{R}_i|) \rho(\mathbf{r}) \right], \end{aligned} \quad (8.6)$$

where

$$\rho'_L(\lambda) = \frac{1}{(2\pi)^{3/2}} \frac{Z_{ion}}{r_{loc}^4} \lambda e^{-\frac{\lambda^2}{2r_\ell^2}},$$

$$V'_S(\lambda) = \frac{\lambda}{r_\ell} e^{-\frac{\lambda^2}{2r_\ell^2}} \left[(2C_2 - C_1) + (4C_3 - C_2) \left(\frac{\lambda}{r_\ell} \right)^2 + (6C_4 - C_3) \left(\frac{\lambda}{r_\ell} \right)^4 - C_4 \left(\frac{\lambda}{r_\ell} \right)^6 \right]. \quad (8.7)$$

Within this formulation, the contribution to the forces from the local part of pseudopotential is written in terms of integrals with localized functions (gaussians times polynomials) times the charge density and the Hartree potential. This allows us to perform the integrals only in a relatively small region around the atom position and to assign different integrations to different processors. Moreover, the calculation is performed with almost linear ($\mathcal{O}(N \log N)$) scaling.

The contribution to the energy that comes from the nonlocal part of the pseudopotential is, as we saw in section 6.2,

$$E_{\text{nonlocal}}(\mathbf{R}_i) = \sum_l \sum_{mn} \langle \Psi | p_m^l(\mathbf{R}_i) \rangle h_{mn}^l \langle p_n^l(\mathbf{R}_i) | \Psi \rangle, \quad (8.8)$$

where we wrote explicitly the dependence of the projector on the atom position \mathbf{R}_i . The contribution of this term to the atomic forces is thus

$$\mathbf{F}_i^{(\text{nonlocal})} = - \sum_l \sum_{m,n} \langle \Psi | \frac{\partial p_m^l(\mathbf{R}_i)}{\partial \mathbf{R}_i} \rangle h_{mn} \langle p_n^l(\mathbf{R}_i) | \Psi \rangle$$

$$- \sum_l \langle \Psi | p_m^l(\mathbf{R}_i) \rangle h_{mn} \langle \frac{\partial p_n^l(\mathbf{R}_i)}{\partial \mathbf{R}_i} | \Psi \rangle. \quad (8.9)$$

Expressing the derivatives of the projectors in the Daubechies basis, the evaluation of the scalar products is straightforward. The scaling functions - wavelets expansion coefficients of the projector derivatives can be calculated with machine precision accuracy in the same way as the projectors themselves were calculated. This is due to the fact that the derivative of the projectors are like the projectors themselves products of gaussians and polynomials.

9. PRECONDITIONING

As already mentioned, direct minimisation of the total energy is used to find the converged wavefunctions. The gradient g_i of the total energy with respect to the i -th wavefunction $|\Psi_i\rangle$ is given by

$$|g_i\rangle = H|\Psi_i\rangle - \sum_j \Lambda_{ij} |\Psi_j\rangle, \quad (9.1)$$

where $\Lambda_{ij} = \langle \Psi_j | H | \Psi_i \rangle$ are the Lagrange multipliers enforcing the orthogonality constraints. Convergence is achieved when the average norm of the residue $(\overline{|g_i| |g_i|})^{1/2}$ is below an user-defined numerical tolerance.

Given the gradient direction at each step, several algorithms can be used to improve convergence. In our method we use either preconditioned steepest-descent algorithm or preconditioned DIIS method [28, 29]. These methods work very well to improve the convergence for non-zero gap systems if a good preconditioner is available.

The preconditioning gradient $|\tilde{g}_i\rangle$ which approximately points in the direction of the minimum is obtained by solving the linear system of equations obtained by discretizing the equation

$$\left(\frac{1}{2} \nabla^2 - \epsilon_i \right) \tilde{g}_i(\mathbf{r}) = g_i(\mathbf{r}). \quad (9.2)$$

The values ϵ_i are approximate eigenvalues obtained by a subspace diagonalization in a minimal basis of atomic pseudopotential orbitals during the generation of the input guess. For isolated systems, the values of the ϵ_i for the occupied states are always negative, therefore the operator of Eq. (9.2) is positive definite.

