

# Computational Physics Homework 5

Stanley "Alex" Breitweiser"

December 14, 2015

## 1 Question 1

### 1.1 Metropolis

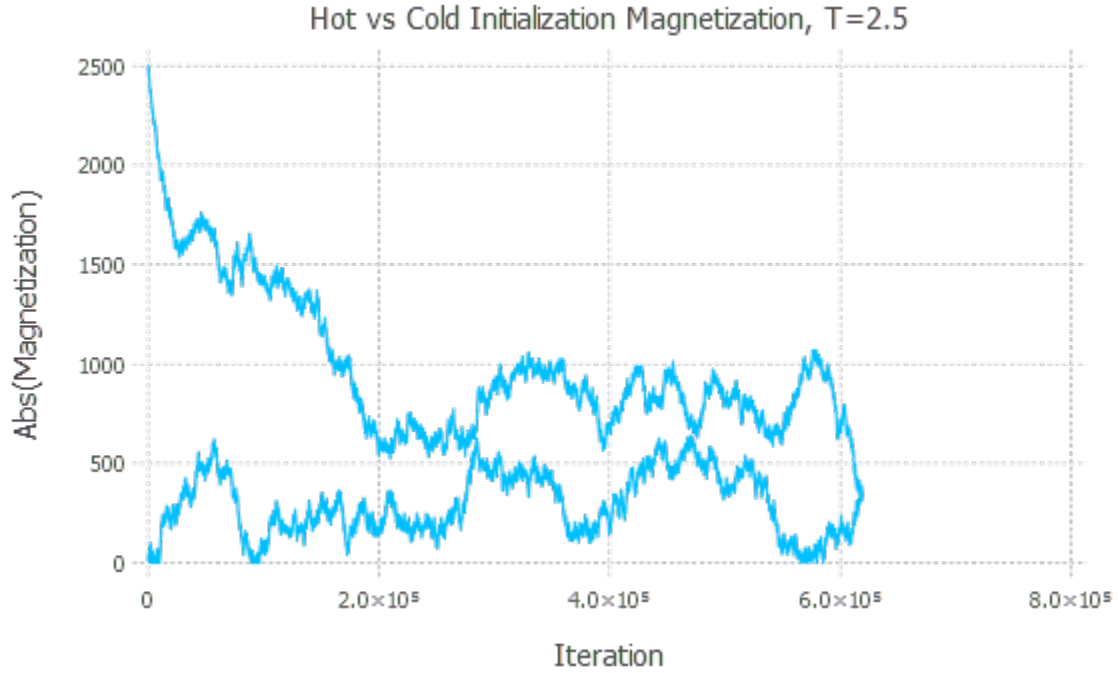
A grid of 50 spins on a side is initialized, and converged towards a steady state solution using the metropolis algorithm. The metropolis algorithm works by randomly picking a spin and proposing to flip it; the energy change,  $dE$ , associated with the change is then calculated. The energy change is the sum of the energy change due to interactions with the four immediate neighbors; for each, the energy change is  $+2J$  if the spins become anti-aligned, or  $-2J$  if they become aligned. If  $dE < 0$ , the change is accepted; if  $dE > 0$ , it is accepted with probability

$$P = e^{-\beta dE},$$

and rejected otherwise (note that  $\beta$  is the inverse temperature,  $\frac{1}{kT}$ ). Probabilistic acceptances are made by the standard method of taking a random number uniform in  $(0,1)$  and accepting if it is less than the probability of acceptance.

To find the number of steps needed for convergence, we start one hot initial condition (random spins) and one cold initial condition (all the same spin), and evolve them at temperatures near the critical temperature ( $T_c = 2.269\frac{J}{k}$ ), where convergence is expected to be slowest. We consider them to have “converged” when they agree within  $L = \sqrt{size}$ , the expected standard deviation of counting measurements. Trying this for a few different temperatures seems to consistently yield convergence within a few hundred thousand steps; an example for  $T = 2.5\frac{J}{k}$  is included below. To be safe, we take the number of iterations needed for convergence to be one million; or, equivalently, the number of steps per spin to be

