Car-Parrinello Molecular Dynamics: Final Project

Paul Yang
Ashwathi Iyer

May 13, 2015

Abstract

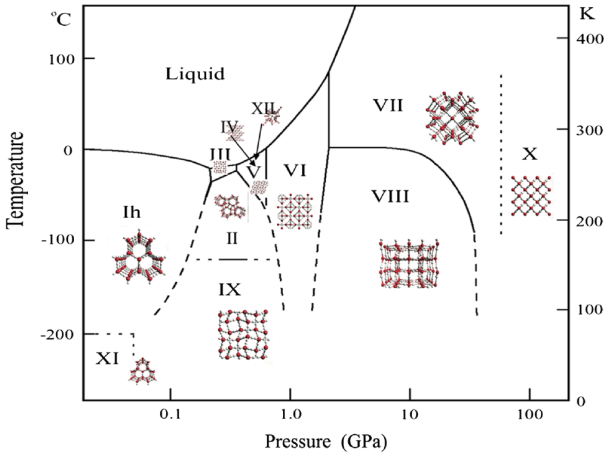
We simulate a monolayer and bilayer of water between graphene sheets using density functional theory (DFT) and Car-Parrinello molecular dynamics (CPMD) and compare our results with a related study on the system [1]. We have also successfully written and implemented a CPMD code, including the SHAKE algorithm that allows for dynamics in the presence of constraints, for a toy system - the hydrogen atom. We present the energy trace and norm of the Kohn-Sham orbital of the electrons from this run. We also present a method for the minimization of the energy of the system that reflects the connection between CPMD and simulated annealing.

1 Introduction

The importance of water cannot be downplayed. Bulk water runs in oceans, lakes and even our very own bodies. It is a crucial ingredient in photosynthesis and can act as a lubricant in many biological and geological processes. Its solid form is even more fascinating, as it varies in structure and property. Ice is of crucial importance in many fields ranging from material science to biology. Despite a large number of experimental and theoretical studies on various phases of water, many of its properties remain elusive. In particular, uncertainties remain on various ground state structures of ice in its phase diagram (Figure 1).

Most ice structures are locally tetrahedral. This is true even in the amorphous phases and in the so-called cubic-ice phase. Interestingly, experiments have shown that the local tetrahedral environment may be broken for confined ice. In particular, we are fascinated by a recent Nature article that ice can form a simple cubic lattice when sandwiched between layers of graphene [1]. This study performed a classical molecular dynamics simulation on the system using an empirical interaction potential and found that the structure of bilayer ice changes from disordered to a cubic lattice at a transition pressure of $1GPa$,

Figure 1: Phase diagram for bulk ice



whereas a monolayer of ice always prefers a square lattice. In this project, we attempt to simulate the same system using DFT as well as an ab-initio molecular dynamics method, Car-Parrinello molecular dynamics and compare our results with those in Ref. [1].

1.1 Why do we care about CPMD?

Car-Parrinello Molecular Dynamics, which was first published in 1985 [3], has found widespread applicability in computational physics and materials science. Up until then, DFT was primarily used for ab-initio electronic structure calculations and classical MD with external force fields was used to study dynamics. The main drawback of DFT is that is works very well only when studying the electronic structure properties of ordered and homogenous systems, such as crystals. While classical MD can deal with disordered or amorphous systems well, the force fields used (which are normally empirical), don't take into account the variation in the electronic structure of the material with the movement of the atoms, which is crucial for forming an breaking chemical bonds for example [6].

The formulation of CPMD allows for the study of disordered systems and the ability to follow the evolution of the electronic potential during the simulation, by combining the best aspects of DFT and classical MD. CPMD is also widely used to study chemical reactions occurring in liquids and large biomolecules. In fact, Car and Parrinello demonstrated their method for amorphous silicon and computational results were for the first time in very good agreement with experimental results [3].

CPMD relies on the separation of time scales between the nuclear and electronic motions, also known as the Born-Oppenheimer or the adiabatic approx-

imation. This means that the motion of the ions only carry physical meaning when the electronic orbitals are always near their ground state as they follow the nuclear motion during the simulation. Starting close to the ground state of the electronic orbitals (obtained through a standard DFT calculation or using a steepest descent/damped dynamics algorithm), CPMD evolves the the ions using Hellman-Feynman forces and the electrons using a fictitious electron dynamics imposed to keep the evolution adiabatic. This approach allows for a great reduction in computation time, in comparison to DFT calculations which involve a matrix diagonalization at every step of the self-consistent loop, an operation that grows in computation time with the number of degrees of freedom. Hence to this day, DFT is typically used for model systems and not for studying dynamical evolution.

1.1.1 Limitations

The adiabatic approximation has to be maintained in order to keep the electrons close to the ground state. If the adiabatic approximation were not maintained, the ions can transfer kinetic energy to the electrons and vice-versa. Since the Hellman-Feynman theorem assumes that the ions move an on adiabatic potential surface, the forces on the ions would end up being inaccurate as well. What this means practically is that the energy gap of the system (energy required to excite the electrons) should be much higher than timescale of the ionic motion. This is generally true for insulators and semiconductors, but not for metals, for which CPMD cannot be directly applied. In addition, a small enough fictitious mass for the electrons must be chosen to allow the orbitals to follow the ions adiabatically.
CPMD also suffers from the typical drawbacks that DFT suffers from, such as the inability to describe van der Waals forces, the band gap problem and the inability to accurately describe highly correlated electrons with localized d and f orbitals. [6].

2 Theory

2.1 Density Functional Theory

The first step of a CPMD calculation is to relax the system close to the electronic ground state so that the subsequent molecular dynamics will be adiabatic. Most widely available software packages such as Quantum Espresso relax to the electronic ground state using steepest descent before starting the CPMD run.
In our implementation of the code, however, we relax to the electronic ground state using a DFT calculation. The principal idea behind a DFT calculation is the Kohn-Sham equation, which represents the total energy of the system as a functional of the electronic density.

