

Investigations of Surface Potentials



SUBMITTED
BY

Senthil SharadKumar Pandian

DIVISION OF PHYSICS & APPLIED PHYSICS
SCHOOL OF PHYSICAL AND MATHEMATICAL SCIENCES

A final year project report
presented to
Nanyang Technological University
in partial fulfilment of the
requirements for the
Bachelor of Science (Hons) in Physics / Applied Physics
Nanyang Technological University

June 2015

Abstract

The potential energy at the surface of materials, like graphene, might be able to trap atoms and molecules. In order to probe this possibility, the theoretical foundations of Density Functional Theory are studied, and the need for numerical calculations is understood. Iterative methods to solve numerical calculations are presented. Quantum ESPRESSO, a Linux package used for numerical calculations is examined, and its important features are explained. Simple simulations involving Aluminium, and Silicon are carried out. The accuracy of quantum ESPRESSO is verified using the Hydrogen molecule, for which we have the analytical solution. Four different attempts at modeling graphene are then presented. The system of a single hydrogen molecule and a graphene sheet, is then studied. Finally, the notion of accuracy in numerical calculations is analyzed in light of the simulations that have been run.

Keywords: Density Functional Theory, Quantum ESPRESSO, iterative methods

Acknowledgements

I would like to express my deepest gratitude to my research supervisors, Professor Tomasz Paterek and Professor Rainer Dumke for their constant support and guidance. In all the classes I have taken under them, as well as the three years I have worked with them on research projects, they have been exceedingly patient and encouraging. It cannot be emphasized enough how crucial they were for this research to have seen the light of day.

I would also like to thank Tran Cong Minh, Margherita Zuppardo, as well as others from SPMS who provided feedback on this report, especially about how to make it more accessible to those who do not work in computational physics.

Finally, I would like to thank my friends and family for keeping me going, even when I wasn't sure I could.

Contents

1	Introduction	4
2	Density Functional Theory	6
2.1	The Schrödinger Equation	6
2.2	Electron Density	7
2.3	The Hohenberg-Kohn Theorems	7
2.4	The Kohn-Sham Approach and Equations	9
3	Numerical Methods	12
3.1	Iterative Method	12
3.2	Bravais lattice and Reciprocal lattice	12
3.3	Choosing k points in the BZ	14
3.4	Kinetic Energy Cut-off	14
3.5	Pseudopotentials	15
4	Quantum ESPRESSO	17
4.1	Introduction to Quantum ESPRESSO	17
4.2	External Tools	17
4.3	Navigating Quantum Espresso	18
4.4	Output in QE	21
5	Running Simulations	22
5.1	Some Simple cells	22
5.2	The Hydrogen Molecule	23
5.3	Modeling Graphene	24
5.4	Modeling Hydrogen and Graphene	26
6	Accuracy and Errors in numerical calculations	29
7	Conclusion	31
	References	32

1 Introduction

The motivation behind this research is to probe the possibility of being able to trap atoms on the surface of materials like graphene, by studying the potential at their surfaces. In order to do this, computational methods will have to be employed to get an idea of what the potentials look like. In particular, Density Functional Theory (DFT), a computational quantum mechanical modeling method will be explored and examined.

DFT is a highly successful approach to deal with the quantum mechanical behavior of atoms and molecules using the Schrödinger equation. From the late eighties and early nineties, DFT has enjoyed enormous popularity among computational physicists and chemists around the world [1]. As the figure below indicates, there are thousands of research papers that use DFT today. The importance of DFT cannot be over-emphasized.

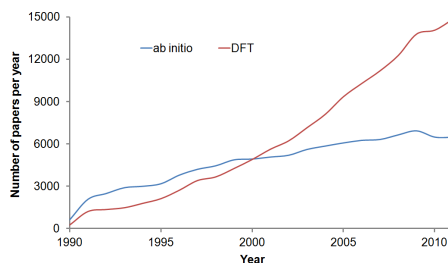


Figure: Number of papers with DFT related keywords.
Source: ISI Web of Knowledge

This thesis has six sections after this introduction. Section 2, will lay out the basic theoretical foundations of Density Functional Theory. It should be noted that this is not meant as a comprehensive introduction, but as a brief look at some of the essential ideas. It turns out that DFT requires numerical calculations, which will be performed using iterative techniques. The general schematic for this iterative technique, as well as some related concepts will be explored in section 3.

Section 4 is a quick introduction to Quantum ESPRESSO (QE), a Linux based package that helps perform the aforementioned calculations. Screenshots will be provided to make the section as clear as possible. Section 5 runs various simulations on QE, as well as checks some of the results obtained against empirical data already available.

Differences between the data and the empirical results prompts a brief inquiry into the origin of these differences, as well as their extent. This is done in section 6. Section 7 is a summary and conclusion.

Before we begin, there are three points to take note of. The first is atomic units are used when talking about DFT [2][3], and so that convention is followed here. The table below gives more details about this system.

Quantity	Atomic unit	Value in SI units	Symbol (name)
mass	rest mass of electron	9.1094×10^{-31} kg	m_e
charge	elementary charge	1.6022×10^{-19} C	e
action	Planck's constant/ 2π	1.0546×10^{-34} J s	$\hbar$
length	$4\pi\epsilon_0 \hbar^2 / m_e e^2$	5.2918×10^{-11} m	a_0 (bohr)
energy	$\hbar^2 / m_e a_0^2$	4.3597×10^{-18} J	E_h (hartree)

Figure: Atomic Units

Second, although I will discuss the programs used to implement numerical calculations, there will be no discussion of how these are installed, or concerning their trouble-shooting. There are accessible guides online, and I will point out these websites wherever necessary.

Third, the chief aim of this work is to gain an understanding of DFT as well as its application in solving for properties of physical systems. As I will explain in section 6, I will not be overly concerned with the precise values obtained in the simulations- what is important is that we understand how they work. Since this is still a developing field, imprecise simulations do not pose a major problem. If there are shortcomings with the current programs, all that is called for is more work on the program.

2 Density Functional Theory

2.1 The Schrödinger Equation

It would not be unreasonable to consider the ultimate problem of quantum mechanical projects to be to find a solution (or at least an approximate solution) to the Schrödinger equation. If $\hat{H}$ is the Hamilton operator for a molecular system with M nuclei and N electrons, we can write the Schrödinger equation as:

$$\hat{H}\Psi_i(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N, \vec{R}_1, \vec{R}_2, \dots, \vec{R}_M) = E_i\Psi_i(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N, \vec{R}_1, \vec{R}_2, \dots, \vec{R}_M) \quad (1)$$

where $\hat{H}$ can be written as

$$\hat{H} = -\frac{1}{2} \sum_{i=1}^N \nabla_i^2 - \frac{1}{2} \sum_{A=1}^M \frac{1}{M_A} \nabla_A^2 - \sum_{i=1}^N \sum_{A=1}^M \frac{Z_A}{r_{iA}} + \sum_{i=1}^N \sum_{j>i}^N \frac{1}{r_{ij}} + \sum_{A=1}^M \sum_{B>A}^M \frac{Z_A Z_B}{R_{AB}} \quad (2)$$

In equation (2), the first two terms represent the kinetic energy of electrons and the nuclei respectively, while the other three terms represent the potential energies in the system, namely, between the electrons and the nuclei, between the electrons, and between the nuclei respectively.

Since the nuclei are a lot more massive than the electrons, one way to simplify calculations is to assume that the nuclei have fixed positions, while the electrons are moving in a field of fixed nuclei. This is known as the *Born-Oppenheimer approximation*. The Hamiltonian from equation (2) reduces to:

$$\hat{H}_{elec} = -\frac{1}{2} \sum_{i=1}^N \nabla_i^2 - \sum_{i=1}^N \sum_{A=1}^M \frac{Z_A}{r_{iA}} + \sum_{i=1}^N \sum_{j>i}^N \frac{1}{r_{ij}} = \hat{T} + \hat{V}_{Ne} + \hat{V}_{ee} \quad (3)$$

The solution to Schrödinger's equation with the above Hamiltonian is Ψ_{elec} , the electronic wave function, dependent on the coordinates of the electrons. The positions of the nuclei are parameters, and do not appear explicitly in the wave function. If the electronic energy is represented by E_{elec} , the total energy E_{tot} is the sum of E_{elec} and the constant nuclear contribution, $E_{nuc} = \sum_{A=1}^M \sum_{B>A}^M \frac{Z_A Z_B}{R_{AB}}$.

We can write this relation as:

$$\hat{H}_{elec}\Psi_{elec} = E_{elec}\Psi_{elec} \quad (4)$$

$$E_{tot} = E_{elec} + E_{nuc} \quad (5)$$

The square of the wave function $|\Psi(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N)|^2 d\vec{x}_1 d\vec{x}_2 \dots d\vec{x}_N$ represents the probability that the electrons are found in the various volume elements ($d\vec{x}_1, d\vec{x}_2, \dots$). As indicated in equation 6, integrating over all volume elements, the probability of finding all N electrons anywhere in space must equal one (assuming the wave function is normalized).

$$\int \dots \int |\Psi(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N)|^2 d\vec{x}_1 d\vec{x}_2 \dots d\vec{x}_N = 1 \quad (6)$$

2.2 Electron Density

We can approximate $\Psi(\vec{x}_1, \dots, \vec{x}_N)$ as a product of individual electron wave functions given by $\Psi = \Psi_1(\vec{x})\Psi_2(\vec{x}), \dots, \Psi_N(\vec{x})$. If there are N electrons, the full wave function solution to the Schrödinger equation will have $3N$ dimensions (considering only the spacial dimensions. Including spin, there are $4N$ dimensions).

To illustrate why this might be a problem, consider a system of 100 Pt atoms (where Pt has an atomic number of 78). The total number of dimensions to the solution to Schrödinger's equation is more than 23,000. Hence we begin to see that solving the equation poses a significant practical problem. To make matters worse, the individual electronic wave function Ψ_i cannot be found without considering all other electronic wave functions. This makes this a many body problem.

