P3800 Project 4: The Protein-Folding Problem

John Healey

Written April 2018

1 Introduction

The Protein-Folding Problem is an inter-disciplinary topic, involving Physics and Biology, so I will start by giving an overview of the biological concepts needed to understand the problem. A protein is a molecule that plays a particular role in the body. Proteins are made up of simple organic compunds called amino acids, connected end to end to form a chain. The types of amino acids that make up the protein, and the way in which they are arranged, determine the protein's ability to perform its biological function. This is not only based on the order of the acids in the chain, which determines its *primary structure*, but also on the shape of the chain itself, i.e., how it is folded. The way in which the chain is folded is known as its *tertiary structure*.

In order for a protein to perform its biological function, then, it must be folded in exactly the right way, and if it becomes unfolded, it must return to its correct tertiary structure. However, for a protein consisting of 300 amino acids, and assuming 4 possible orientation angles between connected acids, there are $4^N \approx 10^{180}$ possible tertiary structures arising from a single primary structure. Despite this inconceivably large amount of possible structures, not only is a protein able to consistently go from an unfolded state to its correct tertiary structure, it is able to do so in seconds. Even if the protein spent as little as 10^{-13}s in each intermediate state, the folding process would still take longer than the age of the universe. This is known as Levinthal's Paradox. How the protein selects the correct tertiary structure is the Protein-Folding Problem.

2 Method of Investigation

To try to understand how protein folding might work, I created a simulation of a protein made up of N amino acids, where N ranged from 15 to 100. I have assumed that there are 20 types of acids available to the protein. Of particular importance in this simulation is the interaction between non-bonded nearest-neighbour (NBNN) amino acids. These are the acids that are not linked in the chain, but are still in nearest- neighbour positions with each other, due to the folding of the protein. The interactions between these acids are important because the primary structure of the protein will not change, so the energy related to the acids connected in the chain is constant. What does change is the protein's tertiary structure, and since the forces between non-connected acids only apply at short distances, NBNN energies determine the energy of the protein's tertiary structure. There are many complicated forces between the acids that determine the interaction energy, so for the purposes of the simulation, I have assumed that the interaction energies between the various types of acids vary randomly within the range of [-4, -2]. For simplicity in the Monte Carlo algorithm, energy is measured in units of k_B.

The total energy of the unfolded protein was taken to be 0, since none of the acids are non-bonded nearest neighbours. As the protein folds, some acids will become NBNN, which will decrease the energy of the protein. The energy of the protein will be given by

$$E = \sum_{i,j}^{NBNN} J_{i,j}, \tag{1}$$

where $J_{i,j}$ is the interaction energy between acid types i and j, and the sum is over all non-bonded nearest neighbours. This use of interaction energies between acids to determine how the protein folds is where physics meets biology, and the problem becomes inter-disciplinary.

3 Simulation

The first step of the simulation was to choose the protein's primary structure, i.e., to randomly choose the type and order of the amino acids that make up the protein. I then created a matrix for the interaction energies between the types of acids, each value of which was a random floating point number between -4 and -2, as mentioned above. The matrix was made to be symmetric, so that the interaction energy between types i and j is the same as that between types j and i. I then initialized an empty lattice on which to place the protein, whose dimensions I made 5N x 5N, to give the protein plenty of space to move around throughout the simulation. I then placed the initialized protein on the lattice, and began the simulation.

The simulation used the Monte Carlo method as follows. A link in the chain is randomly selected. This link has 4 possible positions to move to (the 4 diagonals). One of these positions is selected at random, and the program determines whether or not a move to the position would break the chain. If not, the energy change from the move is calculated, using the formula for energy shown above, using the link's current NBNN, and the NBNN it would have if it were in the position being examined. If the energy change is negative, the link is moved. Otherwise, it is moved if a random number is less than $e^{\frac{-\Delta E}{k_B T}}$. This is done a large number of times, in order to get smooth average values.

