

Tutorial on Path Integral Monte Carlo Simulations of Hydrogen and Warm Dense Carbon

Kevin Driver¹
Burkhard Militzer^{1,2}

University of California, Berkeley
¹Department of Earth and Planetary Science
²Department of Astronomy
307 McCone Hall
Berkeley, CA 94720-4767
kdriver@berkeley.edu
militzer@berkeley.edu

July 24, 2012

Prepared for the University of Illinois at Urbana-Champaign 2012 Summer School on
Computational Materials Science (Quantum Monte Carlo: Theory and Fundamentals).

Overview:

This tutorial covers the basics of using path integral Monte Carlo (PIMC) to study atoms, molecules, and plasmas.

Table of Contents:

- Lesson 1: Hydrogen atom
- Lesson 2: H₂ molecule
- Lesson 3: H₂ triplet state
- Lesson 4: Carbon atom
- Lesson 5: Carbon plasma (static nuclei, PIMC and DFT, 1-atom cell)
- Lesson 6: Carbon plasma (move nuclei, PIMC, 8-atom cell)
- Lesson 7: Hydrogen plasma (move nuclei, PIMC, 52 atom cell)

Obtain files for the tutorial

```
> cd ~/; mkdir PIMCTutorial; cd PIMCTutorial
> cp ~driver/pimc_tutorial.tar.gz .
> tar xfvz pimc_tutorial.tar.gz
```

Note you will need to check the following environment variables for accuracy:

```
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}~/PIMCTutorial/Codes/Cupid/libraries/hdf5-1.6.6/lib
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}~/PIMCTutorial/Codes/Cupid/libraries/sprng-2.0/lib
setenv PATH ${PATH}~/PIMCTutorial/bin
setenv PATH ${PATH}~/PIMCTutorial/scripts
```

The PIMC Code: CUPID

The CUPID code is a PIMC code written in C++. The executables and related analysis tools are located in `Tutorial/bin`. The source code is located at `~/Tutorial/Codes/Cupid`. Executing CUPID without any argument

```
> cupid_serial
```

will reveal details of CUPID keyword commands.

```
> cupid_serial name
```

starts the simulation *name*, which reads input *name.sy* and writes output *name.out*. The statistical averages are written (dumped) into *name.dmp* in HDF5 (Hierarchical Data Format), which can be read by the *h5trace* and *h5analysis* tools in `Tutorial/bin/`. These commands will be automated in scripts provided for ease of use.

Note that the jobs we run below will be submitted from your home directory but run on a scratch directory. If at some point you would like to watch your job running use *cdjob jobID* to go to that scratch directory. Or just wait until the job is completed.

We will use atomic units throughout: $e=a_0=1$

energy: 1Hartree = 27.211396 eV, $1 k_B = 3.166815 \times 10^{-6}$ Ha/K

length: 1 Bohr = 0.52917721 Angstrom

pressure: $1 \text{ Ha/Bohr}^3 = 29421.01 \text{ GPa}$

Lesson 1: Hydrogen Atom

1.1 Isolated Hydrogen Atom

We start with the simplest possible system: the isolated hydrogen atom with the proton at rest. The input file located in `Lesson1/L1.1/cc.sy` reads

```
units Ha a0
```

Use atomic units of Hartrees and Bohr radii a0.

```
boxsize 28.4338
```

The 3dimensional periodically repeated simulation cell with a length 28.4338 a0 in each direction.

```
enorm 1.
```

Divide all energy estimator by 1. (This will be the number of atoms in later examples).

```
beta 40.0
```

Specifies the temperature = $1/k_B T$.

```
nslices 80
```

Use $M=80$ time slices. In conjunction with `,` this determines the time step $\tau = \beta/M$.

```
type p 0.000272308000 0 1 cc.p.ic
```

Define particles: name (proton), $\lambda = \hbar^2/2m_p$, spin*2 (=0 make protons classical), number of particles (here only one proton), file name to read the starting coordinates.

```
type e .5000000000E+00 0 1 cc.e.ic
```

Define particles: name (electron), $\lambda = \hbar^2/2m_e$, spin*2 (leave it at zero since there is only one electron), number of particles (here only one electron), file name to read the starting coordinates.

```
pot pair p e 140.pe.dm 3 3
```

Read the pre-computed electron-proton pair density matrix.

```
pot ewald 1.1911817760 cc.lr
```

```
vtail -0.2050210052
```

```
ptail -0.0000178371
```

Treat the long-range part of the Coulomb interaction in k-space. Note the file cc.lr depends on and must be recomputed before the temperature can be changed.

```
freeze p
```

Do not move the proton.

```
maxblocks 50
```

Run for maximally 50 blocks.

```
RESTART
```

Restart the calculation if an old cc.rs file exists. (If this is commented out all old results will be overwritten.)

```
#CORRSAMPLING
```

By default, the code uses the free particle sampling method to generate the path in the bisection scheme. If the '#' comment is removed the correlated sampling method is invoked.

```
      #Nblocks  Nsteps  level  Nparticles
SELECT    10      50      2        1
```

Parameters: #blocks, #steps per block, #level in bisection, #single particle moves per step (before the code moves to a new window in imaginary time).