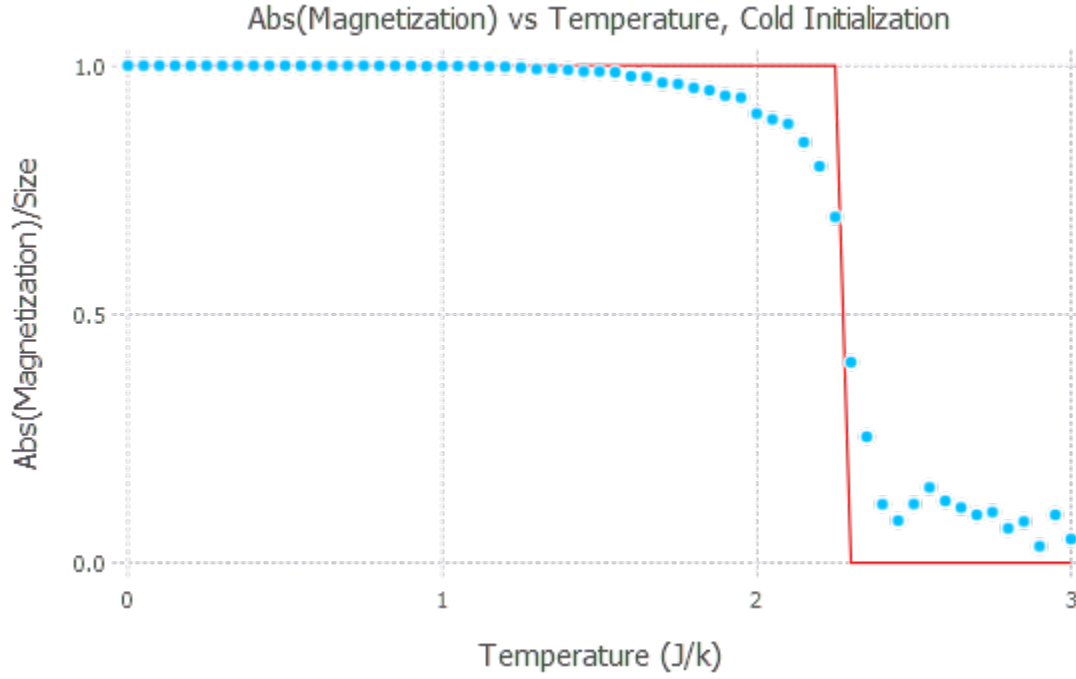
$$t_n = \frac{n}{L^2} = \frac{1,000,000}{2,500} = 400$$



Note that, since we expect behavior for the magnetization to either be near zero or randomly  $\pm L^2$ , we look not at the magnetization itself but its absolute value; this allows us to ignore any difficulties with the sign and only look at the magnitude behavior.