4 Results

To test the accuracy of my simulation, I used it to find the dependence between temperature and the average energy of the chain. I measured this dependence for chains of length 15, 30, and 100, each time starting at $T = 10$, and decreasing T to 0.5. At each temperature, 500,000 MCS were taken. The results of this can be seen in Figures 1-4. Figures 1-3 show that for all of the chain lengths, most of the energy change occured between $T = 5$ and $T = 2$. Figure 4 shows that the length of the chain determined the scale of the range of energies, with a larger chain producing a larger range. Both of these observations agree with expected results. Experiments have shown that real proteins fold abruptly as conditions such as temperature are changed, which explains the rapid decrease in average energy. The second observation makes sense, since the longer the chain, the more NBNN there will be. All interaction energies are negative, so this will lead to a more negative average energy, as observed in the simulation.

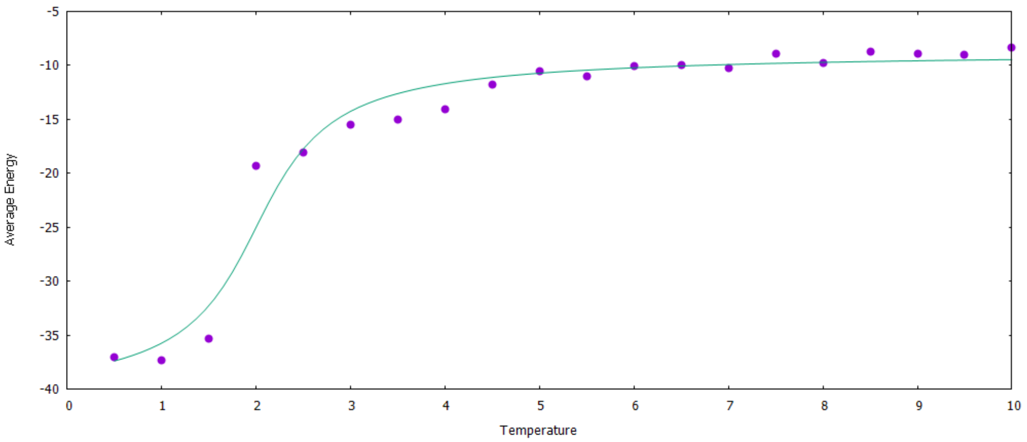

Figure 1: Average energy as a function of Temperature for a protein made up of 15 amino acids

The next step for the simulation was to actually address the Protein-Folding problem. As discussed above, the conundrum is that a protein is able to select the correct tertiary structure out of the many choices, each time it is unfolded. At high temperatures, the protein will often make energy-increasing moves, so it will not settle in a low energy state. So, in order to see if my model protein was able to consistently find the correct tertiary structure, I used a low temperature, and conducted 2 runs on an identical protein. As can be seen in Figure 5, the 2 runs settled in very different final states. This means that the protein ended up in a different tertiary structure when left to unfold in identical conditions. So, my model protein must not be a very accurate model of a real protein.

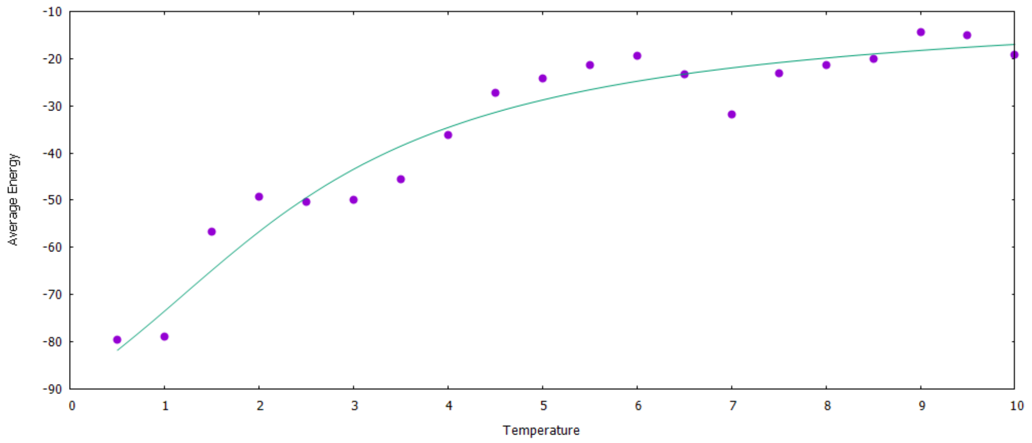

Figure 2: Average energy as a function of Temperature for a protein made up of 30 amino acids

The reason my model protein is not behaving like a real protein in the simulations above is because it is getting stuck in *metastable states*. While searching for the structure corresponding to the global minimum in energy, the protein finds a state that is a local minimum, meaning that all changes in strucure cause an increase in the protein's energy. Since the simulation is carried out at a low temperature, it is unlikely that the protein will make a move that increases its energy, so it gets stuck in the local minimum, never finding the global minimum corresponding to the correct tertiary structure.