Eq. (9.2) is solved by a preconditioned conjugate gradient (CG) method. The preconditioning is done by using the diagonal elements of the matrix representing the operator $\frac{1}{2}\nabla^2 - \epsilon_i$ in a scaling function-wavelet basis. In the initial step we use ℓ resolution levels of wavelets where ℓ is typically 4. To do this we have to enlarge the domain where the scaling function part of the gradient is defined to a grid that is a multiple of 2^ℓ . This means that the preconditioned gradient \tilde{g}_i will also exist in a domain that is larger than the domain of the wavefunction Ψ_i . Nevertheless this approach is useful since it allows us to obtain rapidly a preconditioned gradient that has the correct overall shape. In the following iterations of the conjugate gradient we use only one wavelet level in addition to the scaling functions for preconditioning. In this way we can do the preconditioning exactly in the domain of basis functions that are used to represent the wavefunctions (Eq. 4.5). A typical number of CG iterations necessary to obtain a meaningful preconditioned gradient is 5.

10. ORTHOGONALIZATION

We saw the need of keeping the wavefunctions Ψ_i orthonormal at each step of the minimisation loop. This means that the overlap matrix S , with matrix elements

$$S_{ij} = \langle \Psi_j | \Psi_i \rangle \quad (10.1)$$

must be equal to the identity matrix.

All the orthogonalization algorithms have a cubic complexity causing this part of the program to dominate for large systems, see Fig. 20. We therefore optimized this part carefully and found that a pseudo-Gram-Schmidt algorithm that uses a Cholesky factorization of the overlap matrix S is the most efficient method on parallel computers. In the following, we discuss the reasons for this choice by comparing it to two other orthogonalization algorithms: classical Gram-Schmidt and Loewdin orthogonalizations.

10.1 Gram-Schmidt orthogonalization

The classical Gram-Schmidt orthonormalization algorithm generates an orthogonal set of orbital $\{|\bar{\Psi}_i\rangle\}$ out of a non-orthogonal set $\{|\Psi_i\rangle\}$, by processing separately each orbital. The overlap of the currently processed orbital $|\Psi_i\rangle$ with the set of the already processed orbitals $\{|\bar{\Psi}_j\rangle\}_{j=1,\dots,i-1}$ is calculated and is removed from $|\Psi_i\rangle$. Thereafter, the transformed orbital $|\bar{\Psi}_i\rangle$ is normalized.

$$|\bar{\Psi}_i\rangle = |\Psi_i\rangle - \sum_{j=1}^{i-1} \langle \bar{\Psi}_j | \Psi_i \rangle |\bar{\Psi}_j\rangle \quad (10.2)$$

$$|\bar{\Psi}_j\rangle \longrightarrow \frac{|\bar{\Psi}_j\rangle}{\sqrt{\langle \bar{\Psi}_j | \bar{\Psi}_j \rangle}} \quad (10.3)$$

The algorithm consists of the calculation of $n(n+1)/2$ scalar products and wavefunction updates. If the coefficients of each orbital are distributed among several processors $n(n+1)/2$ communication steps are needed to sum up the various contributions from each processor to each scalar product. Such a large number of communication steps leads to a large latency overhead on a parallel computer and therefore to poor performances.

10.2 Loewdin orthogonalization

The Loewdin orthonormalization algorithm is based on the following equation:

$$|\bar{\Psi}_i\rangle = \sum_j S_{ij}^{-\frac{1}{2}} |\Psi_j\rangle, \quad (10.4)$$

where a new set of orthonormal orbitals $|\bar{\Psi}_i\rangle$ is obtained by multiplying the inverse square-root of the overlap matrix S with the original orbital set.