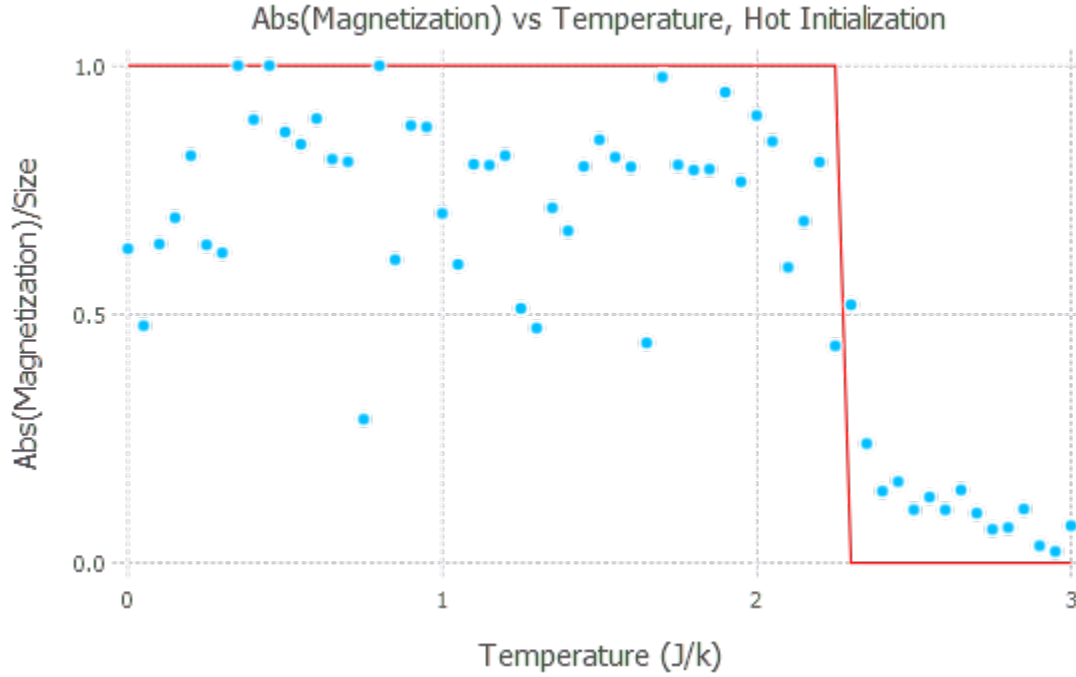
## 1.2 Magnetization

The analytical solution to the Ising model predicts that the magnetization per spin will be almost exactly  $\pm 1$  at temperatures below the critical temperature, and nearly 0 above. To test this we converge our model, using the number of steps needed for convergence from the above part, and take the absolute value of the magnetization. We do this 5 times for every temperature and average the results. We do this for temperatures between 0 and 3, crossing the theoretical threshold of  $T_c = 2.269 \frac{J}{k}$ ; the results are plotted below.



We have included the approximate solution, magnetization =  $\pm 1$  if  $T < T_c$  and 0 otherwise, in red for comparison. While this is not exactly what we expect (in particular, the transition will never be sharp for finite models and we do expect some noise when  $T > T_c$ ), we can use this to see that our model seems to confirm the theoretical prediction well.

Note, however, that we used cold initialization (uniform spins) rather than hot; this is because the hot initial conditions did not converge as nicely at low temperatures. Below we include the results for hot initialization. While it does not fit the approximation as well at low temperatures, it still shows a sudden change from large magnetizations at low temperature to small magnetizations about the critical temperature, confirming the theoretical prediction for the critical temperature.



## 2 Question 2

### 2.1 Setup

200 particles were initialized at position  $x = 1$ ; the temperature was taken such that  $kT = \frac{1}{2}$ , so that  $\beta = \frac{1}{kT} = 2$ . Furthermore, units were taken such that  $m = g = 1$ , so that  $\Phi(x) = x$  for  $x > 0$ .

### 2.2 Metropolis Algorithm

Just like in the Ising problem, the metropolis algorithm here works by (uniformly) randomly picking a particle and “proposing” a change to its state (position). Here, the proposed change,  $dx$ , is taken randomly from a uniform distribution between  $(-\epsilon, \epsilon)$ , where  $\epsilon$  is a chosen parameter of the algorithm. The change in energy associated with this, in our natural units, is just  $dE = dx$ . Then, if  $dE < 0$  the change is accepted; if  $dE > 0$  the change is accepted with probability

$$P = e^{-\beta dE} = e^{-2dx},$$

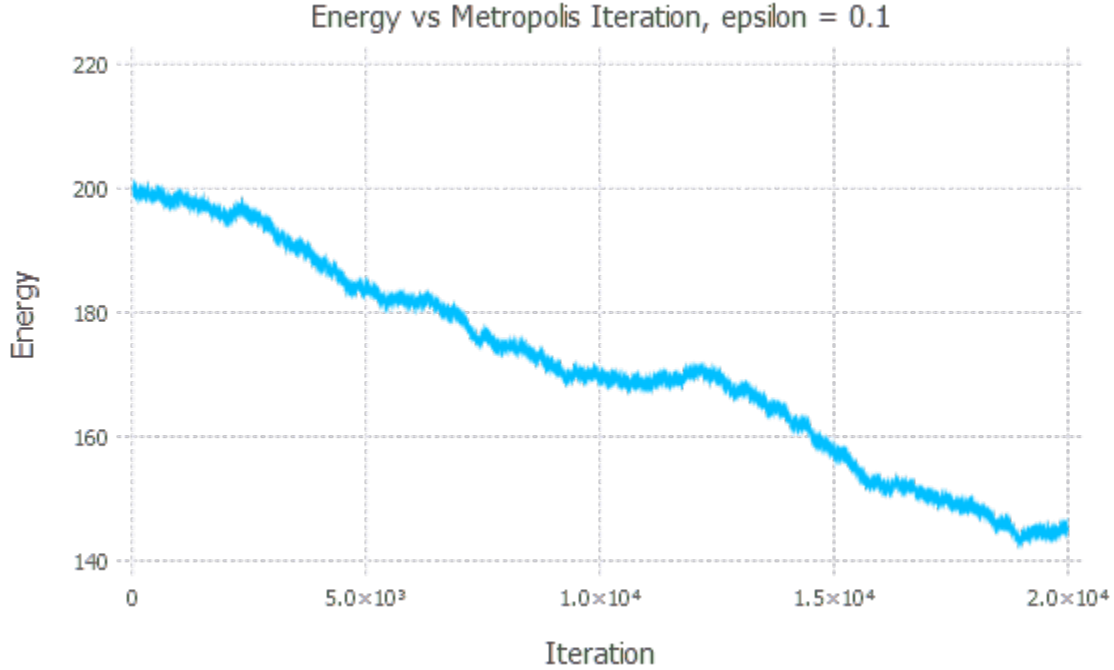
and rejected otherwise. Proposed moves that move the particles below the  $x = 0$  “floor” are rejected, which is equivalent to associating those positions with an infinite (positive) potential energy.