The quantities β , τ (time-step), M (number of time-slices), and T (Temperature), are related by the following equation:

$$\beta = M\tau = 1/\kappa_B T$$

Before we do any computations, it will be instructive to get some intuition for how β , τ , and M are related and what temperature range we can cover with our simulations. These quantities will be the most important and often used quantities as we proceed with PIMC calculations. Please fill in the blanks in the following table. The bisection level, L , determines how many adjacent time-slices are changed each move. A path containing 2^L-1 slices is changed for a given level.

τ (Ha ⁻¹)	M	β (Ha ⁻¹)	$\kappa_B T$ (Ha)	T (Kelvin)	Max Possible Bisection Level, L
0.5	80	40.0			
0.5	40				
0.5	20				
0.5	10				
0.5	4				
0.5	2				
0.25		40.0			
0.125		40.0			
0.0625		40.0			
0.0625	2				

Table 1.1: PIMC parameters [1].

Procedure:

1. Edit the `cc.sy` file and fill in the question marks ("?") such that the simulation is run at a temperature of 7894.367 K and a time step of 0.5.

2. Run the PIMC simulation:

```
> qsub cupid.pbs
```

Please note that `cc.si` tells CUPID how many processors you are using. If you change it in `cupid.pbs`, then also change the number in `cc.si`.

and analyze the results in the `.dmp` files. The HDF5 analysis is automated with this script

```
> Block_Analysis.sh
```

The results are block averaged starting at block 30. The proton-electron pair-correlation function, $g(r)$, function can be plotted using

```
> ./plot_gr.sh
```

The statistical error in a Monte Carlo simulation scales as $[\text{Nblocks} \cdot \text{Nsteps}]^{-1/2}$. Note the error bar for the current values of blocks and steps (40 and 500) and increase them to (50 and 5000).

```
> qsub cupid.pbs
```

Did the statistical error decrease by the amount you expected?

Questions 1.1:

1. What do you find for the internal, kinetic and potential energy (E tot, K tot, P tot)? What are the exact values in a.u. [2]?

2. This simulation uses 80 time slices and the precomputed pair density matrix. How many time slices are in principle necessary to obtain an exact answer (after an infinitely long PIMC run) [3]?

3. In classical simulations where protons and electron are point particles interacting via the Coulomb potential, the electron would fall into the core of the Coulomb potential provided by the protons. This is a fundamental problem of Niels Bohr's model for the atom. Why does this not happen here [4]?

4. How is the proton-electron pair correlation $g(r)$ related to the ground state wave function $\psi_0(r)$ [5]? How would $g(r)$ change if the electron mass were doubled [6]?

1.2 Bisection and Correlated Sampling

In order to calculate the kinetic energy more efficiently, several adjacent time-slices must be changed at once. In the previous calculation, we used a bisection algorithm to perform the *multislice* moves already. Check out the acceptance ratios of the previous simulation

```
> grep ratio cc.1.out
```

You should see the acceptance ratios are around 0.8, which is quite good (larger the better).

Go to `../L1.2` and edit the input file to increase the bisection level to 3 and then 4. Record the acceptance ratio in each case: L=3:_____ L=4:_____

Questions 1.2:

1. You should find that the total acceptance ratio dropped as the level increased. Why [7]?

2. Now enable correlated sampling for the L=3 case and reduce maxblocks to 10 so that the job doesn't run too long. How much does it improve the acceptance ratios [8]?

Leave Lesson #1 and head to Lesson #2:

```
> cd ../../Lesson2
```

Lesson 2: The H₂ molecule

2.1 Find the Ground state Separation of R

In the first part of this lesson, we simulate a H₂ molecule in which the protons are held fixed at a location specified in *cc.p.ic*.