The method I investigated to try to solve this problem is called *annealing*, in which the simulation starts at a temperature that is high enough that the protein will explore a large number of possible structures. The temperature is then slowly lowered, so that the protein starts to spend more and more time in the lowest-energy states. By the time the temperature becomes very low, the hope is that the protein will be stuck in the lowest-energy state. My results using this process can be seen in Figure 6. In this simulation, I performed 2,000,000 MCS, and gradually lowered the temperature from 4 to 1. Compared to my earlier results shown in Figure 5, the protein ended up in a considerably lower-energy final state. The lower run in Figure 5 ended up at an energy of approximately -37, while the run using annealing ended up at an energy of approximately -43.

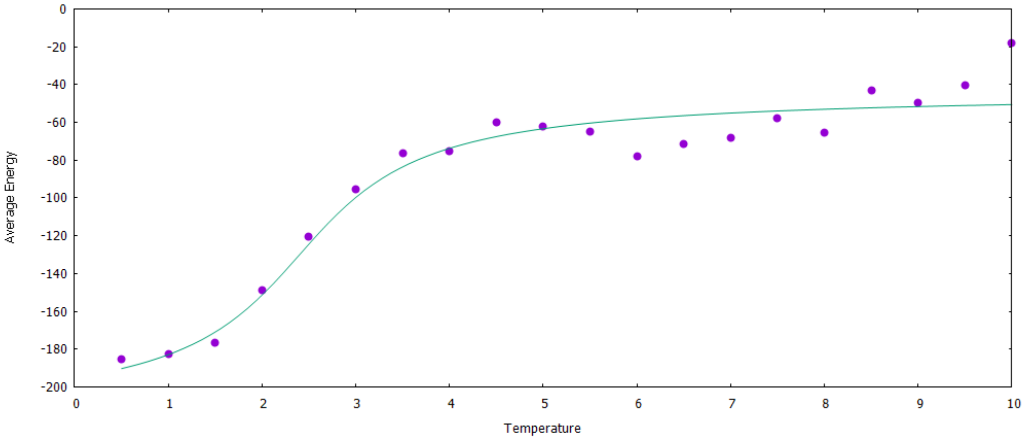

Figure 3: Average energy as a function of Temperature for a protein made up of 100 amino acids

5 Discussion

The process of annealing made my simulated protein behave more like a real protein, but it is not perfect. There is still a chance that the protein will end up in a metastable state, which, biologically, would mean that the protein would be unable to carry out its function. One explanation for this is that real proteins use annealing, and simply do not always fold back to the correct tertiary structure, and only do so most of the time. This would imply that some proteins are not performing their biological function, but as long as many proteins do have the correct structure, this would not be an outrageous claim.

The protein-folding problem has not yet been solved, so I cannot say for sure if my simulated protein behaves the way a real protein does when it comes to re-folding. However, the results I have obtained do agree with experimental results with respect to the abrupt change in average energy as temperature is decreased, and the energy dependence on length. The process of annealing clearly led to a more stable state than simulations without it, demonstrating its possibility as an explanation of the protein-folding problem. This simulation could certainly be built upon to become more accurate, but it does provide a solid framework for investigating this problem.