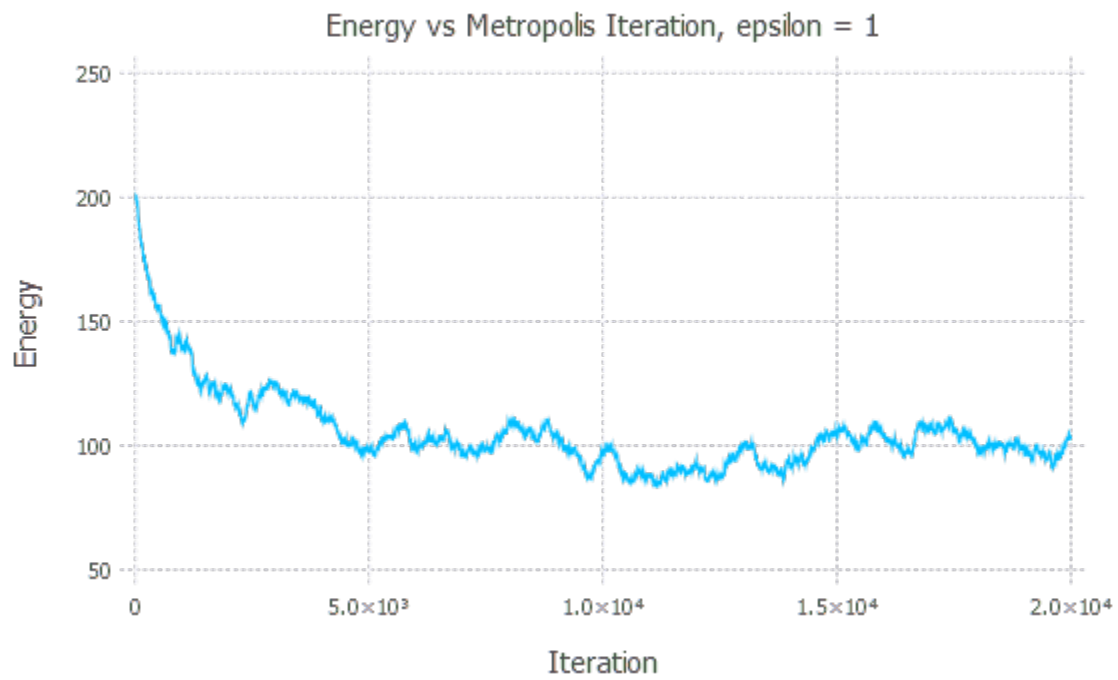
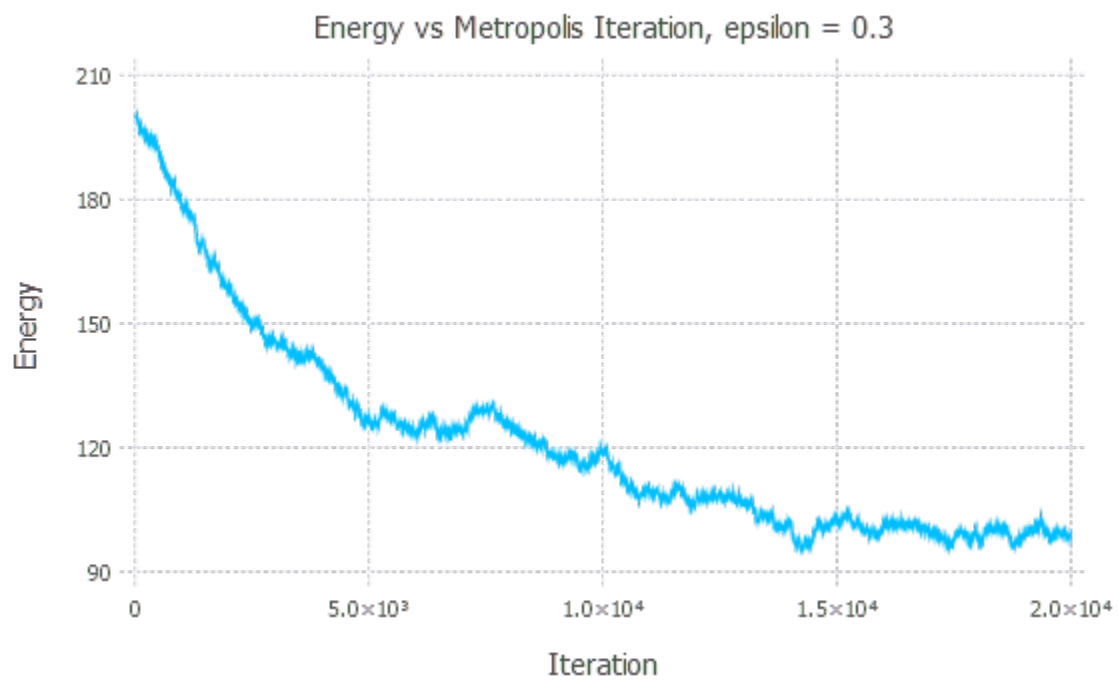
The  $\epsilon$  parameter of the model should be chosen to optimize between two extremes; if  $\epsilon$  is very small, the model will converge very nicely but will take a large number of iterations; if  $\epsilon$  is very large, the model will initially converge much faster but will be subject to much more “noise” from randomness, due to large energy changes associated with proposed changes giving small acceptance probabilities. Furthermore, the model will be unable to produce fine distribution changes (of order much less than epsilon), and will therefore be very slow to converge in small features. Even once it has “converged”, there will be large noise fluctuations about the convergent value.