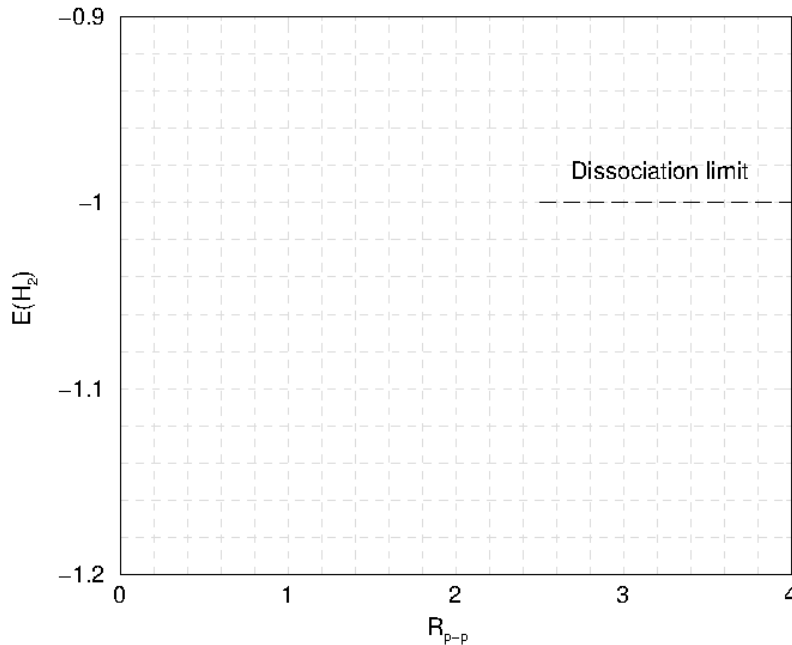


Fig 1. Diagram to plot the energy as a function of proton separation for isolated H₂.

```
> cd L2.1
> qsub cupid.pbs
```

Determine the energy with

```
> Block_Analysis.sh
```

Mark the energy on the plot in Fig. 1 and save your results. You may change the name of the input file to *cc1.sy*, *cc2.sy*, etc., but make sure you change it in the *cupid.pbs* as well. Rerun the simulation for a few different proton positions ranging from 0.5 to 3 Bohr radii until you can estimate the minimum and the dissociation limit.

Questions 2.1

1. What is the equilibrium separation of the protons? What is the binding energy of H₂ [9]?

2. For Coulomb systems, the pressure can be derived from the virial theorem:

$$3PV = 2\langle E_{\text{kin}} \rangle + \langle E_{\text{pot}} \rangle.$$

Looking at your results, does the energy and pressure change sign as you would predict using the virial theorem [10]?

2.2 Move H₂ protons with bisection moves

```
> cd ../L2.2
> qsub cupid.pbs
```

Now we will unfreeze the protons and use the bisection method to generate new paths. After the simulation is done running, check out the proton-proton pair-correlation function:

```
> ./plot_gr.sh
```

Question 2.2

1. The temperature is low enough that the H₂ molecule should be in its ground state and one would expect a clear p-p g(r) peak near the equilibrium location of R=1.5 Bohr. Why do you not see this [11]? (Are the Monte Carlo moves efficient enough?)

2.3 Move H₂ protons with displacement moves

```
> cd ../L2.3
> qsub cupid.pbs
```

Now we repeat L2.2 adding displacement moves for the protons instead of using only bisection moves:

```
displace p 0.1 0.5
```

The displace keyword takes two values: the probability a particle is moved and the distance.

Analyze g(r) again and see if you can find the p-p peak near the equilibrium location.

Questions 2.3

1. Why are displacement moves so much more efficient than bisection moves [12]?
2. Can we forget about bisection moves for protons? How would this change the physics [13]?
3. What determines the time step required for the H₂ simulation? Would 1 slice be enough [14]? (Later in the tutorial we will go through the details of converging time step)

Leave Lesson #2 and head to Lesson #3:

```
> cd ../../Lesson3
```

Lesson 3: H₂ Triplet State

3.1 Restricted Path Integral Monte Carlo Method (free-particle nodes)

In the triplet state of the hydrogen molecule, both electrons are in the same spin state. This is the rest system in which fermionic effects are important and the sign problem shows up because of the electronic exchange effects. In L3.1, we use free-particle nodes to

construct a good and efficient but approximate solution. Run the simulation 'cc.sy'.

Have a look at the e-e $g(r)$:

```
> ./plot_gr.sh
```

Questions 3.1:

1. Determine the total energy. Why is it above -1, the energy of two isolated hydrogen atoms [15]?
2. Can you generalize your observation of the total energy? What can be said about the ground state energy of a fermionic system compared to the corresponding ground state of the system of distinguishable particles [16]?
3. Look at the acceptance ratio (`grep ratio cc.1.out`). What do the two additional "node" accept/reject steps correspond to [17]?