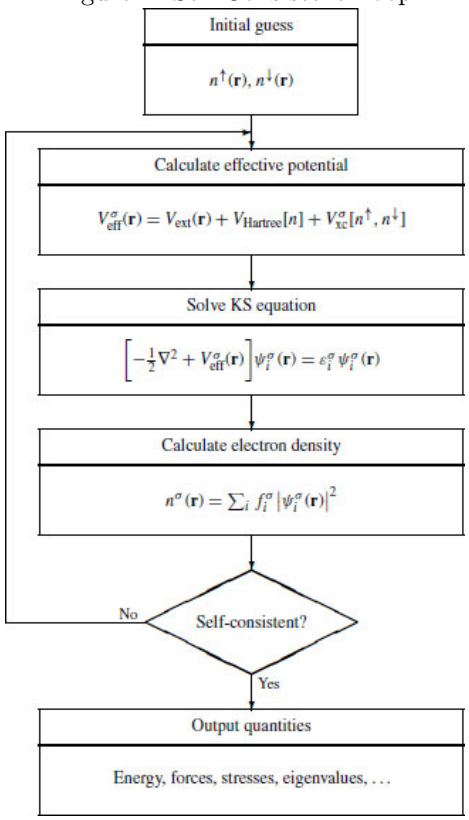
$$H_{KS}(\vec{r}) = -\frac{1}{2}\vec{\nabla}^2 + V_{KS}[n](\vec{r}) \tag{1}$$

$V_{KS}(\vec{r})$ is a functional of the density, given by:

$$V_{KS}(\vec{r}) = V_{ext}(\vec{r}) + \frac{\delta E_{hartree}}{\delta n(\vec{r})} + \frac{\delta E_{XC}}{\delta n(\vec{r})} \tag{2}$$

The idea then is to perform a self-consistent calculation by "guessing" an initial electronic density $n(\vec{r})$ and solving for the energy, and then iterating until we reach the ground state energy and density, as shown in Figure 2.1 [7]. In practice, the eigenvalues of the Kohn-Sham equation are calculated in Fourier space.

Figure 2: Self-Consistent Loop



2.2 Electronic Car-Parrinello

In Carr and Parrinello's original 1985 paper, they described their method as an "unified approach for molecular dynamics and density-functional theory". That is, their theory would allow us to *either* compute ground-state electronic properties or perform *ab initio* molecular dynamics simulation. [3] Over the years, so much focus was put on the molecular dynamics part that the method

has come to be known as CPMD. Here we will present a practical review of the theory starting with its easier half (ground-state electronic structure calculation) and present a mathematical perspective (simulated annealing). We use basis parametrization of the electronic wave functions to connect the theory with practical implementation.

At its heart, the CP method is a constraint optimization scheme based on the Rayleigh-Ritz variational Principle. That is, after a functional is chosen, the ground state of Equation 1 can be obtained by direct minimization of the Kohn-Sham functional, instead of using an SCF procedure. Let us first consider the case where the ions are fixed and electronic orbitals are allowed to vary without any external constraint (there is still the internal constraint that the orbitals being orthonormal). Once a set of N_b basis functions $\{b_k(\vec{r})\}$ is chosen, each of the N_e electronic orbitals can be written as a linear combination of the basis functions

$$\psi_i(\vec{r}) = \sum_{k=1}^{N_b} c_{ik} b_k(\vec{r}), i = 1, 2, \cdots, N_e \tag{3}$$

then the Kohn-Sham functional E is nothing but a multi-variable function of the coefficients $\{c_{ik}\}$. The variational Principle guarantees that the best approximation to the ground state energy and orbital coefficients within the span of the chosen basis set can be obtained by minimizing the objective function

$$\mathcal{O} = E(\{c_{ik}\}) \tag{4}$$

subject to the constraint

$$g = \int \psi_i^*(\vec{r})\psi_j(\vec{r})d\vec{r} - \delta_{ij} = 0 \Rightarrow$$

$$g = \sum_{k,k'=1}^{N_b} c_{ik}^* c_{jk'} \left(\int b_k^*(\vec{r}) b_{k'}(\vec{r}) d\vec{r} \right) - \delta_{ij} = 0 \tag{5}$$

At this point, the problem of obtaining ground-state electronic energy and orbitals is a purely mathematical one, namely minimize (4) subject to (5). Since $E(\{c_{ik}\})$ is a $3N_e$ dimensional function of complex variables, its global minimum is a needle in a cosmic haystack. Fortunately just two years before Car and Parrinello published their method, Kirkpatrick, Delatt and Vecchi brought one of the best mathematical tools for finding a needle in a hay stack (*simulated annealing*) to the physics community. [4] Their method uses Metropolis moves to explore the phase space and can be directly applied to the problem at hand. However, CP realized they could take the idea one step further. Drawing inspiration from classical MD simulation, we see that well-designed Metropolis moves can be a more efficient way to explore the phase space than Lagrangian dynamics, however it loses all information pertaining to dynamical correlation. By switching from Metropolis moves back to Lagrangian dynamics, one loses some efficiency but will be able to capture dynamical correlation. In classical MD, it is obvious that one should use the masses of the particles as damping

factors for the time derivatives of the phase space parameters (positions of the particles). Here on the other hand, this parameter will have to be chosen. That is, the Lagrangian for the electronic degrees of freedom is