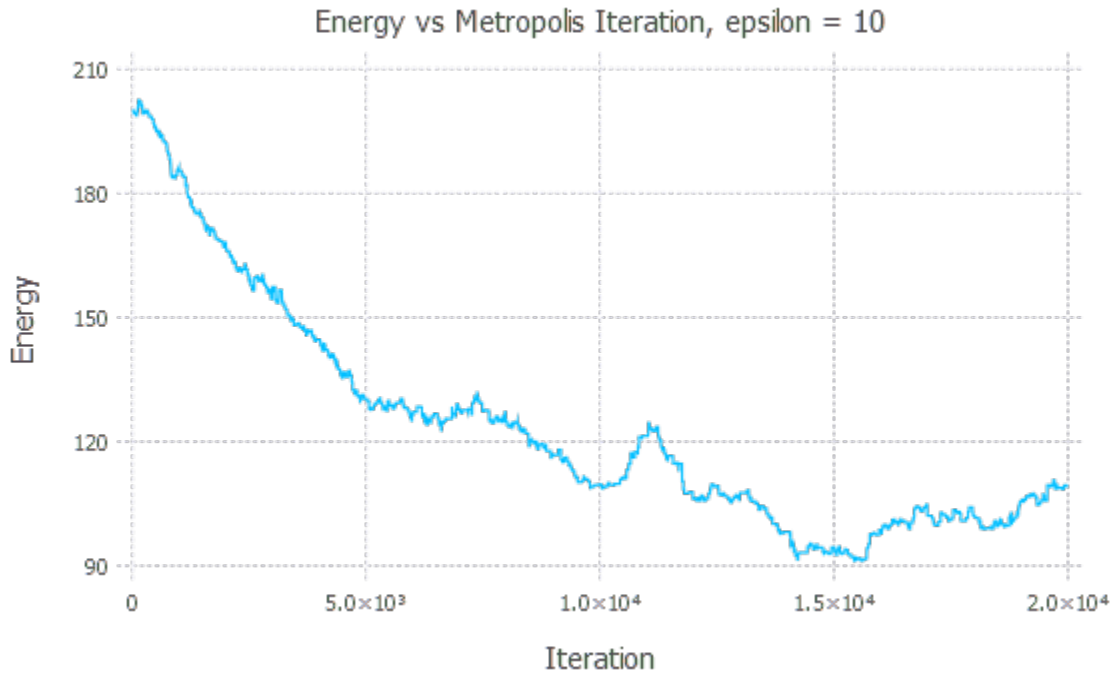
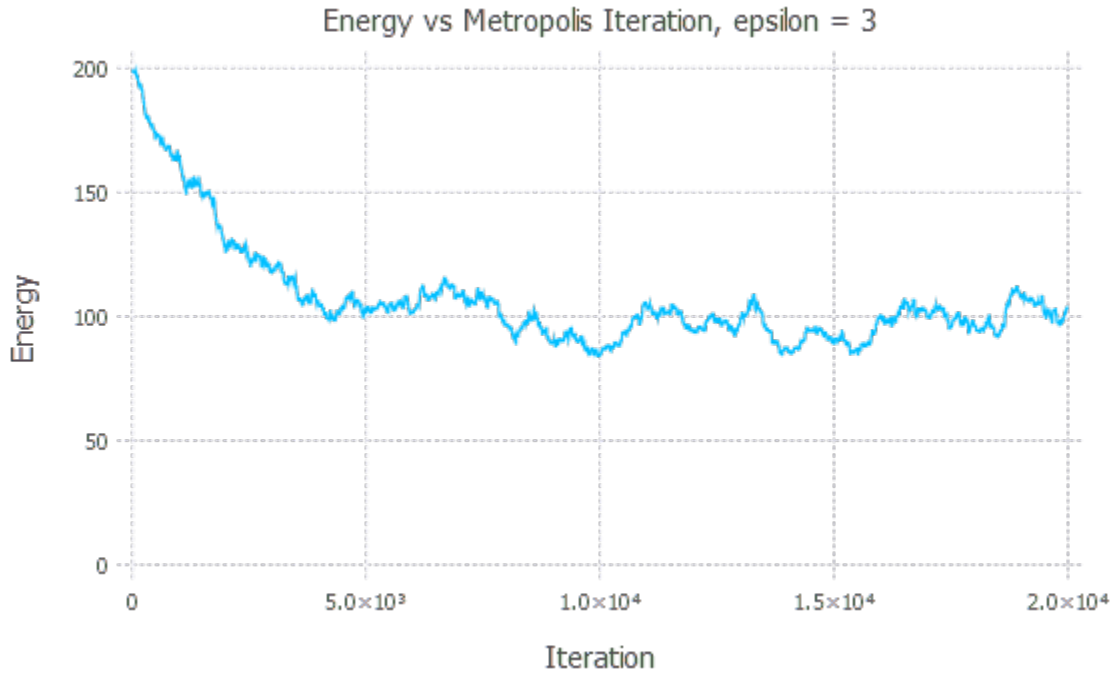
Therefore, we want to pick an  $\epsilon$  between these extremes; we know that  $\epsilon$  should be of roughly the same order as the distribution we expect, which is  $2e^{-2x} \sim 1$ . This agrees with the heuristic that changes should be accepted roughly half of the time (near convergence, the particles will be of characteristic distance  $\sigma \approx \frac{1}{2}$ ,