3.2 Direct Path Integral Monte Carlo Method Method (without nodes)

The previous example showed how the nodal restriction leads to an efficient (but approximate) algorithm. Now, let us imagine this method had not yet been developed and we have to run a direct PIMC simulation, in which we record the signs explicitly. This will show you how bad the sign problem can become even for two particles.

```
> cd ../../L3.2
```

In L3.2 you find a cc.sy which runs the same system but without fixed nodes. (Keyword `directFermionMethod`). Run it and compare the $g(r)$ with L3.1:

```
> ./plot_gr.sh
```

Question 3.2:

1. You should find a noisy $g(r)$ function in the direct case, which is sometimes negative. Why is it negative [18]?
2. Would you expect to get exactly the same answer if you ran L3.1 and L3.2 infinitely long [19]?

3.3 Direct Path Integral Monte Carlo Method Method (without nodes, higher T)

Now let us make life easier and change the temperature from 0.025 Ha to 0.5 Ha. This makes the simulations faster and we expect the sign problem should be less severe since we have fewer permutations. To save time, I ran the simulations in L3.3 (direct PIMC) and L3.4 (restricted PIMC) for you. Use the script in L3.4, to plot both $g(r)$'s together for comparison:


```
> ./plot_gr.sh
```

Questions 3.3:

1. Assume the curves are converged. Why are they different [20]?
2. Which one is the correct answer [20]?

Let's leave hydrogen for now and move on to Carbon:

```
> cd ../../Lesson4
```

Lesson 4: Isolated Carbon Atom

4.1 Isolated Carbon Atom With Non-interacting Electrons

To begin our study, we are going to simulate "hydrogen-like" carbon with one, two, and three non-interacting electrons to ensure we get what we expect as we build up our calculations for heavier atoms.

There are three sub-directories: *One_electron*, *Two_electrons*, and *Three_electrons*. Submit the jobscripts in each

```
> qsub cupid.pbs
```

and analyze the Energy when each is done using

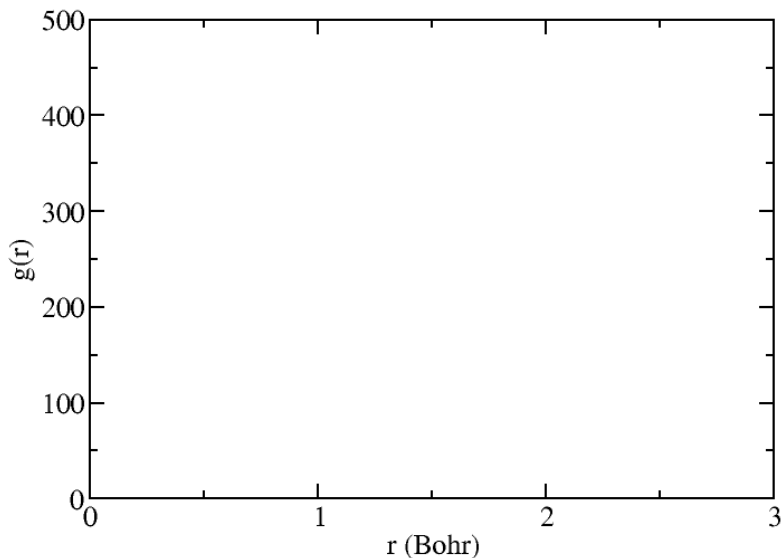
```
> Block_Analysis.sh
```

Questions 4.1:

1. What spin state is specified in the *cc.sy* file [21]?
2. Is the temperature low enough to find the electrons in the ground state [22]?
3. What energies do you get for each of the three cases [23]? Fill in the table below containing the analytical and your computed values. (Hint: What are the energies for a hydrogen-like atom with $Z=6$ and successive electrons in 1s, 2s, and 2p orbitals? Use atomic units: $e=a_0=1$ and recall $E_n \sim Z^2/2n^2$ in atomic units.)

No. of electrons	Analytical Energy (Ha)	PIMC Energy (Ha)
1		
2		
3		

4. Please make a sketch of the expected $g(r)$ functions in the diagram below.



Plot the c-e $g(r)$ in each case and see if your sketch is correct.

```
> ./plot_gr.sh
```

4.2 Isolated Carbon Atom With Interacting Electrons

We now study the simulation of an isolated carbon atom with six interacting electrons. We will use this simulation to converge the time-step ($\tau = \beta/M$), which we will use for a periodic simulation in the next lesson.

```
>cd ../L4.2
```

Before starting, have a look at the files in the directory *tau_0.00390625/*:

We now have a e-e density matrix file in addition to a c-e density matrix file (*631.ee.dm* and *631.ce.dm*). Check that these are referenced in the input file appropriately. The *c.ic* and *e.ic* files contain the initial carbon and electron positions, respectively. The carbon nucleus is held fixed in these simulations to save time. The input file is *cc.sy* as usual.