These sorts of issues lead us to become interested in a quantity other than wave function, namely electron density given by

$$\rho(\vec{r}_1) = N \int \int |\Psi(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N)|^2 ds_1 d\vec{x}_2 \dots d\vec{x}_N \quad (7)$$

where $\rho(\vec{r}_1)$ gives us the probability of finding one of the N electrons within the volume element $d\vec{r}_1$ with arbitrary spin (hence the ds_1 in equation (7)), while the other $(N-1)$ electrons have arbitrary positions in the Ψ state. The probability of finding an electron tapers off and falls to zero as we move to infinity, which can be written as

$$\rho(\vec{r} \rightarrow \infty) = 0 \quad (8)$$

Since $\rho(\vec{r}_1)$ gives the probability of finding one electron, integrating over all $\rho(\vec{r}_i)$ gives us the total number of electrons.

$$\int \rho(\vec{r}) d\vec{r}_1 = N \quad (9)$$

2.3 The Hohenberg-Kohn Theorems

It is important to note that the electron density depends only on the three spatial variables, instead of $4N$ variables like the wave function. This helps us appreciate the appeal of Density Functional Theory- if we could somehow find a way to replace the use of the wave function with the electron density, then our calculations would get considerably simpler. This is precisely what happened in a seminal 1964 paper by Hohenberg and Kohn, where they presented two theorems which became the foundations on which DFT is built. I will now state

the theorems as they appear in the original paper [7]. (Note that the V_{ext} in the theorems refer to V_{Ne} from equation 3, and represents the potential energy due to the interaction between electrons and the fixed nuclei)

The First Hohenberg-Kohn Theorem: The external potential $V_{ext}(\vec{r})$ is (to within a constant) a unique functional of $\rho(\vec{r})$; since, in turn $V_{ext}(\vec{r})$ fixes $\hat{H}$ we see that the full many particle ground state is a unique functional of $\rho(\vec{r})$.

Proof: Hohenberg and Kohn provided a fairly simple proof for this theorem, using reductio ad absurdum. We start off by assuming that there exist two external potentials V_{ext} and V'_{ext} which give rise to the same electron density $\rho(\vec{r})$. The corresponding Hamiltonians are given by $\hat{H} = \hat{T} + \hat{V}_{ee} + \hat{V}_{ext}$ and $\hat{H}' = \hat{T} + \hat{V}_{ee} + \hat{V}'_{ext}$, and we see that they only differ in their external potentials. Since we have non-identical Hamiltonians, we will have different ground state wave functions (call them Ψ and Ψ' respectively), as well as different ground state energies E_0 and E'_0 . Finally, let both wave functions lead to the same electron density $\rho(\vec{r})$. By the variational principle, we have

$$E_0 < \langle \Psi' | \hat{H} | \Psi' \rangle = \langle \Psi' | \hat{H}' | \Psi' \rangle + \langle \Psi' | \hat{H} - \hat{H}' | \Psi' \rangle \quad (10)$$

Writing the Hamiltonians in terms of their individual terms, we get

$$E_0 < E'_0 + \langle \Psi' | \hat{T} + \hat{V}_{ee} + \hat{V}_{ext} - \hat{T} - \hat{V}_{ee} - \hat{V}'_{ext} | \Psi' \rangle \quad (11)$$

which gives

$$E_0 < E'_0 + \int \rho(\vec{r}) \{ \hat{V}_{ext} - \hat{V}'_{ext} \} d\vec{r} \quad (12)$$

In a similar fashion, we interchange the primed and unprimed quantities from equations (10) to (12) to get

$$E'_0 < E_0 - \int \rho(\vec{r}) \{ \hat{V}_{ext} - \hat{V}'_{ext} \} d\vec{r} \quad (13)$$

Adding equations (12) and (13), we get

$$E_0 + E'_0 < E'_0 + E_0 \text{ or } 0 < 0 \quad (14)$$

which is absurd. Hence, by reductio, we cannot have two different external potentials (which in turn fix the electron densities) that give the same ground state energy. In other words, the electron density uniquely determines the ground state energy. We can write this schematically as

$$\rho_0 \Rightarrow \{N, Z_A, R_A\} \Rightarrow \hat{H} \Rightarrow \Psi_0 \Rightarrow E_0 \text{ (and other properties)} \quad (15)$$

Since the total energy is fixed by the electron density, so should its individual components

$$E_0[\rho_0] = T[\rho_0] + E_{ee}[\rho_0] + E_{Ne}[\rho_0] \quad (16)$$

We can now split this into two parts- those depend on the system, and those that are universal (in the sense that they are independent of N , R_A , and Z_A).

$$\int \underbrace{\rho_0(\vec{r})V_{Ne}d\vec{r}}_{\text{system dependent}} + \underbrace{T[\rho_0] + E_{ee}[\rho_0]}_{\text{universally valid}} \quad (17)$$

We collect the system-independent portion into a functional $F_{HK}[\rho_0]$, called the Hohenberg-Kohn functional. Rewriting equation (17), we get

$$E_0[\rho_0] = \int \rho_0(\vec{r})V_{Ne}d\vec{r} + F_{HK}[\rho_0] \quad (18)$$

where

$$F_{HK}[\rho] = T[\rho] + E_{ee}[\rho] = \langle \Psi | \hat{T} + \hat{V}_{ee} | \Psi \rangle \quad (19)$$

At this point, we notice that although the first Hohenberg-Kohn theorem showed that a functional of electron density can be used to obtain the ground state energy, we still have no information about the functional. This is where the second Hohenberg-Kohn theorem comes in. [1]

The Second Hohenberg-Kohn Theorem: $F_{HK}[\rho]$, the functional that delivers the ground state energy of the system, delivers the lowest energy if and only if the input density is the true ground state density, ρ_0

Writing this using the variational principle

$$E_0 \leq E[\tilde{\rho}] = T[\tilde{\rho}] + E_{Ne}[\tilde{\rho}] + E_{ee}[\tilde{\rho}] \quad (20)$$

where $\tilde{\rho}$ represents a trial electron density, which gives an energy E , which is greater than ground state energy E_0

$$\langle \tilde{\Psi} | \hat{H} | \tilde{\Psi} \rangle = T[\tilde{\rho}] + V_{ee}[\tilde{\rho}] + \int \tilde{\rho}(\vec{r})V_{ext}d\vec{r} = E[\tilde{\rho}] \geq E_0[\rho_0] = \langle \Psi_0 | \hat{H} | \Psi_0 \rangle \quad (21)$$

An alternative way to represent the second theorem is

$$E_0 = \min_{\Psi \rightarrow N} \langle \Psi | \hat{T} + \hat{V}_{Ne} + \hat{V}_{ee} | \Psi \rangle \quad (22)$$

In equation (22), the $\Psi \rightarrow N$ represents the fact that we search over all allowed, antisymmetric N -electron wave functions, and the one that gives the lowest expectation value for energy (here the Hamiltonian operator) is the ground state wave function.

2.4 The Kohn-Sham Approach and Equations

In 1965, a year after the Hohenberg-Kohn paper, Kohn and Sham suggested an approach to get the universally valid contribution in equation (17). At the risk of over-simplification, their idea was to calculate the kinetic energy of our

interacting system by looking at a similar, but non-interacting reference system. The reference system is built using certain one-electron function orbitals, called *Kohn-Sham orbitals* or KS orbitals, represented as ϕ_i . We can link this reference system with our actual interacting one by choosing an appropriate potential, such that

$$\sum_i^N \sum_s |\phi_i(\vec{r}, s)|^2 = \rho_0(\vec{r}) \quad (23)$$

where ρ_0 is the density of our interacting system. The kinetic energy of the non-interacting system, T_S , is given by

$$T_S = -\frac{1}{2} \sum_i^N \langle \phi_i | \nabla^2 | \phi_i \rangle \quad (24)$$

Even though the two systems share the same electron density, the kinetic energy of the non-interacting system T_S is not identical to the kinetic energy of the interacting system T (it can be shown that $T_S \leq T$, see [1]). Taking this into consideration, we introduce a functional $F[\rho]$ which contains the individual contributions to kinetic energy, coulombic interaction, and a non-classical portion due to "self-interaction correction, exchange (i.e., antisymmetry), and electron correlation effects" [1]

$$F[\rho(\vec{r})] = T_S[\rho(\vec{r})] + J[\rho(\vec{r})] + E_{XC}[\rho(\vec{r})] \quad (25)$$

where E_{XC} is called the exchange-correlation energy and is given by

$$E_{XC}[\rho] \equiv (T[\rho] - T_S[\rho]) + (E_{ee}[\rho] - J[\rho]) = T_C[\rho] + E_{ncl}[\rho] \quad (26)$$

Here E_{ncl} refers to the non-classical contribution. The difference between the kinetic energy of the interacting and non-interacting system is just added to E_{ncl} . Writing down the expression for total energy of the interacting system, in terms of the functionals in equation (25),

$$E[\rho(\vec{r})] = T_S[\rho] + J[\rho] + E_{XC}[\rho] + E_{Ne}[\rho] \quad (27)$$

Using equation (23) and equation (24), we can rewrite equation (27) as

$$\begin{aligned} E[\rho(\vec{r})] &= T_S[\rho] + \frac{1}{2} \iint \frac{\rho(\vec{r}_1)\rho(\vec{r}_2)}{r_{12}} d\vec{r}_1 d\vec{r}_2 + E_{XC}[\rho] + \int V_{Ne}\rho(\vec{r})d\vec{r} \\ &= -\frac{1}{2} \sum_i^N \langle \phi_i | \nabla^2 | \phi_i \rangle + \frac{1}{2} \sum_i^N \sum_j^N \iint |\phi_i(\vec{r}_1)|^2 \frac{1}{r_{12}} |\phi_j(\vec{r}_2)|^2 d\vec{r}_1 d\vec{r}_2 \\ &\quad + E_{XC}[\rho(\vec{r})] - \sum_i^N \int \sum_A^M \frac{Z_A}{r_{1A}} |\phi_i(\vec{r}_1)|^2 d\vec{r}_1 \end{aligned} \quad (28)$$

In the above equation, we lack an explicit form only for the exchange-correlation term. We now check what sort of conditions ϕ_i must satisfy to obtain the minimum energy. From [9], we obtain the following equation.

$$\begin{aligned}
& \left(-\frac{1}{2}\nabla^2 + \left[\int \frac{\rho(\vec{r}_2)}{r_{12}} d\vec{r}_2 + V_{XC}(\vec{r}_1) - \sum_A^M \frac{Z_A}{r_{1A}} \right] \right) \phi_i \\
& = \left(-\frac{1}{2}\nabla^2 + V_{eff}(\vec{r}_1) \right) \phi_i = \epsilon_i \phi_i
\end{aligned} \tag{29}$$

We notice that the terms in the square bracket are the effective potential V_{eff} , given by

$$V_{eff}(\vec{r}) = \int \frac{\rho(\vec{r}_2)}{r_{12}} d\vec{r}_2 + V_{XC}(\vec{r}_1) - \sum_A^M \frac{Z_A}{r_{1A}} \tag{30}$$

We now have an idea of V_{eff} , which determines the orbitals and hence the ground state energy using equation (28).

3 Numerical Methods

3.1 Iterative Method

Going back to equation (28) and equation (30), we observe a major problem in solving it. In order to calculate the electron density in equation (28), we need to know the external potential V_{eff} . However, we see that the electron density is required to calculate the V_{eff} in equation (30). Hence, we need to solve these iteratively, and obtain a solution which is self-consistent with regard to both equations. A schematic representation of this is shown in figure 1.