so a negative change will be accepted about half the time, when the particle does not go below the floor; positive changes will be about  $\frac{1}{2}$ , and so will be accepted about  $e^{-2*\frac{1}{2}} \approx \frac{1}{3}$  of the time).

More fine-scale tuning of  $\epsilon$  is best done by trial, so we here include convergence graphs of the potential energy for values of epsilon geometrically spaced between 0.1 and 10 by  $\sqrt{10} \approx 3$ . A good confirmation of the algorithm, and the heuristic, is that all of these work reasonably well; however, we choose  $\epsilon = 1$  as giving the best (fast) convergence and (small) variation after convergence. Therefore, we take  $\epsilon = 1$ .

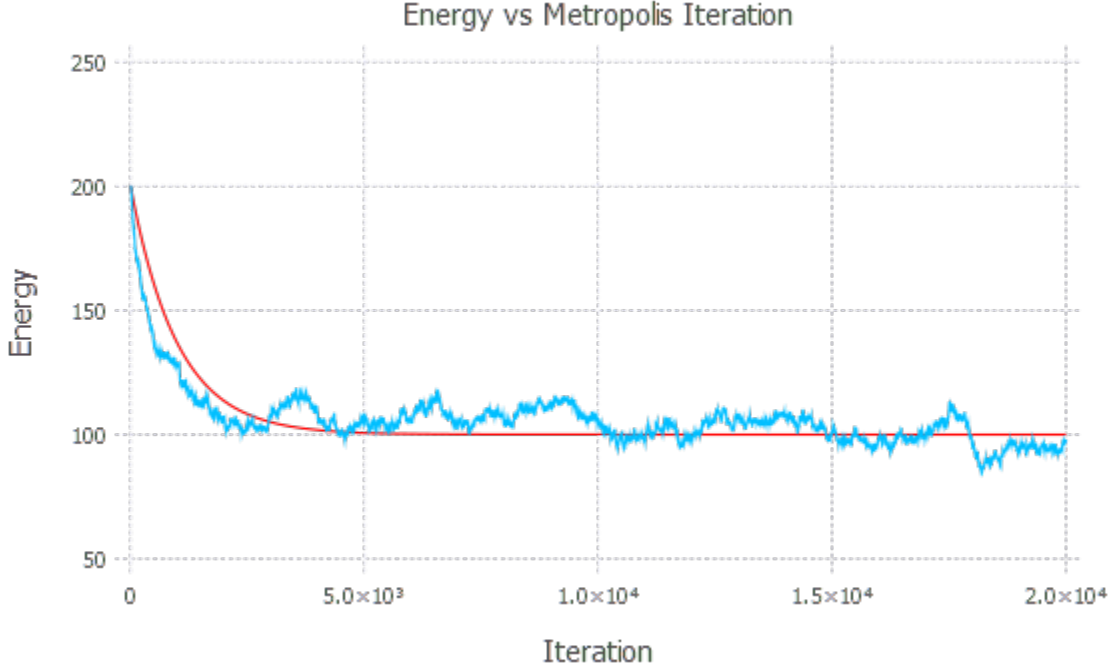






### 2.3 Equilibration Time

The potential energy,  $U = \sum \Phi(x) = \sum x$ , is plotted below as a function of the number of iterations.



The equilibration time,  $\tau$ , is defined such that the expectation value for the potential is approximately

$$E[U(t)] \approx U_{\infty} + (U_{\text{initial}} - U_{\infty})e^{-\frac{t}{\tau}}.$$