Since the density matrix files have been made for you, the specific time-steps at which you may run calculations have also been preselected. To reveal the time steps at which you can perform simulations, `grep "beta" in the .dm file`:

```
> grep beta 631.ee.dm
```

How many available time-steps do you see? How are they related?

Your task is to get the energy at the six time-steps for which directories are set up for you:

tau_0.00390625 -- completed for you
tau_0.0078125 -- completed for you
tau_0.015625 -- for you to run
tau_0.03125 -- for you to run
tau_0.0625 -- for you to run
tau_0.125 -- for you to run

In order to converge the time-step, we fix the temperature to at $\beta=2.0 \text{ Ha}^{-1}$ ($T \sim 157887.34 \text{ K}$). To be unbiased, we must keep the length of the path that the bisection moves modify the same by adjusting the level appropriately.

In order to get an intuitive feel for the role of the bisection level, please complete the drawing in Fig. 4.2.1. The vertical axis in the figure is imaginary time. In the end, each of the four panels should contain a path with a hypothetical bisection drawn as a dashed line. In the first three panels, the temperature is held constant, while time-step, τ , and number of slices vary according to $\beta=M\tau$. In these three panels we want the length of path modified by the bisection move to remain the fixed. Your job is to figure out what level is needed in the third panel to accomplish that goal and then draw a possible bisection path as a dashed line. Finally, in the fourth panel, we have changed the temperature (i.e. the total length of the path). Draw the appropriate path and a possible bisection move for the parameters given. (Remember the level changes 2^L-1 slices.)

