Department of Physics, Chemistry and Biology

Master's Thesis

Orbital-free Density-Functional Theory in a Finite Element Basis

Joel Davidsson

LTH-IFM-A-EX–15/3123–SE

Linköpings universitet
INSTITUTE OF TECHNOLOGY

Department of Physics, Chemistry and Biology
Linköpings universitet, SE-581 83 Linköping, Sweden

Master's Thesis
LTH-IFM-A-EX–15/3123–SE

Orbital-free Density-Functional Theory in a Finite Element Basis

Joel Davidsson

Adviser: **Alexander Lindmaa**
IFM

Examiner: **Rickard Armiento**
IFM

Linköping, 18 September, 2015

Avdelning, Institution
Division, Department

Thesis Division
Department of Physics, Chemistry and Biology
Linköpings universitet, SE-581 83 Linköping, Sweden

Datum
Date

2015-09-18

Språk
Language

☐ Svenska/Swedish
☒ Engelska/English

☐ ______________

Rapporttyp
Report category

☐ Doktorsavhandling
☐ Licentiatavhandling
☒ Examensarbete
☐ C-uppsats
☐ D-uppsats
☐ ______________

ISBN
—

ISRN
LTH-IFM-A-EX–15/3123–SE

Serietitel och serienummer **ISSN**
Title of series, numbering
—

URL för elektronisk version

`http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-121778`

Titel Orbitalfri Täthetsfunktionalsteori med finita elementmetoden
Title Orbital-free Density-Functional Theory in a Finite Element Basis

Författare Joel Davidsson
Author

Sammanfattning
Abstract

In this work, we have implemented an orbital-free density functional theory (OF-DFT) solver using the finite element method. In OF-DFT, the total ground state energy is minimized directly with respect to the electron density, rather than via orbitals like in the standard Kohn-Sham approach. For this to be possible, one needs an approximation of a universal density functional of the non-interacting kinetic energy. Presently available approximations allow for computation with very low computational expense, but which gives inaccurate energies. A stable OF-DFT code can be used as a testbed for new kinetic energy functionals and provide the necessary tool for investigating the accuracy of OF-DFT calculations for complex systems. We have implemented Thomas-Fermi theory with and without nuclear cusp condition, as well as additional exchange terms of Dirac and Amaldi. The program uses an extended version of the steepest descent in order to find the minimizing density in the variational principle. Our results include convergence tests for the hydrogen atom, weak bonding in the H2 molecule, and accurate results for the lightest noble gases (He, Ne, Ar). For heavier atoms (Kr, Xe, Rn), the results are less accurate. In addition, we consider hydrogen in the simple cubic structure without the cusp condition, which is a first attempt to use the code for periodic systems. Lastly, we discuss some possible improvements for the iterative process towards the minimizing density, as well as other possible directions for future development.

Nyckelord Orbital-free Density-Functional Theory, Finite element method, Thomas-Fermi, nuclear cusp condition
Keywords

Abstract

In this work, we have implemented an orbital-free density functional theory (OF-DFT) solver using the finite element method. In OF-DFT, the total ground state energy is minimized directly with respect to the electron density, rather than via orbitals like in the standard Kohn-Sham approach. For this to be possible, one needs an approximation of a universal density functional of the non-interacting kinetic energy. Presently available approximations allow for computation with very low computational expense, but which gives inaccurate energies. A stable OF-DFT code can be used as a testbed for new kinetic energy functionals and provide the necessary tool for investigating the accuracy of OF-DFT calculations for complex systems. We have implemented Thomas-Fermi theory with and without nuclear cusp condition, as well as additional exchange terms of Dirac and Amaldi. The program uses an extended version of the steepest descent in order to find the minimizing density in the variational principle. Our results include convergence tests for the hydrogen atom, weak bonding in the H2 molecule, and accurate results for the lightest noble gases (He, Ne, Ar). For heavier atoms (Kr, Xe, Rn), the results are less accurate. In addition, we consider hydrogen in the simple cubic structure without the cusp condition, which is a first attempt to use the code for periodic systems. Lastly, we discuss some possible improvements for the iterative process towards the minimizing density, as well as other possible directions for future development.

Acknowledgements

First and foremost, I would like to thank my examiner Rickard Armiento, for always taking the time to listen to my problems, for answering my questions and for very good discussions. I would also like to thank my supervisor Alexander Lindmaa, for good discussions as well and for the practical advices.

A great help was also the deal.II discussion group. I would like to thank all the people that took their time to answer my questions about the finite element method, especially Wolfgang Bangerth that usually was the first to answer.

A finally thanks goes to Mikael Kuisma for a good video tutorial online and for answering my mails.

Contents

1 Introduction **1**

2 Density Functional Theory **3**
- 2.1 Introduction . . . 3
- 2.2 Kohn-Sham Equations . . . 4
- 2.3 Orbital-Free Density Functional Theory . . . 5
- 2.4 Functional Derivatives . . . 6
  - 2.4.1 External Potential . . . 6
  - 2.4.2 Hartree Potential . . . 6
- 2.5 The Universal Functional G . . . 7
  - 2.5.1 Thomas-Fermi . . . 7
  - 2.5.2 Weizsäcker . . . 7
  - 2.5.3 Amaldi . . . 8
  - 2.5.4 Dirac . . . 8
- 2.6 Cusp Condition . . . 8
- 2.7 Ewald Summation . . . 10

3 Finite Element Method **11**
- 3.1 Introduction . . . 11
- 3.2 Mesh . . . 12
- 3.3 Galerkin Method . . . 13
- 3.4 Test Function . . . 13
  - 3.4.1 Lagrange Element . . . 13
- 3.5 Weak Formulation to Linear System . . . 14
- 3.6 Boundary Conditions . . . 16
- 3.7 Solver . . . 16
- 3.8 Preconditioner . . . 16
- 3.9 Derivatives . . . 17
- 3.10 Steepest Descent . . . 18

4 Method and Implementation **19**
- 4.1 Main Scheme . . . 19
- 4.2 Flow Chart . . . 20
- 4.3 Material and Mesh . . . 22
- 4.4 External Potential . . . 22

4.5 Initial Density 23
4.6 Hartree Potential 23
    4.6.1 Finite Systems 23
    4.6.2 Periodic Systems 24
4.7 Functional Derivatives 24
4.8 New Density 25
4.9 Calculate Energy 26
    4.9.1 Additional Energy Terms 26

5 Results and Discussion **27**
5.1 Convergence 27
    5.1.1 Element Selection 27
    5.1.2 Mesh and Initial Density 29
    5.1.3 Simulation Box Size 31
    5.1.4 Progression 32
    5.1.5 Soft Coulomb 33
5.2 Benchmark Calculations 34
    5.2.1 Exchange Term 34
    5.2.2 H_2 Molecule 38
    5.2.3 Cusp Condition 40
5.3 Bulk Systems 42
    5.3.1 Periodic Potential 42
    5.3.2 Atomisation Energy 44
5.4 Extension 45
    5.4.1 Gradient Correction 45

6 Outlook **47**
6.1 Density 47
6.2 Mesh 47
6.3 Convergence 48
6.4 Derivatives 48
6.5 Periodic 49
6.6 Miscellaneous 49

7 Conclusions **51**

Appendices **53**

A Input parameters **55**

B Additional Data **59**
B.1 Element Selection 59
B.2 Cusp Condition 60
B.3 H_2 Molecule 61
B.4 Soft Coulomb 63

Bibliography **65**

Chapter 1

Introduction

In the age we live in, a big part of physics is the development of new materials. There are many material design problems everywhere we look. For example, take the glass on a modern smartphone. It is made to be very strong, durable and there is a coating to prevent scratches. To create materials with such properties is possible since we understand the underlying physics down to the atomic scale. A branch of physics is set out to find theories and models to describe materials and their properties and in doing so, open a path to further improved materials.

The objective of this research is to test if finite element method can be used to solve orbital-free density functional theory and in doing so getting a tool to be able to further the development of more accurate functionals.

Another application for this program would be as a help in high-throughput applications to get ballpark figures on materials and compounds since the orbital-free density functional theory has less accuracy but requires less computer power.

In November 1964, Hohenberg and Kohn published the article Inhomogeneous Electron Gas which was the start of the density functional theory. This theory presented a new way to solve the Schrödinger equation. In this paper, the orbital-free approach is outlined and was promising but in the following year, Kohn and Sham published an even more promising article that presented the Kohn-Sham equations. By now, a vast majority of computational materials science rely on the Kohn-Sham equations.

In the same decade as the Hohenberg and Kohn article was published, in the field of engineering, the finite element method became popular. The method was invented before that, but it was rediscovered by engineers in the sixties and put to use for dam calculations. Today the finite element method is one of the most researched and used methods for solving differential equations.

In this thesis, we will cover the background theory. This includes the density functional theory, the steepest descent, optimization schemes and the finite element method. Then we show how we combine these into a program and discuss the problems there was along the way. The results from this program will also be presented along with discussion and comparison with previously known results. Finally, we discuss the method and possible improvements that can be made.

Chapter 2

Density Functional Theory

For the theoretical background of this work, this chapter covers orbital-free density functional theory, functional derivatives and a restricted solution technique that invokes a density cusp condition. In this thesis, all equations are written using Hartree units

$$m_e = e = \hbar = k_e = 1, \tag{2.1}$$

where the m_e is the electron mass, e is the elementary charge, $\hbar$ is the reduced Planck's constant and k_e is the Coulomb's constant. The presentation of density functional theory follows closely the presentation in the work by Hohenberg and Kohn [1] with addition information from Ref. [2, 3].

2.1 Introduction

In 1926, Erwin Schrödinger published his famous equation

$$\hat{H}\Psi = E\Psi. \tag{2.2}$$

This is the time-independent Schrödinger's equation where $\hat{H}$ is the Hamiltonian operator that acts on the wave function Ψ to give the energy E. This equation technically determines all properties of any atomistic system, *e.g.*, a material. A direct solution to a many-body system is very complicated, if not impossible. This is because there are too many degrees of freedom.

In 1964, Hohenberg and Kohn [1] proved that it is possible to change this formulation into one with fewer degrees of freedom by using the density n as the fundamental variable instead of the many possible wave function Ψ. This provides the same energy for the ground state and the density contains the same information as the wave function Ψ. The density is defined as:

$$n = \langle\Psi|\,\hat{n}\,|\Psi\rangle = \int d\mathbf{r}_2 \ldots \int d\mathbf{r}_N \sum_{i=1}^{N} \delta(\mathbf{r} - \mathbf{r}_i)\,|\Psi(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)|^2 \tag{2.3}$$

where N is the number of electrons, $\mathbf{r}_i$ their position and $\hat{n}$ is the density operator. We proceed to show how to rewrite the Schrödinger equation into the density functional theory.

The Hamiltonian $\hat{H}$ for many-body electron problem can be divided into components. In the Born-Oppenheimer approximation [4], these components are:

$$\hat{H} = \hat{T} + \hat{U} + \hat{V}. \tag{2.4}$$

In this equation, $\hat{T}$ is the kinetic energy operator, $\hat{U}$ is the Coulomb interaction between the electrons, and $\hat{V}$ is the Coulomb interaction between nuclei and electrons, also known as the external potential. These operators are defined as:

$$\hat{T} = \frac{1}{2}\int \nabla\Psi^*(\mathbf{r})\nabla\Psi(\mathbf{r})d\mathbf{r} \tag{2.5}$$

$$\hat{U} = \frac{1}{2}\int \frac{1}{|\mathbf{r}-\mathbf{r}'|}\Psi^*(\mathbf{r})\Psi^*(\mathbf{r}')\Psi(\mathbf{r})\Psi(\mathbf{r}')d\mathbf{r}d\mathbf{r}' \tag{2.6}$$

$$\hat{V} = \int v(\mathbf{r})\Psi^*(\mathbf{r})\Psi(\mathbf{r})d\mathbf{r} \tag{2.7}$$

Using Eq. (2.4), it is possible to reformulate the expectation value E of the total energy as a functional of the density n, giving

$$E[n] = \int v(\mathbf{r})n(\mathbf{r})d\mathbf{r} + \frac{1}{2}\int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}d\mathbf{r}' + G[n], \tag{2.8}$$

where the first term corresponds to the external potential functional, the second term is the Coulomb interaction between electrons and the last term is a universal functional. This functional is not known on an explicit form. It includes all the remaining many-body quantum effects of the system. In the paper by Kohn and Sham [3], it is further partitioned:

$$G[n] = T_{\mathrm{s}}[n] + E_{\mathrm{xc}}[n] \tag{2.9}$$

here, $T_{\mathrm{s}}[n]$ is the non-interacting kinetic energy and $E_{\mathrm{xc}}[n]$ is exchange and correlation energy.

2.2 Kohn-Sham Equations

In Kohn-Sham density functional theory, the problem is reformulated in terms of non-interacting particles. The strength of the Kohn-Sham approach is that the non-interacting kinetic energy $T_{\mathrm{s}}[n(\mathbf{r})]$ is calculated exactly. The Kohn-Sham equations are given by:

$$\Big(-\frac{1}{2}\nabla^2 + v_{\mathrm{KS}}\Big)\phi_i(\mathbf{r}) = \epsilon_i\phi_i(\mathbf{r}) \tag{2.10}$$

where the $\phi_i(\mathbf{r})$ are the Kohn-Sham orbitals with corresponding eigenvalues ϵ_i. The v_{KS} is called the Kohn-Sham potential and contains the external potential, the Coulomb interaction between electrons, and the exchange and correlation terms.

$$v_{\text{KS}}(\mathbf{r}) = v(\mathbf{r}) + \int \frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}' + \mu_{xc}(n(\mathbf{r})) \tag{2.11}$$

Eq. (2.10) is solved repeatedly until a self-consistent solution is found. The density of the interacting system can be represented by the Kohn-Sham orbitals so that

$$n(\mathbf{r}) = \sum_i^N |\phi_i(\mathbf{r})|^2. \tag{2.12}$$

2.3 Orbital-Free Density Functional Theory

In contrast to Kohn-Sham density functional theory, in orbital-free density functional theory are both the functionals $T_{\text{s}}[n]$ and $E_{\text{xc}}[n]$ are approximated. In Section 2.5 we will discuss the different ways of approximating these terms. OF-DFT is an active area of research. For an overall discussion, and recent works see, *e.g*, [5, 6, 7, 8]. First, we will discuss how to find the density that gives the lowest energy $E[n]$. The minimization is performed under the constraint of keeping the number of electrons fixed in the system, and which is given by the equation:

$$\int n(\mathbf{r})d\mathbf{r} = N. \tag{2.13}$$

Hence the equation we want to solve:

$$\begin{aligned} &\min_n \; E[n] \\ \text{s.t} &\int n(\mathbf{r})d\mathbf{r} = N \\ &n(\mathbf{r}) \geq 0 \end{aligned} \tag{2.14}$$

where $E[n]$ contains the functionals that are used to describe the nature of the electron, the constraint $\int n(\mathbf{r})d\mathbf{r} = N$ keeps the number of electrons fixed and the constraint $n(\mathbf{r}) \geq 0$ makes sure the solution is physical, since negative density do not exist as physical solution. The constraint $\int n(\mathbf{r})d\mathbf{r} = N$ can be handled by introducing a Lagrange multiplier μ.

$$\begin{aligned} &\min_n \; E[n] - \mu\Big(\int n(\mathbf{r})d\mathbf{r} - N\Big) \\ &\text{s.t } n(\mathbf{r}) \geq 0 \end{aligned} \tag{2.15}$$

The Lagrange multiplier μ is the chemical potential of the system. The density n that solves this equation is given by a stationary state, where the functional derivative is equal to zero.

$$\frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})} - \mu = 0 \tag{2.16}$$

2.4 Functional Derivatives

The definition of a functional derivative is in Ref. [9].

$$\lim_{\epsilon\to 0}\frac{F[f+\epsilon\phi]-F[f]}{\epsilon}=\frac{d}{d\epsilon}(F[f+\epsilon\phi])\Big|_{\epsilon=0}=\int\frac{\delta F}{\delta f(x)}\phi(x)dx \tag{2.17}$$

In this formulation, $F[f]$ is the functional that is to be differentiated, and $\phi(x)$ is an arbitrary function. In most cases, this can be taken as a delta function $\delta(x-x_0)$ to simplify the definition. This definition is similar to the definition of the derivative of a function and it turns out they share many important properties such as the chain rule and linearity.