In other words, the equilibration time is the time it takes a perturbation from the equilibrium value to drop by a factor of  $\frac{1}{e} \approx \frac{1}{3}$ . We conservatively estimate, from our plot, that our equilibration time is about

$$\tau \approx 1000$$

iterations, and plot the corresponding expectation for the potential energy as a red line in our above graph. We see that, indeed, this is a good conservative estimate.

As is standard, we consider our model to have equilibrated after 4 equilibration times have passed; this will give agreement in the expected value to  $\frac{1}{e^4}$ , which is about 1 part in 100, of any perturbation from equilibrium in the initial conditions.

## 2.4 Average Energy and Variation

After we have allowed our model to equilibrate, we find an average energy value of

$$\langle U \rangle = 102.91 \approx NkT = 100,$$

which is in good agreement with theoretical predictions. We find a variation on this average of

$$(dU)^2 = \frac{1}{N} \sum (U(t) - \langle U \rangle)^2 = 33.83.$$

## 2.5 Correlation Time

Similarly to the equilibration time, we want to find a correlation time,  $\tau_{\text{corr}}$ , such that

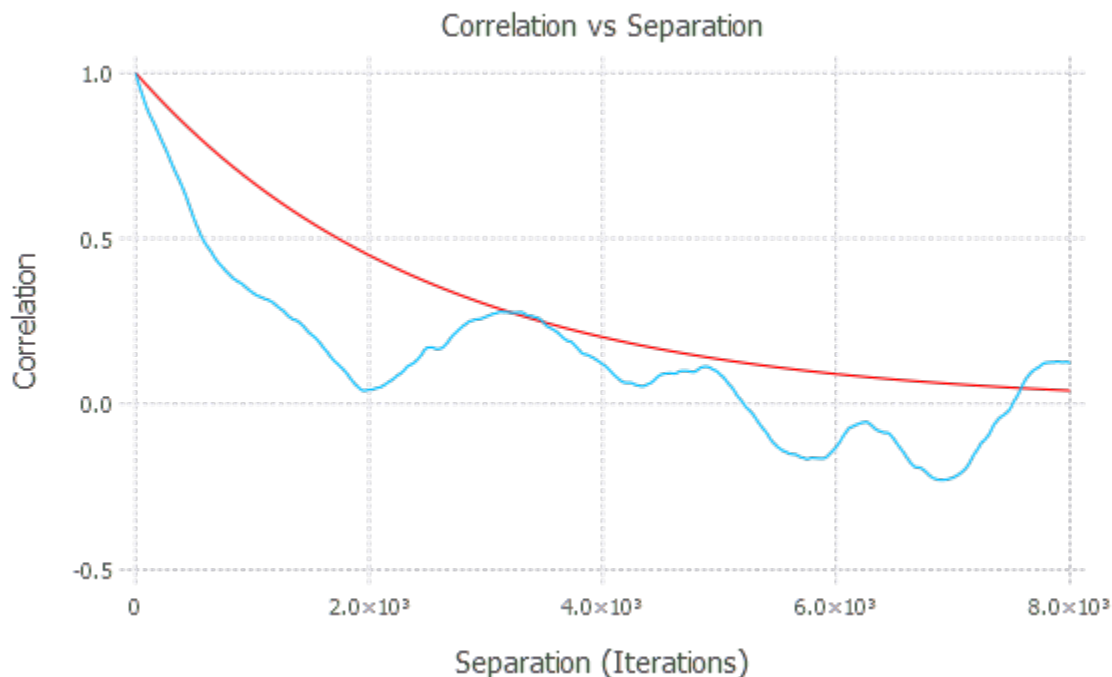
$$E[C(m)] \approx e^{-\frac{t}{\tau_{\text{corr}}}}.$$