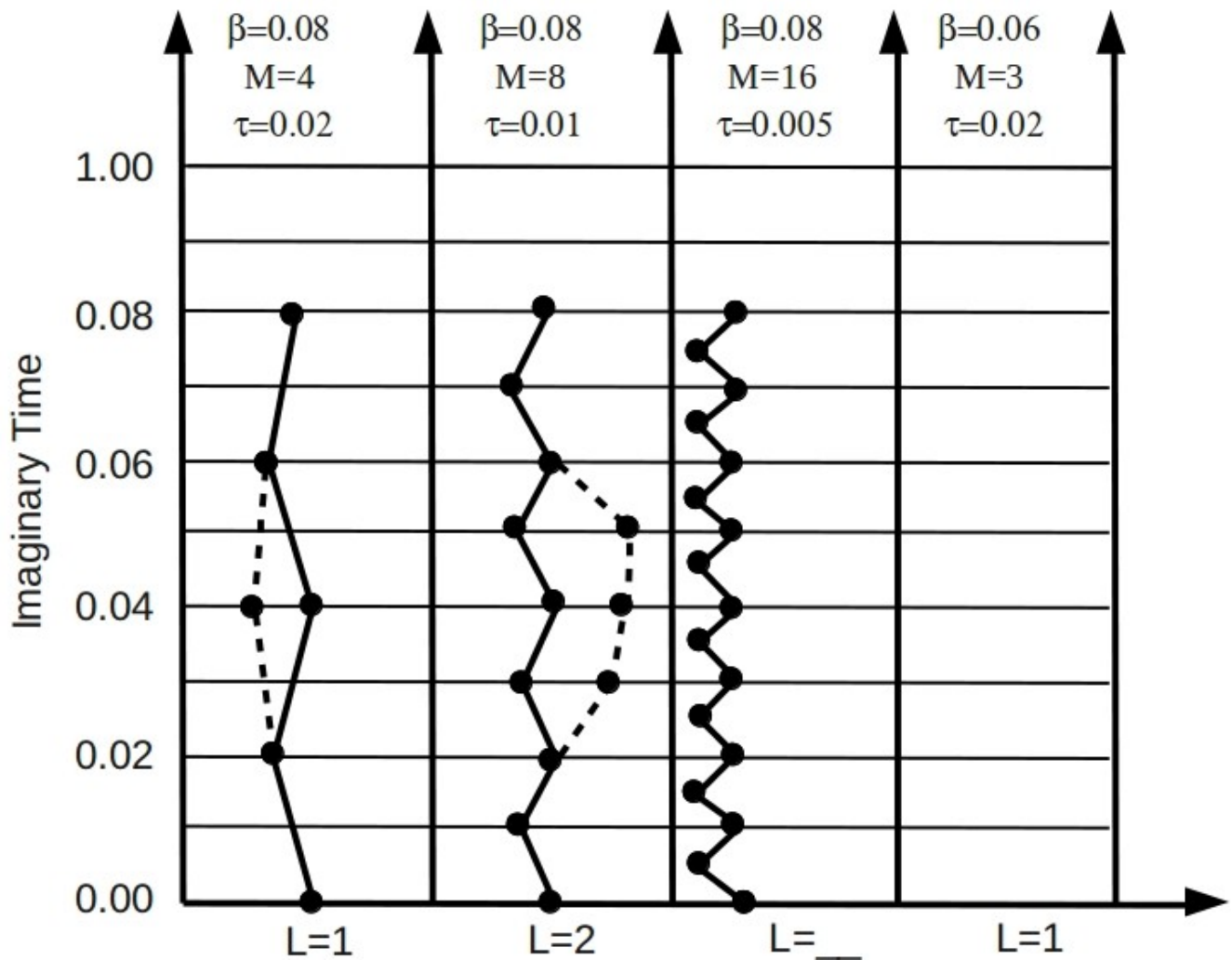


Figure 4.2.1: Path variation with bisection level and temperature.

Now prepare for the calculations by filling in the rest of the following table:

τ (Ha^{-1})	β (Ha^{-1})	Number of Slices, M	Bisection Level, L
0.125	2.0		1
0.00390625			

Table 4.2 Parameters need to converge the PIMC time-step.

Due to time constraints, the results for the smallest two time-steps are completed and provided to you. Your task for the other four time-steps is to complete the following procedure:

1) Edit the input file and fill in the question marks ("?") with appropriate values for β , M , and the level in each simulation. The number of blocks and steps have been set to reasonable values for you.

2) Submit the jobs and record the energies in a text file:

```
> qsub cupid.pbs  
> Block_Analysis.sh
```

Questions 4.2:

What time-step is converged [24]?

Do you expect that energy we converged to is accurate [25]? (Think about ground state nodes)

Leave Lesson #4 and head to Lesson #5:

```
> cd ../../Lesson5
```

Lesson 5: Carbon Atom in Periodic Boundary Conditions at High Temperature

Our goal is to simulate a fully interacting carbon plasma in periodic boundary conditions. However, for the sake of saving time in this tutorial, we limit the cell size to only one. You will have a chance to run a 8 atom cell later if you have time.

5.1 PIMC Simulations

The main goal of this section is to compute the pressure versus temperature equation of state of carbon for temperatures ranging from 0 to 50 million Kelvin by combining PIMC and DFT simulations. Given that PIMC uses free-particle nodes, it is only expected to be accurate for temperature regions where sufficient thermal ionization occurs. The main question to answer is how low in temperature can we go with free-particle nodes?

```
>cd L5.1
```

Have a look at the files in the directory *500thousandK/*. Notice we now have three types of density matrix files for c-c, c-e, and e-e interactions. Check that these are referenced in the input file *cc.sy* appropriately. The initial carbon and electron positions are set in *c.ic* and *e.ic*, respectively. The carbon atoms are fixed in order to save time.

We'll start out using a time-step of $0.0078125 \text{ Ha}^{-1}$. The simulations at temperatures of ~ 100 -, ~ 150 - and ~ 250 -thousand K are completed for you to save time.

Note once you get to a temperature of ~ 5 million K, you will have to decrease the level to accommodate for the smaller number of slices. This is fine since the temperature is becoming quite large and our system becoming more classical-like. At ~ 10 million K you should decrease the time-step by a factor of 2 to keep at least 8 slices. At ~ 50 million K try to figure out how to

set up the calculation on your own.

In preparation for the calculations, please fill out the following table for the additional temperatures of ~500,000, ~750,000, ~1 million, ~5 million, ~10 million, and ~50 million K. The exact temperature will have to be determined based on the fixed time step. Recall $\beta=1/k_B T=M\tau$.

T(K)	τ (Ha ⁻¹)	β (Ha ⁻¹)	M	Level, L
100047.424	0.0078125	3.15625	404	5
149700.590	0.0078125	2.1093750	270	5
249500.983	0.0078125	1.2656250	162	5
	0.0078125			5
	0.0078125			5
	0.0078125			5
	0.0078125			5

A directory for each temperature has already been set up for you. Perform the following procedure:

1. Edit the input file and fill in the question marks ("?") with appropriate values for β , M, and the level in each simulation. The number of blocks and steps have been set to reasonable values for you to complete the simulation quickly.
2. Submit the jobs and record the energies in the text file I've started for you called *PIMC_PvsT.dat*.

```
>qsub cupid.pbs  
>Block_Analysis.sh
```

5.2 DFT Simulations

We will now run the ABINIT code to compute pressure versus temperature for a series of temperatures less than 1million K and see how the result match up with what we just computed in PIMC. The temperatures you are to run are already set up in separate directories as in the previous exercise. The 1million K run is completed for you already to save time.