$$\begin{aligned}\mathcal{L} =& \sum_i \frac{1}{2}\mu \int |\dot{\psi}_i(\vec{r})|^2 d\vec{r} - E[\{\psi_i\}] \\ &+ \sum_{ij} \Lambda_{ij} \left[\int \psi_i^*(\vec{r})\psi_j(\vec{r}) - \delta_{ij} \right] \\ =& \sum_{i=1}^{N_e} \left[\frac{1}{2}\mu \sum_{k,k'=1}^{N_b} \dot{c}_{ik}^* \dot{c}_{jk'} \left(\int b_k^*(\vec{r}) b_{k'}(\vec{r}) d\vec{r} \right) - E(\{\vec{c}_i\}) \right] \\ &+ \sum_{i,j=1}^{N_e} \Lambda_{ij} \left[\sum_{k,k'=1}^{N_b} c_{ik}^* c_{jk'} \left(\int b_k^*(\vec{r}) b_{k'}(\vec{r}) d\vec{r} \right) - \delta_{ij} \right] \end{aligned} \tag{6}$$

where μ is the *fictitious mass* for the electrons, serving as the damping factor for the phase space parameters (orbital coefficients $\{\vec{c}_i\}$). Λ_{ij} is a matrix of Lagrangian multipliers to reinforce the orthonormality of the orbitals. The resulting equations of motion can be integrated with a standard MD integrator such as the Verlet algorithm, and the Lagrangian multipliers can be obtained almost exactly with an iterative method called SHAKE. [5]

2.3 CPMD

There is no reason why (4) and (5) cannot include the ionic degrees of freedom and that is exactly the idea behind CPMD. The full CPMD Lagrangian

$$\begin{aligned}\mathcal{L} =& \sum_i \frac{1}{2}\mu \int |\dot{\psi}_i(\vec{r})|^2 d\vec{r} + \sum_I \frac{1}{2} M_I \dot{\vec{R}}_I^2 + \sum_\nu \frac{1}{2}\mu_\nu \dot{\alpha}_\nu^2 \\ &- E[\{\psi_i\}, \{\vec{R}_I\}, \{\alpha_\nu\}] + \sum_{ij} \Lambda_{ij} \left[\int \psi_i^*(\vec{r})\psi_j(\vec{r}) - \delta_{ij} \right] \end{aligned} \tag{7}$$

where I label the ions with M_I being the mass of ion I and $\vec{R}_I$ being the position of ion I. α_μ are any additional parameters the Hamiltonian depends on, which could arise from external volume or pressure constraints for example. μ_ν are the damping factors for these extra parameters. The equations of motion arising from (7) are

$$\begin{cases} \mu\ddot{\psi}_i = -\frac{\delta E}{\delta \psi_i^*} + \sum_k \Lambda_{ik}\psi_k \\ M_I \ddot{\vec{R}}_I = -\vec{\nabla}_{\vec{R}_I} E \\ \mu_\nu \ddot{\alpha}_\nu = -\frac{\partial E}{\partial \alpha_\nu} \end{cases} \tag{8}$$

When the wave function are put into a basis, the electronic equation of motion becomes

$$\mu\ddot{\vec{c}}_i = -\left(H\vec{c}_i - \sum_k \lambda_{ik}\vec{c}_k\right) \tag{9}$$

which can be integrated with Verlet to obtain [7]

$$\vec{c}\,_i^{\,n+1} = 2\vec{c}\,_i^{\,n} - \vec{c}\,_i^{\,n-1} - \frac{(\Delta t)^2}{\mu}\left(H\vec{c}\,_i^{\,n} - \sum_k \lambda_{ik}\vec{c}\,_k^{\,n}\right) \tag{10}$$

2.3.1 SHAKE algorithm

The following discussion of the SHAKE algorithm has been adapted from Ref. [8]. The equation of motion for the electronic degrees of freedom in CPMD, when using the plane wave basis is:

$$c_i^{n+1}(\vec{G}) = 2c_i^n(\vec{G}) - c_i^{n-1}(\vec{G}) - \frac{(\Delta t)^2}{\mu}\left(\sum_{\vec{G}} H(\vec{G},\vec{G}')c_i^n(\vec{G}') - \sum_k \lambda_{ik}c_k^n(\vec{G})\right) \tag{11}$$

In Equation 11, n denotes the current time step, $c_i(\vec{G})$ are the plane wave coefficients of the i^{th} Kohn-Sham orbital, λ_{ik} is the matrix of Lagrangian multipliers and μ is the pseudo-mass of the electron. Remember that the Lagrange multipliers were introduced in order to satisfy the constraint that the orbitals are orthonormal during every simulation step of the CMPD calculation. We need to ensure that the constraints are indeed satisfied, i.e. we need to figure out how to time evolve the Lagrange multipliers. Also, note that when you have just one orbital, as in the case of a hydrogen atom, the above equation can be rewritten as:

$$c^{n+1}(\vec{G}) = 2c^n(\vec{G}) - c^{n-1}(\vec{G}) - \frac{(\Delta t)^2}{\mu}\left(\sum_{\vec{G}} H(\vec{G},\vec{G}')c^n(\vec{G}') - \lambda c^n(\vec{G})\right) \tag{12}$$

The common algorithm used to time evolve the Lagrange multipliers is called the SHAKE algorithm. We'll give a brief overview of it here.

When you have constraints, the Lagrangian is: $L_c = L_u - \sum_\alpha \lambda_\alpha \sigma_\alpha(\vec{q})$, where $\sigma_\alpha = 0$ is the constraint to be satisfied for each α. Note that in our case the generalized coordinates $\vec{q}$ will be the plane wave coefficients of the Kohn-Sham orbitals.
This Lagrangian leads to the equation of motion

$$\mu\ddot{\vec{q}}_i = -\vec{\nabla}_i U - \sum_\alpha \lambda_\alpha \vec{\nabla}_i \sigma_\alpha \tag{13}$$

The second term in Equation 13 can be interpreted as the force applied to keep the constraints satisfied.

