

## Metropolis Algorithm – Nonideal gases

Consider a gas of  $N$  particles in a box of volume  $V$ . At any moment, the gas can be described by its position in “phase space”: the position *and* velocity of each particle. This is thus a  $6N$ -dimensional problem.

If the interaction between any two particles is  $U(\mathbf{x}_1, \mathbf{x}_2)$ , the energy of the system at a point in phase space is

$$E = \sum_{i=1}^N \frac{1}{2} m \bar{v}^2 + \sum_{i < j} U(\vec{x}_i, \vec{x}_j). \text{ The } \textit{thermodynamic average} \text{ energy is}$$

$$\langle E \rangle = \frac{\int E \exp(-E/kT) d^{3N}x d^{3N}v}{\int \exp(-E/kT) d^{3N}x d^{3N}v}. \text{ Here } k \text{ is Boltzmann's constant, } 8.617 \times 10^{-5} \text{ eV K}^{-1}.$$

If one has no interaction ( $U = 0$ ), then one has an ideal gas, and  $\langle E \rangle_{\text{ideal}} = (3/2) N kT$ . Most usefully, one computes the *equation of state* for the energy per unit volume, that is,  $\epsilon(\rho, T) = \langle E \rangle / V = (3/2) \rho kT$ , where  $\rho$  is the density (particles per unit volume). From the equation of state one can derive the pressure and other useful quantities.

If the interaction is not zero, but does not depend on the velocity, then the velocity and the position integrals factorize and one has  $\langle E \rangle = \langle E \rangle_{\text{ideal}} + \langle E \rangle_{\text{int}}$ , where

$$\langle E \rangle_{\text{int}} = \frac{\int U \exp(-U/kT) d^{3N}x}{\int \exp(-U/kT) d^{3N}x}, \text{ where } U = \sum_{i < j} U(\vec{x}_i, \vec{x}_j). \text{ This is a very complicated}$$

multidimensional integral to evaluate.

Your project is to determine the interacting contribution to the equation of state for a gas of particles interacting via the Coulomb interaction,  $U(\mathbf{x}_1, \mathbf{x}_2) = e^2 / |\mathbf{x}_1 - \mathbf{x}_2|$ . Specifically, find  $\epsilon_{\text{int}}(\rho, T) = \langle E_{\text{int}} \rangle / V$  as a function of the (number) density  $\rho$  and temperature  $T$ . The appropriate units are  $e^2 = 14.4 \text{ eV-A}$  (Angstrom), and  $\rho$  is in  $\text{A}^{-3}$  and  $T$  is in eV (in these units the Boltzmann constant  $k = 1$ ). Work in the region where  $\rho$  is between 0.1 and  $1.0 \text{ A}^{-3}$  and  $T$  between 0.1 and 1.0 eV. (Warning: sometimes for low values, around 0.1, I had difficulty getting reasonable acceptance rates--you might have to set 0.2 as the lower limit for both density and temperature.) Express your answer as  $\epsilon_{\text{int}}(\rho, T) = C \rho^\alpha T^\beta$ ; what I really want are the exponents  $\alpha$  and  $\beta$  (which you can extract from a log-log plot of your results).

To simplify things, you may modify the programs `metatron.f` and `metrolib.f` found on the class webpage (as well as `ranlib.f`). Also, use the program `correlation.f` to examine the correlation function. You do not need to examine the correlation function for each run, but you should do it for a range of values.

The main difference here from the original `metatron.f` and `metrolib.f` is that  $\mathbf{x}$  is now six-dimensional. You will need to write an energy function  $U(\mathbf{x}_1, \mathbf{x}_2) = e^2/|\mathbf{x}_1 - \mathbf{x}_2|$  and a weight function  $W = \exp(-U/T)$ . Be sure that you pass variables of the correct type and dimension (e.g, `real x(6)`).

When you revise `metrostep`, be sure your particles do not go outside the box.

*Sweeping.* The naïve algorithm is to change all the coordinates of  $\mathbf{x}_1, \mathbf{x}_2$  simultaneously. This tends to give high rejection rates. A better strategy is to *sweep*, that is, to change just one coordinate (e.g.,  $y_1$  or  $z_2$  or ...) at a time and then accept or reject. This is less important for low-dimensional integrals but for high dimensional problems is critical. Sweeping is not required for this project but is recommended.

Step 1: Write an energy routine. Suggestion: rather than inputting  $\mathbf{x}_1, \mathbf{x}_2$  as two separate variables, create a single 6-dimensional vector that contains both, and only internally in `U` do you separate them. This will simplify your Metropolis implementation.

Step 2: Write the weight routine.

Step 3. Revise `metrostep` and `metropolis` subroutines to accommodate higher dimensional vectors.

Step 4. Debug and run and see if you get reasonable results. Be sure to example the correlation function. You will have to experiment to see what is a reasonable step size to get a reasonable acceptance rate.

For full credit, your program should be well-commented, and input and output clear and easy to use.

\*\*\*\*\*

Submission: your code should do the following (not necessarily in this order):  
density and temperature (from the density you can derive the size of your box)  
random seed

(average) step size

# of thermalization steps

# of samples to be taken

# of steps between samples

Output should be  $\epsilon_{\text{int}}(\rho, T) = \langle E_{\text{int}} \rangle / V$  including error bars! You should also write data to file that can be used by the correlation routine (write denominator and numerator to separate files, see my example).

Project is due Midnight on Sunday, Dec 19, 2010.

Advanced project: Do the above but using four particles (a 12-dimensional integral). Here sweeping is more essential. Compare with two particle simulation—do you get similar results? Are your exponents the same?