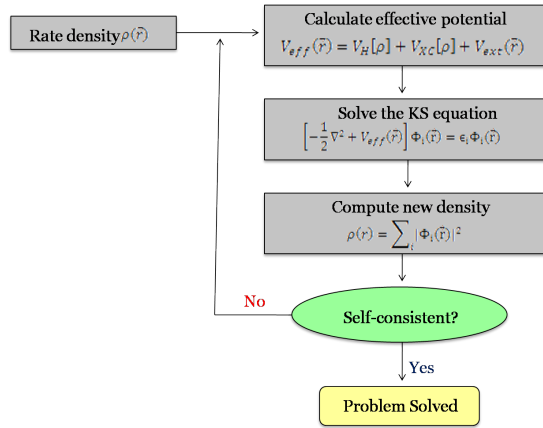


Figure 1: Schematic representation of the iterative method

We start off with a trial electron density and calculate the corresponding V_{eff} , and then use it to solve the Kohn-Sham equations. We then calculate a new electron density and check if it falls within a previously-defined tolerance range. If it doesn't, we run a new iteration with a mixture of the old and new densities. We can control the amount of the new density that blends with the old at each iteration, which helps determine the speed of convergence.

3.2 Bravais lattice and Reciprocal lattice

In DFT we are often interested in calculations involving periodic arrangements of atoms or groups of atoms. We begin by defining the unit cell, the smallest unit that can be reproduced in all directions to recreate the crystal structure. We then define the lattice vectors, which span the whole crystal, and represented in 3 dimensional space as $\vec{a}_1, \vec{a}_2, \vec{a}_3$. Any position vector $\vec{R}$, can be written in term of a sum of multiples of the basis vectors.

$$\vec{R} = n_1 \vec{a}_1 + n_2 \vec{a}_2 + n_3 \vec{a}_3 \quad (31)$$

The solution to Schrödinger's equation for this periodic arrangement must satisfy *Bloch's Theorem*, according to which it should be expressible as a sum

of terms of the form

$$\phi_k(\vec{r}) = \exp(i\vec{k} \cdot \vec{r}) u_k(\vec{r}) \quad (32)$$

where $u_k(\vec{r})$ has the same period as the cell. It was found that it is easier to solve many of the mathematical problems in DFT in terms of $\vec{k}$, the momentum vector, than in terms of $\vec{r}$. The space of $\vec{r}$ vectors is known as real space, while the space of $\vec{k}$ vectors is known as reciprocal space, or k space. Analogous to the lattice vectors $\vec{a}_1, \vec{a}_2, \vec{a}_3$, we define reciprocal vectors $\vec{b}_1, \vec{b}_2, \vec{b}_3$, which fulfill the following conditions

$$\vec{b}_1 = 2\pi \frac{\vec{a}_2 \times \vec{a}_3}{\vec{a}_1 \cdot (\vec{a}_2 \times \vec{a}_3)}, \quad \vec{b}_2 = 2\pi \frac{\vec{a}_3 \times \vec{a}_1}{\vec{a}_2 \cdot (\vec{a}_3 \times \vec{a}_1)}, \quad \vec{b}_3 = 2\pi \frac{\vec{a}_1 \times \vec{a}_2}{\vec{a}_3 \cdot (\vec{a}_1 \times \vec{a}_2)} \quad (33)$$

To illustrate these concepts in a way that's intuitive, let us look at the fcc primitive cell. It is easily shown that the lattice vectors and reciprocal vectors are given by

$$\vec{a}_1 = a(\frac{1}{2}, \frac{1}{2}, 0), \quad \vec{a}_2 = a(0, \frac{1}{2}, \frac{1}{2}), \quad \vec{a}_3 = a(\frac{1}{2}, 0, \frac{1}{2}) \quad (34)$$

$$\vec{b}_1 = \frac{2\pi}{a}(1, 1, -1), \quad \vec{b}_2 = \frac{2\pi}{a}(-1, 1, 1), \quad \vec{b}_3 = \frac{2\pi}{a}(1, -1, 1) \quad (35)$$

Figure 2 shows the vectors for the fcc cell in real and reciprocal space respectively.

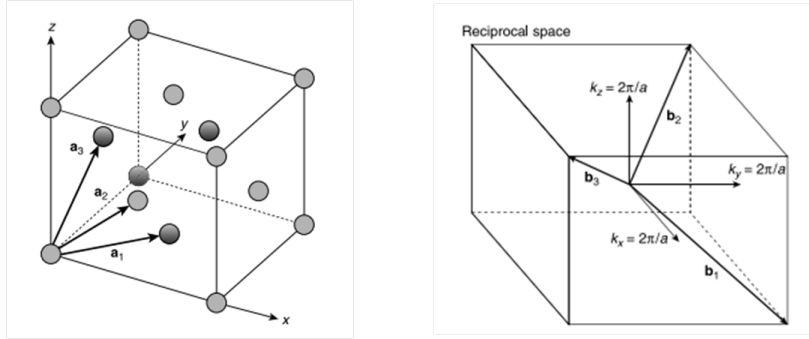


Figure 2: Real space and reciprocal space lattice vectors for the fcc cell

Like the primitive cell in real space, we define a primitive cell in reciprocal space called the Brillouin Zone (BZ). The volume of BZ is given by

$$V_{BZ} = \frac{(2\pi)^3}{V_{cell}} \quad (36)$$

where V_{cell} is the volume of the primitive cell in real space. The BZ is relevant for us as the integral for the evaluation of the electron density from equation (28) reduces to equations of the form

$$\bar{g} = \frac{V_{cell}}{(2\pi)^3} \int_{BZ} g(\mathbf{k}) d\mathbf{k} \quad (37)$$

It is important to note that this integrates over reciprocal space, in particular, integrates over the $\mathbf{k}$ values inside the BZ. I won't be going into the details of how we can get this integral, but instead will focus on how to solve it numerically.

3.3 Choosing k points in the BZ

In 1976, Monkhorst and Pack came up with a method to solve equations like equation (37). According to this, we need to specify how many k points to use in each direction in reciprocal space. If N number of points are used in all three directions, we label the calculations as using $N \times N \times N$ k points. A calculation using $M \times M \times M$ k points will give us a better result after integration than one using $N \times N \times N$ k points, if $M > N$ (note that although it is natural to use the same number of k points along each direction for a cell with equal lengths for each lattice vector, this is not necessary).

One important feature of how the Monkhorst-Pack method works is its utilization of symmetries that exist in the system. As a result of these symmetries, the calculations do not need to be carried for the entire BZ, but only a portion of it. This smaller region in reciprocal space is called the irreducible Brillouin Zone (IBZ). This helps explain why simulations with symmetry converge much faster than those without any symmetry. For tables that vividly illustrate the time difference produces, look at figure [37] in the appendix.

3.4 Kinetic Energy Cut-off

In section 3.2, Bloch's theorem was mentioned, according to which solutions to Schrödinger's equation are of the form

$$\Phi_k(\vec{r}) = \exp(i\vec{k} \cdot \vec{r}) u_k(\vec{r}) \quad (38)$$

where the $u_k(\vec{r})$ has the same period as the cell. Because of this periodicity, we can expand $u_k(\vec{r})$ in terms of plane waves in the following fashion

$$u_k(\vec{r}) = \sum_{\vec{G}} c_{\vec{G}} \exp[i\vec{G} \cdot \vec{r}] \quad (39)$$

where we sum over all vectors $\vec{G}$ defined by $\vec{G} = m_1 \vec{b}_1 + m_2 \vec{b}_2 + m_3 \vec{b}_3$ where m_i is any integer value. Here the set of vectors given by $\vec{G}$ are defined such that $\vec{G} \cdot \vec{a}_i = 2\pi m_i$ where $\vec{a}_i$ can represent any lattice vector in real space.

Combining equation (38) and equation (39), we get

$$\Phi_k(\vec{r}) = \sum_{\vec{G}} c_{\vec{k}+\vec{G}} \exp[i(\vec{k} + \vec{G}) \cdot \vec{r}] \quad (40)$$

Note that since we have infinite possible set of values for $\vec{G}$, evaluating Φ_k seems practically impossible. To tackle this, we recognize that the functions in equation (40) have a physical interpretation we make use of- they have kinetic energies equal to

$$E = \frac{\hbar^2}{2m} |\vec{k} + \vec{G}|^2 \quad (41)$$

Since our goal is to calculate the ground state energy, it seems reasonable to suspect that lower energies will be more crucial to the calculation than higher energies. With this in mind, we can truncate the infinite series to include values only until a certain cut-off value

$$E_{cut} = \frac{\hbar^2}{2m} G_{cut}^2 \quad (42)$$

Equation (40) now reduces to

$$\Phi_k(\vec{r}) = \sum_{|\mathbf{G}+\mathbf{k}| < G_{cut}} c_{\mathbf{k}+\mathbf{G}} \exp[i(\vec{k} + \vec{G})\vec{r}] \quad (43)$$

From the limit in the sum, we see that the number of terms to be summed differs according to $\vec{k}$. We now have a parameter that needs to be defined when we carry out the DFT calculation: the cut-off energy E_{cut} .

3.5 Pseudopotentials

The wave functions of electrons near the core are not very important to chemical bonds, nor the physical properties associated with them. Taking this into consideration, one way to speed up calculations is to substitute the real electron density with a smoothed density selected to match important mathematical and physical properties of the real core. This is known as the *frozen-core approximation*.

We come up with appropriate pseudopotentials, ideally, by observing a single isolated atom of whichever element we're interested in. The resulting pseudopotentials can be reliably used in all kinds of calculations without any adjustment. This is known as the transferability of the pseudopotential.

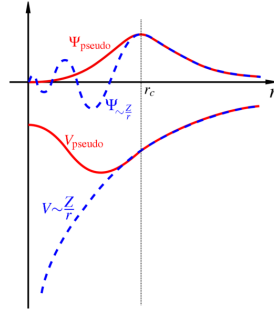


Figure 3: Pseudopotentials, Source: Wikipedia Commons

It has been found that using pseudopotentials give us good results in agreement with calculations involving all the electrons. As we can see in figure 3, after a certain cut-off radius, the pseudopotential and actual potential are almost identical. Pseudopotentials are used widely in this research, as will be pointed out in the next section.

4 Quantum ESPRESSO

4.1 Introduction to Quantum ESPRESSO

Quantum ESPRESSO (QE) is "an integrated suite of computer codes for electronic-structure calculations and materials modeling, based on density-functional theory, plane waves, and pseudopotentials" [13]. The acronym ESPRESSO stands for "opEn Source Package for Research in Electronic Structure, Simulation and Optimization". QE uses a number of methods to enable the probing of various nano-structures. Although both amorphous and crystalline structures maybe studied, this research will be restricted to only crystalline solids.

QE was developed as an open-source project where researchers working on electronic structure calculations could come together, and develop the code together. Figure 4 is a screenshot of the QE website, from where the package can be downloaded for free.



Figure 4: Screenshot from <http://www.quantum-espresso.org/>

The core package for QE is PWscf (PW), used for Plane-Wave Self-Consistent Field calculations using DFT, pseudopotentials and plane waves, all of which have been introduced (briefly) in the previous sections.