Now, the idea is that we require that the constraints stay exactly satisfied at every time step of the CPMD simulation. Some algebra shows that in the velocity Verlet picture, the discretized equation of motion reads as:

$$\vec{q}_i^c(t+\Delta t) = \vec{q}_i^u(t+\Delta t) - \frac{\Delta t^2}{m}\sum_\alpha \lambda_\alpha \vec{\nabla}_i \sigma_\alpha(t) \tag{14}$$

In Equation 14, the subscripts c and u denote the constrained and unconstrained solutions. We want the constraints to be satisfied at $t+\Delta t$, i.e. $\sigma_\alpha^c(t+\Delta t) = 0$. We can perform a Taylor expansion of this expression around the "unconstrained" constraint function, i.e. the constraint function evaluated after evolving the generalized coordinates according to Equation 14.

$$\sigma_\alpha^c(t+\Delta t) = \sigma_\alpha^u(t+\Delta t) + \sum_{i=1}^N \vec{\nabla}_i \sigma_\alpha^u(t+\Delta t).(\vec{q}_i^c(t+\Delta t) - \vec{q}_i^u(t+\Delta t)) + O(\Delta t^4) \tag{15}$$

Using Equation 14, Equation 15 can be rewritten as:

$$\sigma_\alpha^c(t+\Delta t) = \sigma_\alpha^u(t+\Delta t) - \sum_{i=1}^N \sum_\beta \frac{\Delta t^2}{m}\lambda_\beta \vec{\nabla}_i \sigma_\beta^c(t).\vec{\nabla}_i \sigma_\alpha^u(t+\Delta t) \tag{16}$$

Equation 16 is a complicated matrix equation. The idea of the SHAKE algorithm is the decouple the constraints and treat each σ_α independently. This is done by the following approximation (compare to Equation 14):

$$\vec{q}_i^c(t+\Delta t) - \vec{q}_i^u(t+\Delta t) = \frac{-\Delta t^2 \lambda_\alpha}{m}\vec{\nabla}_i \sigma_\alpha(t)$$

This approximation leads to:

$$\sigma_\alpha^u(t+\Delta t) = \Delta t^2 \lambda_\alpha \sum_{i=1}^N \frac{1}{m}\vec{\nabla}_i \sigma_\alpha^u(t+\Delta t).\vec{\nabla}_i \sigma_\alpha^c(t) \tag{17}$$

So at every time step, the new Lagrange multiplier is calculated by:

$$\lambda_\alpha(\Delta t^2) = \frac{\sigma_\alpha^u(t+\Delta t)}{\sum_{i=1}^N \frac{1}{m}\vec{\nabla}_i \sigma_\alpha^u(t+\Delta t).\vec{\nabla}_i \sigma_\alpha^c(t)} \tag{18}$$

So the idea is that you treat all the constraints successively at each time step and repeat this process until the constraints are satisfied with the desired accuracy, and then move on to the next CPMD time step.

3 Simulation of Water between Graphene Sheets

We simulate a monolayer and bilayer of water at different coverages between two fixed graphene sheets, following Ref. [1]. The authors of Ref. [1] found that a monolayer of water relaxes to a square lattice, whereas a bilayer of water transitions from a disordered state (with local tetrahedral coordination) to a cubic lattice at a pressure of 1 GPa. An empirical force field called the extended simple point charge model (SPC/E) is used in Ref. [1] to describe the interactions between the water molecules. This is a sum of a long-range Coulomb potential and a short-range Lennard Jones potential. This force field takes into account the dipole interactions between water molecules.
In this project, we model the same system using the ab-initio methods of DFT and CPMD and probe if we get similar results to those in Ref. [1].

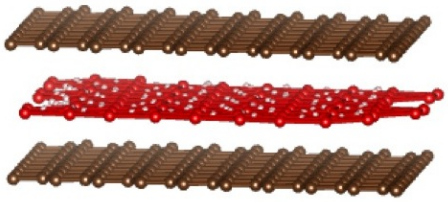
3.1 Simulation Setup

Our simulation setup consists of 4×4 sheets of graphene separated by 6.5 Å for a monolayer of ice and by 9 Å for a bilayer of ice, as shown in Figure 3.

For the DFT run, the plane wave cutoff used is 520 eV and the k-space grid is $5 \times 5 \times 1$. It is worthwhile to note that if we are interested in the band structure of the graphene-water system, a much finer k-grid will be required.
The CPMD run uses the following convergence parameters:
Fictitious electron mass (μ) = 100 a.u.
Time step (Δt) = $4.84 \times 10^{-17} s$
Number of simulation steps = 10000

We initialize the system in either a square or a triangular lattice and see how the system evolves and compare the final stable structures. If we didn't have a limitation on the computer time available to us, we would be able to start at completely random positions of the water molecules as was done in Ref. [1].

Figure 3: Simulation setup. Brown atoms are carbon, red atoms are the oxygen atoms of water.



3.2 Monolayer of Water

3.2.1 Quarter Coverage

We perform DFT and CPMD calculations on a monolayer of water at quarter coverage, in square and triangular configurations as shown in Figures 4 and 5. The setup consists of 64 carbon atoms and 4 water molecules.
For the DFT run, we observe that the square ice configuration is more stable and is lower in energy than the triangular configuration by 5.94 eV, whereas our CPMD results show that the square ice is more stable by 6.53 eV.
CPMD calculations performed at 0, 0.5 and 1 GPa all show the same qualitative differences, matching the result in the Ref [1]. At low coverage, our results agree with results from a standard MD simulation.