Before starting, take a look at the files in one of the directories. There is a *input* file, which has the cut-off energy, k-points, structural information, number of bands and various parameters for

convergence and temperature. There is a *files* file, which ABINIT uses to identify the name of the input files. There is also a pseudopotential *C.fhi*, which was constructed with the Opium code.

Just for your awareness, the key features for running a $T > 0$ DFT simulation are the following:

- a) Use Fermi smearing with $tsmear = k_B * T = (3.166815 * 10^{-6} \text{ Ha/K}) * T$.
- b) Set *nband* large enough so that the occupation of the highest orbital is less than $\sim 10^{-4}$. (Check *occ* at the end of the output file)
- c) The energy cutoff of 30 Ha has already been converged for you.
- d) We run at a single *k*-point to compare with PIMC, which is accurate at high temperatures.

We ask you to perform only the following tasks for each simulation:

1. Edit the input file and fill in the question mark ("?") for the Fermi smearing temperature, *tsmear*. It should correspond to the temperature used in the PIMC simulation. The simulations you are to do are for temperatures of ~ 50 -, ~ 100 -, ~ 150 -, ~ 250 -, ~ 500 -, ~ 750 -thousand K.

2. Submit the jobs and record the pressures in the text file I've started for you called *DFT_PvsT.dat*.

```
>qsub abinit.pbs
>grep Pressure output
```

Finally, we will plot the PIMC and DFT results along with the classical Debye-Huckle results provided for you in *L5.1/*. I've set up a script for you that should work provided you are sitting in *L5.2/* and named the files as specified (*L5.1/PIMC_PvsT.dat* and *L5.2/DFT_PvsT.dat*).

Examine the results:

```
>./plot_PvsT.sh
```

To see that PIMC converges to the classical Debye-Huckle result, you will need to zoom out.

Questions 5.1 and 5.2:

Why are lower temperatures more time consuming and difficult in PIMC [26]?

Do the PIMC and DFT equations of state agree? At which temperature do they start to disagree? What does imply for the underlying physics of the PIMC simulation [27]?

Why can't we simply use DFT for all temperatures? What is the computational limitation of DFT [28]?

Do exchange-correlation functions work at high temperature [29]?

Lesson 6: PIMC Simulations of Carbon with 8 Moving Nuclei (optional)

In the previous lesson we froze the ions positions to save time. In this lesson we give you the opportunity to run a production-quality calculation in which the ions are treated as quantum

particles and undergo molecular dynamics in a 8-atom simulation cell. The task is to perform a PIMC simulation at $\sim 750,000$ K and compare with the DFT-MD result, which is provided.

```
> cd Lesson6
> cd PIMC
```

Take a look at the input file `cc.sy`. Notice we are using displacement moves:
`displace c 0.15 0.06`

The values are chosen such that the acceptance ratio is at least 50% or larger.

You will also see we are using a new keyword called `gamma`:

```
gamma e 3 1.0
gamma c 1 0.05
```

This keyword affects the quantum nature of the particle. The first number sets the number of permutation weave moves. The second number sets the frequency such moves are made. Doubling the number doubles the frequency of moves.

Fill in the 6 spaces labeled with a "?" in the `cc.sy` so that they are appropriate for a 8-atom calculation at $\sim 750,000$ K.

The simulation is started for you because we will not have enough time to run from scratch. The RESTART flag is set in the `cc.sy` file to give you an opportunity to run the simulation also. The simulation will run for 30 minutes. If you don't have enough time, simply analyze the results provided.

```
> qsub cupid.pbs
```

Questions 6:

1. The DFT pressure for the corresponding system is 9300 GPa. Does your PIMC result match [30]?

Lesson 7: PIMC Simulation of Hydrogen Plasma with 54 Moving Nuclei

7.1 Plasma state

In this lesson we simulate a hot, dense hydrogen plasma using a 54-atom periodic simulation cell.

```
> cd L7.1
> qsub cupid.pbs
```

Examine the input file and determine the simulation's temperature. Try to identify the state of hydrogen under these conditions by plotting the $g(r)$ functions.

```
> ./plot_gr.sh
```


Questions 7.1

1. What can you tell about the interaction for the different pairs of particles (pp, pe, ee) from the $g(r)$ [31]? Is there any indication for the formation of atoms or molecules [32]?
2. There are 2 electron-electron pair correlation functions: one for parallel spin and one for anti-parallel spin. Why are they different [33]? What would they look like if all interaction effects but not spin statistics were switched off (ideal plasma) [34]?