2.4.1 External Potential

The external potential has the form

$$E_v[n]=\int v(\mathbf{r})n(\mathbf{r})d\mathbf{r} \tag{2.18}$$

which can be rewritten as

$$E_v[n+\delta n]=\int v(\mathbf{r})(n(\mathbf{r})+\delta n(\mathbf{r}))d\mathbf{r}=\int v(\mathbf{r})n(\mathbf{r})d\mathbf{r}+\int v(\mathbf{r})\delta n(\mathbf{r})d\mathbf{r} \tag{2.19}$$

which gives the derivative

$$\frac{\delta E_v[n(\mathbf{r})]}{\delta n(\mathbf{r})}=v(\mathbf{r}). \tag{2.20}$$

2.4.2 Hartree Potential

The Coulomb interaction between electrons is

$$J[n]=\frac{1}{2}\int\frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}d\mathbf{r}' \tag{2.21}$$

which has the derivative

$$\frac{\delta J[n(\mathbf{r})]}{\delta n(\mathbf{r})}=\int\frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}' \tag{2.22}$$

which defines the Hartree potential $v_i(\mathbf{r})$.

$$v_i(\mathbf{r})\equiv\int\frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}' \tag{2.23}$$

is equivalent to solving Poisson's equation

$$\nabla^2 v_i(\mathbf{r})=-4\pi n(\mathbf{r}). \tag{2.24}$$

2.5 The Universal Functional G

In this section, we will present the functionals that approximate the kinetic energy $T_{\mathrm{s}}[n]$ and exchange and correlation energy $E_{\mathrm{xc}}[n]$ and their functional derivatives.

The simplest approximation of the kinetic energy $T_{\mathrm{s}}[n]$ is the Thomas-Fermi model. Terms of higher order, like the Weizsäcker term, can then be added in order to increase the accuracy.

In this work, we will consider two kinds of approximation to the exchange and correlation energy $E_{\mathrm{xc}}[n]$: the so-called Amaldi functional, and the Dirac exchange term.

2.5.1 Thomas-Fermi

The Thomas-Fermi approximation is derived from a uniform electron gas [10, 11, 12]. In this case, one finds for the kinetic energy:

$$T_{\mathrm{TF}}[n] = \frac{3}{10}(3\pi^2)^{2/3} \int \big(n(\mathbf{r})\big)^{5/3} d\mathbf{r} \tag{2.25}$$

with the functional derivative

$$\frac{\delta T_{\mathrm{TF}}[n(\mathbf{r})]}{\delta n(\mathbf{r})} = \frac{1}{2}(3\pi^2)^{2/3}\big(n(\mathbf{r})\big)^{2/3}. \tag{2.26}$$

2.5.2 Weizsäcker

Weizsäcker [13] proposed a simple term that takes into account the kinetic energy of a single orbital, this is added to the Thomas-Fermi functional to improve the accuracy

$$T_{\mathrm{W}}[n] = \frac{\lambda}{8} \int \frac{\big(\nabla n(\mathbf{r})\big)^2}{n(\mathbf{r})} d\mathbf{r} \tag{2.27}$$

where the parameter λ was originally set to unity, but other values have been proposed based on various arguments. For instance, in the article by Stich, Gross, Malzcher and Dreizler [14], they present an $\lambda = 0.2$ to be optimal.

When applying the functional derivative to the Weizsäcker functionals one gets:

$$T_{\mathrm{W}}[n + \delta n] = \frac{1}{8} \int \frac{\nabla(n(\mathbf{r}) + \delta n(\mathbf{r})) \cdot \nabla(n(\mathbf{r}) + \delta n(\mathbf{r}))}{n(\mathbf{r}) + \delta n(\mathbf{r})} d\mathbf{r}$$

$$= T_{\mathrm{W}}[n] - \frac{1}{4} \int \frac{\nabla^2 n(\mathbf{r})}{n(\mathbf{r})} \delta n(\mathbf{r}) d\mathbf{r} + \frac{1}{8} \int \frac{\nabla n(\mathbf{r}) \cdot \nabla n(\mathbf{r})}{n(\mathbf{r})^2} \delta n(\mathbf{r}) d\mathbf{r} \tag{2.28}$$

and thus

$$\frac{\delta T_{\mathrm{W}}[n(\mathbf{r})]}{\delta n(\mathbf{r})} = -\frac{1}{4}\frac{\nabla^2 n(\mathbf{r})}{n(\mathbf{r})} + \frac{1}{8}\frac{\nabla n(\mathbf{r}) \cdot \nabla n(\mathbf{r})}{n(\mathbf{r})^2}. \tag{2.29}$$

2.5.3 Amaldi

To approximate the exchange energy, Amaldi [15] used the classical Coulomb repulsion

$$T_{\mathrm{A}}[n] = -\frac{1}{2N}\int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}d\mathbf{r}'. \tag{2.30}$$

This functional is $T_{\mathrm{A}}[n] = -J[n]/N$ so its derivative is simply

$$\frac{\delta T_{\mathrm{A}}[n(\mathbf{r})]}{\delta n(\mathbf{r})} = -v_i(\mathbf{r})/N. \tag{2.31}$$

2.5.4 Dirac

Another exchange term was introduced by Dirac [16]:

$$T_{\mathrm{D}}[n] = -\frac{3}{4}\Big(\frac{3}{\pi}\Big)^{1/3}\int \big(n(\mathbf{r})\big)^{4/3}d\mathbf{r}. \tag{2.32}$$

This functional is the equivalent to local-density approximation in Kohn-Sham density functional theory.

The functional derivative of the exchange Dirac functional is

$$\frac{\delta T_{\mathrm{D}}[n(\mathbf{r})]}{\delta n(\mathbf{r})} = -\Big(\frac{3}{\pi}\Big)^{1/3}\big(n(\mathbf{r})\big)^{1/3}. \tag{2.33}$$

2.6 Cusp Condition

When Thomas-Fermi theory is used for atomic systems, the density at the nucleus diverge. The reason is that the external potential for atomic system diverges at the nucleus.

$$v(\mathbf{r}) = -Z/\mathbf{r} \tag{2.34}$$

where Z is the atomic number for the atom. One solution is to use the softened Coulomb potential,

$$v(\mathbf{r}) = -Z/(\mathbf{r}+c), \tag{2.35}$$

where c is the softening parameter. Another alternative approach is presented in the paper by Parr and Ghosh [17]. They resolve this issue with the nuclear cusp condition:

$$\left.\frac{dn(\mathbf{r})}{dr}\right|_{r=0} = -2Zn(0). \tag{2.36}$$

This condition can be enforced via introduction of the following constraint:

$$\int e^{-2k\mathbf{r}}\nabla^2 n(\mathbf{r})d\mathbf{r} < \infty, \tag{2.37}$$

where k is a constant that will be determined later. This constraint is added to the Thomas-Fermi equations by another Lagrange multiplier $-\lambda$,

$$E[n] = \int v(\mathbf{r})n(\mathbf{r})d\mathbf{r} + \frac{1}{2}\int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}d\mathbf{r}' + \frac{3}{10}(3\pi^2)^{2/3}\int \big(n(\mathbf{r})\big)^{5/3}d\mathbf{r}$$

$$-\mu\Big(\int n(\mathbf{r})d\mathbf{r}-N\Big)-\lambda\Big(\int e^{-2k\mathbf{r}}\nabla^2 n(\mathbf{r})d\mathbf{r}\Big). \tag{2.38}$$

Rederiving the variational equation now gives,

$$\frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})}=v(\mathbf{r})+v_i(\mathbf{r})+(5/2)(3\pi^2)^{2/3}\big(n(\mathbf{r})\big)^{2/3}-\mu-\lambda\nabla^2(e^{-2k\mathbf{r}})=0, \tag{2.39}$$

where $\lambda\nabla^2(e^{-2k\mathbf{r}})$ can be rewritten to $(4\lambda k e^{-2k\mathbf{r}})/\mathbf{r}-4\lambda k^2 e^{-2k\mathbf{r}}$. Inserted in the functional derivative together with the external potential gives,

$$-(Z-4\lambda k e^{-2k\mathbf{r}})/\mathbf{r}+v_i(\mathbf{r})+(5/2)(3\pi^2)^{2/3}\big(n(\mathbf{r})\big)^{2/3}-\mu-4\lambda k^2 e^{-2k\mathbf{r}}=0. \tag{2.40}$$

If we now use Maclaurin series expansion on $-(Z-4\lambda k e^{-2k\mathbf{r}})/\mathbf{r}$ we can determine λ. The Maclaurin series is $e^x=1+x+\mathcal{O}(x^2)$ we get:

$$-(Z-4\lambda k(1-2kr+\mathcal{O}(r^2)))/r. \tag{2.41}$$

If we choose $\lambda=Z/4k$, the results are $-(2Zkr+\mathcal{O}(r^2))/r=-2Zk+\mathcal{O}(r)$ which has no singularity as $r\to 0$. Here we have the opportunity to enforce the nuclear cusp condition by choosing $k=\sqrt{(5/9)(3/10)(3\pi^2)^{2/3}n(0)^{2/3}}$.

In Figure 2.1, the converged potential for Helium with the nuclear cusp condition enforced as discussed is compared to the Coulomb and softened Coulomb potentials (convergence is necessary since the method gives a potential with a density dependence).

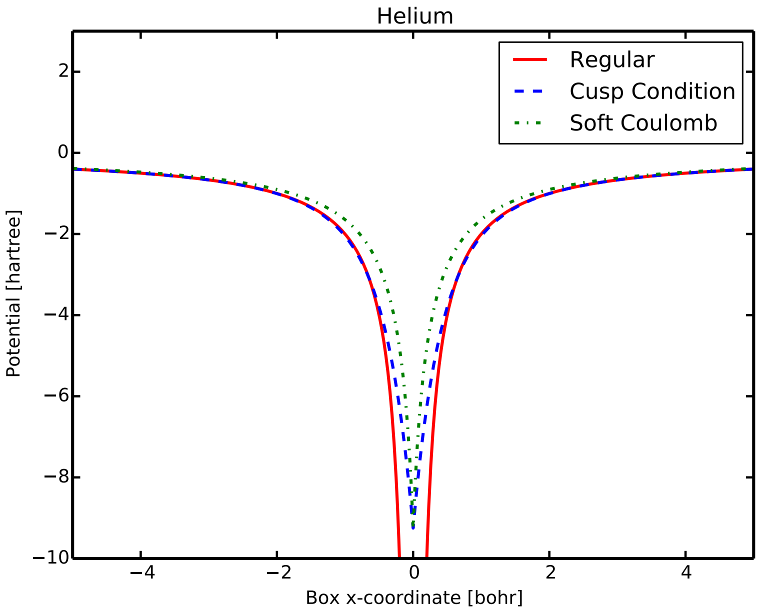


Figure 2.1. Different potentials for a Helium nuclei. The red curve is the $-1/r$, the blue curve is with the cusp condition and the green is the softened Coulomb. The softened Coulomb has the form $1/(\mathbf{r}+0.2162)$ to match up with the cusp condition at the nuclei.

2.7 Ewald Summation

In bulk material with periodic boundary conditions, there is a contribution to the total energy from the ion-ion energy between a nucleus and every infinite copy of this nucleus

$$E_{\text{ion}-\text{ion}} = \frac{1}{2}\sum_{i,j}\frac{1}{N}Z_iZ_j\sum_{k,l,k\neq l}\varphi(||\mathbf{r}_k - \mathbf{r}_l||_2). \tag{2.42}$$

The standard method for calculating this energy is the Ewald summation. We summarize this method following the presentation by Toukmaji and Board [18] and the notation from Ref. [19]. The sum is divided into three parts: one long range, one short range, and a constant

$$E_{\text{ion}-\text{ion}} = E^r + E^m + E^0. \tag{2.43}$$

The short range interaction E^r is calculated in real space,

$$E^r = \frac{1}{2}\sum_{i,j,i\neq j} Z_iZ_j\sum_{\mathbf{L}}\frac{\text{erfc}(a||\mathbf{r}_i - \mathbf{r}_j + \mathbf{L}||_2)}{||\mathbf{r}_i - \mathbf{r}_j + \mathbf{L}||_2)}, \tag{2.44}$$

where $\mathbf{L}$ is the lattice vectors inside a sphere with radius L_{max} and a is a convergence parameter that determines the division between the long and short range. The long range interaction E^m is calculated in the Fourier space

$$E^m = \frac{1}{2}\sum_{i,j,i\neq j}\frac{Z_iZ_j}{\pi V}\sum_{\mathbf{G}}\frac{e^{||\mathbf{G}||_2^2/(2a)^2}}{||\mathbf{G}||_2^2}\cos(\mathbf{G}\cdot(\mathbf{r}_i - \mathbf{r}_j)) \tag{2.45}$$

where $\mathbf{G}$ is the reciprocal lattice vectors inside a sphere with radius G_{max}. The constant E^0 is given by:

$$E^0 = -\frac{9}{\sqrt{\pi}}\sum_i Z_i^2 - \frac{\pi}{2Va^2}\left(\sum_i Z_i\right)^2 \tag{2.46}$$

where V is the cell volume. The convergence parameter a do not affect the results only the speed of convergence. In Pymatgen [20], a code for running ewald sums, it is evaluated by the following form to give a fast convergence:

$$a = \sqrt{\pi}\left(\frac{0.01M}{V}\right)^{\frac{1}{6}} \tag{2.47}$$

where M is the number of atoms.

Chapter 3

Finite Element Method

This chapter is a brief introduction to the finite element method and in particular as implemented in the open-source library deal.II [21, 22] which we use in this work. There is, of course, an extensive amount of literature on this subject. The presentation in this chapter follows closely that of the manual, tutorials and video lectures that accompany deal.II [23].

3.1 Introduction

The main reason we explore the use of the finite element method to solve the density functional theory equations is that finite element method is very good at solving Poisson's equation by providing good numerical integrating. Which could be used to solve Eq. (2.24). The general scheme for solving a partial differential equation with the finite element method is:

1. Generate a mesh.
2. Write the partial differential equation into weak form with a set of test functions.
3. Write the weak form in terms of a linear set of equations.
4. Solve this system of equations.

Below we will look closer at these steps and how they are done in detail especially for solving Poisson's equation.

3.2 Mesh

The first step in the finite element method is to generate a mesh. A mesh is an approximation of the geometric domain using polygons or polyhedrons. This can be done in many different ways depending on the dimension and the choice of polygons or polyhedrons one uses. Deal.II has only implemented quadrilateral polygons, *i.e.*, four-sided polygons.

The way this is done in deal.II is that one marks a cell for refinement. Then the marked cell is subdivided into smaller cells. In a problem formulated in n dimensions a cell marked for refinement will be divided into 2^n cells.

If there are two cells next to each other and one is refined and the other is not, then one will get a hanging node as shown in Figure 3.1.

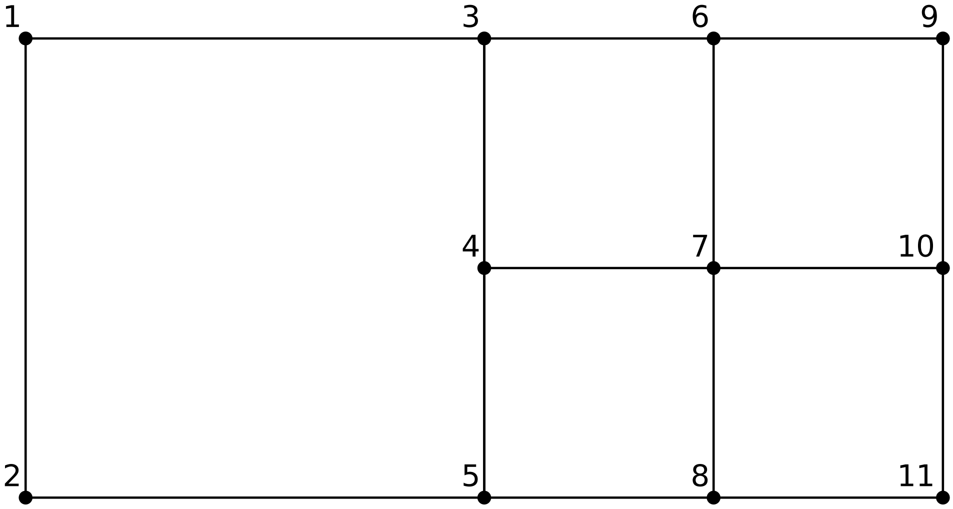


Figure 3.1. A mesh with a hanging node in node 4. The right cell (nodes 3,5,9,11) is refined one step further than the left cell (nodes 1,2,3,5).