Figure 4: Top view of the square lattice at quarter coverage of a single layer of water

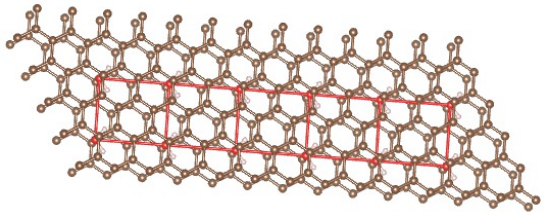
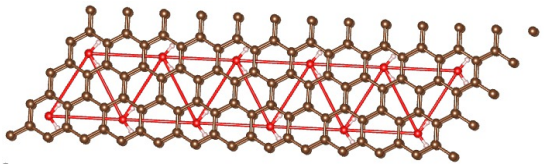


Figure 5: Top view of the triangular lattice at quarter coverage of a single layer of water



3.2.2 Full Coverage

At full coverage, there are 16 water molecules instead of 4. Again, we perform DFT and CPMD calculations at full coverage for a monolayer of water.
Interestingly, in the DFT calculation, the square lattice relaxes to a triangular lattice, indicating that the triangular lattice is more stable by 5.8 eV, whereas our CPMD results show that the triangular lattice is 0.033 eV more stable than the square lattice. A plausible reason for this discrepancy between our results and the results in Ref [1] is discussed in Section 3.6.

The square and triangular structures are shown in Figures 6 and 7.

Figure 6: Top view of the square lattice at full coverage of a single layer of water

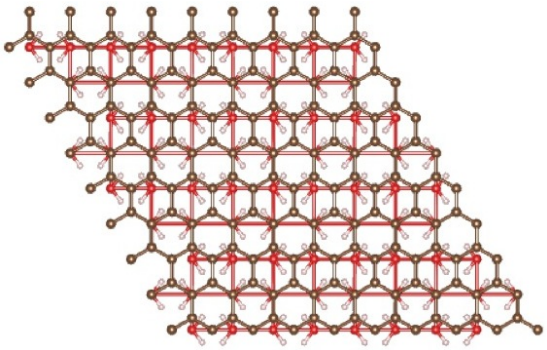
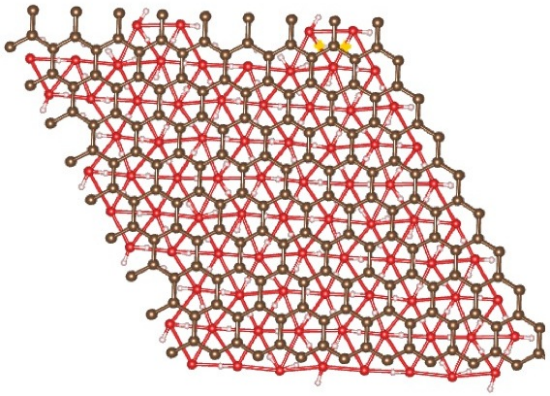


Figure 7: Top view of the triangular lattice at full coverage of a single layer of water



3.3 Bilayer of Water

3.4 Full Coverage

With a bilayer of water, we are interested in the effect of water layer stacking on the ground state energy. Ref. [1] found that the ice molecules arrange themselves in an AB stacking with no in-plane ordering, but transition to a cubic lattice at 1 GPa. However, experiments have observed an AA stacking, with the water molecules arranged in a cubic lattice.
In the DFT calculation, we start with water arranged in a cubic lattice with AA and AB stacking, in order to compare the final energies of both systems. Again, following the same pattern as in the monolayer ice at full coverage, the water molecules arrange themselves in a triangular lattice when starting with AA stacking (Figure 8). When starting with AB stacking, in-plane order is lost, but the water molecules maintain local tetrahedral configuration (Figure

9). Since our pure DFT run does not include a pressure contribution, these results are at 0 GPa. Therefore, our DFT results seem to agree with Ref. [1] for AB stacking. For AA stacking, it is interesting to note that in-plane order is more or less maintained. The ground state energy of ice in AA stacking lower than that of the system in AB stacking by 7.9 eV.

Figure 8: Relaxed structure when starting with a bilayer of ice with AA stacking

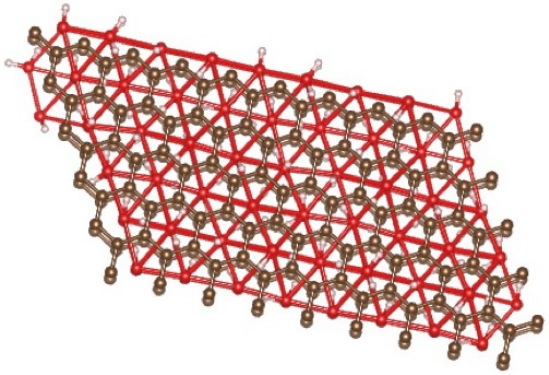
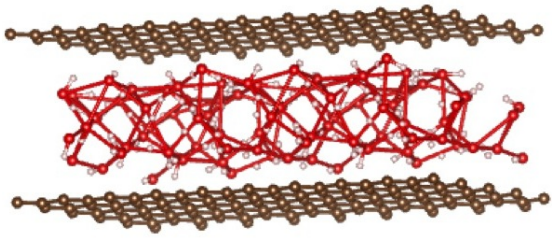


Figure 9: Relaxed structure when starting with a bilayer of ice with AB stacking

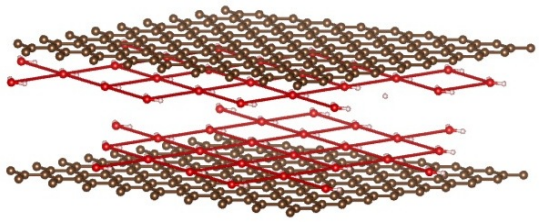


Our CPMD results for AA and AB stacking show that the AA stacked structure is more stable by 0.154 eV. However since we were limited to running the CPMD calculation for a short amount of time, it remained more or less in the cubic lattice that we started with. Comparing this result with the DFT result tells us that the AA stacking is only a local minimum, because the AB stacking in the DFT run eventually relaxes to an amorphous, disordered state with much lower energy.
Hence our results again disagree with Ref [1] with the water molecules in our case seeming to prefer a triangular lattice as opposed to a cubic lattice. An explanation for this discrepancy is given in Section 3.6.