The implementation of this algorithm requires that the symmetric overlap matrix S is calculated. In contrast to the classical Gram-Schmidt algorithm the matrix elements S_{ij} depend on the original set of orbitals and can be calculated in parallel in the case where each processor holds a certain subset of the coefficients of each wavefunction. At the end of this calculation a single communication step is needed to sum up the entire overlap matrix out of the contributions to each matrix element calculated by the different processors. Since S is an hermitian positive definite matrix, there exist a unitary matrix U which diagonalizes $S = U^* \Lambda U$, where Λ is a diagonal matrix with positive eigenvalues. The inverse square-root of S is then given by $S^{-\frac{1}{2}} = U^\dagger \Lambda^{-\frac{1}{2}} U$. Hence, an eigenvalue problem must be solved in order to find U and Λ .

10.3 Pseudo Gram-Schmidt using Cholesky Factorization

In this scheme a Cholesky factorization of the overlap matrix $S = LL^T$ is calculated. The new orthonormal orbitals are obtained by

$$|\bar{\Psi}_i\rangle = \sum_j (L_{ij}^{-1}) |\Psi_j\rangle, \quad (10.5)$$

and are equivalent to the orbitals obtained by the classical Gram-Schmidt. The procedure for calculating the overlap matrix out of the contributions calculated by each processor is identical to the Loewdin case. Instead of solving an eigenvalue problem we have however to calculate the decomposition of the overlap matrix. This can be done much faster. This algorithm also requires only one communication step on a parallel computer but has a lower pre-factor than the Loewdin scheme.

11. PARALLELIZATION

Two data distribution schemes are used in the parallel version of our program. In the orbital distribution scheme, each processor works on one or a few orbitals for which it holds all its scaling function and wavelet coefficients. In the coefficient distribution scheme (see Fig. 16) each processor holds a certain subset of the coefficients of all the orbitals. Most of the operations such as applying the Hamiltonian on the orbitals, and the preconditioning is done in the orbital distribution scheme. This has the advantage that we do not have to parallelize these routines and we therefore achieve almost perfect parallel speedup. The calculation of the Lagrange multipliers that enforce the orthogonality constraints onto the gradient as well as the orthogonalization of the orbitals is done in the coefficient distribution scheme (Fig. 17). For the orthogonalization we have to calculate the matrix $\langle \Psi_j | \Psi_i \rangle$ and for the Lagrange multipliers the matrix $\langle \Psi_j | H | \Psi_i \rangle$. So each matrix element is a scalar product and each processor is calculating the contribution to this scalar product from the coefficients it is holding. A global reduction sum is then used to sum the contributions to obtain the correct matrix. Such sums can easily be performed with the very well optimized BLAS-LAPACK libraries. Switch back and forth between the orbital distribution scheme and the coefficient distribution scheme is done by the MPI global transposition routine `MPI_ALLTOALL`. For parallel computers where the cross sectional bandwidth [33] scales well with the number of processors this global transposition does not require a lot of CPU time. The most time

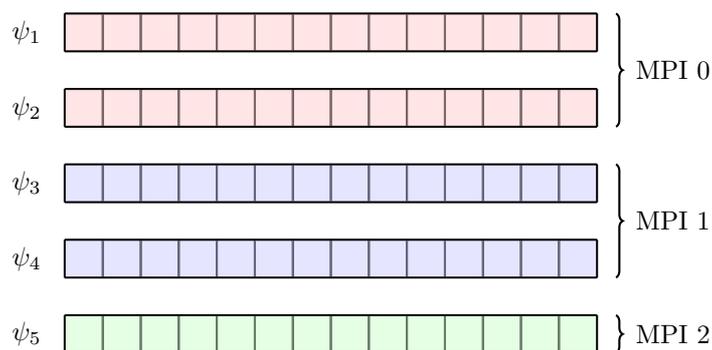


Figure 16. Orbital distribution scheme.



Figure 17. Coefficient distribution scheme.

consuming communication is the global reduction sum required to obtain the total charge distribution from the partial charge distribution of the individual orbital.

11.1 OpenMP parallelisation

In the parallelisation scheme of the BigDFT code another level of parallelisation was added via OpenMP directive. In particular, all the convolutions and the linear algebra part can be executed in multi-threaded mode. This add furthe flexibility on the parallelisation scheme. At present, several strategies are under analysis for systems with different sizes to understand the best repartition of the data between nodes such as to minimise the computational overhead.