7.2 Formation of bound states

```
> cd L7.2
```

This simulation is same as the previous one except the temperature is much lower. See if you can find any indication for bound states. The simulation will take too long to run in this lab, so I've run it for you and provide the results for you to analyze.

```
> ./plot_gr.sh
```

Answers to the Questions

- [1] Check your table values with your neighbor.
- [2] $E_{\text{tot}}=-0.5$, $K_{\text{tot}}=+0.5$, $P_{\text{tot}}=-1.0$
- [3] 1, since the pair density matrix is the exact solution of the two particle problem.
- [4] Because of Quantum Mechanics, which is correctly incorporated in the pair density matrix
- [5] At $T=0$, $\rho(R,R) \propto |\psi(r)|^2 \rightarrow g(r) = |\psi(r)|^2$
- [6] The electron would be closer to the nucleus.
- [7] Interaction effects modify the distribution of paths (more so at lower the temperature). Consequently the acceptance ratio for each level drops by a certain amount if we use the free particle distribution to sample the paths.
- [8] About 20%.
- [9] $R_{\text{eq}}=1.4$ Bohr. $E_{\text{bind}}=E_{\text{eq}} - E_{\text{diss}} = -1.17 - (1) = -0.17$ Ha.
- [10] Yes: $2\langle E_{\text{kin}} \rangle + \langle E_{\text{pot}} \rangle$ should be positive for $R < 1.4$ and negative for $R > 1.4$.

[11] Protons are 1836 times heavier than electrons. Their paths are like rods. The bisection moves can only move by a small amount proportional to the $\text{Sqrt}[2^L \lambda_p \tau]$.

[12] It's more efficient to displace the whole paths at once for the rod-like path.

[13] This would be equivalent to the Born-Oppenheimer approximation in which the nuclei are infinite in mass and classical.

[14] Correlation effects beyond the 2 particle interaction are present in H_2 . We need many time slices to incorporate those exactly.

[15] $E = -0.778(3)$ Ha. E is larger than $E_{\text{diss}} = -1$ Ha because the triplet is an anti-bonding state.

[16] The ground state energy of a fermionic system is always above (or equal) in energy to that of a corresponding system of Boltzmannians.

[17] $\text{ratio} \approx 0.8$. The fermion node is enforced in two steps: 1) check for a node crossing at a bead and 2) check for the leaking probability between beads.

[18] $g(r)$ represents a probability and should never be negative. However, in the direct PIMC method, positive and negative contributions enter. Because of the statistical noise, the negative do often outweigh the positive ones.

[19] No, do the following exercise to understand why.

[20] If we ran long enough, we eventually would sample all the negative and positive pockets accurately and DPIMC gives the exact answer. RPIMC is approximate (but more efficient). This explains why the two curves are different.

[21] All electrons are spin-up in each case.

[22] Yes. $\beta = 3.15$ corresponds to $T = 100,000$ K. Therefore, $(3/2)k_B T = 0.475$ Ha, which is much less than, $E_{n=1} = -18$ Ha and $E_{n=2} = -4.5$ Ha.

[23] 1e: $E = -18$ Ha, 2e: $E = -18 - 4.5 = -22.5$, 3e: $E = -18 - 2 * 4.5 = -27$

[24] $0.0078125 \text{ Ha}^{-1}$

[25] Since we are using free-particle nodes, we should not expect that we are describing ground state nodes correctly.

[26] The paths representing the particles are grow longer as temperature decreases. The number of time-slices grow larger and paths become more time consuming to simulate. Challenges of sampling paths efficiently and ensuring ergodicity become amplified.

[27] PIMC and DFT agree in the range 100-750,000 K. DFT breaks down above 750,000 K due to the pseudopotential inadequately describing core excitations. The PIMC calculations are all-electron. PIMC perhaps agrees at lower temperatures, but calculations grow more costly and unlikely to agree with DFT due to the use of a free particle nodal structure.

[28] In addition to pseudopotentials breaking down with increasing temperature, the degree of excited state occupation becomes unwieldy and too computationally expensive to proceed much further than 1million K.

[29] Cross-validated by the PIMC results, exchange-correlation functions appear to work perfectly fine at high temperature, even though they were designed for 0 K calculations.

[30] yes (as we should expect from Lesson 5)

[31] Protons repel each other as do the electrons. One finds that electrons are attracted to protons but they cannot bind at this temperature.

[32] No.

[33] The Pauli exclusions principle leads to an extra repulsion between same-spin electrons.

[34] For electrons the with opposite spin $g(r)$ is a constant equal 1. For electrons the with the same spin is must go to zero for small r .