These hanging nodes need to be handled in a special way. The value V at node 4 is linear interpolated between the neighbouring nodes.

$$V_4 = \frac{1}{2}V_3 + \frac{1}{2}V_5 \tag{3.1}$$

3.3 Galerkin Method

The Galerkin method states that it is possible to rewrite the partial differential equation into a weak formulation using a test function. For instance, consider Poisson's equation. The strong formulation is:

$$-\nabla^2 u = f \ in \ \Omega \tag{3.2}$$

where Ω is the simulation space where the solution to Poisson's equation exists. The first step in rewriting the partial differential equation into weak formulation is to multiply with a test function ψ and integrate, which gives

$$-\int\limits_{\Omega} \psi \nabla^2 u = \int\limits_{\Omega} \psi f. \tag{3.3}$$

A partial integration then leads to the equation

$$\int\limits_{\Omega} \nabla\psi \cdot \nabla u - \int\limits_{\delta\Omega} \psi n \cdot \nabla u = \int\limits_{\Omega} \psi f. \tag{3.4}$$

If we now choose a test function ψ that has the property $\int_{\delta\Omega} \psi n \cdot \nabla u = 0$ we end up with the following equation:

$$\int\limits_{\Omega} \nabla\psi \cdot \nabla u = \int\limits_{\Omega} \psi f \tag{3.5}$$

This is called the weak formulation of Poisson's equation. For a unique solution to Poisson's equation to exist, one requires a set of boundary conditions. This is explained further in Section 3.6.

3.4 Test Function

The choice of the test function ψ depends on the problem at hand. For Poisson's equation, the common choice is to use the Lagrange elements, which is also what we use in this work. This element has the property $\int_{\delta\Omega} \psi n \cdot \nabla u = 0$, which can be seen in the definition provided in the next section.

3.4.1 Lagrange Element

A Lagrange element is defined to be unity at one node, but zero at all other nodes.

$$\psi(x_i, x_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{3.6}$$

In order to define the functions between two nodes, one uses the Lagrange polynomials. These are defined as:

$$\psi_j(x) = \prod_{i=0,\ i\neq j}^{k} \frac{x - x_i}{x_j - x_i} \tag{3.7}$$

where each x_i and x_j are the positions of the nodes. In Figure 3.2, one can see the first orders of the Lagrange polynomials on a 1D line from 0 to 1. In this figure there are two nodes; one at 0 and one at 1, but the elements will have different degrees of freedom depending on the order of the Lagrange polynomials. Degrees of freedom is the number of variables that are free to vary.

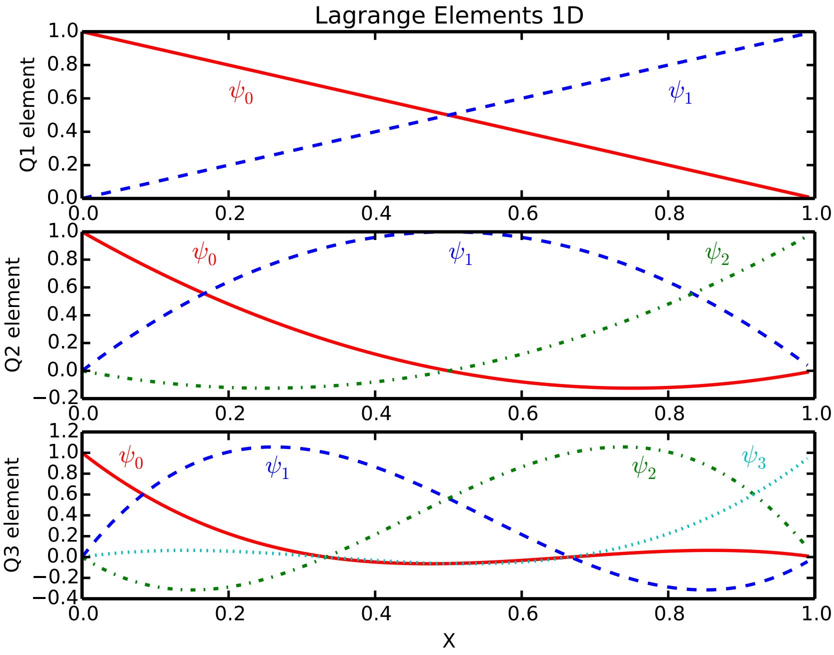


Figure 3.2. Lagrange polynomials for order 1,2 and 3 in 1D. Q1 elements has 2 degrees of freedom at 0 and 1. Q2 has three degrees of freedom at 0, 0.5 and 1. Q3 has four degrees of freedom at 0, 1/3, 2/3 and 1.

With higher order elements, the line gets divided more and the system get more degrees of freedom when saving this information. These Lagrange polynomials have a similar appearance in 2D and 3D.

A good rule of thumb when choosing the order of an element is order 2 or 3 for simulations in 3D.

3.5 Weak Formulation to Linear System

Having defined the partial differential equation in the weak formulation, and having chosen suitable test functions, we can look for the solution u_h to this equation. The h indicates that this is the solution on the mesh is not the real solution u. u_h

can be written a linear combination of test functions:

$$u_h(x) = \sum_j U_j \psi_j(x). \tag{3.8}$$

Hence, we need to find a U that solves the equation:

$$AU = F \tag{3.9}$$

where each A_{ij} is given by

$$A_{ij} = \int\limits_{\Omega} \nabla\psi_i \cdot \nabla\psi_j \tag{3.10}$$

and F_i is presented as

$$F_i = \int\limits_{\Omega} \psi_i f. \tag{3.11}$$

Now we look at each contribution from each cell K:

$$A_{ij} = \sum_K \int\limits_K \nabla\psi_i \cdot \nabla\psi_j \tag{3.12}$$

$$F_i = \sum_K \int\limits_K \psi_i f \tag{3.13}$$

In order to solve these integrals, we use a standard tool in numerical integration, namely the quadrature formula. The integrals are approximated as follows

$$A_{ij}^K = \int\limits_K \nabla\psi_i \cdot \nabla\psi_j \approx \sum_q \nabla\psi_i(x_q^K) \cdot \nabla\psi_j(x_q^K) w_q^K \tag{3.14}$$

$$F_i^K = \int\limits_K \psi_i f \approx \sum_q \psi_i(x_q^K) f(x_q^K) w_q^K \tag{3.15}$$

where each x_q^K and w_q^K are the position and weight respectively that are used in the quadrature formula. The are many different quadrature formulas to choose from, but the most common is the Gauss quadrature formula. When using Lagrange elements and the Gauss quadrature, there is an important relation to keep in mind when choosing the order of elements n and the order of quadrature K. It turns out that K needs to be such that $K = n + 1$, in order to give the best approximation.

This will result in a system of equations given on the standard form $AU = F$, where the size of the matrix is the degrees of freedom. Most entries in the matrix A will be zero, and is hence a sparse matrix. This occurs since the entries for the i:th row only depends on the i:th node and the nodes that are connected to that node.

3.6 Boundary Conditions

To converge to a unique solution, we must specify boundary conditions for the simulation space Ω. This is required for Poisson's equation, otherwise the linear system $AU = F$ will be underdetermined.

The two most common boundary conditions are Dirichlet boundary condition $u = g2$ on $\delta\Omega$ and Neumann boundary condition $n \cdot \nabla u = g2$ on $\delta\Omega$ where $g1$ and $g2$ are arbitrary functions.

One can also use periodic boundary conditions but then, the system still is underdetermined. The solution to this system differs by a constant $u_h = u_h + C$. To determine this constant we must add an additional condition. An arbitrary point needs to have a Dirichlet boundary condition $u = 0$ at one node to determine C. In order to find a periodic solution to Poisson's equation, the problem have to be well-posed, *i.e.*, the right-hand side needs to integrate to zero.

3.7 Solver

Having defined the linear system in Eq. (3.9), we need to choose a solver to find the solution at all nodes. There are two kinds of solvers, direct or iterative.

Direct solvers solves this system of equations by means of Gaussian elimination or by finding the inverse. These solvers provide the exact solution to the system but takes a lot of time and are thus computationally cumbersome.

Iterative solvers will find an approximate solution to the equation system by taking iterative steps to get closer to the correct solution. The conjugate gradient solver is a common iterative method. This solver has the feature that if you have an NxN system it is guaranteed to converge in N steps, but usually it finds the threshold value faster than that. One way to improve the speed of iterative solver is to use a preconditioner.

3.8 Preconditioner

Consider a system $Ax = B$ being solved with an iterative solver. A preconditioner P^{-1} is used to relax the system $P^{-1}Ax = P^{-1}B$, which will reduce iterations to find the solution. Clearly, the optimal preconditioner is $P^{-1} = A^{-1}$ since no iterations would be needed, but it takes a lot of computations to calculate the inverse of a matrix. Thus we need to approximate the inverse and this is done by dividing the matrix A into three different parts: $A = L + D + U$. Here L is an lower triangle matrix, D is an diagonal matrix and U is an upper triangle matrix. A simple choice is the Jacobi preconditioner $P^{-1} = D^{-1}$. This is faster than calculating the inverse and still reduces the number of iterations needed.

The preconditioner used most frequently in the deal.II tutorials is the Symmetric Successive Over-Relaxation (SSOR) which has the following form:
$P^{-1} = (D + \omega U)^{-1} * D * (D + \omega L)^{-1}$ with the value for ω set to 1.2 as default. This is the preconditioner used in this work.

3.9 Derivatives

In many applications, the derivative of the solution in every node is needed. We follow the approach presented in the documentation of another finite element library, FEniCS [24]. Given the solution

$$u_h(x) = \sum_j U_j \psi_j(x), \tag{3.16}$$

the derivative is given by

$$\nabla u_h(x) = \sum_j U_j \nabla \psi_j(x). \tag{3.17}$$

This may introduce problems depending on the choice of test function ψ. For example, the derivative of the Lagrange element of order 1 (Q1) is discontinuous, *i.e.*, it belongs to another function space than the regular elements. There are two possible solutions; either one chooses a better test function or one projects the derivative $\nabla u_h(x)$ on to the same function space as the solution $u_h(x)$.

The projection method is done by setting up a weak formulation of the projection and solving this problem similar as we did in Section 3.5 for Poisson's equation. Since the derivative of a scalar field is a vector field and we are not interested in the individual vector components but the norm of the derivative. The strong formulation is $u = |\nabla u_h|^2$ which will give a linear system $AU = F$. This time, the matrix A and the right-hand side F are different:

$$A_{ij} = \int\limits_\Omega \psi_i \cdot \psi_j \tag{3.18}$$

$$F_i = \int\limits_\Omega \psi_i (\nabla u_h(x) \cdot \nabla u_h(x)) \tag{3.19}$$

Solving this system gives the derivative in the same function space as the solution $u_h(x)$, and this equation does not require any boundary conditions since the system is well defined.

A similar setup is done if we need the second order derivatives. This can be found by solving the inverse to Poisson's equation $u = \nabla^2 u_h$. Hence, now the Matrix A and the right-hand side F are switched:

$$A_{ij} = \int\limits_\Omega \psi_i \cdot \psi_j \tag{3.20}$$

$$F_i = -\int\limits_\Omega \nabla \psi_i \cdot \nabla u_h(x) \tag{3.21}$$

The minus sign that appear in the right-hand side F_i, is due to the fact that Poisson's equation is defined like $-\nabla^2 u = f$. Though, the problem we are solving is $u = \nabla^2 u_h$. This equation does not require any boundary conditions as well.

3.10 Steepest Descent

Finally, we will discuss optimization schemes. This is not specific to the finite element method, but can be used on the nodal value given by the finite element method. The simplest algorithm in optimization to solve the problem like Eq. (2.14) is the steepest descent. The general method of steepest descent for minimizing a function $f(x)$ is taken from the book [25] and proceeds in the following steps:

1. Choose a starting point $x^{(k=0)}$
2. Calculate the gradient $d(x)^{(k)} = \nabla f(x^{(k)})$
3. Check the abort criteria $\|d^{(k)}\|^2 < \epsilon$
4. Calculate the step size $t^{(k)}$
5. Find the new point $x^{(k+1)} = x^{(k)} - t^{(k)} d(x)^{(k)}$
6. Repeat step 2 with $k = k + 1$

It is known that the steepest descent will find a global solution if the function is convex and the set is convex.

Chapter 4

Method and Implementation

This chapter discusses the implementation of an orbital-free density functional theory solver. The code was written to solve the orbital-free equations for atoms, molecules and bulk systems in 3D space. We name the program DeFuSE which is an acronym for **De**nsity **Fu**nctional **S**olver for finite **E**lement method. In this chapter will we discuss the basic flow of the program, issues that arose when combining the orbital-free equations with the finite element method and how the equations were solved.

4.1 Main Scheme

The following scheme describes how Eq. (2.15) is solved in our program. This is done by implementing the steepest descent scheme, in Section 3.10, over functional space (instead of for an ordinary function).

1. Choose a starting density $n(\mathbf{r})^{(k=0)}$
2. Calculate the functional derivative $d(\mathbf{r})^{(k)} = \frac{\delta E[n(\mathbf{r})^{(k)}]}{\delta n(\mathbf{r})}$
3. Check the abort criteria $\|d(\mathbf{r})^{(k)} - \mu^{(k)}\|^2 < \epsilon$
4. Determine the step size $t^{(k)}$
5. Find the new density $n(\mathbf{r})^{(k+1)} = n(\mathbf{r})^{(k)} - t^{(k)}(d(\mathbf{r})^{(k)} - \mu^{(k)})$
6. Repeat step 2 with $k = k + 1$

The steps shown above constitute the steepest descent implementation devised for finding the density that minimizes the energy. We assume that the initial density is sufficiently close to the solution for the scheme to converge to the global minimum.

These steps are visualized in Figure 4.1.

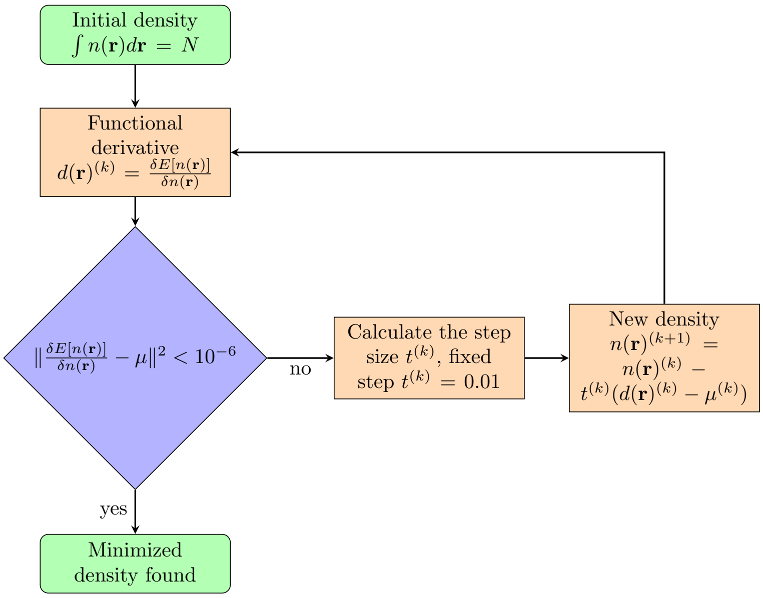

Initial density
$\int n(\mathbf{r})d\mathbf{r} = N$
Functional derivative
$d(\mathbf{r})^{(k)} = \frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})}$
$\|\frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})} - \mu\|^2 < 10^{-6}$
no
Calculate the step size $t^{(k)}$, fixed step $t^{(k)} = 0.01$
New density $n(\mathbf{r})^{(k+1)} = n(\mathbf{r})^{(k)} - t^{(k)}(d(\mathbf{r})^{(k)} - \mu^{(k)})$
yes
Minimized density found

Figure 4.1. The modified steepest descent algorithm for functionals with the tolerance level $\epsilon = 10^{-6}$ and a fixed step size $t^{(k)} = 0.01$.

4.2 Flow Chart

The basic structure of the program can be viewed in Figure 4.2. In this flow chart, one can see the different stages that the program goes through and each step will be discussed in separate sections.

The program use a conjugate gradient solver with an SSOR preconditioner as discussed in Sections 3.7 and 3.8.

All the input parameters for DeFuSE can be found in Appendix A.