In this section, some features of QE will be briefly described. Since there are plenty of sources of information about how to install and use QE, this section will focus instead on how some of the features mentioned in the previous section are implemented in the QE package.

4.2 External Tools

There are useful tools in QE that make working with it significantly more convenient. Two such tools used in this research are discussed here.

1. PWgui

PWgui is a graphical user interface, developed by Anton Kokalj, which is helpful for entering input for the program. Each input has a corresponding box (or button) which makes entering it very convenient. Extensive information is given

under conveniently located "help" buttons, which explain the significance of each corresponding input.

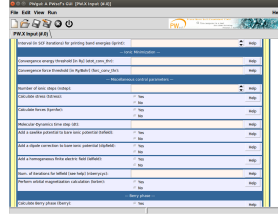


Figure 5: PWgui screenshot

Figure 5 is a screenshot of a part of the PWgui interface, which shows the various input boxes, as well as the corresponding help boxes.

2. XCrySDen

XCrySDen is another tool developed by Anton Kokalj, which stands for Crystalline Structures and Densities. It is used to visualize the atomic structure of periodic crystals. This program reads the number of atoms, their positions, and their properties, and presents them in a 3-D model. Figure 6 is a sample model to give an idea of how this program works.

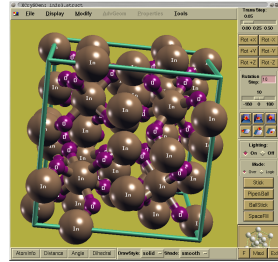


Figure 6: XCrySDen screenshot

Source: http://www.xcrysden.org/img/xcrysden_in2o3.png

As is seen in figure 6, we can translate, rotate and zoom from all directions. Additionally, there are other options which can aid in understanding the system. For example, the covalent bonds can be observed, the unit cell can be repeated in different directions, the Wigner-Seitz cell can be shown, etc. In the next section, each model will have its corresponding crystal structure displayed using XCrySDen.

4.3 Navigating Quantum Espresso

In this section, the QE package will be explored with an emphasis on how data needs to be inputted. In the next section, some models will be simulated, and

the results will be analyzed.

Figure 7 shows the first PWgui screen that one encounters when they start QE. Multiple tabs are seen at the top, which we can use to navigate. The control tab is used to set the input and output files, and well as to choose what sort of calculation we want (in this project, we only run calculation for self-consistency).

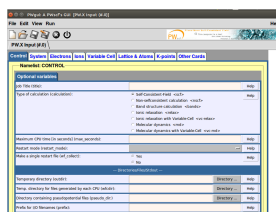


Figure 7: QE Screenshot- The Control Tab

Figure 8 shows the System Tab. This is used to input information about the structure of the system under investigation. The `ibrav` option is used to indicate the symmetry of the crystal, and hence the kind of crystal. The figure on the left is a screenshot of the help button for `ibrav`, and gives us an idea of the range of options available. (For a complete list, go to http://www.quantum-espresso.org/wp-content/uploads/Doc/INPUT_PW.html). In QE, all lengths are inputted as a ratio of the cell dimension (keyed in at the Systems tab).



Figure 8: QE Screenshot- The System Tab

At the bottom of the Systems tab (as seen in figure 9), we encounter a smaller box where we input the number and type of atoms in the cell. Notice also that we input the kinetic energy cut-off, which was talked about in section 3.4.

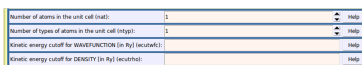


Figure 9: QE Screenshot- The System Tab continued

We see the Lattice and Atoms tab in figure 10. We can explicitly input the lattice vectors here, although there are default vectors provided for some values

of ibrav (as mentioned in the help box in figure 8).

We also need to key in which atomic species we are considering, its mass, a pseudopotential file for each species (where the electron density for core electrons is found, as discussed in section 3.5), as well as the positions for each atom in the cell.

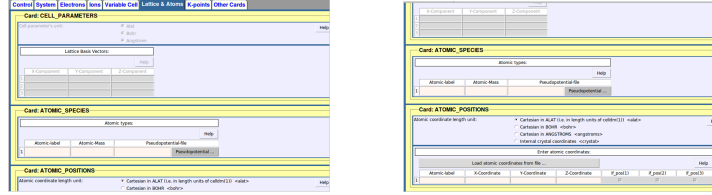


Figure 10: QE Screenshot- The Lattice and Atoms Tab

We can find files containing various pseudopotentials for different elemental species at the QE website (<http://www.quantum-espresso.org/pseudopotentials/>) as shown in figure 11.

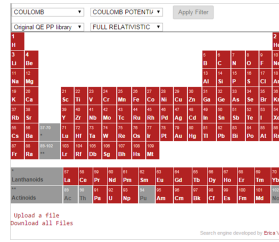


Figure 11: Pseudopotential Library

The K-points tab, as shown in figure 12, is where we key in the momentum coordinates in reciprocal space, over which we integrate (numerically), as mentioned in section 3.3.

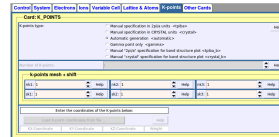


Figure 12: K point grid

These are only some of the essential input cards that need to be filled in for the simulations that were run for this research. There are others that will need to be filled up for other calculations.

4.4 Output in QE

Once the appropriate input is filled in we can run the program. A dialogue box as seen in figure 13 is seen.

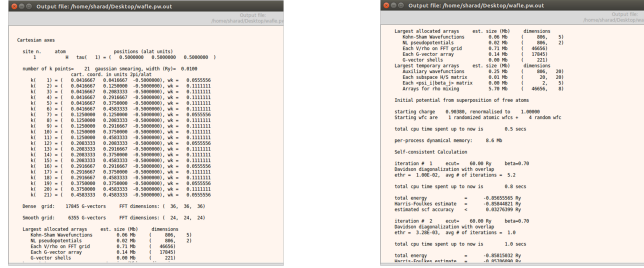


Figure 13: Comparing the QE results with the theoretical prediction

In figure 13, the image on the left shows the output listing the various k points that make up the grid specified in the input. The image on the right shows the program running multiple iterations until convergence is reached. At the end of the run, once convergence has been reached, we see the image in figure 14.

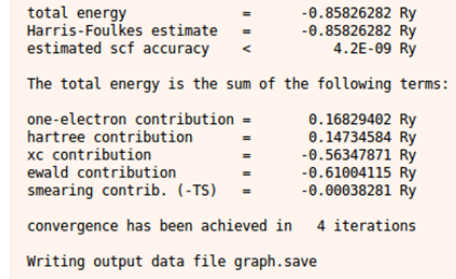


Figure 14: Energy values

At the end of the calculation, we get a value for total energy, as well as the various contributions in Rydberg. For this research, only the total energy will be used.

5 Running Simulations

In this section, various simulations are run to check the structures, as well as the energy values against analytic solutions (wherever available).

5.1 Some Simple cells

First, to examine how the visualization of the crystals work, we simulate the unit cells of Aluminium, silicon, and a modified silicon supercell.

Aluminium forms face-centered cubic structures (fcc), and so we need to key in only one cell dimension- the cell length of 404.95 pm, which is approximately 7.5 bohr.

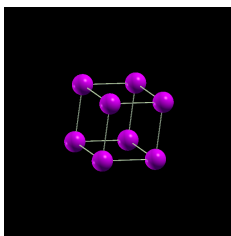


Figure 15: Aluminium cell

For silicon, the crystal structure is Tetrahedral Packing, and so we again input only one parameter- the cell dimension of 10.26 bohr.

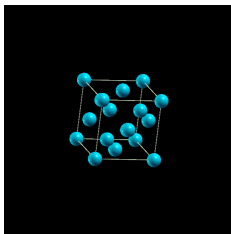


Figure 16: Silicon cell

Both simulations above used a standard numbers of atoms in the cell, and so generated the usual cell. However, given all the input options we have, there is a certain amount of versatility that can be put to use with regard to input options. Figure 15, for instance is a Silicon supercell for which 16 distinct atomic positions are specified explicitly.

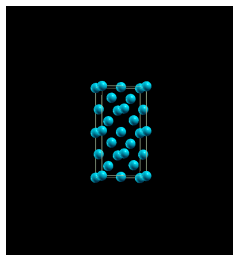


Figure 17: Silicon Supercell

5.2 The Hydrogen Molecule

Now that there is a degree of familiarity with the model visualizations, we need to look at the values of energy generated. Since one of the few multi-atom systems for which we have an analytical solution for ground state energy is the hydrogen molecule, we build a model for it, and compare the energy values obtained with the analytical solution. Figure 18 shows the schematic diagram for the hydrogen molecule. The analytical solution can be found in [6].

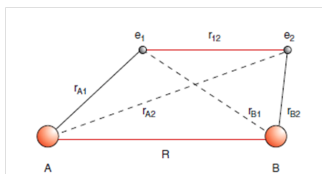


Figure 18: Schematic of Hydrogen molecule

Figure 19 shows two hydrogen atoms separated by a distance inside the unit cell. We see that the cell is elongated along the z-axis to allow for us to vary the distance between the atoms without fear of interaction from adjacent cells in the z-direction.

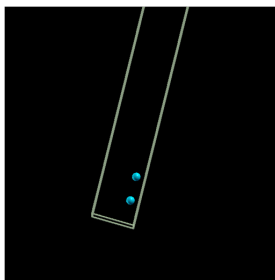


Figure 19: QE simulation of two hydrogen atoms approaching each other

The distance is varied, and the energies at different distances are subtracted

from a control value (two times the energy of a single hydrogen atom), so we can plot the energy difference versus distance on our graph. This is done in figure 20, and the image on the right shows the analytical solution (along with the solutions from some approximations. Source:[6]).

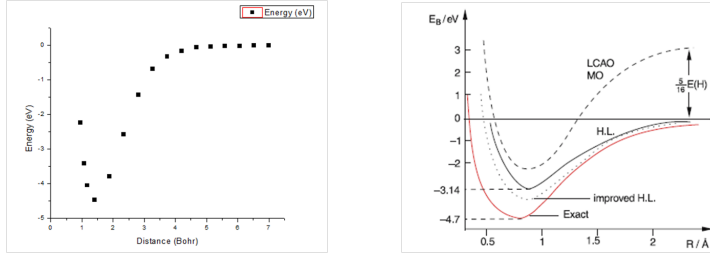


Figure 20: Comparing the QE results with the theoretical prediction

It is seen that there is a minimum energy difference of -4.7 eV between 0.5 and 1 Angstrom in the analytical solution. We see that the QE simulation gives us a minimum of approximately -4.7 eV, although at approximately 0.5 bohr further. The reasons for these differences will be examined in section 6.

5.3 Modeling Graphene

Graphene will be modeled in this section, and it will be used again later to test its properties. Four attempts to simulate graphene were made, and are listed below.