3.5 Quarter Coverage

Since we were able to recover the same results as Ref. [1] in the case of a monolayer of water at quarter coverage, but not at full coverage, we surmise that the same may be true for bilayers. So we did a BOMD run at quarter coverage at different pressures: 0, 1 and 1.5 GPa. The reason we didn't do a CPMD simulation is because we were not able to converge the fictitious kinetic energy of the electron despite many different choices for the fictitious mass and time step.
We again initialize the system in AA and AB stacking. As expected, AA stacking is higher in energy than AB stacking at all pressures by 0.122 - 0.131 eV. Just as in the case of full coverage, the AB stacking disorders at low pressure, including at 1GPa, the transition pressure observed in Ref. [1]. However at P=1.5 GPa, AB stacking maintains in-plane order in a square lattice as shown in Figure 10.

Figure 10: Relaxed structure when starting with a bilayer of ice with AB stacking



3.6 Discussion of Results

Our results indicate that we are in good agreement with Ref [1] at low coverage of water on graphene, but not at high coverage. This can be interpreted as an effect of the Van der Waals forces between water molecules, a virtue of their dipole moment. It is well-known that DFT cannot capture Van der Waals forces very well [6] and our implementation of CPMD did not include empirical Van der Waals interaction potentials.
At low coverage, there is 4-5Ådistance between the water molecules, presumably leading to weak Van der Waals interactions. This explains why we discovered that square ice is more stable than triangular ice, similar to Ref [1].
At high coverage, the maximum separation between water molecules is around 2.8Å, comparable to a separation of 2.7Åbetween water molecules in bulk ice. This presumably leads to strong Van der Waals interactions, which aren't captured by our CPMD calculations. Ref [1] uses an empirical force field to describe the interactions between water molecules. It was however not clear from the work that Van der Waals interactions could have been responsible for the observed structure of ice. Our results seem to suggest that the greater relative stability of square and cubic ice could be an effect of the dipole interactions between the water molecules and could have little to do with the dynamics of

the system.
However, when we simulated a bilayer of water at quarter coverage at different pressures, we were able to recover the same trends as Ref. [1]. As expected, we found that AA stacking was less stable than AB stacking. AB stacking relaxes to a disordered structure at low pressures, just as in the case of full coverage. But at 1.5 GPa, it maintains in-plane ordering in a square lattice. We think that the higher transition pressure that we observe can be explained as follows: At high coverage, we argue that the water molecules are too close, resulting in strong Van der Waals interactions. At low coverage, we circumvent that problem by moving the water molecules far away from each other. But now, they are too far away (compared to the side length of the square lattice in Ref. [1]). So, we think that a higher transition pressure is required to compensate for the larger distance between the water molecules.

4 CPMD implementation for Hydrogen Atom

4.1 SHAKE algorithm in the context of the hydrogen atom

For a hydrogen atom, there is just one Kohn-Sham orbital, ψ which can be expanded in the plane wave basis as: $\psi = \sum_{\vec{G}} c(\vec{G}) e^{i\vec{G}.\vec{r}}$. Our equations of motion are formulated in terms of the expansion coefficients (look at Equation 11).
Since there is just one orbital, there is only one constraint function, given by:

$$\sigma = \sum_{\vec{G}} c^*(\vec{G}) c(\vec{G}) - 1 = 0$$

Also, Equation 18 simplifies to:

$$\lambda(\Delta t^2) = \frac{\sigma^u(t+\Delta t)}{\frac{1}{m}\vec{\nabla}_{c(\vec{G})}\sigma^u(t+\Delta t).\vec{\nabla}_{c(\vec{G})}\sigma^c(t)} \tag{19}$$

The steps for implementing the SHAKE algorithm in this case are:

1. Time evolve the coefficients using Equation 12. The initial Lagrange multiplier is just the DFT energy of the system.

2. From these coefficients, you can calculate a new constraint function (which will not in general satisfy the required constraint). Let's call this

$$\sigma^u(t+\Delta t) = \sum_{\vec{G}} c_u^*(\vec{G}) c_u(\vec{G}) - 1$$

The subscripts u as usual denote the "unconstrained" case.

3. Now, we can calculate the correct Lagrange multiplier using Equation 19. But to do this, we need to know how to compute the gradient of the constraint function with respect to the expansion coefficients. Thankfully this can be done analytically and is just:

$$\vec{\nabla}_{c(\vec{G})}\sigma = c^*(\vec{G})$$

4. So, at every time step we can calculate the correct Lagrange multiplier using:

$$\lambda(\Delta t^2) = \frac{\sigma^u(t+\Delta t)}{\frac{1}{m}\vec{c^*}_{t+\Delta t}\cdot\vec{c^*}_t}$$

. The numerator is obtained as per Step 2. The first term in the denominator is obtained from the coefficients at $t+\Delta t$ (which correspond to an orbital that is not orthonormal in general) and the second term from the coefficients at t (which correspond to an orthonormal orbital). The vector of these coefficients is just a vector in the reciprocal space, i.e. each term of the vector corresponds to an allowed $\vec{G}$ value.

5. Recalculate the coefficients such that they satisfy the constraint using:

$$c^c(t+\Delta t) = c^u(t+\Delta t) - \frac{\Delta t^2}{m}\lambda\nabla_c\sigma(t)$$

for every $c(\vec{G})$. Most importantly note that the σ in this equation is evaluated at t and not $t+\Delta t$ (obviously since we're only now calculating the coefficients at $t+\Delta t$ which will let us compute the new constraint function).