12. PERFORMANCE RESULTS

We have applied our method on different molecular systems in order to test its performances. As expected, the localization of the basis set allows us to reduce considerably the number of degrees of freedom (i.e. the number of basis functions which must be used) to attain a given absolute precision with respect to a plane wave code. This fact reduces the memory requirements and the number of floating point operations. Figure 18 shows the comparison of the absolute precision in a calculation of a 44 atom molecule as a function of the number of degrees of freedom used for the calculation. In table 5 the

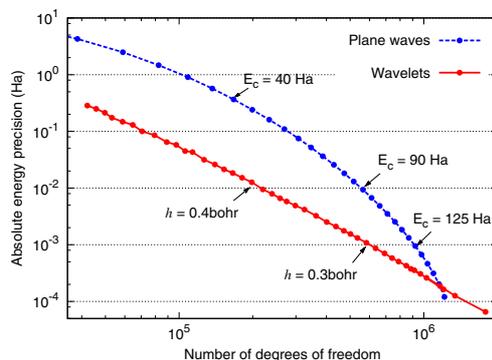


Figure 18. Absolute precision (not precision per atom) as a function of the number of degrees of freedom for a cinchonidine molecule (44 atoms). Our method is compared with a plane wave code. In the case of the plane wave code the plane wave cutoff and the volume of the computational box were chosen such as to obtain the required precision with the smallest number of degrees of freedom. In the case of our wavelet program the grid spacing h and the localization radii were optimized. For very high accuracies the exponential convergence rate of the plane waves beats the algebraic convergence rate of the wavelets. Such high accuracies are however not required in practice. Since convolutions can be executed at very high speed the wavelet code is faster than the plane wave code at any accuracy even if the number of degrees of freedom are similar (see table 5).

Table 5. Computational time in seconds for a single minimization iteration for different runs of the cinchonidine molecule used for the plot in figure 18. The timings for different cutoff energies E_c for the plane waves runs are shown. The input parameters for the wavelet runs are chosen such as to obtain the same absolute precision of the plane wave calculations. The plane wave runs are performed with the ABINIT code, which uses iterative diagonalization and with CPMD code [37] in direct minimization. These timings are taken from a serial run on a 2.4GHz AMD Opteron CPU.

E_c (Ha)	ABINIT (s)	CPMD (s)	Abs. Precision	Wavelets(s)
40	403	173	$3.7 \cdot 10^{-1}$	30
50	570	207	$1.6 \cdot 10^{-1}$	45
75	1123	422	$2.5 \cdot 10^{-2}$	94
90	1659	538	$9.3 \cdot 10^{-3}$	129
145	4109		$2 \cdot 10^{-4}$	474

comparison of the timings of a single SCF cycle with respect to two other plane wave based codes are shown. Since the system is relatively small the cubic terms do not dominate. For large systems of several hundred atoms the gain in CPU time compared to a plane wave program is proportional to the reduction in the number of degrees of freedom (compare Eq. (12.1)) and can thus be very significant as one can conclude from Fig. 18.

The parallelization scheme of the code has been tested and has given the efficiency detailed in Figure 19. The overall efficiency is always higher than 88%, also for large systems with a big number of processors.

It is also interesting to see which is the computational share of the different sections of the code with respect to the total execution time. Figure 20 shows the percentage of the computational time for the different sections of the code as a function of the number of orbitals while keeping constant the number of orbitals per processor. The different sections considered are the application of the Hamiltonian (kinetic, local plus nonlocal potential), the construction of the density, the Poisson solver for creating the Hartree potential, the preconditioning-DIIS, and the operations needed for the orthogonality constraint as well as the orthogonalization, which are mainly matrix-matrix products or matrix decompositions.