Trial #1 Our first attempt involves trying to model graphene as a free cell. For free cells, we assign a single cell dimension, and all other values are specified as a ratio to this single value.

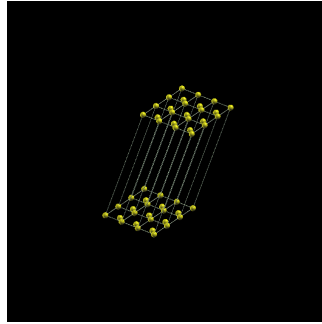


Figure 21: Graphene Trial 1

Clearly, we did not take seriously the repetition of the unit cell. This leads

to atoms from adjacent cells overlapping, and this needs to be fixed.

Trial #2

Here we use the in-built bravais lattice for Hexagonal and Trigonal lattices (ibrav=4). We key in two cell dimensions- 4.65 bohr and 2.00 bohr (for the z-direction). The result is seen in figure 22.



Figure 22: Graphene Trial 2

Clearly, figure 22 is a significant improvement over figure 21. One issue that arises is its small size along the xy-axis. Consider figure 23.

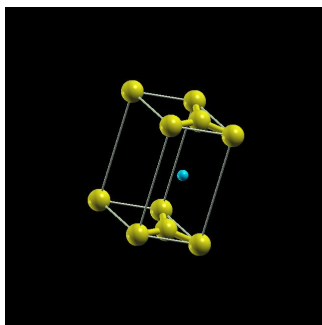


Figure 23: Parking hydrogen on trial graphene 2

Figure 23 represents a common calculation using DFT- to determine the potential when we bring two different configurations of atoms close. Here a single hydrogen is being moved along the z-axis to determine how the ground state energy of the system varies. However, given the small size of the cell, it appears as though it is possible that neighboring cells might be influencing hydrogen significantly, and so might give us wrong results.

Trial #3

To remedy the issue highlighted above, we move to a free lattice and explicitly specify the position of 8 carbon atoms, as well as the appropriate lattice

vectors. The result is figure 24.

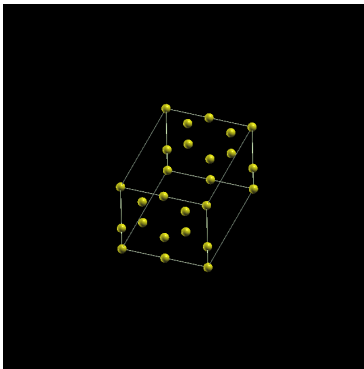


Figure 24: Graphene Trial 3

Trial #4

In a similar spirit, we build a graphene cell with 32 carbon atoms whose positions are explicitly specified.

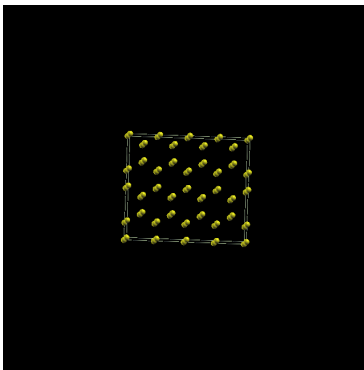


Figure 25: Graphene Trial 4

Although in principle, the trial 4 graphene minimizes external influences the most, being large and unsymmetrical, it takes a very long time for numerical calculations involving it. For this reason, we will use trial graphene 3 for the subsequent section.

5.4 Modeling Hydrogen and Graphene

Now that a graphene simulation is here, we can test what happens when a hydrogen molecule is brought near it. As seen in figure 26, graphene trials 2 and 4 are not suitable for this task. Trial 2 is too small to give reliable results, while trial 4 is too massive to be computable practically (it doesn't converge

even after several hours.) Hence, for this section, we use graphene trial 3, as seen in figure 24.



Figure 26: Energy difference vs. Distance

A 2000 paper titled *Density functional study of adsorption of molecular hydrogen on graphene layers* by J. S. Arellano, et. al., ran the system shown in figure 27 and generated graphs concerning the ground state energy with varying distance. [9]

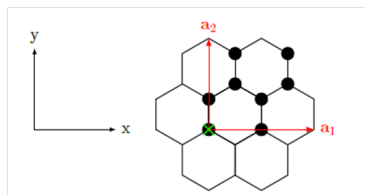


Figure 27: Schematic

Figure 28 shows a hydrogen molecule on trial graphene 3.

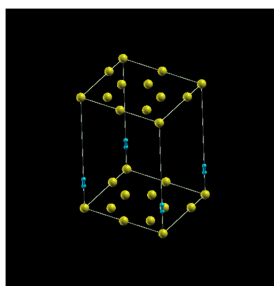


Figure 28: Hydrogen on graphene

Figure 29 (left) is the plot of the graph obtained, while the image on the left

is what was obtained in another similar simulation in a previous paper [4].

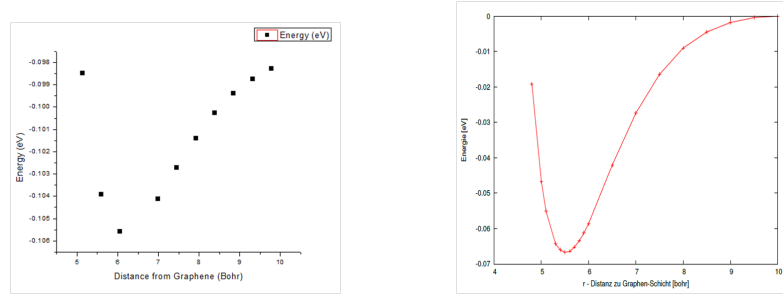


Figure 29: Energy difference vs. Distance

We see that the graph that was obtained shows the same general shape as the previous one, but differs in the location of the minimum energy. (the exact values are not very important, as will be explained in the next section) Since both graphs come from numerical calculations, we now look into the potential sources of errors in numerical calculations in section 6.

6 Accuracy and Errors in numerical calculations

When talking about the accuracy of calculations, it is important to separate physical accuracy, and numerical accuracy. Physical accuracy is to ask whether the predictions generated in a DFT calculation are precise when compared to the value of the property measured under ideal (possibly hypothetical) conditions. On the other hand, numerical accuracy is concerned about whether our attempts at solving the Kohn-Sham equations gave a well-converged solution. In this paper, we have not concerned ourselves with numerical accuracy, as this comes down to coding. We have assumed that QE is a blackbox throughout this project, and so it is physical accuracy that needs to be considered.

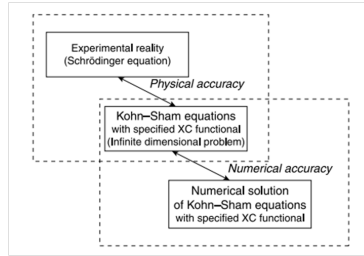


Figure 30: Numerical accuracy and physical accuracy

To determine the extent of physical accuracy, we need to examine the approximations we have made from the beginning. For our plain-wave calculations, these include the number of k-points integrated over and the kinetic energy cut-off. The Born-Oppenheimer approximation used where we assumed that the nuclear distance was just a parameter might also be an issue, since the nuclei are not actually stationary.

To illustrate why we need to be concerned with these, consider the simulation of a single lithium atom shown in figure 31.

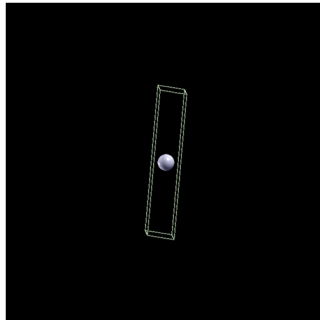


Figure 31: Lithium Atom

The total ground state energy of this atom, according to QE was -201.11 eV. However, the expected ionization energies of lithium's three electrons are -5.39 eV, -75.64 eV, and -122.45 eV, which gives a total of -203.48 eV. Clearly, the values obtained through ESPRESSO do not match up precisely with those obtained experimentally.

For another example, consider the helium atom in figure 32.

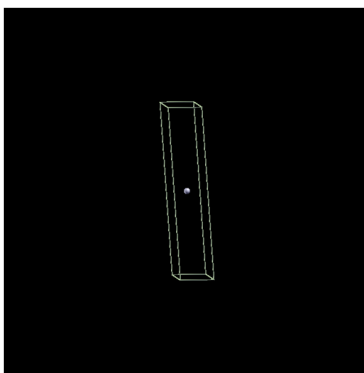


Figure 32: Helium Atom

The total ground state energy from QE was -76.93 eV. But according to the experimental values, the first and second ionization energies of Helium are -24.58 eV and -54.42 eV, which gives a total of -79.00 eV. Once again, we see that the results from QE are not precisely those expected. And so, while QE might predict the ground state value of hydrogen (section 5.2), it seems to give inaccurate results for smaller orders of magnitudes.

Taking all of this into account, it is unsurprising that different values were obtained in section 5.4. The energy values being plotted were of the order of magnitude of 10^{-2} , which are smaller than QE seems to accurately provide.

This is potentially a major worry in using this package as we require very precise values when doing numerical calculations involving atoms and molecules. Note that this issue is exacerbated by the fact that we do not know which approximation might have gone wrong out of those listed above in figure 29. No analytic solution is present, and no experiment can isolate the kind of 8 carbon graphene system we want. It then seems unclear how we can have access to which graph is the right one. However, it should be noted that all the graphs we have got possess the right form, and are approximately accurate. The issue here is precision. More work will have to be done to show how QE can be made to give more precise results.

7 Conclusion

With Density Functional Theory being used increasingly in diverse fields like engineering, material science, chemistry, and physics, it becomes critical to understand its theory, and well as its applications. This research was an attempt at trying to do both. Section 2 offered a brief look at some of the crucial portions of DFT's theoretical foundations including the Schrödinger's equation, the two Hohenberg-Kohn theorems, as well as the Kohn-Sham Approach. It was realized that iterative methods would be needed to carry out the relevant calculations, and the relevant schematic was developed.

One powerful tool to carry out these calculations is Quantum ESPRESSO, a Linux package freely available online. The two external tools PWgui and XCrySDEN were introduced, and then some of the major features of QE were described. These were then tested out using simple crystals like Aluminium and Silicon. The accuracy of the QE energy values was verified using a model of two hydrogen atoms approaching each other. Four different graphene models were built, and their benefits were discussed. Model 3 was selected for further work as it struck a balance between calculation ease, and complexity of design. A hydrogen molecule was then placed near this graphene sheet and the energy variation of the system was observed.

Although the graphs that were plotted resembled those previously obtained, we observed that there were deviations from the exact values expected. This seems to imply that QE doesn't not provide precise predictions for values below a certain order of magnitude, as pointed out in section 6. This also seems to imply that we need to take a closer look at the exact way the program calculates the energies, i.e., we can not treat it as a black box which generates some output when we input values.

Future work should consider experimental verification of the predictions made. It should be verified that the graphene models accurately capture all, or at least the essential properties, of real-life graphene (such as actual dimensions, potential, etc). If a minimum appears on the graph in figure 29, this might mean we can eventually "park" a hydrogen molecule on the surface of graphene using the surface potential. These, as well as other similar experiments will have been made possible by research like that done in this project.