(Remember that  $C(0) = 1$  and  $C(\infty) = 0$ ). Below we plot the correlation as a function of separation; we conservatively estimate

$$\tau_{corr} \approx 2500$$

and plot the expectation value for the correlation from that in red, seeing that it is a good conservative estimate.

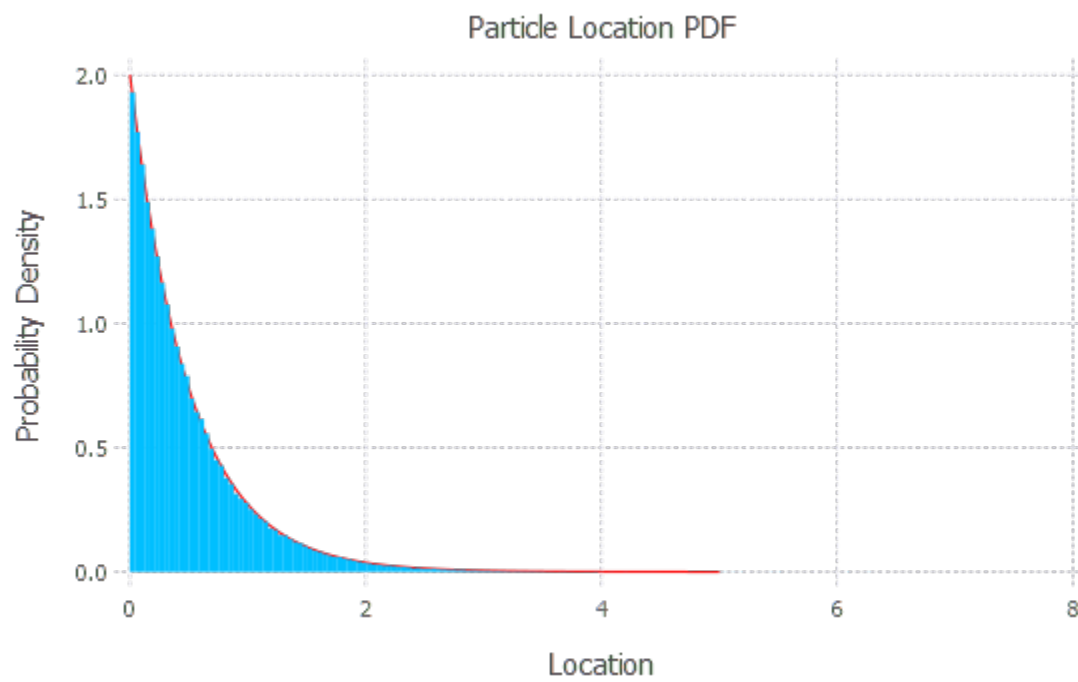


## 2.6 Boltzmann Distribution

For good statistics, we sample the positions of particles from 1000 groups, all iterated from the same initial conditions but separated by the four times the correlation time (and therefore approximately uncorrelated). We use these to build a probability density histogram for the position of the particles, and compare it with the theoretical prediction, the Boltzmann distribution,

$$f(x) = \frac{mg}{kT} e^{-\frac{mgx}{kT}} = 2e^{-2x}.$$

Below we include the graph, with the Boltzmann distribution shown in red



We see that our model produces a distribution in excellent agreement with the theoretical prediction.