MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Mathematical Methods for Materials Scientists and Engineers

3.016 Fall 2010

W. Craig Carter

Department of Materials Science and Engineering
Massachusetts Institute of Technology
77 Massachusetts Ave.
Cambridge, MA 02139

PROBLEM SET 9: **Out: 22 Nov.** AND **Due: 1 Dec. 2011**

INDIVIDUAL ASSIGMENTS SHOULD BE A COMBINATION OF YOUR HAND-WORKED SOLUTIONS AND OTHER PRINTED MATERIAL—THEY SHOULD BE PLACED IN THE MAILBOX OUTSIDE PROF. CARTER'S DOOR. EMAIL GROUP ASSIGNMENTS TO 3016-psets(the symbol at)pruffle.mit.edu

For the individual problems indicated as "Handworked", you should work your solutions by hand and show your work. Print the results of software-worked solutions, and staple them to your hand-worked assignments before turning them in.

Individual Exercise I9-1

In this problem, you will explore how do compute an approximation to the entropy of a simple system using the Boltzmann relation $S(E) = k \log[\Omega(E)]$. Your answer to this problem will be improved if you do some background reading on your own about Boltzmann's equation and how to compute the temperature of the system from Boltzmann's equation. These are topics from elementary statistical thermodynamics. You will also need to make use of physical insight and the nature of statistical approximation.

Consider a system of non-polar atoms, such as He, that interact only through the van der Waals potential. We will assume no two atoms can overlap (i.e., the minimum possible distance between any two He atoms is twice the radius of a He-atom: $2R_{\text{He}}$); this is known as a hard-sphere approximation. The van der Waals interaction potential energy, U, between any two such non-polar atoms can be approximated as

$$U(r) = \frac{-A}{r^6}$$

where r is the distance between two atoms. This energy is "shared" by the two atoms.

We will consider a two-dimensional collection of such atoms that are constrained to be inside a square region of fixed area.

1. Non-dimensionalize this van der Waals interaction energy. Scale all of the energies with U_{min} defined as the minimum energy that any two atoms can share. Scale all of the distances with R_{He}.

We will use this scaled energy and distance to answer the questions below.

Furthermore, we will assume that all of the energy of the system is accounted for by the potential energy of the particles—we will not count the kinetic energy (or we will assume that none of the atoms are in motion).

2. What is the minimum possible energy per atom in a system of very many atoms? Use this energy as the scaled reference energy $U_{\mathrm{ref}}/U_{\mathrm{min}}$. All energies below will be with respect to this reference energy (in other words, the minimum energy state of a collection of atoms will always greater than zero).

3. Here is some code that places a new atom (of radius `radius`) inside a square region of side-length `sidelength` that is occupied by a set of like-particles with positions give by `particles`. The particles have been pre-computed to have a total energy `totalEnergy`.

   The code uses some advanced features that I would like you to learn about by exploration.

```
placeParticle[{totalEnergy_, particles_}, radius_ , sidelength_] :=

 Module[{testpos, distfunc, insertionEnergy},
  distfunc = Nearest[particles];
  testpos = RandomReal[{-sidelength/2, sidelength/2}, {2}];
  (*Print[{testpos,distfunc[testpos]}];*)
  While[
   (*Print[{"smallest distance",Norm[distfunc[testpos]-{testpos}]}];*)

      Apply[And, Norm[distfunc[testpos] - {testpos}]] < 2 radius,
   testpos = RandomReal[{-sidelength/2, sidelength/2}, {2}]];
  insertionEnergy =
   3 + Sum[N[-1/
       SquaredEuclideanDistance[testpos, particles[[i]]]^3], {i, 1,
      Length[particles]}];
  {totalEnergy + insertionEnergy, Join[particles, {testpos}]}
  ]
```

and here are examples of its use for 50 particles of (scaled) radius 0.5 inside a box of (scaled) size 5.

```
{energy, ps} = Nest[placeParticle[#1, 0.5, 5] &, {0, {{0, 0}}}, 50]

Timing[energies =
   Table[First[
     Nest[placeParticle[#1, 0.5, 5] &, {0, {{0, 0}}}, 10]], {10000}];]

Histogram[energies, 400]

BinCounts[energies, {Min[energies], Max[energies], 0.01}]
```

Provide a detailed narrative explaining how this code works and what the various examples show. In particular, what does the histogram represent?

Are there cases where the function `placeParticle` will fail to produce a result? Explain.

4. It would be very difficult to find an expression for the number of states at each energy $\Omega(U)$. However, the functional form can be numerically simulated by randomly constructing states and then computing the energy of each state. As the number of data points from the random samples is increased, this method provides an ever-better method of estimating Ω. This method is known as the "Monte Carlo" method and there are good web-resources to get a more detailed description.

5. The number of states that one randomly samples using a Monte Carlo method is a compromise between the accuracy of the results (which is obtained by many "experiments") and the computer time and memory that it takes to perform each computation.

   Obtain Monte Carlo data for this model for subsequent analysis.

6. Fit your resulting statistical approximation of $\Omega(\Delta u)$ (where $\Delta u \equiv (U - U_{\text{ref}})/U_{\text{min}})$) to an approximating model curve $\Omega_{\text{fit}}(\Delta u)$.

   It can be a bit difficult to find a very satisfying fit. For example, I tried to fit the model to $A\exp(-B(\Delta u - C)^2)$, *but you may wish to find a better model.*

```
hgram = Histogram[energies, 1000]

omegas = BinCounts[
  energies, {Min[energies],
   Max[energies], (Max[energies] - Min[energies])/1000}]

ebins = Table[
  e, {e, Min[energies],
   Max[energies] - (Max[energies] - Min[energies])/
     1000, (Max[energies] - Min[energies])/1000}]
```

   There is a numerical difficulty with trying to find a fit because the algorithm attempts to minimize and also use numbers like $\exp(20)$. *The way to get around this is to visually center the data by subtracting an energy value and rescaling the* Ω *so that its maximum value is near one, for example:*

```
scaledData = Transpose[{(ebins - 27), omegas/500.0}]
```

   Then, the fit to the scaled data is behaves better and you can rescale the model with something like: `500 fitmodel /. x -> (q - 27).`

7. What is your model of the normalized entropy of the system, $(S_{\text{fit}}(\Delta u))/k$?

8. The equilibrium temperature, T, of the system is given by $(1/T) \equiv \partial S/\partial\Delta u$, what is the temperature of the system as a function of its total energy?

   Only consider the range of of energies Δu from where $S_{\text{fit}}/k = 0$ up to the value of energy where the second derivative $S''_{\text{fit}} = 0$.

9. This system is physical, but a bit unusual. At higher energies, the number of states begins to diminish and the temperature becomes *negative*. Why do the number of state saturate and what is the meaning of negative temperature?

   Wikipedia has a very brief discuss of negative temperatures—negative temperatures are possible when there is an upper bound on the allowed energies.

10. In a large system composed of many copies of our simple system, the individual systems will exhange energy while keeping the total energy of constant. The Boltzman factor, $P(E_s)$, is the probability of finding a subsystem at an energy E_s and is given by:

    $$P(E_s) = \frac{\exp(E_s/kT)}{\sum_{\text{all states} E_i} \exp(-E_i/kT)} \equiv \frac{\exp(-E_s/kT)}{Z(T)}$$

    where the normalization term $Z(T)$ is the *partition function* which plays a central role in statistical thermodynamics.

    Using your fitted model for $T(\Delta u)$, Calculate and visualize this Boltzmann probability factor $P(u_s)$ at several different temperatures.