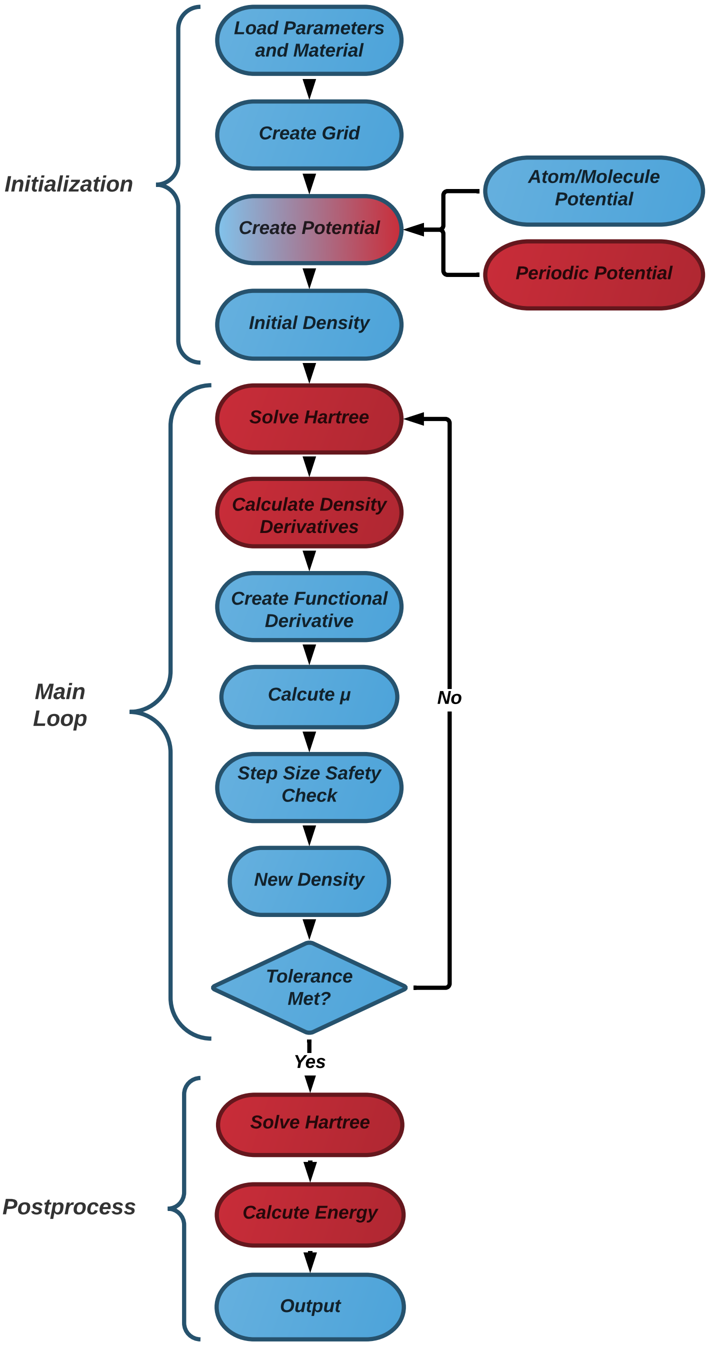

Initialization
Load Parameters and Material
Create Grid
Create Potential
Atom/Molecule Potential
Periodic Potential
Initial Density
Main Loop
Solve Hartree
Calculate Density Derivatives
Create Functional Derivative
Calcute μ
Step Size Safety Check
New Density
Tolerance Met?
No
Yes
Postprocess
Solve Hartree
Calcute Energy
Output

Figure 4.2. The flow chart of DeFuSE. The red parts is where the finite element method is used.

4.3 Material and Mesh

The first step in DeFuSE is to read the material species file and input parameters. The material is loaded using a vasp poscar file [26]. In this poscar file, the size of the simulation box (or unit cell for periodic systems) is specified and where atoms should be placed. The simulation box is the geometric domain we use for the simulation, and corresponds to Ω as discussed in Section 3.3. For atoms and molecules, we would like to simulate an infinite space but this is not possible in a computer hence a simulation box is required. For periodic systems, the simulation box is the size of the unit cell. When the information in the poscar file has been uploaded, the simulation box is created and the refinement begins.

In the program, there are three levels of refinement choices: Pre-, atom- and post-refinement. The pre-refinement is used to make sure that there is a node at each atom position. This has to be done prior to the actual first step in the calculation since the program does not do this automatically. The pre-refinement refines all cells in the mesh. After that, the atom-refinement is made. This refines additional cells around the nucleus inside a given radius. Finally, the post-refinement can be used to refine all cells again. The different refinement levels have the following notation: 3:(5,1):0 where 3 is the pre-refinement level, 5 is the atom-refinement level with radius 1 bohr and 0 for the post-refinement level. Since the post-refinement is usually omitted the notation 3:(5,1) is also used.

After the mesh has been constructed the atoms are placed at their positions.

4.4 External Potential

After loading the material we need to determine the external potential since this is the driving force for finding the density for the specific system.

For atoms and molecules it is easy to define the external potential, since we know the analytic description $v(\mathbf{r}) = -Z/\mathbf{r}$ for every atom. Hence, in the program the positions of the nodes are entered into this equation and a potential is made for each atom in the material. It is also possible to use a soften coulomb, *i.e.*, an external potential on the form $v(\mathbf{r}) = -Z/(\mathbf{r} + c)$.

The external potential for periodic systems is less straightforward. One needs to make an external potential that is periodic and takes into account all the infinite copies of itself. In DeFuSE, this is done by solving Poisson's equation, with delta functions centred at each atomic site times the nuclear charge, with periodic boundary conditions. This results is the equation

$$-\nabla^2 v(\mathbf{r}) = \sum_i 4\pi Z_i \delta(\mathbf{r} - \mathbf{r}_i) \tag{4.1}$$

where Z_i is the nuclear charge and r_i the position of the atom in the material. For a non-periodic system, the solution would simply reproduce the usual atomic potential.

$$v(\mathbf{r}) = \int \frac{Z_i \delta(\mathbf{r}' - \mathbf{r}_i)}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' = \frac{Z_i}{|\mathbf{r} - \mathbf{r}_i|} \tag{4.2}$$

4.5 Initial Density

There are two choices that have been implemented for the initial density: A constant distribution across the simulation box and a Gaussian distribution. Both of these obey the condition $\int n(\mathbf{r})d\mathbf{r} = N$, where N is the total number of electrons.

The constant distribution is simply $n(\mathbf{r}) = N/V$ where V is the volume of the simulation cell.

The case of the Gaussian distribution is a little more complicated. This adds Gaussian distribution at all the nucleus in the simulation box. These are weighted so that a heavier nucleus gets a higher density that a lighter ones:

$$n(r) = \kappa \sum_i Z_i e^{-r^2}. \tag{4.3}$$

Where κ is a normalization constant and Z_i is the nuclear charge.

4.6 Hartree Potential

The most time consuming part of the code is to solve for the Hartree potential. This has to be done in each iteration after a new density has been calculated. In Eq. (2.24) the Hartree potential $v_i(\mathbf{r})$ is defined in terms of Poisson's equation.

$$-\nabla^2 v_i(\mathbf{r}) = 4\pi n(\mathbf{r}) \tag{2.24}$$

where the boundary conditions are chosen appropriately depending on whether the system is finite or periodic.

4.6.1 Finite Systems

The finite element method needs boundary conditions to converge, but the definition of the hartree potential do not provide any. A solution to this problem is presented in the work by Rostgaard [27], which is called the Gaussian neutralization. The principle of this method is that you add a negative Gaussian distribution $g(\mathbf{r})$, for each of the atoms in the material, so the average density is zero. This means we can solve Poisson's equation with Dirichlet boundary condition equal to zero. Since the solution to Poisson's equation with a Gaussian distribution $v_g(\mathbf{r})$ is given analytically, we can subtract it afterwards.

$$g(\mathbf{r}) = -Ze^{-r^2/2}, \; -\nabla^2 v_g(\mathbf{r}) = 4\pi g(\mathbf{r}), \; v_g(\mathbf{r}) = -Zerf(\mathbf{r}/\sqrt{2})/\mathbf{r} \tag{4.4}$$

Hence, the scheme for solving the Hartree potential for an atom or molecule is now determined.

$$-\nabla^2 v_i(\mathbf{r}) = 4\pi(n(\mathbf{r}) + g(\mathbf{r})) \; in \; \Omega \tag{4.5}$$

$$v_i(\mathbf{r}) = 0 \; on \; \delta\Omega \tag{4.6}$$

$$v_i(\mathbf{r}) = v_i(\mathbf{r}) + v_g(\mathbf{r}) \tag{4.7}$$

4.6.2 Periodic Systems

In order to use periodic boundary conditions as discussed in Section 3.6, the right-hand side needs to integrate to zero. Hence, a uniform background $-N/V$ is added to Poisson's equation for periodic systems.

$$-\nabla^2 v_i(\mathbf{r}) = 4\pi(n(\mathbf{r}) - N/V) \; in \; \Omega \tag{4.8}$$

After the solution $v_i(\mathbf{r})$ has been obtained, it is displaced so it integrates to zero.

4.7 Functional Derivatives

When the density and a corresponding Hartree potential is determined, the program moves on to compose the functional derivatives. In the case of Thomas-Fermi theory we get the following equation:

$$\frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})} = v(\mathbf{r}) + v_i(\mathbf{r}) + \frac{1}{2}(3\pi^2)^{2/3}\big(n(\mathbf{r})\big)^{2/3} \tag{4.9}$$

where $v(\mathbf{r})$ is the potential, $v_i(\mathbf{r})$ is the Hartree potential and the last term is the kinetic energy. If one is using a cusp condition, this contribution shows up in the potential like we discussed in Section 2.6.

When the functional derivative is formed, the chemical potential μ is found by integrating over the simulation space Ω:

$$\mu = \frac{1}{V}\int\limits_{\Omega} \frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})} d\mathbf{r} \tag{4.10}$$

where V is the volume of the simulation space Ω. Now μ can be subtracted from the functional derivative and the stopping criterion is checked:

$$\left|\left|\frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})} - \mu\right|\right|^2 < \epsilon. \tag{4.11}$$

If the requirement is not met, the steepest descent continues and a new density is found. How the new density is found is described in the next section. However, a side remark about what happens when the energy functional is extending with Amaldi, Dirac or Weizsäcker terms, as described in Section 2.5.

The exchange functional derivatives of Amaldi Eq. (2.30) and Dirac Eq. (2.32) are only depended on the density and can be added the same way as the Thomas-Fermi term. The Weizsäcker term is more complicated. First, one needs to calcute the first $|\nabla n(\mathbf{r})|^2$ and second derivate $\nabla^2 n(\mathbf{r})$ to form the Weizsäcker functional derivate as in Eq. (2.29). This is done by solving two separate finite element problems as discussed in Section 3.9.

4.8 New Density

To construct the next density we use the equation

$$n(\mathbf{r})^{(k+1)} = n(\mathbf{r})^{(k)} - t^{(k)}\Big(\frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})} - \mu\Big) \tag{4.12}$$

as discussed in Section 3.10. In the steepest descent, one of the steps is to calculate the step size $t^{(k)}$. An even simpler way is to have a fixed step size. This approach is chosen for the DeFuSE code, this only affects the speed of convergence. A bigger step size means faster convergence but if it is too big, problems could occur when trying to keep the density positive. A number of step sizes were tested and a good value that holds this balance is $t = 0.01$.

However, as the density is updated, it is possible that the density becomes negative in some spacial regions. To avoid this problem, the step size is always taken as the smaller value of (i) the intended step size or (ii) half the step size that would turn any point of the density negative. If this criterion limit the step size to be below 10^{-12} the program terminates. A schematic diagram of this routine is shown in in Figure 4.3.

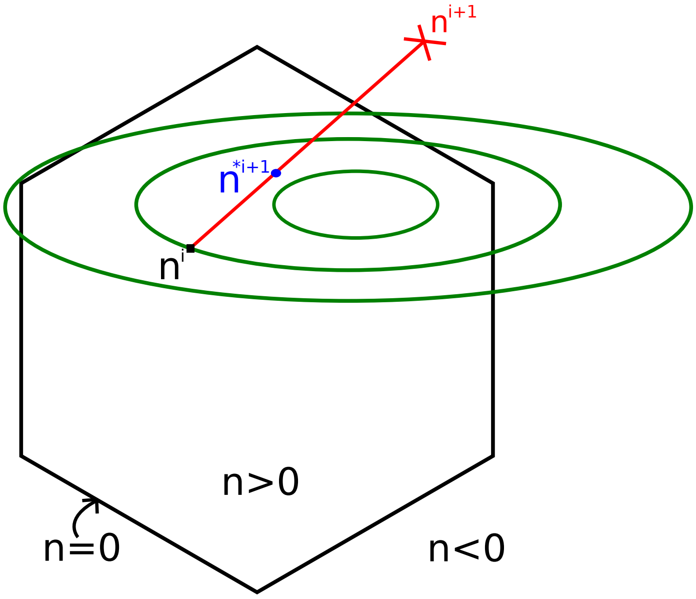


Figure 4.3. A schematic illustration of calculating a new density. Here the black hexagon represents the allowed densities and the green ellipses represent contour lines for the functionals. If a step size moves from a allowed density, the black square n^i, to a non allowed density, the red cross n^{i+1}, a density which is outside the hexagon. Then the step correction changes the step size so that the density becomes allowed, to the blue circle n^{*i+1}.

The steepest descent loop continues to run until the tolerance level is met. For all simulation in this thesis, the tolerance is set to 10^{-6}.

4.9 Calculate Energy

When the steepest descent is finished and the minimized density is found. The final step in DeFuSE is to calculate the energy. For the case of Thomas-Fermi, the energy is given by:

$$E[n] = \int v(\mathbf{r})n(\mathbf{r})d\mathbf{r} + \frac{1}{2}\int v_i(\mathbf{r})n(\mathbf{r})d\mathbf{r} + \frac{3}{10}(3\pi^2)^{2/3}\int \big(n(\mathbf{r})\big)^{5/3}d\mathbf{r}. \tag{4.13}$$

The two last terms are simple to calculate by appropriate routines in the Deal.ii library but the first term has singularities that are difficult to integrate. Hence, we have instead devised an alternative method using the Hartree potential. Starting from the definition, Eq. (2.23).

$$v_i(\mathbf{r}) = \int \frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}' \tag{2.23}$$

If we set $\mathbf{r} = \mathbf{r}_i$ to the position of the atom and multiply with the atomic number Z_i we get the desired integral

$$Z_i v_i(\mathbf{r}_i) = \int n(\mathbf{r}')\frac{Z_i}{|\mathbf{r}_i-\mathbf{r}'|}d\mathbf{r}' = \int n(\mathbf{r}')v(\mathbf{r}')d\mathbf{r}'. \tag{4.14}$$

The external potential integral is replaced by a sum of Hartree potentials.

$$E[n] = \sum_i Z_i v_i(\mathbf{r}_i) + \frac{1}{2}\int v_i(\mathbf{r})n(\mathbf{r})d\mathbf{r} + \frac{3}{10}(3\pi^2)^{2/3}\int \big(n(\mathbf{r})\big)^{5/3}d\mathbf{r} \tag{4.15}$$

This equation is the energy for the Thomas-Fermi model. If one uses more functional terms these energies are calculated as shown in Section 2.5.

4.9.1 Additional Energy Terms

There are some energy contributions that are not included in the functional description. These are the interaction between nuclei in molecules and bulk simulation. For atoms, the contribution is the Coulomb interaction

$$E_{\text{ion}-\text{ion}} = \frac{1}{2}\sum_{i,j,i\neq j}\frac{Z_i Z_j}{|\mathbf{r}_j - \mathbf{r}_i|} \tag{4.16}$$

where i and j are the numbers of atoms for the material. This term is zero for a single atom.

When we do bulk simulations, *i.e.*, periodic conditions, we use the Ewald summation as discussed in Section 2.7. This is implemented using the materials science library Pytmatgen [20].

Chapter 5

Results and Discussion

In this chapter, the results from the DeFuSE program is presented. First, a couple of convergence tests are performed to determine appropriate parameters for further computations. These tests primarily use a hydrogen atom setup since this is the simplest system possible and the results can be compared with the exact solution.

The exact energy for a single hydrogen atom is $E = -13.6$ eV $= -0.4998$ hartree for the ground state with the corresponding density $n(\mathbf{r}) = e^{-2r}/\pi$ bohr^{-3}.

