Particle Simulation I: Assignment II

Analysis of molecular dynamics trajectories

In this assignment the ouput of a one-dimensional Lennard-Jones particle simulation was analysed to determine the behaviour of the system as a function of T. From these results the constant volume specific heat capacity (C_v) was calculated by graphical means and from the microcanonical fluctuation formulae. These results were then compared with those predicted by theory, leading to the discovery that while most of the results agreed well with theoretical predictions, the fluctuation approach to calculating C_v was somewhat flawed. Suggestion for possible explanations and improvements for this approach are given in the conclusion.

Introduction

The aim of this assignment is to assess my ability to use a molecular dynamics simulation and to analyse the data it produces. The specific case investigated here is the behaviour of a one-dimensional Lennard-Jones substance at a range of temperatures. I have been given the basic FORTRAN code to carry out this simulation (from Molecular Dynamics Simulation by J M Haile (Wiley, 1992)), and I am free to alter the code in any way I see fit in order to assist the analysis of the data.

The first step in the analysis is to run a series of simulation in order to determine a reasonable time step for the finite difference calculation. Once an acceptable accurate and stable time step is found, it will be possible to examine the following unitless variables at equilibrium:

- $<U^*>$ - The average internal energy.
- $<P^*>$ - The average pressure.
- $<E^*>$ - The average total system energy (should be equal to the instantaneous system energy, E^*.

By finding these equilibrium parameters for a range of temperatures it should be possible to describe the behaviour of the system as a function of $<T^*>$, and from this information to work out the dimensionless isometric heat capacity, C_v^*. This can also be calulated from the fluctuations in the system at equilibrium, via the fluctuation formulae given in the Particle Simulation I lecture course.

An estimate of the errors in my results is also required.

Theory

• *Molecular Dynamics code:*

While it is not the aim of this assessment for me to explain the inner workings of the MD simulation, it is wise at this point to explain what information the code needs, and what information it generates (the more general aspects of the simulation have already been explained in the introduction).

INPUT:
- NATOM • number of spheres in the simulation (N = 500).
- STEP • time-step ($\Delta t^* \sim 0.004$).
- TEMP • set-point temperature ($T^* \sim 1.2$).
- DENSTY • number density ($N\sigma^3/V = 0.65$).
- MEQSTP • number of time-steps to execute during equilibration (2000).
- MAXSTP • number of time-steps to execute during data production (10000).
- KWRITE • time-step increment at which to output data to file (20).

The temperature range to be investigated is the range of T^* from 1.0 to 2.5 in steps of 0.1. Some pilot experiments should also be performed to find a satisfactory value of Δt^*, although the value of 0.004 which the program already uses should work reasonably well. This part of the analysis is carried out by altering the time-step and seeing at what level it starts giving results that deviate from those results at much smaller time-steps. Whilst doing this it is important to remember that the number of equilibration steps (MEQSTP) must increased at Δt^* is decreased, and vice-versa, so that the same amount of simulation time has passed during the equilibration stage. If this is not done then small time-steps lead to simulations that think equilibriation has been given enough time, when in fact it hasn't, and is still trying to settle down. To do this I must assume that Haile is correct in giving 2000 steps as being sufficient time for equilibration to occur give that $\Delta t^*=0.004$, meaning that I can use the following relation:

$$MEQSTP = 2000 \times 0.004/STEP$$

ie a STEP of 0.002 performs 4000 iterations during equilibration.

OUTPUT:
- Every KWRITE steps the program writes the following data to a file:
   T* U* P* E*
   ie the dimensionless form of the temperature, internal energy, pressure and total system energy, all written on one line separated by spaces.

The program originally gave the running averages of these variables as well, but I decided to make my analysis program use the instantaneous values to calculate it's own average (along with the standard deviation error) and so these running averages were not require. This alteration also enabled me to make the program output the results to more decimal places than before whilst keeping the data in neat tables whose rows were short enough to be displayed easily on the screen. Previously, having more decimal places would have forced the rows to be too long to be displayed on the screen in an easily understandable format.

I also modified the program so that output to the file only occured after the equilibration stage, as the data during equilibration is also not required.

• *Analysis of the Data:*

Once the MD code has been executed for the range of T* indicated above (ie 1.0-2.5 in steps of 0.1), the next step is to write an analysis program to read in the data files given by the MD code and calculate <E*>, <U*>, <P*> as a function of <T*>. To do this, my program (see Appendix A) simply needs to load in the data table and take the average of the numbers in each column. While the program is doing this, it can also calculate the standard deviation. For a set of values for a variable X, the mean ($\bar{X}$) and standard deviation (δX) are calculated as follows:

$$\bar{X} = \frac{\sum_{i=1}^{N} X_i}{N}$$

$$(\delta X)^2 = \overline{X^2} - \bar{X}^2$$

where

$$\overline{X^2} = \frac{\sum_{i=1}^{N} X_i^2}{N}$$

The data for a particular temperature is then appended to 3 output files simultaneously, each file having the form:

$$<T^*> \qquad <F(<T^*>)> \qquad \delta F(<T^*>)$$

Where the three files refer to <E*>, <U*> and <P*> as a function of <T*>.