4.2 Description of the Code

For our toy system, the hydrogen atom, there is only one classical ion and one Kohn-Sham orbital with no external constraint. The Lagrangian for the hydrogen atom is

$$\mathcal{L} = \frac{1}{2}\mu\int|\dot{\psi}(\vec{r})|^2 d\vec{r} + \frac{1}{2}M\dot{\vec{R}}^2 - E(\psi,\vec{R}) + \lambda\left(\int|\psi|^2 - 1\right) \tag{20}$$

where λ is the Lagrange multiplier to be determined by SHAKE. The Kohn-Sham energy functional of hydrogen atom only has two contributions, namely kinetic and external potential

$$E[\psi,\vec{R}] = \int\psi^* H\psi d\vec{r} = -\frac{1}{2}\int\vec{\nabla}^2_{\vec{r}}\psi d\vec{r} + \int V_{\text{ext}}(\vec{r})n(\vec{r})d\vec{r} \tag{21}$$

$$V_{\text{ext}}(\vec{r}) = -\frac{1}{|\vec{r}-\vec{R}|} \tag{22}$$

The equations of motions are

$$\begin{cases} \mu\ddot{\psi} = -\frac{\delta E}{\delta \psi^*} + \lambda\psi = -(H\psi - \lambda\psi) \\ M\ddot{\vec{R}}_I = -\vec{\nabla}_{\vec{R}_I} E \end{cases} \tag{23}$$

which can be integrated with the Verlet algorithm

$$\begin{cases} \psi^{n+1} = 2\psi^n - \psi^{n-1} - \frac{(\Delta t)^2}{\mu}(H\psi^n - \lambda\psi^n) \\ \vec{R}^{n+1} = 2\vec{R}^n - \vec{R}^{n-1} - \frac{(\Delta t)^2}{M}\vec{\nabla}_{\vec{R}} E \end{cases} \tag{24}$$

For practical implementation, we have chosen a set of a finite number of plane wave functions commensurate with a given box size of L within a *plane wave cutoff* of E_{cut}. That is

$$\vec{B}(E_{\text{cut}}) = \{b_{\vec{k}}(\vec{r}) = \frac{1}{\sqrt{V}} e^{-i\vec{k}\cdot\vec{r}} \mid \vec{k} = \frac{2\pi}{L^3}(n_1\hat{k}_x + n_2\hat{k}_y + n_3\hat{k}_z); \\ n_1, n_2, n_3 \in \mathbb{N}; \frac{1}{2}|\vec{k}|^2 < E_{\text{cut}}\} \tag{25}$$

This basis set is orthonormal and has the additional benefit of being the only set needed for an extended system with periodic boundary. The ground state electronic wave function as well as the Hamiltonian are expanded in this basis $\vec{B}_1 = \vec{B}(E_{\text{cut}})$. For reason that will become apparent later, we expand density and external potential in a basis with twice the plane wave cutoff $\vec{B}_2 = \vec{B}(2E_{\text{cut}})$

$$\begin{cases} |\psi\rangle = \sum_{i=1}^{N_{b1}} c_i |B_{1i}\rangle \\ H = \sum_{i,j=1}^{N_{b1}} h_{ij} |B_{1i}\rangle \langle B_{1j}| \\ n(\vec{r}) = \sum_{i=1}^{N_{b2}} n_i B_{2i}(\vec{r}) \\ V_{\text{ext}}(\vec{r}) = \sum_{i=1}^{N_{b2}} v_i B_{2i}(\vec{r}) \end{cases} \tag{26}$$

To construct the Hamiltonian, we only need to know the expansion coefficients of the external potential

$$\begin{aligned} h_{ij} &= \langle B_{1i}| -\frac{1}{2}\nabla^2 + \hat{V}_{\text{ext}}|B_{1j}\rangle \\ &= -\frac{1}{2}|\vec{k}_{1j}|\delta_{ij} + \int V_{\text{ext}}(\vec{r}) B^*_{1i}(\vec{r}) B_{1j}(\vec{r}) d\vec{r} \\ &= -\frac{1}{2}|\vec{k}_{1j}|\delta_{ij} + \frac{1}{V}\int \left(\sum_{p=1}^{N_{b2}} v_p \frac{1}{\sqrt{V}} e^{i\vec{k}_{2p}\cdot\vec{r}}\right) e^{i(\vec{k}_{1j} - \vec{k}_{1i})\cdot\vec{r}} d\vec{r} \\ &= -\frac{1}{2}|\vec{k}_{1j}|\delta_{ij} + \frac{1}{\sqrt{V}}\sum_{p=1}^{N_{b2}} v_p \delta(\vec{k}_{2p} - (\vec{k}_{1i} - \vec{k}_{1j})) \end{aligned} \tag{27}$$

It is now clear why the plane wave basis set for external potential needs to be bigger than that for the wave function and Hamiltonian, namely a plane wave function with wave vector $\vec{k}_{1i} - \vec{k}_{1j}$ may not be in $\vec{B}_1$. The expansion coefficients for the external potential can be obtained through a Fourier transform of (22). It is worth noting that once the Hamiltonian is fully written in the plane wave basis, one could directly diagonalize it to obtain the ground state wave function in the plane wave basis, aka the coefficients $\vec{c}_o$ that corresponds to the ground state wave function.

4.3 Relation to Simulated Annealing

To demonstrate the CP method's ability to solve for the electronic ground state wave function, we use an initial guess for the ground state coefficients $\vec{c}$ that is perturbed from those of the true ground state $\vec{c}_o$ by a random vector. Then we evolve the electronic degrees of freedom keeping the ions fixed. This effectively minimizes the Kohn-Sham energy function through a simulated annealing procedure which should reproduce the ground state eigenvalue and eigenvector obtained using DFT. This is indeed what we observed. Figures 11 and 12 show plots of the energy trace and the norm of the electronic orbital in a typical run. The energy trace converges from some higher value to the ground state value of $-0.5Ha$ and the norm of the orbital is kept constant at 1 within 10^{-4}, which verifies the correctness of our implementation of the SHAKE algorithm. The final converged energy is always higher than the DFT ground state because of the finite size of CPMD timestep.