After these tests, we will see if we can replicate results from the papers [17, 28] and test the new bulk simulation method. The last test is to see how well the gradient correction terms, like the Weizsäcker, works.

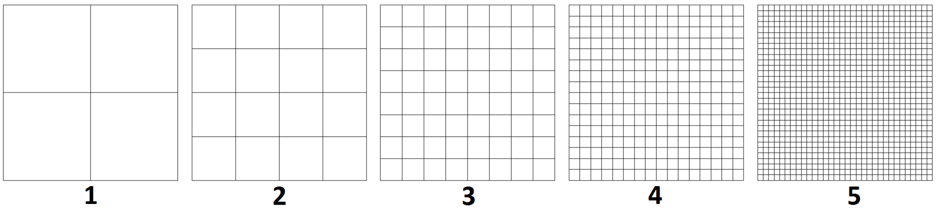
Additional data for certain tests are found in Appendix B.

5.1 Convergence

In this section, we will decide what convergence parameters are appropriate for the hydrogen atom. These tests will decide what finite element (Q1,Q2,Q3), mesh, starting density and size of the simulation box to use for the Thomas-Fermi functional with the cusp condition. The final test is for the Thomas-Fermi functional with a softened Coulomb potential.

5.1.1 Element Selection

First, we need to decide the order of element to use. The options are tri-linear Q1, tri-quadratic Q2 or tri-cubic Q3 Lagrange elements. This test is executed with a uniform mesh with a simulation box size of $5x5x5$ bohr using the Thomas-Fermi functional Eq. (2.25) with cusp condition Section 2.6. The initial density is set to a constant distribution. Then a series of refinement steps will be carried out to see how the total energy and density behaves. The different refinement levels can be viewed in Figure 5.1.



1 2 3 4 5

Figure 5.1. The different levels of refinement that were tested for the Thomas-Fermi functional with cusp condition. The hydrogen atom is located at the center of the simulation box.

In Figure 5.2, the total energies and total time of the program for the different elements are shown and in Figure 5.3, the density for Q2 elements is shown.

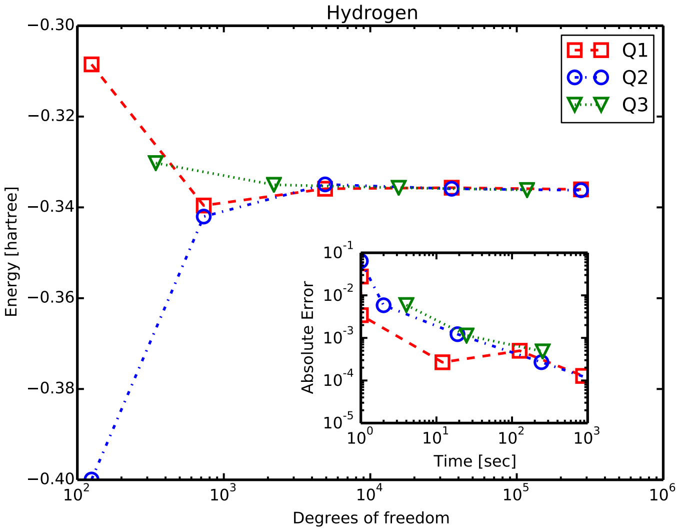


Figure 5.2. The total energy of the hydrogen atom as a function of the number of degrees of freedom, for the different kinds of Lagrange elements (Q1,Q2,Q3), using the Thomas-Fermi functional with the cusp condition. It is shown that the total energy converges to a value of about −0.34 hartree. The inset shows how the error depends on the time for the different kinds of Lagrange elements (Q1,Q2,Q3). The data that was used for this plot can be found in Appendix B.

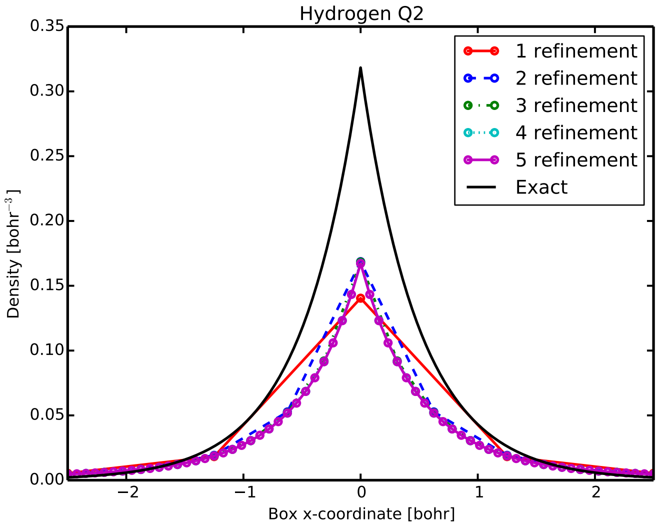


Figure 5.3. Calculated densities of the hydrogen atom using the Thomas-Fermi functional with the cusp condition, for different levels of refinement. In this case, the Q2 Lagrange element have been considered. The black curve shows the exact density of hydrogen.

The density at the nuclei in Figure 5.3 seems to converge to roughly one half of the exact density of the hydrogen atom.

Discussion

As can be seen from Figure 5.2, the energy converges to about -0.34 hartree for each of the different choices of elements. In this figure, one can also see that the Q1 elements have some oscillations in the error. It is generally recommended to use Q2 or Q3 elements when doing simulations in 3D although Q1 seems to give a fast rough estimate in this case. The Q1 elements converges first, but the other elements provide better results for the final error. The Q2 elements achieves a good trade-off between speed and accuracy and seems to converge without any oscillations.

In the following tests, Q1 elements are used for most applications to determine if the results are reasonable and Q2 elements are used for results that require greater accuracy.

5.1.2 Mesh and Initial Density

In order to converge the results faster, a better mesh should be used in addition to having a more suitable initial density. If one study how the density changes in Figure 5.3 one can see that the most change occurs inside a sphere of radius one bohr around the atomic nucleus. Hence, to concentrate the mesh around the nuclei seems to be a good idea.

In order to illustrate that this does not affect the results but only the speed of convergence, a test was carried out with the simulation box size set to $5x5x5$ bohr and using Q2 elements for the two different meshes. In Figure 5.4, one can see the two meshes that were used. The uniform mesh 5 is comprised of elements that have the volume 0.15625^3 bohr. The concentrated mesh 3:(5,1) also have elements of volume 0.15625^3 bohr at the nucleus but bigger element, with volume 0.625^3 bohr, at the boundary.

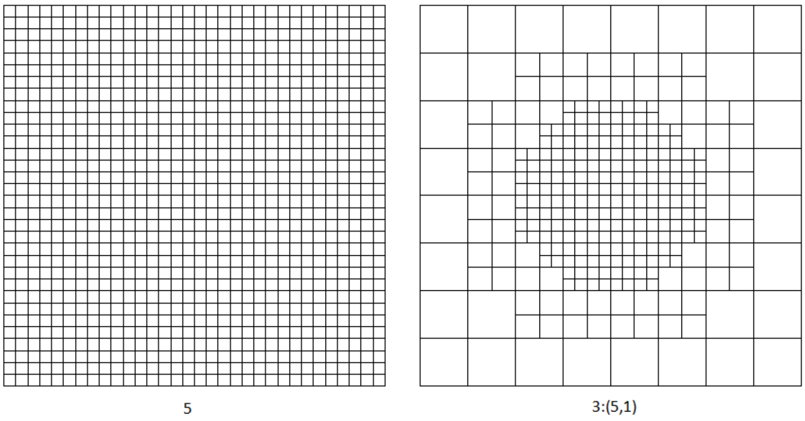


Figure 5.4. The two meshes used in the test. The uniform mesh 5 has 274625 degrees of freedom and the concentrated mesh 3:(5,1) has 32829.

Table 5.1 shows the influence of the mesh and starting density on the energy and total run time.

Mesh	Initial density	Final error	μ	Energy [hartree]	Time [sec]
5	constant	$9.63073 \cdot 10^{-7}$	0.118646	-0.3362	3064
5	Gaussian	$9.56019 \cdot 10^{-7}$	0.118661	-0.336221	1752
3:(5,1)	constant	$9.63118 \cdot 10^{-7}$	0.118644	-0.336205	244
3:(5,1)	Gaussian	$9.56912 \cdot 10^{-7}$	0.118646	-0.3362	145

Table 5.1. The total energy values and total time for the program for the different meshes and starting densities using the Thomas-Fermi functional with the cusp condition.

Discussion

Table 5.1 shows that both the chemical potential and the energy has the same significant figures to a precision of 10^{-4}. That the result varies beyond this point could be due to the final error of the program, even running a simulation with the same mesh but different initial density gives a different final error which could affect the value slightly. There may also be some numerical error in the software.

A trend is clear that with a better constructed mesh and starting value, the program converges faster without affecting the values beyond a reasonable point, an error less than 10^{-4}. It is a coincidence that the mesh 5 with constant density and the mesh 3:(5,1) with Gaussian density has the same value for μ.

5.1.3 Simulation Box Size

In this section, we discuss the impact on the energy as the simulation box size is changed. The tests so far have been carried out with a $5x5x5$ bohr simulation box. This test have been carried out with the improved mesh 3:(5,1) and as we doubled the size of the simulation box, we also added an extra level of refinement. The end result is that the actual size of the elements remain the same and the only thing that changes is the simulation box. The starting density was set to the constant distribution and the element was set to Q2. The results can be viewed in Figure 5.5 and in Table 5.2.

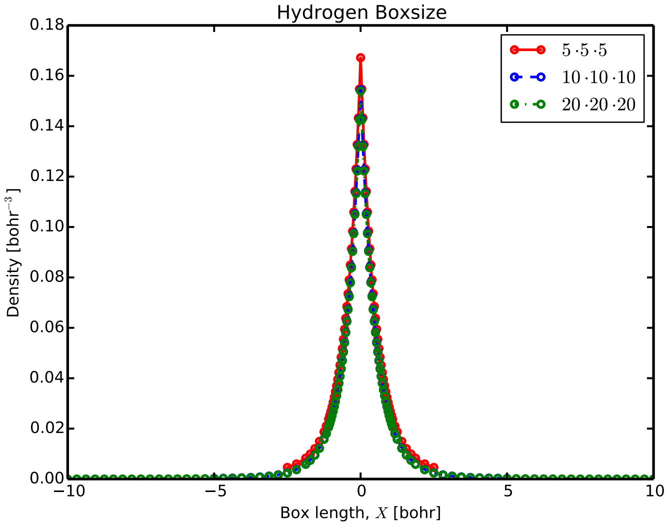


Figure 5.5. Calculated densities of the hydrogen atom, for simulation box sizes $5x5x5$, $10x10x10$ and $20x20x20$. The biggest change between the different simulation box sizes is the value at the nuclei.

Simulation box size	Refinement	μ	Energy [hartree]	% change
5x5x5	3:(6,1)	0.118646	-0.3362	-
10x10x10	4:(7,1)	0.0188453	-0.359031	6.9 %
20x20x20[1]	5:(8,1)	0.00239274	-0.359208	0.05%

Table 5.2. The result from changing the simulation box size for the Thomas-Fermi functional with the cusp condition.

Discussion

One can clearly see that the simulation box size affects the total energy. As the simulation box size increases, the value of the total energy levels out even though the density looks more or less the same for all simulation box sizes. A $10x10x10$ bohr is sufficient for Thomas-Fermi theory with the cusp condition.

It can also be seen that the value of the chemical potential μ is decreasing as the size of the simulation box is increasing. For hydrogen (and any other neutral atom), the chemical potential is $\mu = 0$. Hence, μ can be viewed as an indicator of a sufficiently sized simulation box.

A final observation that is worth noticing is that during iteration for convergence, the limitation on step size to avoid negative densities was applied and reduced the step size for the run with the $20x20x20$ bohr simulation box. This limitation was probably necessary due to the fact that the value for the density near the boundary is very close to zero.

5.1.4 Progression

With the convergence parameters for hydrogen now determined, we study how the energy is changing during the run of the program. The following calculations use a $10x10x10$ bohr simulation box with refinement 3:(7,1) with Q2 element and a constant density as initial density. The energy levels are shown in Figure 5.6.

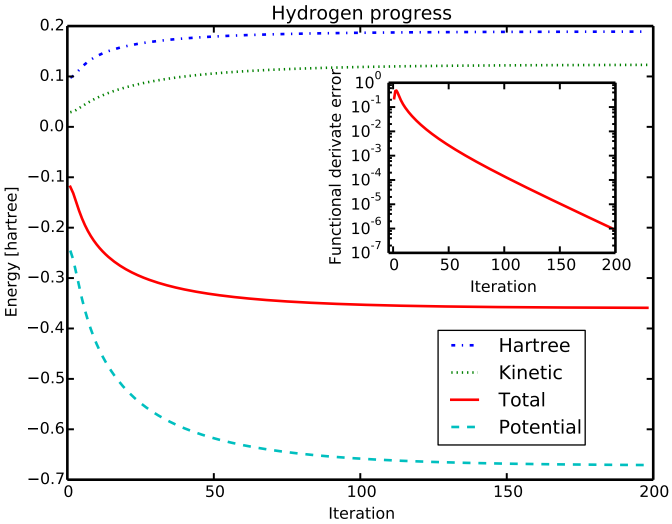


Figure 5.6. Values of the Hartree energy, the kinetic energy, the potential energy, and the total energy as function of the iteration. This run uses the Thomas-Fermi functional with the cusp condition. In the inset, one can see the functional derivative error Eq. (4.11) in each iteration.

---

[1] The step safety routine changed the step size from 0.01 to 0.005 during the program run

Discussion

A note to make is that the error is slightly increasing in the beginning. This is probably due to the initial density is constant in the beginning and this is a very poor representation of the correct density.

5.1.5 Soft Coulomb

We have so far presented calculations using the cusp condition. This section explores the results of using a softened Coulomb potential Eq. (2.35). The main question is whether hydrogen with a softened Coulomb potential gives a result closer to the exact energy of Thomas-Fermi $E = -0.7687Z^{7/3}$. This test was carried out using a $5x5x5$ bohr simulation box with Q1 elements. The results are shown in Figure 5.7, it is seen that the energy is getting closer to the exact value as the softening parameter increases. Calculations with a further reduced Coulomb softening parameter, $c = 0.001$, did not converge. The density for the softened Coulomb can be seen in Figure 5.8.

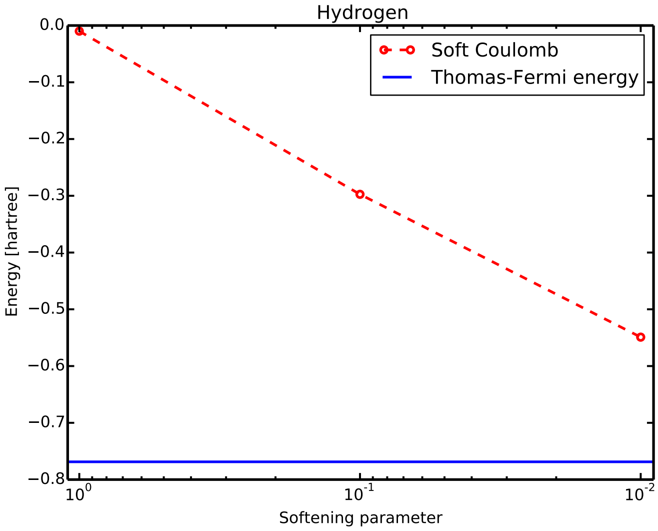


Figure 5.7. The energy for the 1.0, 0.1 and 0.01 softened Coulomb vs. the exact value of the Thomas-Fermi energy. Convergence test data is presented in Appendix B.