References

- [1] Koch, Wolfram, and Max C. Holthausen. *A Chemist's Guide to Density Functional Theory*. Weinheim: Wiley-VCH, 2001.
- [2] "Why Model with Density Functional Theory?" *SCM: Feature List ADF Program*. N.p., n.d. Web. 12 Apr. 2015.
- [3] Sholl, David S., and Janice A. Steckel. *Density Functional Theory: A Practical Introduction..* Hoboken, NJ: Wiley, 2009.
- [4] Nüßle, Jochen. *Adsorption von Rubidium an Kohlenstoff-Nanoröhren* (Doctoral dissertation). University of Tübingen (Germany). 2013.
- [5] Gross, E. K. U., and Reiner M. Dreizler. *Density Functional Theory*. Berlin: Springer, 2006.
- [6] Demtroder, W. *Atoms, Molecules and Photons: An Introduction to Atomic-, Molecular-, and Quantum-physics*. New York: Plenum, 1995.
- [7] Hohenberg, P., and W. Kohn. "Inhomogeneous Electron Gas." *Physical Review*. 136.3B (1964): B864-871.
- [8] Kohn, W., and L. J. Sham. "Self-Consistent Equations Including Exchange and Correlation Effects." *Physical Review*. 140.4A (1965): A1133-1138.
- [9] Parr, Robert G., and Weitao Yang. *Density-functional Theory of Atoms and Molecules*. New York: Oxford UP, 1989.
- [10] Arellano, J. S., L. M. Molina, A. Rubio, and J. A. Alonso. "Density Functional Study of Adsorption of Molecular Hydrogen on Graphene Layers." *The Journal of Chemical Physics*. 112.18 (2000): 8114.
- [11] Perdew, J. P. "Self-interaction Correction to Density-functional Approximations for Many-electron Systems." *Physical Review B*. 23.10 (1981): 5048-079.
- [12] Fuchs, Martin, and Matthias Scheffler. "Ab Initio Pseudopotentials for Electronic Structure Calculations of Poly-atomic Systems Using Density-functional Theory." *Computer Physics Communications* 119.1 (1999): 67-98.
- [13] Giannozzi, Paolo, Stefano Baroni, Nicola Bonini, Matteo Calandra, Roberto Car, Carlo Cavazzoni, Davide Ceresoli, Guido L. Chiarotti, Matteo Cococcioni, Ismaila Dabo, Andrea Dal Corso, Stefano De Gironcoli, Stefano Fabris, Guido Fratesi, Ralph Gebauer, Uwe Gerstmann, Christos Gougoussis, Anton Kokalj, Michele Lazzeri, Layla Martin-Samos, Nicola Marzari, Francesco Mauri, Riccardo Mazzarello, Stefano Paolini, Alfredo Pasquarello, Lorenzo Paulatto, Carlo Sbraccia, Sandro Scandolo, Gabriele Scanzzero, Ari P. Seitsonen, Alexander Smogunov, Paolo

- Umari, and Renata M. Wentzcovitch. "QUANTUM ESPRESSO: A Modular and Open-source Software Project for Quantum Simulations of Materials." *Journal of Physics: Condensed Matter* 21.39 (2009): 395502, <http://arxiv.org/abs/0906.2569>
- [14] We used the pseudopotentials C.pbe-rrkjus.UPF, Si.pbe-rrkj.UPF, Al.pz-vbc.UPF, and H.pbe-rrkjus.UPF from <http://www.quantum-espresso.org>.
- [15] "Ionization Energies of the Elements (data Page)." *Wikipedia*. Wikimedia Foundation, n.d. Web. 12 Apr. 2015.
- [16] "American Mineralogist Crystal Structure Database." *American Mineralogist Crystal Structure Database*. N.p., n.d. Web. 12 Apr. 2015.

Appendix A: Miscellaneous Figures

There were some figures that might be of interest to anyone interest in the details of the simulations, but weren't necessary to be a part of the thesis itself. Some of these are found here.

Figure 33 shows the input in PWgui for Graphene trial 1.

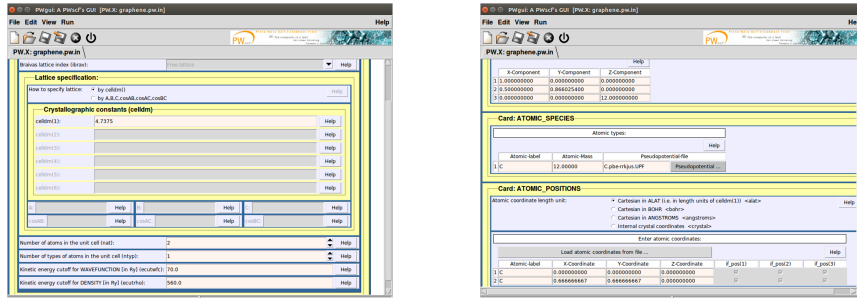


Figure 33: Graphene trial 1

Figure 34 shows the input in PWgui for Graphene trial 2.

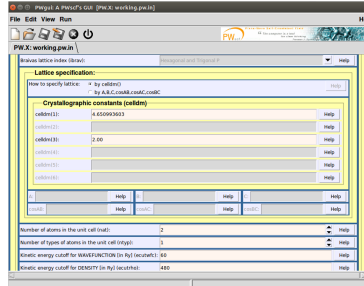


Figure 34: Graphene trial 2

Figure 35 shows the input in PWgui for Graphene trial 3, including a portion of the locations of each of the 8 carbon atoms.

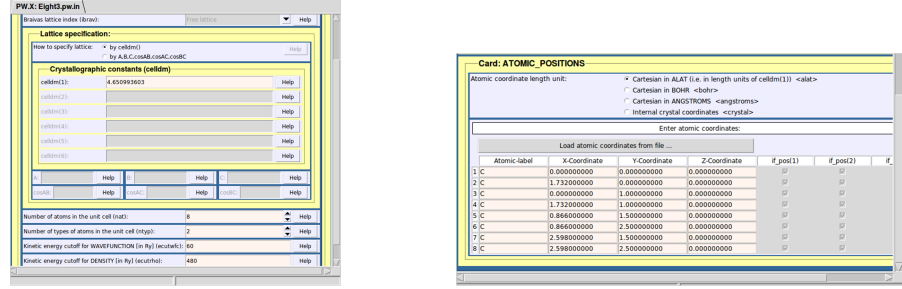


Figure 35: Graphene trial 3

Figure 36 shows the input in PWgui for Graphene trial 4, including a portion of the locations of each of the 32 carbon atoms.

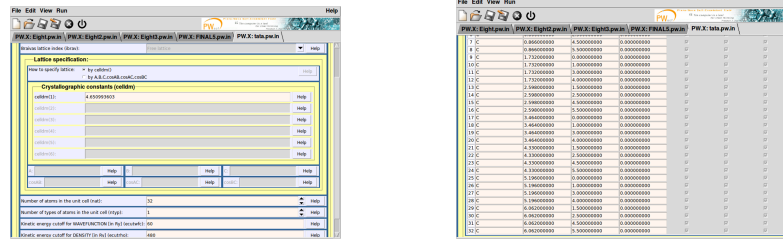


Figure 36: Trial Graphene 4

Figure 37 shows the time difference in calculation caused by asymmetry in the system, as explained in section 3.3.

M	E/atom (eV)	No. of k Points in IBZ	τ_M/τ_1
1	-1.8061	1	1.0
2	-3.0997	1	1.1
3	-3.6352	4	2.3
4	-3.7054	4	2.6
5	-3.7301	10	5.2
6	-3.7541	10	6.0
7	-3.7676	20	10.4
8	-3.7671	20	11.2
9	-3.7680	35	16.9
10	-3.7676	35	17.1
11	-3.7662	56	31.2
12	-3.7665	56	28.5
13	-3.7661	84	40.0
14	-3.7659	84	39.7

M	E/atom (eV)	$\Delta E/\text{atom}$ (eV)	No. of k Points in IBZ	τ_M/τ_1
1	-1.8148	-0.009	1	1.0
2	-3.0900	0.010	4	2.1
3	-3.6272	0.008	14	5.6
4	-3.6969	0.009	32	12.3
5	-3.7210	0.009	63	21.9
6	-3.7446	0.010	108	40.1
7	-3.7577	0.010	172	57.5
8	-3.7569	0.010	256	86.8

Figure 37: Time difference in executing

Appendix B: Screenshots

This appendix contains screenshots of the output that QE yields. The screenshots are instructive as they give us an idea of what goes into the calculations, as well as how the final energy values are displayed.

Figure 38 displays the k -points created as per the instructions input in the k -point grid box.

```

Output file: /home/charnd/Desktop/wfn/qe.out
Output file: /home/charnd/Desktop/wfn/qe.out

Cartesian axes
site N.      atom      positions (a.u.)
1          1          0.000000  0.000000  0.000000
number of k points= 21  gaussian smearing width (Ry)= 0.0000
k1 1 = 0.000000  0.000000  0.000000  wk = 0.000000
k2 2 = 0.000000  0.000000  0.000000  wk = 0.111111
k3 3 = 0.000000  0.000000  0.000000  wk = 0.111111
k4 4 = 0.000000  0.000000  0.000000  wk = 0.111111
k5 5 = 0.000000  0.000000  0.000000  wk = 0.111111
k6 6 = 0.000000  0.000000  0.000000  wk = 0.111111
k7 7 = 0.000000  0.000000  0.000000  wk = 0.000000
k8 8 = 0.125000  0.000000  0.000000  wk = 0.111111
k9 9 = 0.125000  0.000000  0.000000  wk = 0.111111
k10 10 = 0.125000  0.000000  0.000000  wk = 0.111111
k11 11 = 0.125000  0.000000  0.000000  wk = 0.111111
k12 12 = 0.000000  0.125000  0.000000  wk = 0.000000
k13 13 = 0.000000  0.125000  0.000000  wk = 0.111111
k14 14 = 0.000000  0.125000  0.000000  wk = 0.111111
k15 15 = 0.000000  0.125000  0.000000  wk = 0.111111
k16 16 = 0.000000  0.125000  0.000000  wk = 0.000000
k17 17 = 0.000000  0.125000  0.000000  wk = 0.111111
k18 18 = 0.000000  0.125000  0.000000  wk = 0.111111
k19 19 = 0.000000  0.125000  0.000000  wk = 0.111111
k20 20 = 0.125000  0.000000  0.000000  wk = 0.111111
k21 21 = 0.125000  0.000000  0.000000  wk = 0.000000
Density grid: 1380 G-vectors  FFT dimensions: ( 24, 24, 24)
Smooth grid: 825 G-vectors  FFT dimensions: ( 24, 24, 24)
Largest allocated arrays  est. size (MB)  dimensions
Wave-Function            0.00 Mb  ( 800, 1)
H. eigenvalues            0.00 Mb  ( 800, 1)
Each V-vec. in FFT grid  0.14 Mb  ( 17000, 1)
Each V-vec. in FFT grid  0.14 Mb  ( 17000, 1)
G-vector shells           0.00 Mb  ( 17000, 1)
Largest temporary arrays  est. size (MB)  dimensions
Auxiliary wavefunctions   0.25 Mb  ( 800, 1)
Each wavefun. in matrix  0.00 Mb  ( 800, 1)
Each wavefun. in matrix  0.00 Mb  ( 800, 1)
Arrays for the solving    0.00 Mb  ( 17000, 1)
Initial potential from superposition of free atoms
starting charge  0.00000, re-normalized to  1.00000
Starting wfn. str. 1  randomized atomic wfn.  0  random wfn.
total cpu time spent up to now is  0.5 secs
per-process physical memory:  0.0 Mb
self-consistent calculation
Iteration # 1  wfn=  0.00 Ry  before= 70
Residual Eigenvalue with norm= 0
etot = 1.00E-02  avg # of iterations = 5.2
total cpu time spent up to now is  0.0 secs
total energy  =  0.00000000 Ry
Harris-Pulsen estimate =  0.00000000 Ry
estimated of memory  =  0.00000000 Ry
Iteration # 2  wfn=  0.00 Ry  before= 70
Residual Eigenvalue with norm= 0
etot = 3.18E-03  avg # of iterations = 5.8
total cpu time spent up to now is  1.8 secs
total energy  =  0.00000000 Ry
Harris-Pulsen estimate =  0.00000000 Ry