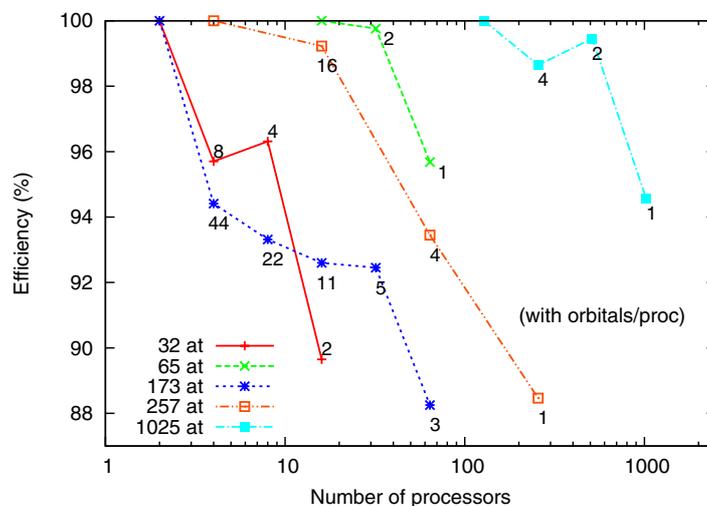


Figure 19. Efficiency of the parallel implementation of the code for several runs with different number of atoms. The number close to each point indicates the number of orbitals treated by each processors, in the orbital distribution scheme.

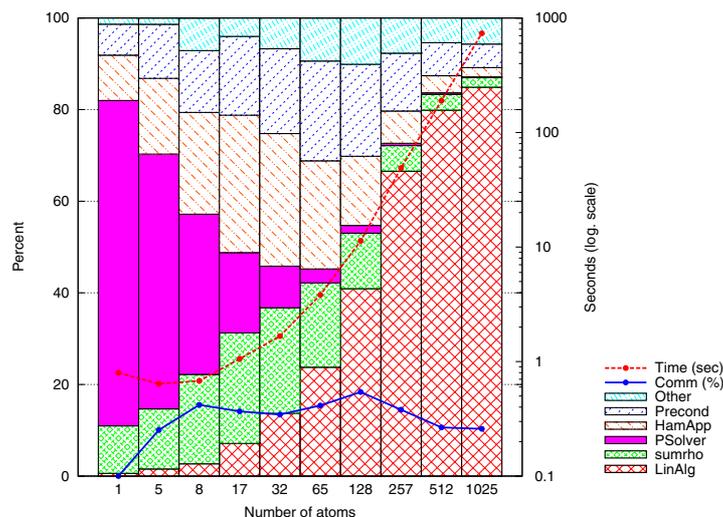


Figure 20. Relative importance of different code sections as a function of the number of atoms of a simple alkane chain, starting from single carbon atom. The calculation is performed in parallel such that each processor holds the same number of orbitals (two in this figure). Also the time in seconds for a single minimization iteration is indicated, showing the asymptotic cubic scaling of present implementation.

These operations are all performed by linear algebra subroutines provided by the LAPACK libraries [38]. Also, the percentage of the communication time is shown. While for relatively small systems the most time-dominating part of the code is related to the Poisson solver, for large systems the most expensive section is by far the calculation of the linear algebra operations. The operations performed in this section scales cubically with respect to the number of atoms. Apart from the Cholesky factorization, which has a scaling of $\mathcal{O}(n_{\text{orb}}^3)$, where n_{orb} is the number of orbitals, the cubic terms are of the form

$$\mathcal{O}(n \cdot n_{\text{orb}}^2), \quad (12.1)$$

where n is the number of degrees of freedom, i.e. the number of scaling function and wavelet expansion coefficients. Both the calculation of the overlap matrix in Eq. (10.1) and the orthogonality transformation of the orbitals in Eq. (10.5) lead to this scaling. The number of the coefficients n is typically much larger than the number of orbitals.

13. CONCLUSIONS

In this contribution we have shown the principal features of an electronic structure pseudopotential method based on Daubechies wavelets. Their properties make this basis set a powerful and promising tool for electronic structure calculations. The matrix elements, the kinetic energy and nonlocal pseudopotentials operators can be calculated analytically in this basis. The other operations are mainly based on convolutions with short-range filters, which can be highly optimized in order to obtain good computational performances. Our code shows high systematic convergence properties, very good performances and an excellent efficiency for parallel calculations. This code is integrated in the ABINIT software package and is freely available under GNU-GPL license. At present, several developments are in progress to improve the features of this code. Mainly, they concern the extension of this formalism to fully periodic systems and surfaces, as well as the inclusion of non-collinear spin-polarized XC functionals. A linear scaling version of this wavelet code is also under preparation.