Discussion

Using a softened Coulomb potential shows that the calculations converge towards the exact Thomas-Fermi result when the softening of the Coulomb is decreased. We cannot use the exact atomic potential as discussed in Section 2.6. The problem is the exact Thomas-Fermi density diverges at the nuclei, the tendency can be

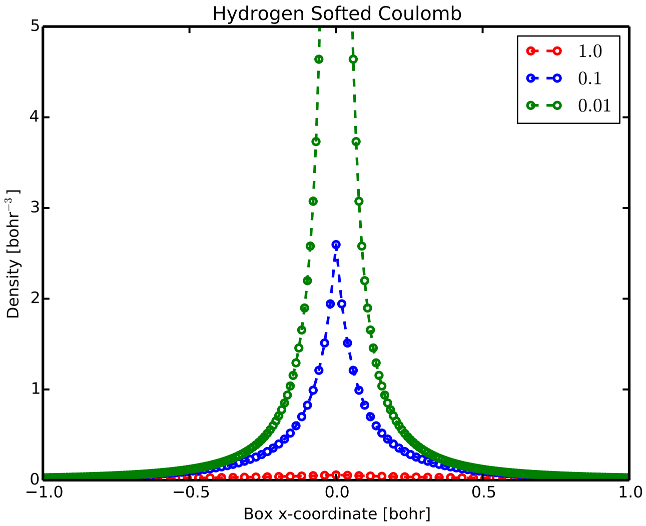


Figure 5.8. Calculated densities with various softened Coulomb terms, with values of c equal to 1.0, 0.1, and 0.01.

seen in Figure 5.8. Hence, a very tight mesh is required to capture the correct behaviour as the softening parameter gets smaller. To make calculations converge with a parameter value of 0.001 may be possible in a smaller simulation box. The present test is executed for hydrogen which is the easiest to converge. For heavier elements the calculations are likely to be difficult to converge.

5.2 Benchmark Calculations

In this section, we compare the results from our code with the results from the work by Kim, Youn and Kang [28] for the H_2 molecule. To do this comparison, we first investigate how the exchange terms Amaldi and Dirac influence the solution.

Finally, we perform calculations of noble gases using the Thomas-Fermi functional with cusp condition and compare with results of Parr and Ghosh [17].

5.2.1 Exchange Term

First, we investigate the result of using the Amaldi exchange functional. A simulation with the Amaldi functional Eq. (2.30) in a $5x5x5$ bohr simulation box with the 3:(5,1) mesh did not converge, so the simulation box size was reduced to $2x2x2$ bohr. Then the simulation box size was gradually increased as can be seen in Figure 5.9. These tests were carried out with Q1 elements.

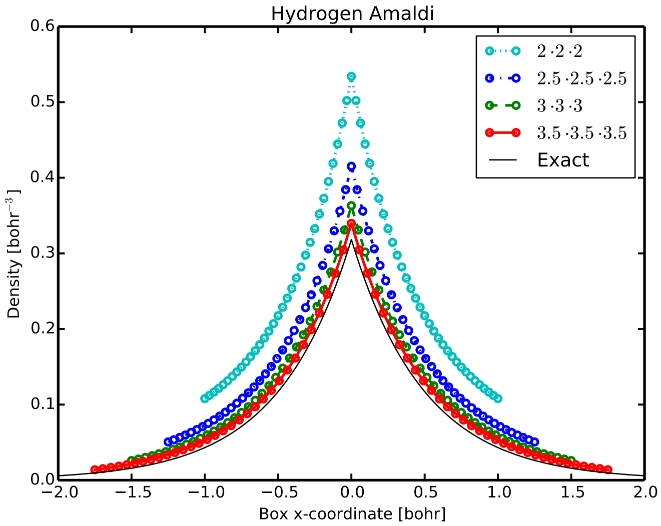


Figure 5.9. The density gets closer to the exact solution as the simulation box size increases for the Thomas-Fermi-Amaldi functional with the cusp condition.

The energy values and chemical potential for the different simulation boxes can be viewed in Table 5.3.

Simulation boxsize	μ	Energy [hartree]
2x2x2	0.0813442	-0.14876
2.5x2.5x2.5	-0.158784	-0.401915
3x3x3	-0.263315	-0.536374
3.5x3.5x3.5	-0.310098	-0.613706

Table 5.3. The effects of simulation box size using the Thomas-Fermi-Amaldi functional with the cusp condition.

The results approach the exact density as the simulation box size increases. The $4x4x4$ bohr simulation box did not converge. A similar test was carried out with the Dirac functional Eq. (2.32) instead. These results are shown in Figure 5.10.

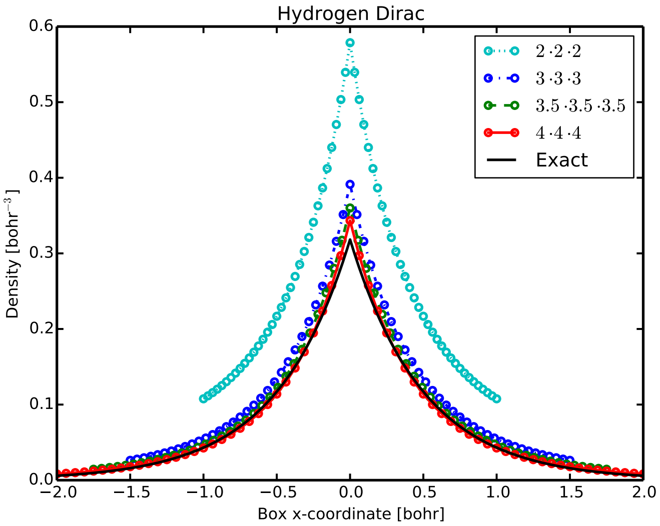


Figure 5.10. The density gets closer to the exact solution as the simulation box size increases for the Thomas-Fermi-Dirac functional with cusp condition.

These results are similar to the Amaldi results, but now the calculations converge with a larger simulation box, the limit is $5x5x5$ bohr. In Table 5.4, the chemical potential and the total energy is shown.

Simulation box size	μ	Energy [hartree]
2x2x2	0.290743	-0.172238
3x3x3	0.0247714	-0.474284
3.5x3.5x3.5	-0.0156867	-0.531774
4x4x4	-0.0371524	-0.564679

Table 5.4. The effects of simulation box size using the Thomas-Fermi-Dirac functional with the cusp condition.

To verify that these energies are correct, a convergence test was preformed for the Dirac functional. This convergence test was carried out for the $3.5x3.5x3.5$ bohr simulation box with Q2 element in order to increase the accuracy. The results of this test are shown in Table 5.5.

Refinement	μ	Energy [hartree]
3:(4,1)	-0.0156898	-0.531786
3:(5,1)	-0.0156873	-0.532106
3:(6,1)	-0.0156892	-0.532161

Table 5.5. Converged total energies for different levels of refinement for the Thomas-Fermi-Dirac functional with the cusp condition.

Discussion

As one increase the size of the simulation box for the Amaldi functional the density gets closer to the exact value and the energy decreases. For this functional, the density is consistently larger than the exact density and the chemical potential μ decrease with the size of the simulation box.

When using the Dirac exchange functional instead, the density also approaches the exact solution, but for the largest simulation box ($4x4x4$), the converged density ends up below the exact density in some spacial regions. Furthermore, the chemical potential μ is positive for small simulation boxes and becomes negative for larger simulation boxes. In Table 5.4, one see that $\mu = 0$ is somewhere between the values for the simulation box size $3x3x3$ and $3.5x3.5x3.5$ and the exact energy value is also between these simulation boxes. Our results suggest that μ still can be viewed as an indicator to see if the size of the simulation box is correct.

We also presented a convergence test for the Dirac exchange functional. Both the density and the total energy are much closer to the exact solution when using the Thomas-Fermi-Dirac with the cusp condition than just the Thomas-Fermi with the cusp condition.

5.2.2 H_2 Molecule

In the following test, we aim to reproduce the bonding energy for H_2 molecule as calculated by Kim, Youn, and Kang Ref. [28]. They used the Thomas-Fermi-Amaldi functional with the cusp condition in their article. First, we used a $6x3x3$ bohr simulation box with a uniform mesh refinement 6 and with different positions of the two atoms. In the second test, we changed the length of the simulation box in such a way that there always is a 3 bohr margin from the atoms to the edge of the simulation box. This mesh has the refinement of 6:(7,1). To make the element less rectangular, an extra refinement in only the x-dimension was carried out first. This means that the first test has cubic cells but the second test has more rectangular cells as the simulation box increases. The meshes used in the calculations are shown in Appendix B.

The energy value is shown in Figure 5.11 and the densities for our two tests are shown Figure 5.12 and Figure 5.13.

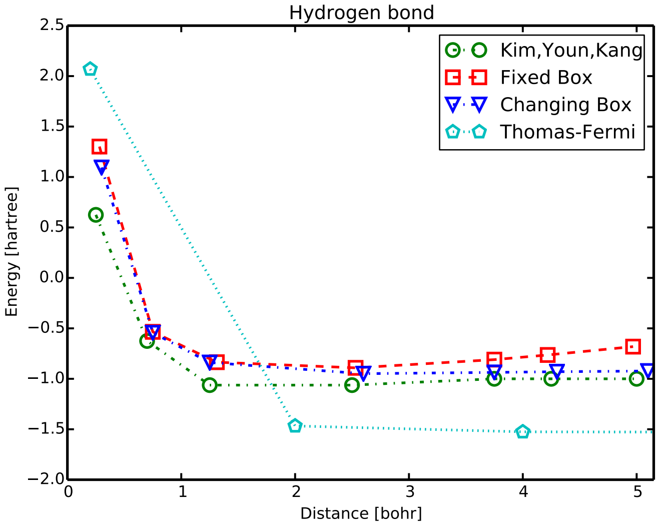


Figure 5.11. Bonding energy in an H_2 molecule. The results from the Ref. [28], our results using a fixed and changing simulation box for the Thomas-Fermi-Amaldi functional with the cusp condition and the traditional Thomas-Fermi energy taken from the Ref. [29].

The energies in Figure 5.11 have a slight absolute offset relative to values of Ref. [28], but the shape is roughly the same. Similar to the findings of Ref. [28] we do get bonding in H_2, giving a strong bond with the fixed simulation box and a very weak for the changing simulation box setup.

In Figure 5.12, we show the density for the fixed $6x3x3$ bohr simulation box and in Figure 5.13, we show the density for the simulation box with x-dimension that change as function of hydrogen-hydrogen distance.

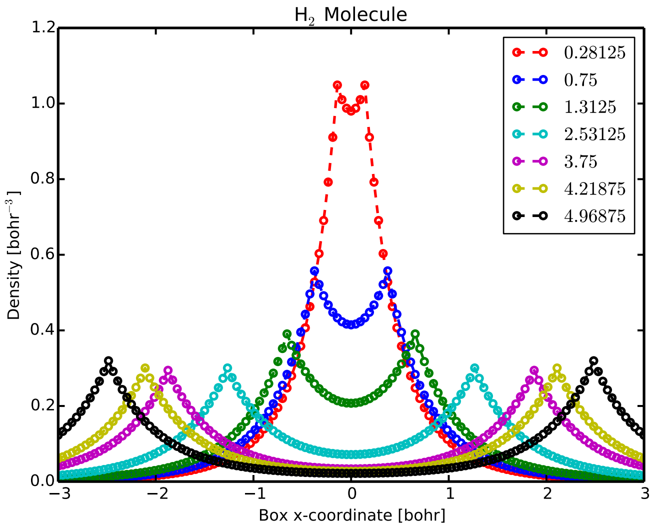


Figure 5.12. The densities for different bond length in the 6x3x3 simulation box.

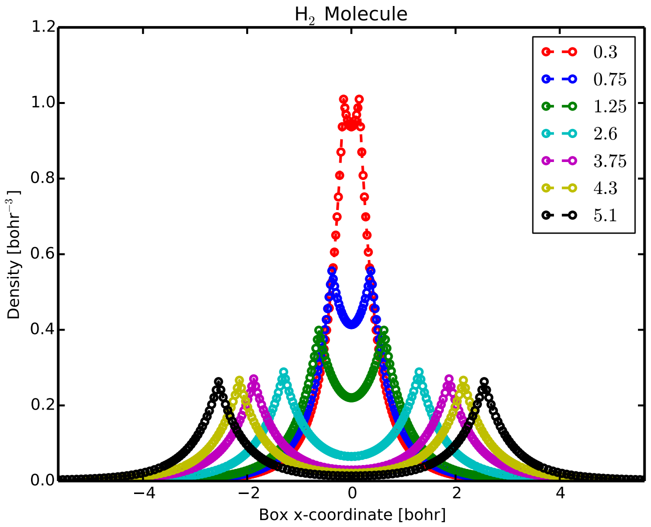


Figure 5.13. The densities for different bond length in the simulation box with varying x size.

Discussion

There are several possible explanations for the absolute offset found relative to the total energies in Ref. [28]. They do not specify what size of the simulation box or the element that are used, so it is difficult to compare the exact same setup to investigate the differences further.

The reason for the strong binding in the $6x3x3$ bohr simulation box could be due to that the simulation box is too small and influence the solution. This is corroborated by the density plot in Figure 5.12, where it is seen that the density at the nucleus increases as the nucleus move closer to the edge of the simulation box. This was the motivation for carrying out the second test. In this setup, a much weaker bonding is observed. Our conclusion is that we are uncertain if this bonding is a physical result. It is possible that the bonding is caused by numerical errors, a too coarse mesh or loss of accuracy due to non-cubic elements. We have also seen that the total energy for The Amaldi functional is very dependent on the simulation box size as seen in Table 5.3, so varying the box size may influence the total energy as well.

One may also speculate how a better exchange functional, *i.e.*, Dirac exchange, would affect the results.

5.2.3 Cusp Condition

Parr and Ghosh [17] tested the Thomas-Fermi functional with the cusp condition for the noble gases. In this section, we compare the results of our implementation with theirs. First, a convergence test was made for He, this test can be found in Appendix B. The result from this test shows that using a $10x10x10$ bohr simulation box and the refinement 3:(7,1) with Q2 element gives very good values in a reasonable amount of time. This setup was then used for all the noble gases. The results are shown in Table 5.6 and the densities are shown in Figure 5.14.

Element (N)	μ	$-E/Z^{7/3}$	Ref. [17]	$n(0)/Z^3$	Ref. [17]
He (2)	0.024	0.4395	0.4397	0.2262	0.2274
Ne (10)	0.040	0.5746	0.5763	0.3671	0.3672
Ar (18)	0.046	0.6165	0.6110	0.4023	0.4020
Kr (36)	0.054	0.6642	0.6439	0.4293	0.4325
Xe (54)	0.059	0.5924	0.6599	0.4363	0.4460
Rn (86)	0.064	-0.049	0.6756	0.4358	0.4581

Table 5.6. The total energies for the noble gases using Thomas-Fermi with cusp condition. The N in the element column is for the number of electrons.

The densities for the different atoms can be view in Figure 5.14.

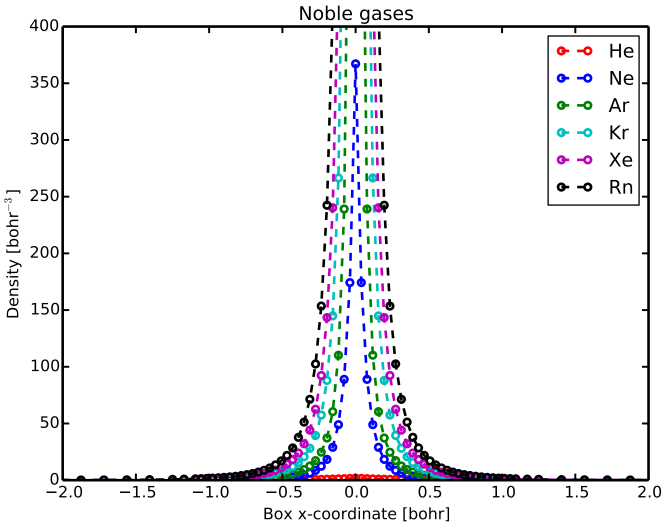


Figure 5.14. The densities for the noble gases using Thomas-Fermi with the cusp condition.

Discussion

The results for the lighter elements fit the data well but as the elements get heavier the data do not agree that well. The values for the energy differ more than the values for the density at the nuclei. This would indicate that the density at nuclei is reasonable correct and that the difference in energy values is most likely due to a too coarse mesh. The mesh does not capture the shape of the decreasing density correctly. The reason for this is probably that the density greatly increases in the region around the nucleus when the elements are getting heavier. This is shown in Figure 5.14. The density at the nuclei for the Rn atom is about 150000 times higher that of the He atom. The Rn atom has a density at the nucleus of 277214 bohr^{-3} compared with He which has 1.81007 bohr^{-3}. The mesh that worked fine for He cannot give an equally good value for Rn since the difference in density between the nodes are too great. This could be solved with a tighter mesh to capture the behaviour around the nuclei better, but this is very costly for heavy elements.

Another related issue is that the chemical potential increases for heavier elements. Since the density increases and while the simulation box remains the same size, the chemical potential μ increases. To counter this problem one would need to use bigger simulation boxes for heavier elements.