```

Figure 38: Screenshot of QE output: k points

Figure 39 shows the iterations of the calculations.

```

Output file: /home/charnd/Desktop/wfn/qe.out
Output file: /home/charnd/Desktop/wfn/qe.out

Largest allocated arrays  est. size (MB)  dimensions
Wave-Function            0.00 Mb  ( 800, 1)
H. eigenvalues            0.00 Mb  ( 800, 1)
Each V-vec. in FFT grid  0.14 Mb  ( 17000, 1)
Each V-vec. in FFT grid  0.14 Mb  ( 17000, 1)
G-vector shells           0.00 Mb  ( 17000, 1)
Largest temporary arrays  est. size (MB)  dimensions
Auxiliary wavefunctions   0.25 Mb  ( 800, 1)
Each wavefun. in matrix  0.00 Mb  ( 800, 1)
Each wavefun. in matrix  0.00 Mb  ( 800, 1)
Arrays for the solving    0.00 Mb  ( 17000, 1)
Initial potential from superposition of free atoms
starting charge  0.00000, re-normalized to  1.00000
Starting wfn. str. 1  randomized atomic wfn.  0  random wfn.
total cpu time spent up to now is  0.5 secs
per-process physical memory:  0.0 Mb
self-consistent calculation
Iteration # 1  wfn=  0.00 Ry  before= 70
Residual Eigenvalue with norm= 0
etot = 1.00E-02  avg # of iterations = 5.2
total cpu time spent up to now is  0.0 secs
total energy  =  0.00000000 Ry
Harris-Pulsen estimate =  0.00000000 Ry
estimated of memory  =  0.00000000 Ry
Iteration # 2  wfn=  0.00 Ry  before= 70
Residual Eigenvalue with norm= 0
etot = 3.18E-03  avg # of iterations = 5.8
total cpu time spent up to now is  1.8 secs
total energy  =  0.00000000 Ry
Harris-Pulsen estimate =  0.00000000 Ry

```

Figure 39: Screenshot of QE output: Iterations

The integration required for the Kohn Sham approach involves an integral over volume. This is accomplished in QE by integrating over the k-points. This is shown in figure 40.

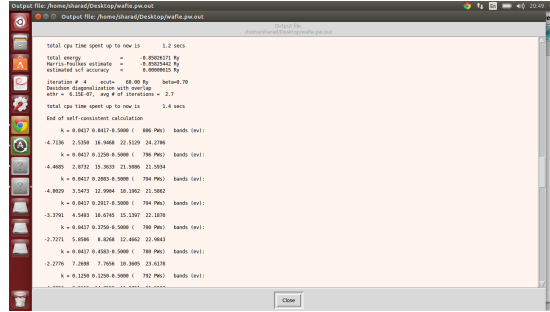


Figure 40: Screenshot of QE output: k point calculation

Figure 41 shows the ground state energy calculated by QE. Note that the total energy is displayed, along with all the components which were discussed in section 1. The final total energy (at the end of the final iteration) can be located by the exclamation mark at the margin.

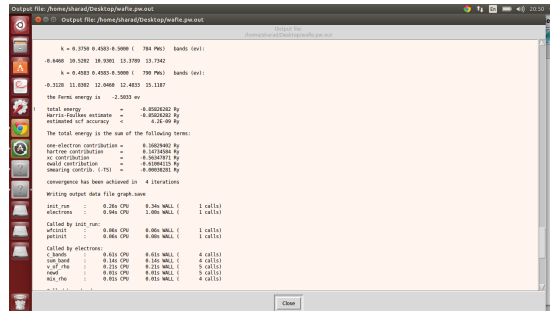


Figure 41: Screenshot of QE output: Total Energy

Notice also that the number of iterations, as well as the time taken for convergence is listed.

Figure 42 shows the screen at the end of the calculation. It has the date and time at the bottom of the screen.

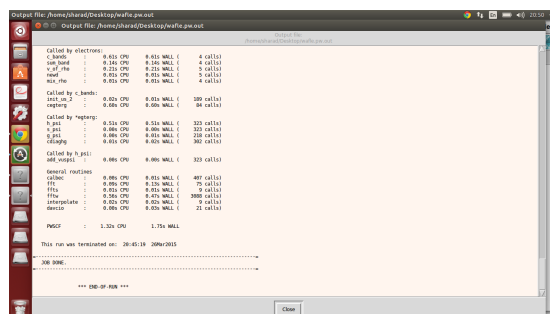


Figure 42: Screenshot of QE output: End of scf calculations

Appendix C: Miscellaneous Models

This appendix contains screenshots of models and the corresponding energy graphs that did not make it into the main report for various reasons.

Figure 43 shows a cell with a single hydrogen atom.

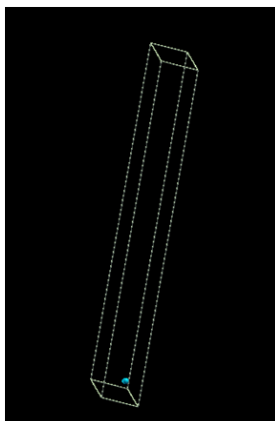


Figure 43: Single Hydrogen atom

Figures 44 and 45 show how the simulations that led up to Figure 19, where the hydrogen molecule was studied. Figure 44 shows two hydrogen atoms close enough to generate the default bond in the XCrySDEN view.

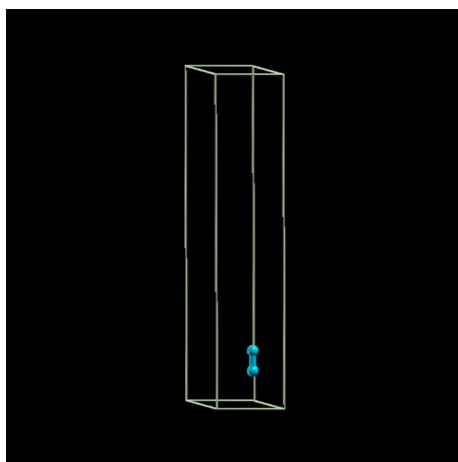


Figure 44: Hydrogen atoms at a distance of 3.65 bohr

Figure 45 shows the atoms from figure 44 at a distance where the bond is no longer generated.

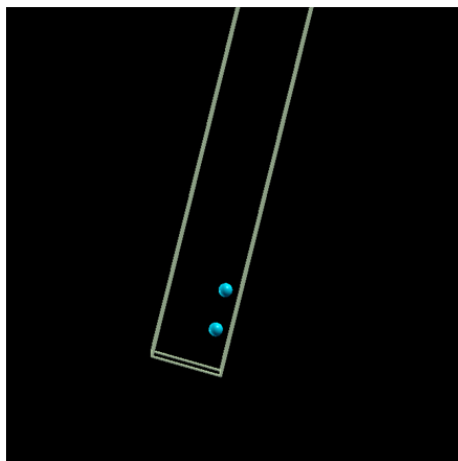


Figure 45: Hydrogen atoms at a distance of 4.65 bohr

Figure 46 shows the hydrogen molecule placed at a variable distance from a carbon atom. The hydrogen molecule is modeled by placing two hydrogen atoms at a distance equal to bond length.

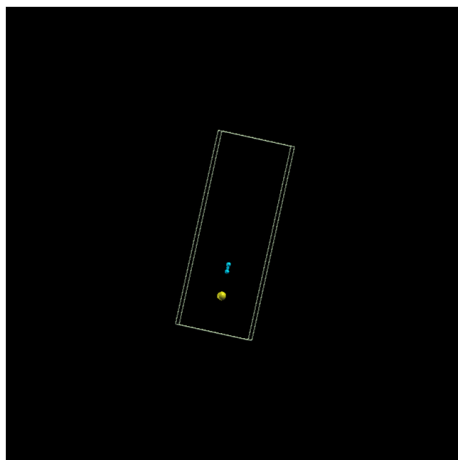


Figure 46: Single carbon atom and a Hydrogen molecule

Figure 47 shows a hydrogen atoms being placed too close to a carbon atom, causing it to overlap. We can see that this places a limit of the variable distance between different atoms.

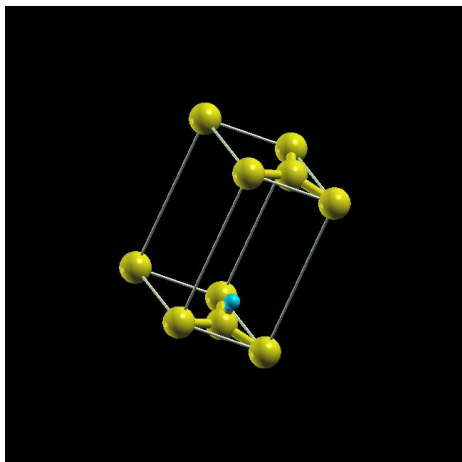


Figure 47: Hydrogen atom parked too close on trial graphene 2

Figure 48 shows the Hydrogen atoms at a larger distance from the graphene sheet.