References

- [1] J. Perdew, K. Burke and M. Ernzerhof, Phys. Rev. Lett **77**, 3865 (1996)
- [2] M. Dion, H. Rydberg, E. Schröder, D. C. Langreth, and B. I. Lundqvist, Phys. Rev. Lett. **92**, 246401 (2004)
- [3] I. Daubechies, “*Ten Lectures on Wavelets*”, SIAM, Philadelphia (1992)
- [4] X. Gonze, J.-M. Beuken, R. Caracas, F. Detraux, M. Fuchs, G.-M. Rignanese, L. Sindic, M. Verstraete, G. Zerah, F. Jollet, M. Torrent, A. Roy, M. Mikami, Ph. Ghosez, J.-Y. Raty, D.C. Allan. Computational Materials Science **25**, 4780–492 (2002). <http://www.abinit.org>
- [5] http://inac.cea.fr/sp2m/L_Sim/BigDFT
<http://www.unibas.ch/comphys/comphys/SOFTWARE>
- [6] Thomas L. Beck, Rev. Mod. Phys. **72**, 1041 (2000)
- [7] J. E. Pask, B. M. Klein, C. Y. Fong, and P. A. Sterne Phys. Rev. B **59**, 12352 (1999)
- [8] J. J. Mortensen, L. B. Hansen, and K. W. Jacobsen Phys. Rev. B **71**, 035109 (2005)
- [9] J. R. Chelikowsky, N. Troullier, Y. Saad, Phys. Rev. Lett. **72**, 1240 (1994).
- [10] Stefan Goedecker, Rev. Mod. Phys. **71**, 1085 (1999)
- [11] T. A. Arias, Rev. Mod. Phys. **71**, 267 (1999)
- [12] T. Yanai, G. I. Fann, Z. Gan, R. J. Harrison, and G. Beylkin, J. Chem. Phys. **121**, 6680 (2004)
- [13] S. Goedecker, “*Wavelets and their application for the solution of partial differential equations*”, Presses Polytechniques Universitaires Romandes, Lausanne, Switzerland 1998, (ISBN 2-88074-398-2)
- [14] G. Beylkin, R. Coifman and V. Rokhlin, Comm. Pure and Appl. Math. **44**, 141 (1991)
- [15] S. Goedecker, M. Teter, J. Hutter, Phys. Rev. B **54**, 1703 (1996)
- [16] C. Hartwigsen, S. Goedecker and J. Hutter, Phys. Rev. B **58**, 3641 (1998)
- [17] M. Krack, Theor. Chem. Acc. **114**, 145 (2005)
- [18] M. Payne, M. Teter, D. Allan, T. Arias and J. Joannopoulos, Rev. of Mod. Phys. **64**, 1045 (1992)
- [19] G. Beylkin, *SIAM J. on Numerical Analysis* **6**, 1716 (1992)
- [20] J. Strang, G. J. Fix, *An analysis of the Finite Element Method*, Wellesley-Cambridge Press, 1988
- [21] C. J. Tymczak and Xiao-Qian Wang, Phys. Rev. Lett. **78**, 3654 (1997)
- [22] Ka-Sing Lau and Qiyu Sun, Proceedings of the American Mathematical Society, **128**, 1087 (2000)
- [23] A. I. Neelov and S. Goedecker, J. of Comp. Phys. **217**, 312–339 (2006)