A test with mesh 3:(8,1) and a convergence test for Ne are shown in Appendix B. The test with mesh 3:(8,1) produces results closer to the reference data.

5.3 Bulk Systems

In this section, we will test how well the program works for periodic systems. We will first investigate how well the potential is produced and then we will investigate the convergence of the atomization energy of a cubic hydrogen crystal.

5.3.1 Periodic Potential

In order to produce good results for periodic systems, an accurate periodic potential is needed. To test the efficiency of the periodic potential Eq. (4.8), we made a convergence test with a $5x5x5$ bohr simulation box with Q1 element. This test uses hydrogen in a simple cubic structure. The reason we use hydrogen is that we want to be able to compare the test results with previous data. We emphasize that this is a completely artificial structure, since hydrogen does not have the structure in nature, used only as a straightforward convergence test. The resulting potentials are shown in Figure 5.15.

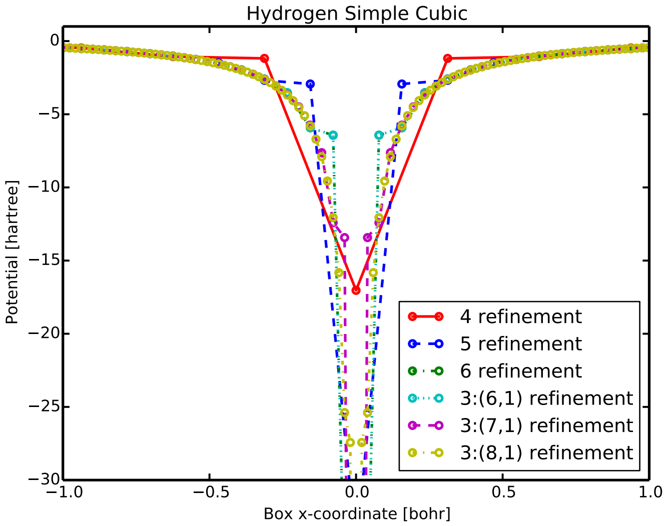


Figure 5.15. Periodic potentials for hydrogen in a simple cubic structure, for different levels of refinement.

The periodic potentials do no converge to the appropriate $-1/r$ potential shape at the nuclei, but rather to a finite value. If one would compare the potential with softened Coulomb potentials, the values at the nuclei would correspond to the following softening parameter that can be seen in Table 5.7.

Refinement	Soften Parameter
4	0.0588
5	0.0288
6	0.0143
3:(6,1)	0.0143
3:(7,1)	0.0071
3:(8,1)	0.0035

Table 5.7. The softened Coulomb parameters corresponding to the periodic potential solution for different levels of refinement.

Potentials obtained by solving Eq. (4.8) become periodic. To illustrate this point, we show the result when a smaller simulation box $2x2x2$ with a refinement of 6 was used. The result is shown in Figure 5.16.

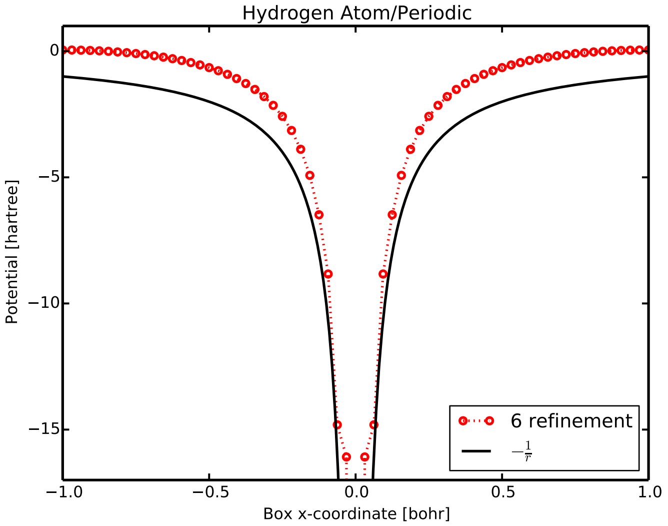


Figure 5.16. The red curve is the periodic solution from DeFuSE and the black curve is the atomic potential for a hydrogen atom. Note that the red curve have been offset to integrate to zero. The red curve is horizontal at the boundary compared with the atomic potential that is inclined.

Discussion

Since the periodic potential does no capture the correct $-1/r$ potential the bulk simulation cannot run with cusp condition. The singularity from the cusp condition does not get cancelled by the potential singularity. Hence, if one would like to run a simulation for a bulk material, one would have to run it without the cusp condition.

Table 5.7 shows that the softening parameter is cut in half with an increased mesh. This suggests that a tight enough mesh may have an equivalent softening parameter small enough to use the cusp condition.

Even though the periodic system do not converge to the correct absolute value, one could still look at the atomisation energy. These results can be found in the next section.

5.3.2 Atomisation Energy

In this section, we study the atomisation energy for simple cubic hydrogen by comparing two different sized simulation boxes, one with $1x1x1$ bohr and one with $4x4x4$ bohr. The atomisation energy is given by: $E_{\text{atomisation}} = E_{1x1x1} - E_{4x4x4}$. This test is similar to the simulation box size test in Section 5.1.3, that in we use the same sized mesh and only changing the simulation box size. The $4x4x4$ bohr simulation box is refined two times more that $1x1x1$ bohr. The results are shown in Figure 5.17.

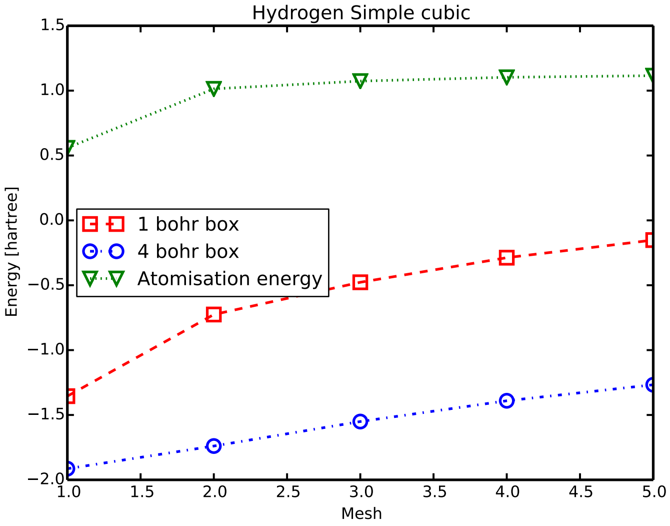


Figure 5.17. The energy from calculations using a $1x1x1$ bohr simulation box, a $4x4x4$ bohr simulation box and the atomisation energy for the Thomas-Fermi functional with periodic boundary conditions.

The atomisation energy in Figure 5.17 seems to converge to a value around 1.1 hartree.

Discussion

The absolute energies for the $1x1x1$ bohr simulation box and for the $4x4x4$ bohr simulation box have not converged as seen in Figure 5.17. This is similar to the results in Section 5.1.5 where the absolute value for the softened Coulomb have not converged as well. This is mentioned since the periodic potential could be compared to the softened Coulomb as shown in Table 5.7. Both simulation box sizes have an offset relative to the exact value but when we take the difference, this offset should cancel out between the two simulation boxes. The atomisation energy that can be seen in Figure 5.17 gives therefore give an indication that the behaviour is correct, but further testing is needed.

There is also an error from the mesh but since we use the same sized element, we should also have the same sized error. These errors should also cancel out in the atomisation energy.

5.4 Extension

In this chapter, we will discuss functionals with a dependence on the electron density gradient. However, within the present diploma work there was not sufficient time to complete this study.

5.4.1 Gradient Correction

This test was carried out to find out if the code produces correct derivatives. First, a test was preformed by taking the derivatives of a Gaussian function in a $5x5x5$ bohr simulation box with a uniform 5 mesh and Q2 element. The result is shown alongside the exact solution in Figure 5.18.

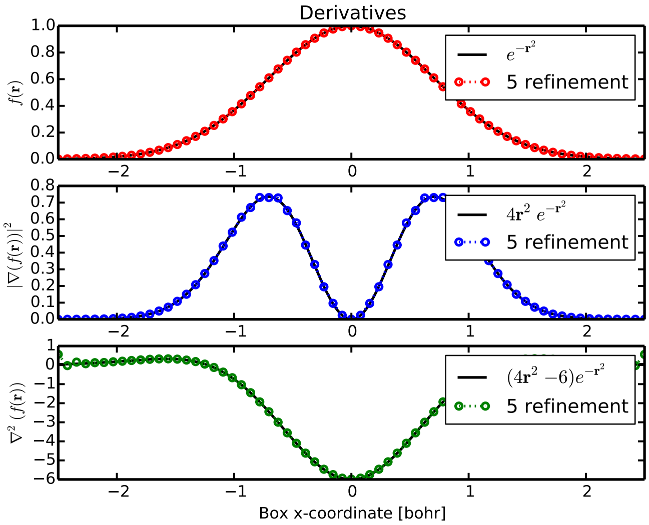


Figure 5.18. The first and second derivative of the Gaussian function $f(\mathbf{r}) = e^{-\mathbf{r}^2}$ in a $5x5x5$ bohr simulation box.

After this, we added the Weizsäcker term Eq. (2.29) to the Thomas-Fermi-

Dirac setup. Since this required a smaller simulation box, this test was carried out with $3x3x3$ bohr simulation box and a 3:(6,1) refinement, since this worked well for the Dirac case. This run did not converge and the error kept getting larger as the simulation went on. The result of the first step is shown in Figure 5.19

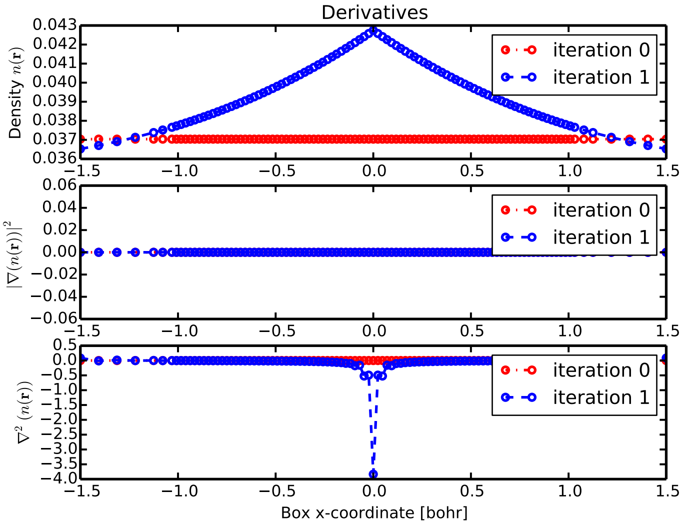


Figure 5.19. The density and density derivatives for the first step in the Thomas-Fermi-Dirac-Weizsäcker with cusp condition run.

Discussion

It appears that that the first and the second derivative work fine for smooth functions as seen in Figure 5.18. For functions that have a point that is non-differentiable, the derivatives appear to diverge. The density has these kinds of points at every nucleus. Hence, we need to find a way to handle such points before functionals that depend on the density gradient can be implemented.

Another observation in Figure 5.19 is that the gradient seems to handle these non differentiable points but this is most likely due to the fact that the difference between values in this plot is too small to see any difference in the gradient.

Chapter 6

Outlook

In this chapter we discuss remaining issues with the present implementation and ideas for solving them.

6.1 Density

As seen in Section 5.2.3 the representation of the density could be crucial. We discussed using the logarithm of the density $\chi(r) = e^{-n(r)}$ as the representation instead. This would solve the problem with negative densities and one would not have to worry about the step size. At the time, we considered it too be much work for a slight improvement, as one would need to redefine how the hartree potential is solved but it could be worth taking another look at. Perhaps there exists an even better representation of the density.

A better initial density would also aid the program to find the solution faster.

6.2 Mesh

There are several improvements that could be interesting. First, let's look at the placement of the atoms.

In the way the atoms are placed in the method now, they have to be placed at a node. An improvement of this approach would be to have a routine that places nodes automatically instead of having the need of calculating this beforehand. Another approach would be to make a code where the position of the atom does not matter.

Another interesting idea would be to use adaptive mesh instead of a fixed grid as the program has now. This idea has been tested for the Schrödinger's equation by Young and Armiento [30] and looks promising.

A final remark would be to change the element from quadrilaterals to triangles or hexagons. This could improve the generation of the mesh which in deal.II has some drawbacks. In other finite element codes there are many optimization schemes for the placement of polygon especially for triangles. This could solve the problem for placing atoms on nodes, since one would specify where certain nodes would be placed. Another advantage would be to specify the size of the element so there is a smooth transition from very finite element at the nucleus to coarser element at the boundary.

6.3 Convergence

The overall convergence of the main scheme depends on the step to find a new density, as discussed in Section 4.8. The step size is now fixed in the code, in the original steepest descent, the step size is calculated using a separate minimization problem:

$$n(\mathbf{r})^{(k+1)} = n(\mathbf{r})^{(k)} - t^{(k)}\Big(\frac{\delta E[n(\mathbf{r})]}{\delta n(\mathbf{r})} - \mu\Big) \tag{2.24}$$

We are looking for the $t^{(k)}$ that $\min_{t^{(k)}\geq 0}\ n(r)^{(k+1)}$ which occurs when the derivative of $\frac{\delta n(r)^{(k+1)}}{\delta t^{(k)}} = 0$. Calculations using this technique did not appear to converge well, but further testing is needed. We also implemented the conjugate gradient method, but this method gave problems with preserving the number of particles across iterations. Both of these approaches were taken from the book [25].

Another approach we did not have time to implement is the Newton method also described in the book [25]. An implementation of this method for a finite element code could be found at the FEniCS website [24]. The idea is to convert the non-linear problem into a series of linear problems.

One more bottleneck for the speed of the program is solving Poisson's equation to get the Hartree potential. In the code right now, the Hartree problem is solved with a tolerance of 10^{-6} every time. At the beginning of the program, this could be higher since the error from the starting density is greater and as the density converge the tolerance could be lowered.

6.4 Derivatives

As we mentioned in Section 3.9, it could be easier to obtain the derivatives of a function when they are represented with some types of elements. It is possible that a different choice of elements than the Lagrange elements could give better approximations of the density derivatives. The two properties that the elements should have are the following: provide good results for solving Poisson's equation and easy to find the first and the second derivatives.

Another issue is that the derivatives have non-differentiable points at the nucleus. A solution to this could be to add constraints in these points. In Section 2.6, we talked about the nuclear cusp condition. It is possible that this constraint, with some modifications, could solve the non-differentiable problem.

6.5 Periodic

The potential does not give the right correct behaviour at the nuclei for periodic systems. It is hard to say how this could be improved. One possibility is to try if a direct solver gives better accuracy or maybe the problem need to be divided into an analytical solution around the nuclei and Poisson's equation at the boundary.

The Ewald sum is not included in DeFuSE but uses the Pymatgen code [20]. This could be added or perhaps, another way to calculate the energy could be devised. In theory, one should be able to calculate the Ewald sum using $E = -\nabla(v(r))$.

6.6 Miscellaneous

In this section, we will mention a few other possible improvements.

There are other ways the nuclear cusp condition has been implemented and perhaps, there is a better way. For instance, in the article by Parr and Ghosh [17] they mention another way to enforce this condition by using $\int n(\mathbf{r})\nabla^2 n(\mathbf{r})d\mathbf{r} < \infty$. They got similar results with this constraint as $\int e^{-2k\mathbf{r}}\nabla^2 n(\mathbf{r})d\mathbf{r} < \infty$. As they discuss in their paper this means that the condition in not unique.

The software only works for systems with the same number of electrons as protons. This is probably not difficult to change so it is possible to test positive and negative ions. One would need to add an addition Gaussian in the Hartree potential to compensate for this charge.

At the moment DeFuSE can only do atoms, molecules and bulk materials. It would also be interesting to do surfaces. This would require an external potential that is periodic in one direction and finite in another. This would also need a different compensation term in the calculation of the Hartree potential to get the right solution.

In all simulations in the work, the simulation box has been a cube. It is possible that one could get better results to run the simulation in a spherical volume instead.

Another idea is to run the code in parallel to increase the performance of deal.ii solvers.

Chapter 7

Conclusions

We set out to make a tool that could be used for testing new functionals using finite elements. As a proof of concept, the program works and provides results that one would expect. We have shown that it is possible to add more complicated functionals to the Thomas-Fermi model and get convergence. To try more advanced functionals, the routines to calculate the electron density gradient and laplacian need to be completed. Once that is done, the program should be very useful for testing new functionals with gradient corrections for atoms and molecules.