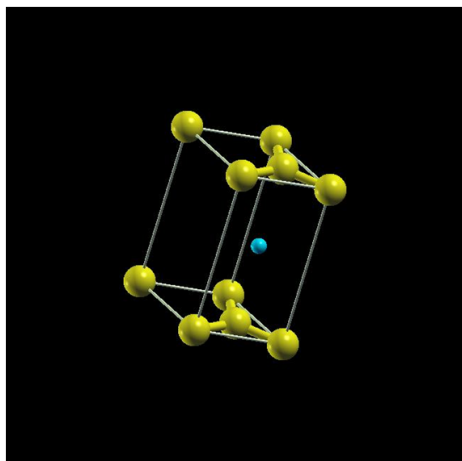


Figure 48: Atom on trial graphene 2: position 1

The ground state energy for different distances in figure 48 are plotted in figure 49 (left). The Hydrogen atom is then replaced with a Rubidium atom and the energies are plotted in figure 49 (right).

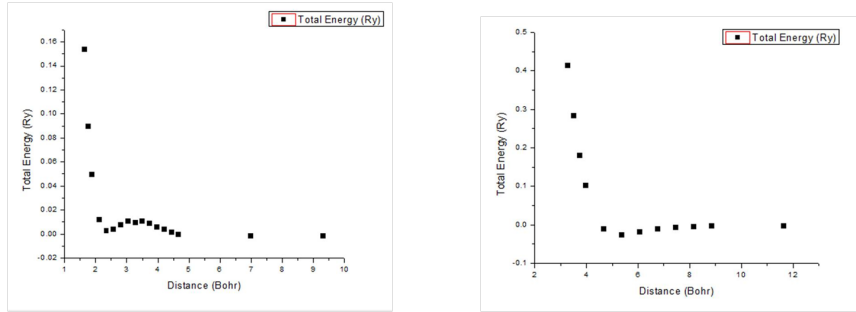


Figure 49: Energy graphs for a hydrogen atom (left) and a Rubidium atom (right) in position 1

The position of the hydrogen atom is shifted, as is shown in figure 50.

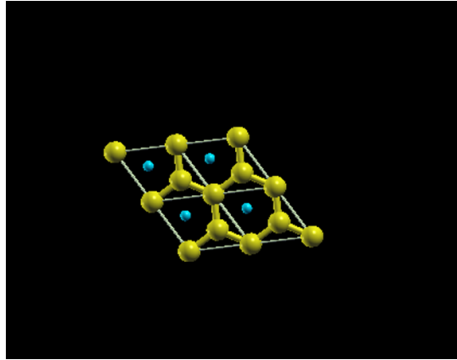


Figure 50: Atom on trial graphene 2: position 2

The ground state energy for different distances in figure 50 are plotted in figure 51 (left). The Hydrogen atom is then replaced with a Rubidium atom and the energies are plotted in figure 51 (right).

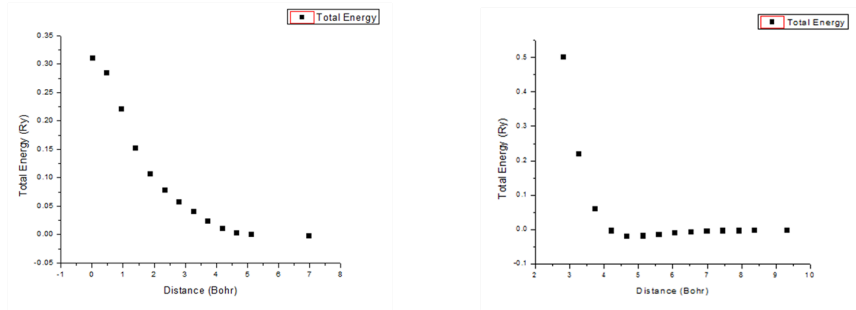


Figure 51: Energy graphs for a hydrogen atom (left) and a Rubidium atom (right) in position 2

The position of the hydrogen atom is again shifted, as is shown in figure 52.

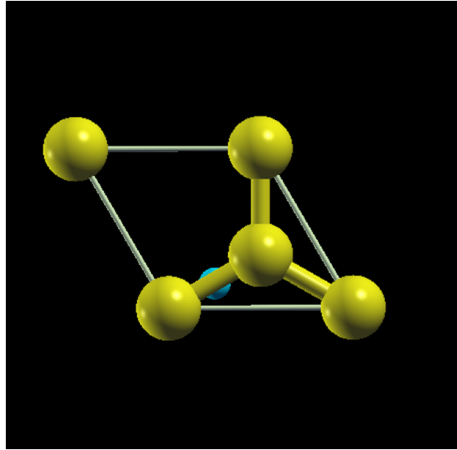


Figure 52: Atom on trial graphene 2: position 3

The ground state energy for different distances in figure 52 are plotted in figure 53 (left). The Hydrogen atom is then replaced with a Rubidium atom and the energies are plotted in figure 53 (right).

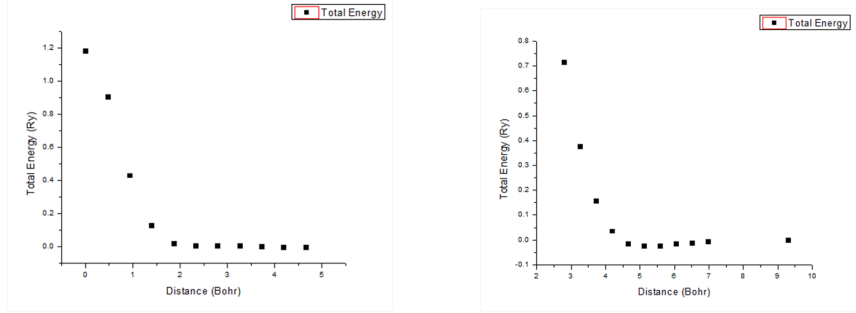


Figure 53: Energy graphs for a hydrogen atom (left) and a Rubidium atom (right) in position 3

Figure 54 shows a rubidium atom placed on a variation of trial graphene 2. It is now clear why these graphs were excluded from the main report- with the rubidium going outside the cell, it is unclear if the energy values represent the ground state energy of a single rubidium atom anymore.

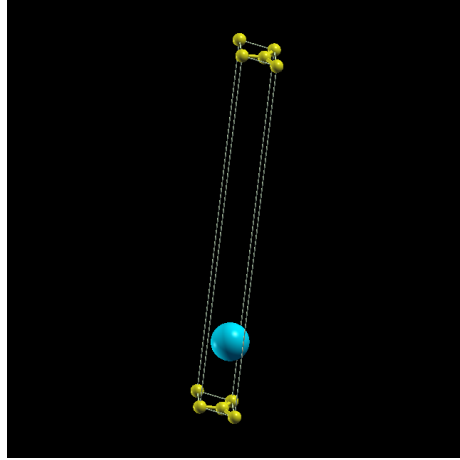


Figure 54: Rubidium atom on graphene

Figures 55 and 56 represent the model of a single hydrogen atom on graphene trial 3. Since the model of graphene with a hydrogen molecule was shown in figure 28, these models were excluded since they did not add to the discussion.

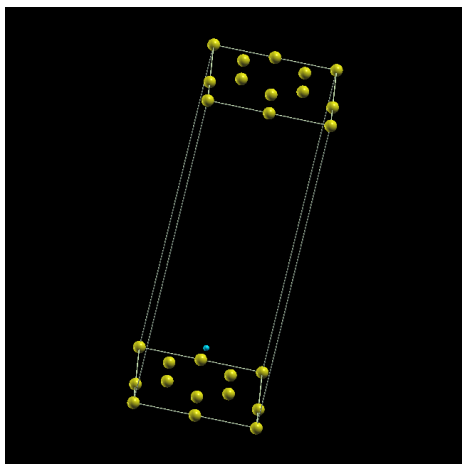


Figure 55: Hydrogen atom on graphene trial 3: View 1

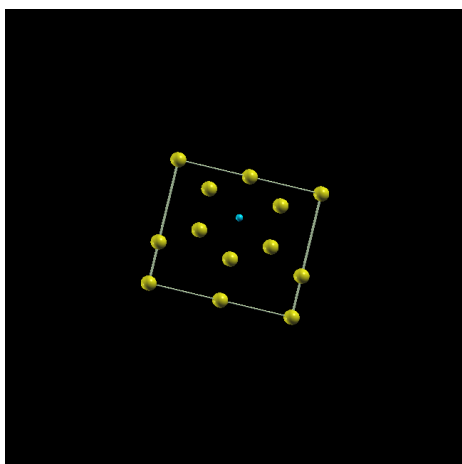


Figure 56: Hydrogen atom on graphene trial 3: View 2

Figures 57 and 58 represent the model of a single hydrogen atom on graphene trial 4. These systems were too large and unsymmetrical to converge in a reasonable amount of time.

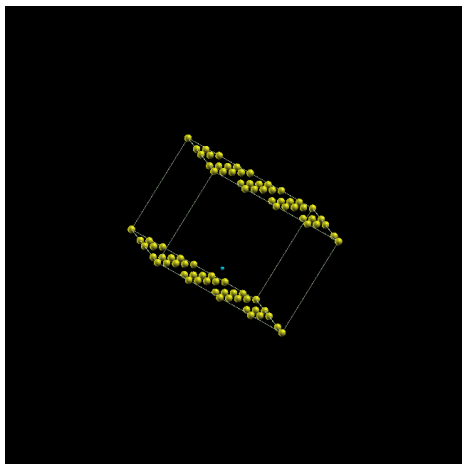


Figure 57: Hydrogen atom on graphene trial 4: View 1

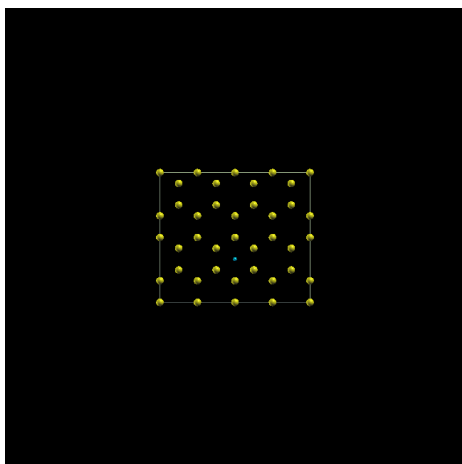


Figure 58: Hydrogen atom on graphene trial 4: View 2

Figure 59 shows a single rubidium atom on graphene trial 4. Note that unlike figure 54, the rubidium atom here is completely contained within the graphene supercell. However, as in the case of figures 57 and 58, this system is too large to converge quickly.

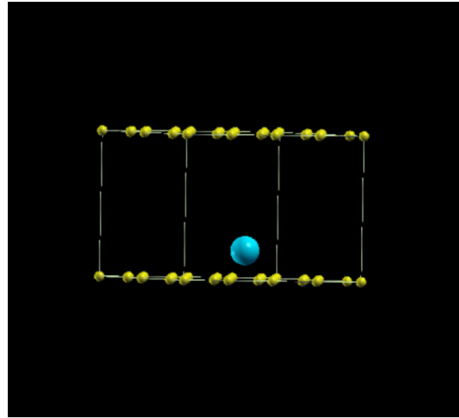


Figure 59: Rubidium atom on graphene trial 4