- [24] L. Genovese, T. Deutsch, A. Neelov, S. Goedecker, G. Beylkin, J. Chem. Phys. **125**, 074105 (2006)
- [25] L. Genovese, T. Deutsch, S. Goedecker, J. Chem. Phys. **127**, 054704 (2007)
- [26] J. A. White and D. M. Bird, Phys. Rev. B **50**, 4954 (1994)
- [27] B. R. Johnson, J. P. Modisette, P. J. Nordlander and J. L. Kinsey, J. Chem. Phys. **110**, 8309 (1999)
- [28] P. Pulay, Chem. Phys. Lett., **73**, 393 (1980)
- [29] J. Hutter, H.P. Lüthi and M. Parrinello, Comp. Mat. Sci. **2** 244 (1994)
- [30] P. Pulay, in *Modern Theoretical Chemistry*, H. F. Schaefer editor, (Plenum Press, New York) (1977)
- [31] <http://physics.nist.gov/PhysRefData/DFTdata/Tables/ptable.html>
- [32] M. M. Morrell, R. G. Parr and M. Levy, J. Chem Phys **62**, 549, (1975)
- [33] S. Goedecker, A. Hoisie, “*Performance Optimization of Numerically Intensive Codes*”, SIAM publishing company, Philadelphia, USA 2001 (ISBN 0-89871-484-2)
- [34] E. R. Davidson, J. Comp. Phys. **17**, 87 (1975)
- [35] G. Kresse, J. Furthmüller, Phys. Rev. B **54**, 11169 (1996)
- [36] F. Bottin, S. Leroux, A. Knyazev, G. Zérah, Computational Materials Science **42**, 329 (2008)
- [37] CPMD Version 3.8: developed by J. Hutter, A. Alavi, T. Deutsch, M. Bernasconi, S. Goedecker, D. Marx, M. Tuckerman and M. Parrinello, Max-Planck-Institut für Festkörperforschung and IBM Zürich Research Laboratory (1995–1999)
- [38] E. Anderson *et al.*, “*LAPACK Users’ Guide*”, SIAM publishing company, Philadelphia, USA 1999 (ISBN 0-89871-447-8)
- [39] R. W. Hockney, The potential calculations and some applications, Methods Comput. Phys. **9** (1970) 135–210.
- [40] G. J. Martyna, M. E. Tuckerman, A reciprocal space based method for treating long range interactions in *ab initio* and force-field-based calculations in clusters, J. Chemical Physics **110** (6) (1999) 2810–2821.
- [41] L. Füsti-Molnar, P. Pulay, Accurate molecular integrals and energies using combined plane wave and gaussian basis sets in molecular electronic structure theory, J. Chem. Phys. **116** (18) (2002) 7795–7805.
- [42] Marie-Catherine Desjonquères, Daniel Spanjaard “*Concepts in Surface Physics*” Springer Series in Surface Sciences (1998)
- [43] Peter Minary, Mark E. Tuckerman, Katianna A. Pihakari, and Glenn J. Martyna, “*A new reciprocal space based treatment of long range interactions on surfaces*”, J. Chem. Phys. **116**, 5351 (2002)
- [44] J. J. Mortensen and M. Parrinello, “*A density functional theory study of a silica-supported zirconium monohydride catalyst for depolymerization of polyethylene*”, J. Phys. Chem. B **2000**, **104**, 2901–2907
- [45] R. W. Hockney and J. W. Eastwood, “*Computer Simulation Using Particles*” (McGraw-Hill, New York, 1981)
- [46] W. Hackbusch and U. Trottenberg, “*A Multigrid Method*”, Springer, Berlin, 1982
- [47] N. Saito, G. Beylkin, G., Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] Vol. **41** (12), (1993) 3584–3590
- [48] G. Deslauriers and S. Dubuc, Constr. Approx. **5**, 49 (1989)
- [49] G. Beylkin and L. Monzon, Applied and Computational Harmonic Analysis, **19** (2005) 17–48; Algorithms for numerical analysis in high dimensions G. Beylkin and M. J. Mohlenkamp, SIAM J. Sci. Comput., **26** (6) (2005) 2133–2159; G. Beylkin, M. J. Mohlenkamp, Numerical operator calculus in higher dimensions, in: Proceedings of the National Academy of Sciences, Vol. **99**, 2002, pp. 10246–10251
- [50] S. Goedecker, Comput. Phys. Commun. **76**, 294 (1993)