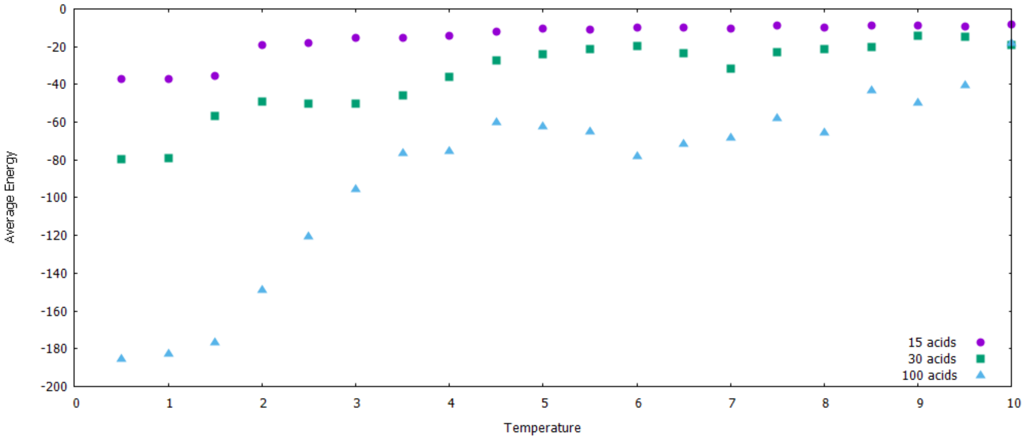

Figure 4: Average energy as a function of Temperature for proteins made up of 15, 30, and 100 amino acids

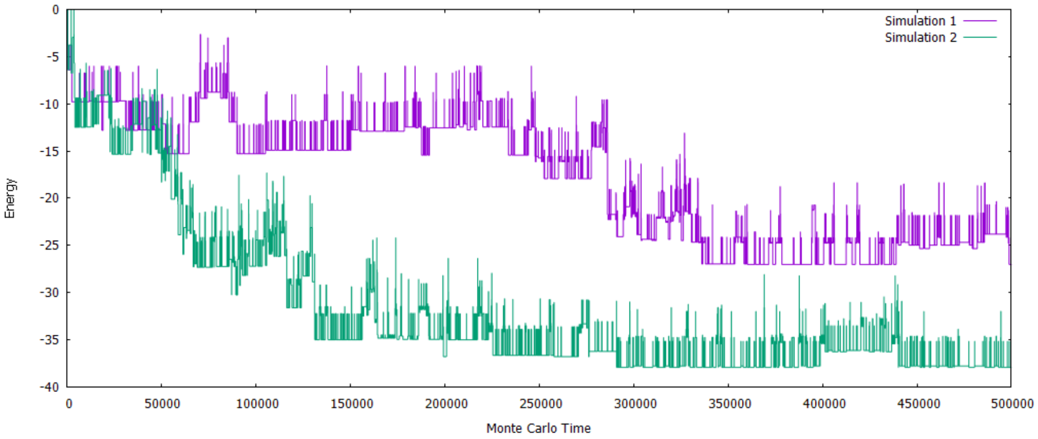

Figure 5: Energy as a function of Monte Carlo Time for two simulations using a protein made up of 30 amino acids

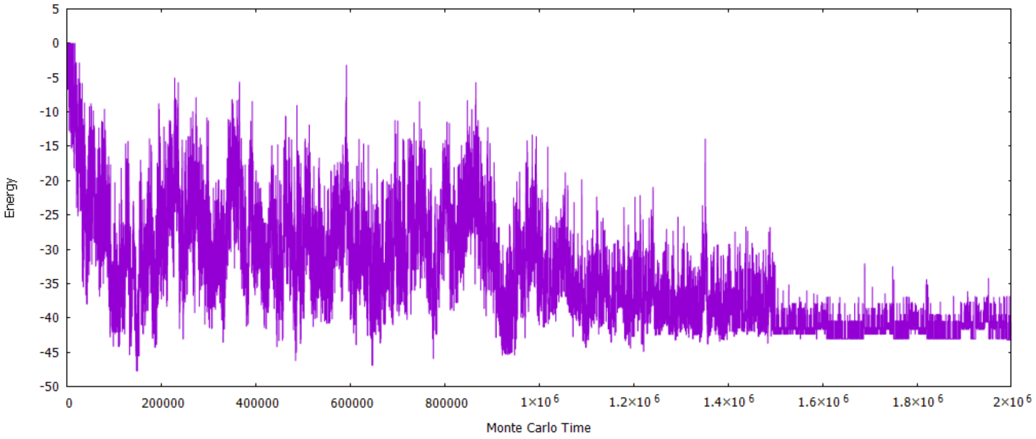

Figure 6: Energy as a function of Monte Carlo Time using a protein made up of 30 amino acids with annealing