Calculation of C_v^* is to be carred out in two ways, the first is to fit a line to the data in the file which describes <E*> as a function of <T*>, the gradient of which is equal to C_v^*. The second method uses the fluctuation formula mentioned before. This has the form:

$$C_v^* = \frac{3}{2}\left[1 - \frac{2}{3NT^{*2}}<(\delta U^*)^2>\right]^{-1}$$

Where δU^* is the standard deviation error in U* calculated by my analysis program (as outlined above). This also has a form based on the kinetic energy fluctuation, δK^*, but this is not implemented here. These C_v calculation were carried out by a modified version my analysis code, and as the modification were only slight this second program is not included in the code appendix.

• *Comparison of behaviour with thermodynamic results:*

The internal energy of the system can be described by:

$$dU = dW + dQ$$

Given that we do no work on the system, and only increase the temperature at constant volume:

$$dU = dQ = |dE/dT|_v \, dT = C_v \, dT$$

integration wrt T yields:

$$U = C_v T + c$$

where c is a constant of integration. In other words we should find that $<U*>$ varies linearly with $<T*>$.

The Lennard-Jones potential creates a system that is closely approximated by the Van der Waals gas equation, ie:

$$(p+a/V^2)(V-b) = nkT$$

By inspection it is clear that for constant volume, $<P*>$ should be linearly related to $<T*>$.

Assuming that the Lennard-Jones potential is not vastly different from an ideal gas, it is possible to estimate the value of C_v*. For a one-dimensional system:

$$E \sim \tfrac{1}{2}kT$$

Converting to our dimensionless simulation variables:

$$E* \sim \tfrac{1}{2} T*$$

And as $C_v* = |dE*/dT*|_v$:

$$C_{v*} \sim \tfrac{1}{2}$$

The results of the simulation can be checked against these theoretical results in order to ascertain the value of the model.

• *Estimation of Errors:*

The output from the analysis program already gives an error for the data for $<E*>$, $<U*>$ and $<P*>$ as a function of $<T*>$, and so this can be quoted in the results. For the linear fit to the $<E*>$ against $<T*>$ data, the curve fitter I will use will take into account the given errors in $<E*>$ and given an error margin for the linear fit, thus giving an error for C_v*. All that remains is to find a way of estimating errors for the fluctuation method for C_v*. If a linear fit is a good fit for the $<E*>$ against $<T*>$ data, then that means that C_v* is constant over the temperature range (because the curve has a constant gradient). This implies that the range of values for $dU*$ as a function of $<T*>$, which I have already calculated, can be used to create a range of estimates for C_v*, from which the mean and standard deviation error can be taken. If the linear fit is poor, no estimation of the error in the fluctuation formula result is possible.

Results

1 • Identifying a satisfactory time-step:

I tested a rang of time-steps and recorded the value of <U*> after 10,000 production steps, expecting the internal energy to settle down to a particular value as the time-step was decreased. Table 1 below shows the actual results obtained:

Time Step	<U*> after 10,000 steps
0.1	no result
0.05	no result
0.01	-0.4379
0.005	-0.4363
0.001	-0.4345
0.0005	-0.4370
0.0001	-0.4352

Table 1: Identifying a satisfactory time-step.

From this information it is clear that the simulation is either drastically unstable, leading to no results at all (ie all columns in the data are lines of asterisks), or reasonable good. All the simulation which have results are giving roughly the same answer. As the simulation took little time to execute, I decided to use a time-step of 0.0001, as there is no harm in having this extra accuracy if there is no real decrease in simulation speed.

