AEP 4380 Final Project: The 3D lattice and efficiencies using Monte Carlo

Vineet Kamat

December 1, 2020

Abstract

In homework 9, the class was tasked with using Monte Carlo to find a 2D lattice structure using the Lennard-Jones potential, the partition function, and a random number generator. In this project, the 3D lattice is explored and the Monte Carlo method will be modified for efficiency.

1 Leonard Jones Potential

This lattice will make sure of the Leonard Jones Potential given by:

$$U(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

The non-dimensionalized form gives:

$$U(r) = 4\left[\left(\frac{1}{r}\right)^{12} - \left(\frac{1}{r}\right)^{6}\right]$$

Below is a plot of the non-dimensionalized Lennard-Jones Potential:

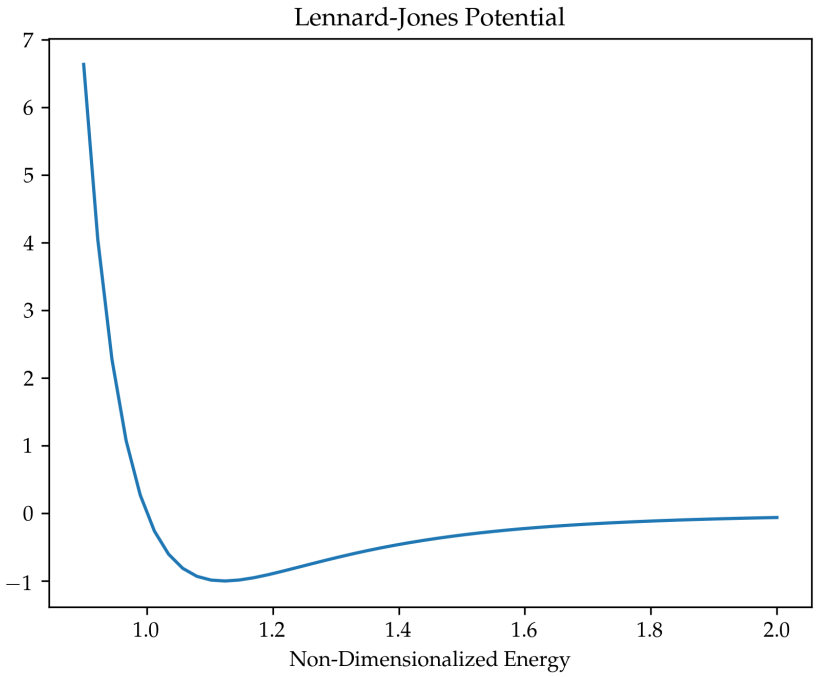

Figure 1: Lennard-Jones Potential

It is important to note that the potential reaches a minumum of -1 at a distance of 1. This is crucial to ensuring that a crystalline structure is reached. This also implies that the total energy of the crystal will be a negative.

2 3D lattice

The Monte Carlo method was used in homework 9 to find the following lattice structure:

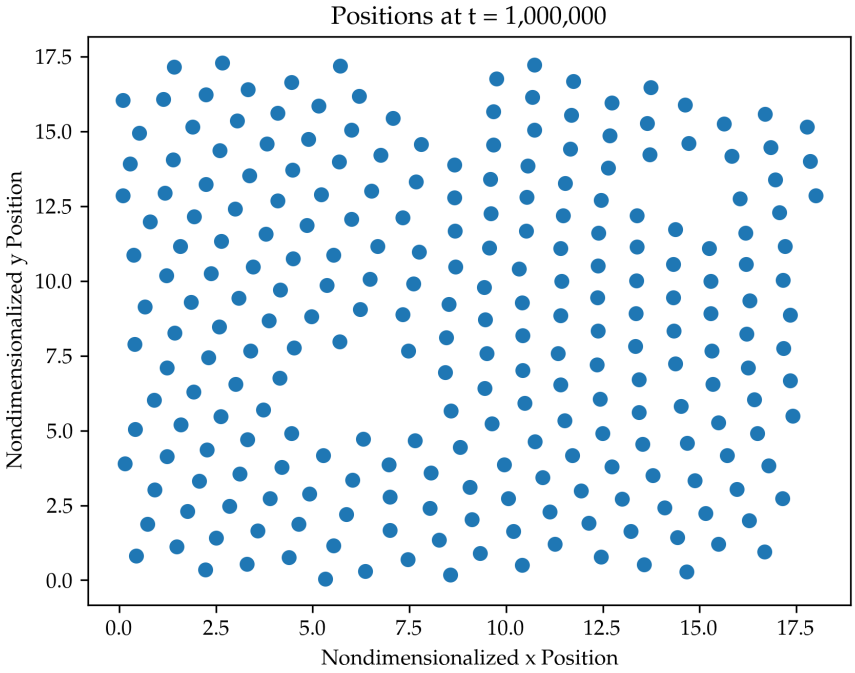

Figure 2: Positions after 1,000,000 time steps

In this subsequent project, the 3D structure of this lattice will be explored. However, to do this for more than 250 atoms, each iteration will take much longer due to the n^2 calculation time, making the method essentially not viable.

3 Proposed Speedup

As proposed earlier, it is possible to skip many iterations of the Energy calculation. Instead of recalculating all energies, one could keep track of a matrix of energies between atoms, making it so instead of doing $n * (n-1)$ calculations for each update to the lattice, one would only have to do $n-1$. The total energy will be calculated by summing over the matrix at the end of the solution. However, at each individual iteration, the previous algorithm only needed ΔE to perform an update. In this new algorithm, instead of finding total energy, one can simply find ΔE by finding the changes in the updated elements rather than the whole. This will make the update much faster and hopefully allow the program to create a much more accurate calculation for a small number of atoms with many more iterations or make the calculation much faster for a larger number of atoms for the same number of iterations.

4 Conclusion

In previous attempts, this faster algorithm did not work since there is an issue with pointer arithmetic with multidimensional arrays. Once these issues are worked out, the implementation of this algorithm will be quite similar to the previous implementation, but this time

with the update step running only a single for loop over the n atoms and updating an the adjacency matrix of Energies.

5 Bibliography