Figure 11: Energy trace of a CPMD run

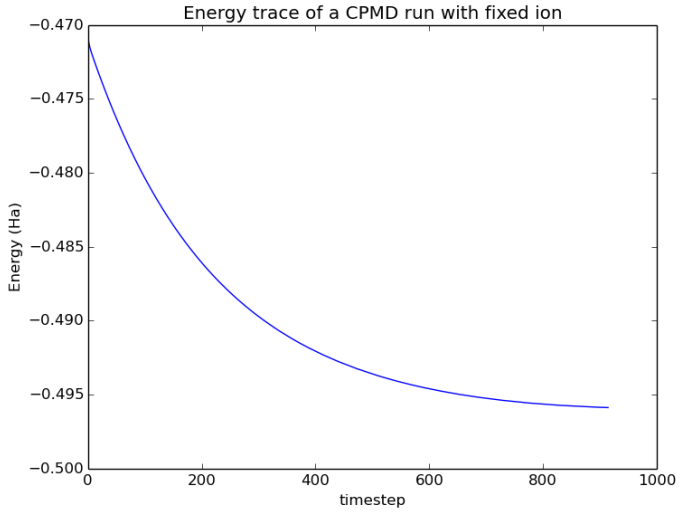


Figure 12: Norm of the hydrogen orbitals during a CPMD run

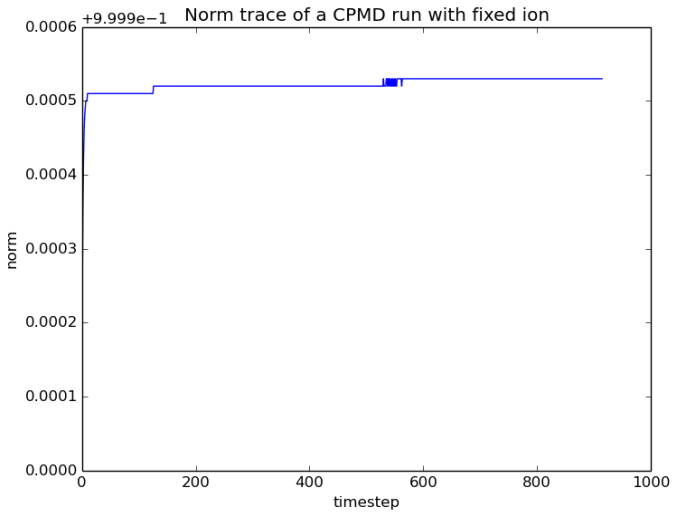
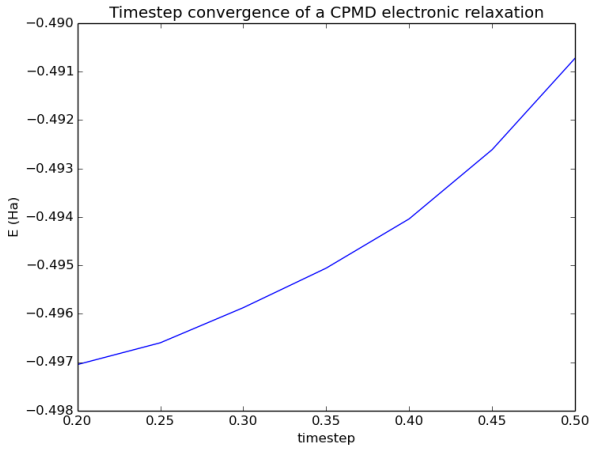


Figure 13 shows the final CPMD energy as a function of timestep demonstrating a converge to the true DFT ground state in the zero timestep limit.

Figure 13: Timestep convergence



5 Conclusions

We simulated a monolayer and bilayer of water between two sheets of graphene, at a quarter and full coverage and discovered that the ground state is a square or cubic ice structure at low coverage (in agreement with Ref [1]), but is triangular or disordered (with local tetrahedral coordination) at higher coverage. However, for a bilayer of water at quarter coverage, we are able to recover the same trends as in Ref. [1]. We reason that this might be a result of the strong Van der Waals interactions between water molecules that are not captured by CPMD or DFT calculations. We also conclude that our results suggest that the dynamics of the water molecules don't necessarily play a huge role in square ice formation between graphene sheets.
We also implemented a CPMD code for a hydrogen atom and demonstrate energy minimization that is effectively equivalent to a Hartree-Fock approach, using the simulated annealing interpretation of CPMD.

References

[1] G. Algara-Siller et al. *Square Ice in Graphene Nanocapillaries*, Nature 519 (2015), pp. 443-457.

[2] W. Ritz, *ber eine neue Methode zur Lsung gewisser Variationsprobleme der mathematischen Physik*, J. Reine Angew. Math., 135 (1909), pp. 1–61.

[3] R. Car and M. Parrinello, *Unified Approach for Molecular Dynamics and Density-Functional Theory*, Phys. Rev. Lett., 55 (1985).

[4] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi, *Optimization by Simulated Annealing*, Science, 220 (2006), pp. 671–680.

[5] J.P. Ryckaert, G. Ciccotti and H.J.C. Berendsen, *Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes*, J. Comp. Phys., 23 (1977), pp. 327–341.

[6] Nature Editorial, *A Model Approach to Modelling*, Nature Materials, 9, 687 (2010).

[7] Richard M Martin, *Electronic Structure - Basic Theory and Practical Methods*, Cambridge University Press (2010).

[8] Daan Frenkel, Berend Smit, *Understanding Molecular Simulations*, Academic Press (2002).