2 • Behaviour of the system as a function of <T>:*

The behaviour of the system, in terms of <U*>, <P*> and <E*>, is illustrated by figures 1, 2 and 3 respectively.
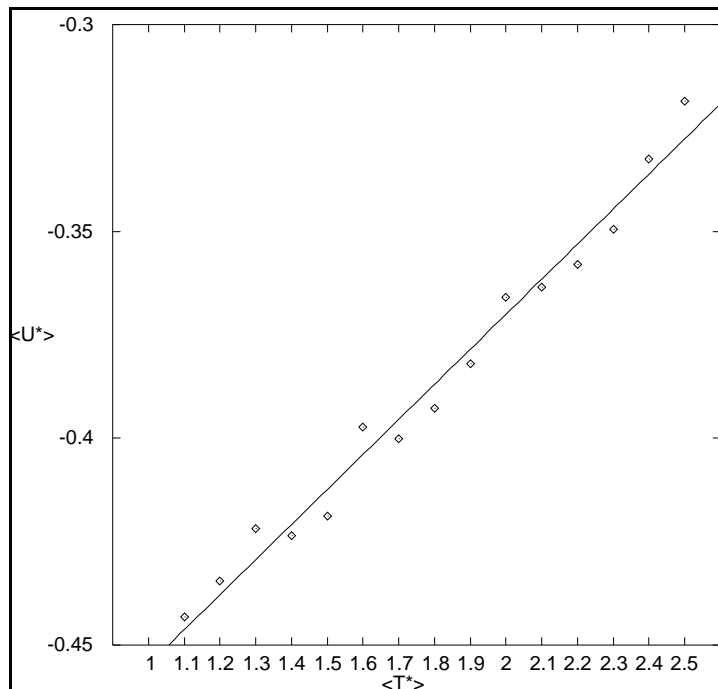
Figure 1: <U*> as a function of <T*>.
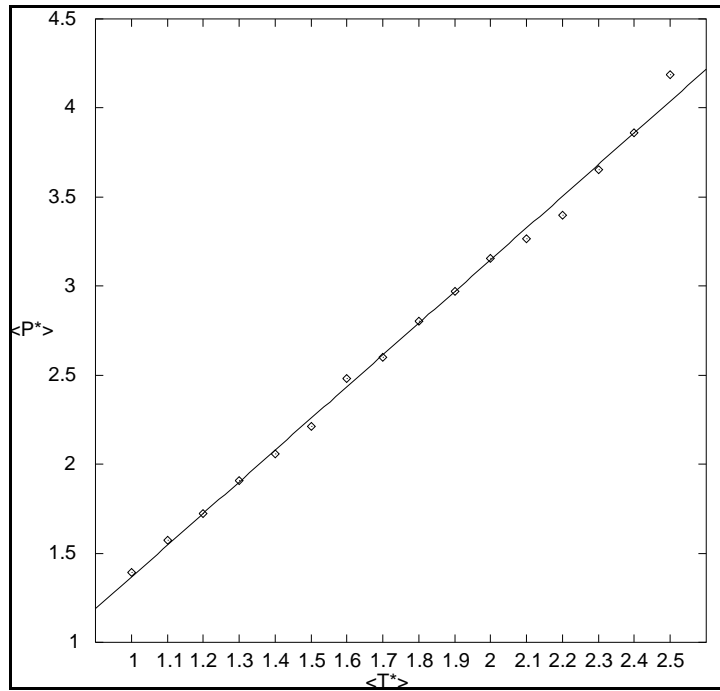
Figure 2: <P*> as a function of <T*>.
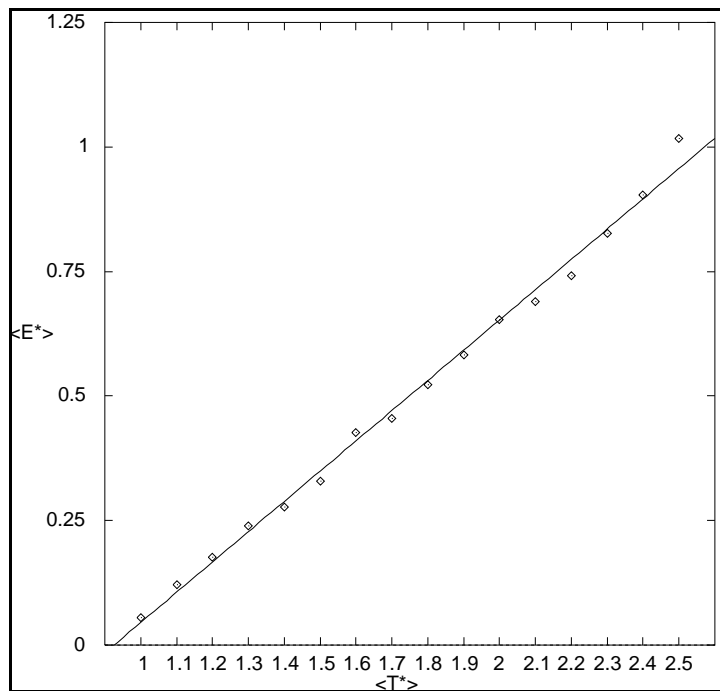
Figure 3: <E*> as a function of <T*>.

I have used the data to fit a linear curve for all three sets of data, and the linear approximation is a good one, as can be seen from the above figures. This means that the internal energy, the total energy and the pressure all vary linearly with respect to temperature as is expected from the theory.

3 • Calculation of C_v*:

The value of C_v* is first determined from the gradient of the $<E*>$ against $<T*>$ graph. The linear fit in figure 3 is described by the equation:

$$<E*> = (0.58753\pm1.9\times10^{-8}).<T*> - (0.53069\pm2.2\times10^{-8})$$

where the errors given are based on the error bars of the values of $<E*>$ as defined by the standard deviation from the set of instantaneous values $E*$.