Simulation of materials with periodic boundary conditions is still in the development phase. It is possible to perform such calculations, however, the calculation of the external potential needs to be improved in order to get results that are accurate.

To be able to use this software to estimate formation energies as a pre-screening step for thermodynamic stability in high-throughput calculations is still a long way to go. Many of the issues that need to be resolved before the program can be used in this way are discussed in Section 6.

As a final remark, one of the great benefits of solving orbital-free density functional theory with the finite element method is that it does not require any extra input than the essential information about the system. If we have a kinetic energy functional that could describe the nature of the electron well, this type of program would become a very useful general tool for atoms, molecules and materials involving all elements in the periodic table.

Appendices

Appendix A

Input parameters

Here are the input parameters with their default values for DeFuSE. This file loaded in the program using the class ParameterHandler in deal.II [21, 22].

```
# Load material.vasp from input folder
set Load material = true

# The path where the material should be loaded from
subsection Load material
set Path to material = input/material.vasp
end

# The path where the output files will be saved
set Output path = output/

# Choose between to have a cusp condition or a soft coulomb
to handle singulaties
set Potential factor = cusp condition

subsection Potential factor
set Soft coulomb = 0.01
end

# Activates the Dirac functional
set Dirac = false

# Activates the Amaldi functional
set Amaldi = false

# Activates the Weizsacker functional
set Weizsacker = false
```

```
subsection Weizsacker
set Weizsacker parameter = 0.2
end

# Activate on periodic boundary conditions
set Periodic boundary conditions = false

subsection Periodic boundary conditions
set Periodic boundary condition X = true
set Periodic boundary condition Y = true
set Periodic boundary condition Z = true
end

# Activate options for refinement of the mesh
set Set mesh parameters = true

subsection Set mesh parameters
set Pre refinement = 3
set Atomic refinement level = 4
set Atomic refinement size = 1
set Post refinement = 0
end

# Adds a gaussian compensation to the hartree potenial
set Gaussian density compensation = true

# The starting density
set Initial guess = constant

# The maximum number of iteration
set Iteration = 10000

# The chosen error tolerance
set Tolerance = 1e-6

# Save output files from a run
set Output = true

subsection Output
set Density = true
set Hartree = true
set Log density = false
set Electrostatic = true
set Save progress = false
```

```
subsection Save progress
set Frequency = 10
set Progress path = output/progess/
set Density = true
set Hartree = true
set Functional derivative = true
end
end

# Save output in text files
set Output text = true

subsection Output text
set Density = true
set Potential = false
set Hartree = false
set Energy = false
end
```

Appendix B

Additional Data

In this Appendix, data that did not fit in the results and discussion chapter is shown.

B.1 Element Selection

Here follows the values for Q1 in Table B.1, Q2 in Table B.2 and Q3 in Table B.3 elements that was used to make Figure 5.2.

Refinement	Degrees of freedom	μ	Energy [hartree]	Runtime [sec]
1	27	0.0411674	-0.0449898	1
2	125	0.116505	-0.308512	1
3	729	0.118897	-0.339592	1
4	4913	0.118743	-0.3359	12
5	35937	0.118675	-0.335669	126
6	274625	0.118664	-0.33604	871
6:(7,1)	352095	0.118651	-0.336202	1285
6:(7,1)	921069	0.118648	-0.336258	4291

Table B.1. The tests for Q1 element.

Refinement	Degrees of freedom	μ	Energy [hartree]	runtime [sec]
1	125	0.131417	-0.399937	1
2	729	0.119248	-0.342022	2
3	4913	0.118656	-0.334944	19
4	35937	0.118654	-0.335899	245
5	274625	0.11866	-0.336226	3064
3:(5,1)	32829	0.118644	-0.336205	244
3:(6,1)	132713	0.118644	-0.33626	1547

Table B.2. The tests for Q2 element.

Refinement	Degrees of freedom	μ	Energy [hartree]	Runtime [sec]
1	343	0.120543	-0.33024	4
2	2197	0.11889	-0.335025	25
3	15625	0.118664	-0.335685	254
4	117649	0.11866	-0.336168	2730
3:(4,1)	28309	0.118657	-0.336147	620
3:(5,1)	105211	0.118659	-0.336262	2520
3:(6,1)	433429	0.118659	-0.336279	14843

Table B.3. The tests for Q3 element.

The idea was to increase the uniform mesh until the solution converged, but the computer I used could only handled about 1 million degrees of freedom so the improved mesh had to be used for the last entries.

B.2 Cusp Condition

The convergence test for He is presented in Table B.4.

Refinement	Box size	μ	$-E/Z^{7/3}$	$n(0)/Z^3$	runtime [sec]
3:(5,1)	10x10x10	0.024135	0.4378	0.2272	185
3:(6,1)	10x10x10	0.0241297	0.4390	0.2264	724
3:(7,1)	10x10x10	0.024131	0.4395	0.2262	3888
4:(7,1)	12x12x12	0.0140706	0.4394	0.2262	3104
4:(8,1)	12x12x12	0.0140705	0.4394	0.2262	19,840

Table B.4. The convergence test for He for Thomas-Fermi with cusp condition.

It is clear that a tight mesh is required to get a good convergence. The choice of 3:(7,1) seems like a good compromise between size, mesh and time.

In Table B.5, one can see the results for a $10x10x10$ simulation box with refinement of 3:(8,1).

Element (N)	μ	$-E/Z^{7/3}$	Ref. [17]	$n(0)/Z^3$	Ref. [17]
Ne (10)	0.039	0.5756	0.5763	0.3666	0.3672
Ar (18)	0.046	0.6095	0.6110	0.4017	0.4020
Kr (36)	0.054	0.6486	0.6439	0.4325	0.4325
Xe (54)	0.059	0.6753	0.6599	0.4455	0.4460
Rn (86)	0.065	0.6765	0.6756	0.4552	0.4581

Table B.5. The results for the noble gases using Thomas-Fermi with cusp condition. The N in element is the number of electrons.

In Table B.6, a convergence test for Ne is presented.

Refinement	Box size	μ	$-E/Z^{7/3}$	$n(0)/Z^3$	runtime [days]
3:(8,1)	14x14x14	0.013	0.5749	0.3668	1
4:(9,1)	12x12x12	0.022	0.5760	0.3666	17

Table B.6. The convergence test for Ne for Thomas-Fermi with cusp condition.

B.3 H_2 Molecule

Here are the meshes that were used for the fixed simulation box and the changing simulation box. In Figure B.1, one can see the fixed $6x3x3$ simulation box and in Figure B.2 is the changing simulation box.

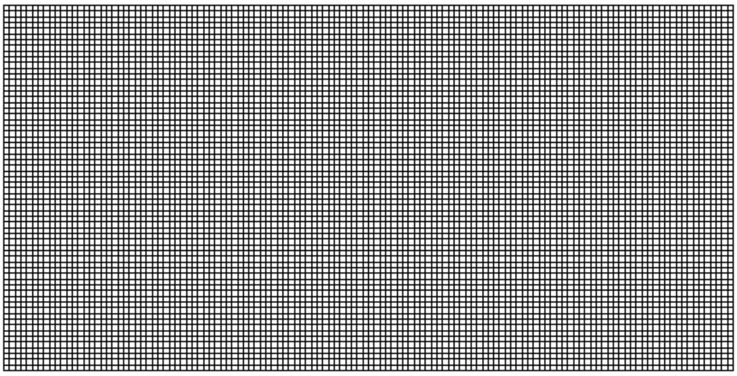


Figure B.1. This is the fixed $6x3x3$ simulation box that was used in the H_2 test.

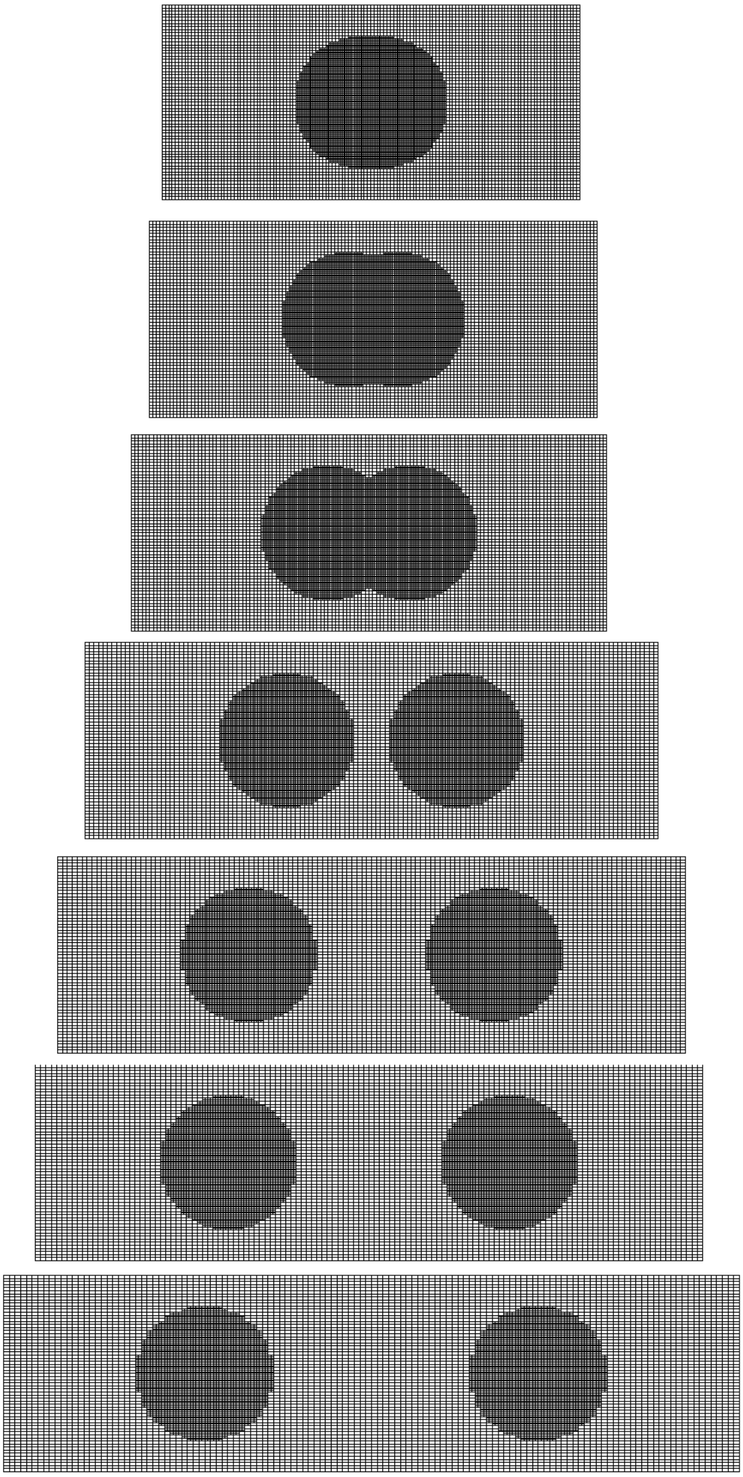


Figure B.2. These are the simulation box and the meshes that was used for the simulation box with changing x-dimension for the H_2 test.

B.4 Soft Coulomb

The convergence test for soft Coulomb potential can be viewed in Table B.7.

Refinement	Softening	μ	Energy [hartree]
3:(5,1)	1.0	0.266104	-0.00995078
3:(6,1)	1.0	0.266148	-0.0099448
3:(7,1)	1.0	0.266172	-0.0099627
3:(7,1)	0.1	0.136899	-0.297737
3:(8,1)	0.1	0.136937	-0.297415
3:(7,1)	0.01	0.111463	-0.744589
3:(8,1)	0.01	0.111716	-0.56772
3:(9,1)	0.01	0.111745	-0.549058

Table B.7. The convergence test for hydrogen for Thomas-Fermi with soft Coulomb.

Bibliography

[1] Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. *Physical review*, 136(3B):B864, 1964.

[2] Kieron Burke et al. The abc of dft. *Department of Chemistry, University of California*, 2007.

[3] Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140(4A):A1133, 1965.

[4] Max Born and Robert Oppenheimer. Zur quantentheorie der molekeln. *Annalen der Physik*, 389(20):457–484, 1927.

[5] Reiner M Dreizler and Eberhard KU Gross. Density functional theory: An approach to the quantum many-body problem springer, 1990.

[6] Valentin V Karasiev, Debajit Chakraborty, and SB Trickey. Progress on new approaches to old ideas: Orbital-free density functionals. In *Many-Electron Approaches in Physics, Chemistry and Mathematics*, pages 113–134. Springer, 2014.

[7] Yan Alexander Wang, Niranjan Govind, and Emily A Carter. Orbital-free kinetic-energy density functionals with a density-dependent kernel. *Physical Review B*, 60(24):16350, 1999.

[8] Donghyung Lee, Lucian A Constantin, John P Perdew, and Kieron Burke. Condition on the kohn–sham kinetic energy and modern parametrization of the thomas–fermi density. *The Journal of chemical physics*, 130(3):034107, 2009.

[9] Robert G Parr and Weitao Yang. *Density-functional theory of atoms and molecules*, volume 16. Oxford university press, 1989.

[10] Llewellyn H Thomas. The calculation of atomic fields. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 23, pages 542–548. Cambridge Univ Press, 1927.

[11] E Fermi. Z. physik 36, 902, 1926; dirac. In *Proc. Roy. Soc. A*, volume 112, page 661, 1926.

[12] E Fermi. Z. physik, 48, 73, 1928. *Nature*, 121:502, 1928.

[13] CF Weizsäcker. Z. physik 96, 431 (1935); ha bethe and rf bacher. *Rev. Mod. Phys*, 8:82, 1936.

[14] W Stich, EKU Gross, P Malzacher, and RM Dreizler. Accurate solution of the thomas-fermi-dirac-weizsäcker variational equations for the case of neutral atoms and positive ions. *Zeitschrift für Physik A Atoms and Nuclei*, 309(1):5–11, 1982.

[15] E Fermi and G Amaldi. *Men R Acad Italia*, 6:117, 1934.

[16] PAM Dirac. Cambridge philos. soc. 26, 376 1930; sj vosko, l. wilk, and m. nusair. *Can. J. Phys*, 58:1200, 1980.

[17] Robert G Parr and Swapan K Ghosh. Thomas-fermi theory for atomic systems. *Proceedings of the National Academy of Sciences*, 83(11):3577–3579, 1986.

[18] Abdulnour Y Toukmaji and John A Board. Ewald summation techniques in perspective: a survey. *Computer physics communications*, 95(2):73–92, 1996.

[19] Felix Faber, Alexander Lindmaa, O Anatole von Lilienfeld, and Rickard Armiento. Crystal structure representations for machine learning models of formation energies. *International Journal of Quantum Chemistry*, 2015.

[20] Pymatgen.org. pymatgen.analysis.ewald pymatgen 3.1.6 documentation, 2015.

[21] W. Bangerth, T. Heister, L. Heltai, G. Kanschat, M. Kronbichler, M. Maier, B. Turcksin, and T. D. Young. The `deal.II` library, version 8.2. *Archive of Numerical Software*, 3, 2015.

[22] W. Bangerth, R. Hartmann, and G. Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.

[23] the authors. The deal.ii finite element library, 2015.

[24] Fenicsproject.org. Fundamentals fenics project, 2015.

[25] Jan Lundgren, Mikael Rönnqvist, and Peter Värbrand. *Optimeringslära*, volume 3:3. Studentlitteratur, 2003.

[26] Cms.mpi.univie.ac.at. 5.7 poscar file, 2015.

[27] Carsten Rostgaard, Jens Jørgen Mortensen, and Karsten Wedel Jacobsen. Exact exchange in density functional calculations. *Master's thesis, Technical University of Denmark*, 2006.

[28] Min Sung Kim, Sung-Kie Youn, and Jeung Ku Kang. Grid-based thomas-fermi-amaldi equation with the molecular cusp condition. *The Journal of chemical physics*, 124(12):124107, 2006.

[29] JR Townsend and GS Handler. Thomas—fermi homonuclear diatomic molecule. i. method of solution and atomic interaction potential. *The Journal of Chemical Physics*, 36(12):3325–3329, 1962.

[30] Tony D Young and Rickard Armiento. Strategies for h-adaptive refinement for a finite element treatment of harmonic oscillator schrödinger eigenproblem. *Communications in Theoretical Physics*, 53(6):1017, 2010.