This means the curve fitting method leads to a values of $C_v* = 0.58753$, with a very small error (of the order of 3.2×10^{-6}%). This result compares favourably with the expected result of 0.5.

As we have seen, the behaviour of $<E*>$ wrt $<T*>$ is linear, and so it is possible to determine C_v* via the fluctuation formula. Table 2 shows the results I took for C_v* for the whole range of $<T*>$.

$<T*>$	Cv
1.0	1.50000043579
1.1	1.50000032629
1.2	1.50000031437
1.3	1.50000030744
1.4	1.50000026027
1.5	1.50000032680
1.6	1.50000028626
1.7	1.50000031153
1.8	1.50000031482
1.9	1.50000025373
2.0	1.50000031346
2.1	1.50000026297
2.2	1.50000021669
2.3	1.50000023922
2.4	1.50000027161
2.5	1.50000032311

Table 2: C_v* for the range of $<T*>$..

Averaging of these results lead to:

$$C_v* = 1.5000002978 \pm 0.0000000463$$

If the theory is correct then this is clearly wrong, and something must be going wrong with the fluctuation formula calculation.

Discussion/Conclusion

While the majority of the results clearly agree well with the theory, the fluctuation approach is clearly flawed. This is in part due to the equation being that for a three dimensional Lennard-Jones system, and so is not suitable for our one-dimensional simulation. Correcting this would mean that the fluctuation formula should have the form:

$$C_v^* = \frac{1}{2}\left[1 - \frac{2}{NT^{*2}}<(\delta U^*)^2>\right]^{-1}$$

At this I should point out that I do not know if this is correct, and that I have not been able to verify the one-dimensional micro-canonical ensemble derivation.

While this correction would give values in the right range, the fluctuations (δU^*) would still be far to small to push the value of C_v^* far from 0.5. This implies that the system is fluctuating in the same manner as an ideal gas, although the curve fitting result of 0.59 indicates the system is not ideal. This in turn leads to the conclusion that while the formation of equilibrium occurs in an non-ideal manner (leading the non-ideal behaviour diplayed in the $<E^*>$ against $<T^*>$ plot), the non-ideal characteristics somehow cancel out once the system has settled down, leading to ideal fluctuation behaviour.

Appendix A: FORTRAN data analysis code written for this assignment.

```
        program accessfile
        character*12 userstr
        character*24 filename
        integer c,r,i,n
        double precision data(10,1000),temp,sum(10),sumsqr(10)
        double precision mean,var
        OPEN (UNIT=21, FILE='t.dat', STATUS='APPEND')
        OPEN (UNIT=22, FILE='u.dat', STATUS='APPEND')
        OPEN (UNIT=23, FILE='p.dat', STATUS='APPEND')
        OPEN (UNIT=24, FILE='e.dat', STATUS='APPEND')
 666    FORMAT(A)
        WRITE(*,*) 'Enter file to access: '
        READ(*,666) userstr
        c=4
        filename=userstr//'              '
        CALL lfDP(filename,data,c,10,r,1000,temp)
c zero summation array:
        DO n=1,c
         sum(n)=0.0d0
         sumsqr(n)=0.0d0
        ENDDO
c sum up things:
        DO i=1,r
         DO n=1,c
          sum(n)=sum(n)+data(n,i)
          sumsqr(n)=sumsqr(n)+data(n,i)*data(n,i)
         ENDDO
        ENDDO
c show mean and varience and STD:
        WRITE(*,*) 'At temperature: ',temp
        DO n=1,c
         mean=sum(n)/r
         var=ABS(sumsqr(n)/r-mean*mean)
         WRITE(*,43) n,mean,SQRT(var)
         WRITE(20+n,44) temp,mean,SQRT(var)
        ENDDO
  43    FORMAT('column:',I4,', mean:',F15.10,', std:',F15.10,'.')
  44    FORMAT(F8.1,2F15.10)
c
        CLOSE(UNIT=21)
        CLOSE(UNIT=22)
        CLOSE(UNIT=23)
        CLOSE(UNIT=24)
        stop
        end
c
        subroutine lfDP(filestr,A,c,cp,r,rp,ref)
        integer c,cp,r,rp,n
        character*12 filestr
        character*128 equstr
        double precision A(cp,rp),ref
        OPEN (UNIT=11, FILE=filestr, STATUS='OLD')
        READ(11,*) ref
  25    FORMAT(A)
        READ(11,25) equstr
        r=1
  5     READ(11,*,END=15) (A(n,r),n=1,c)
         r=r+1
         IF (r.GT.rp) GOTO 15
        GO TO 5
  15    r=r-1
        CLOSE (UNIT=11)
        return
        end
```