PHY407 Lab10

(Yonatan Eyob Q1), (Kivanc Aykac Q2,Q3)
Student Numbers: 1004253309, 1004326222

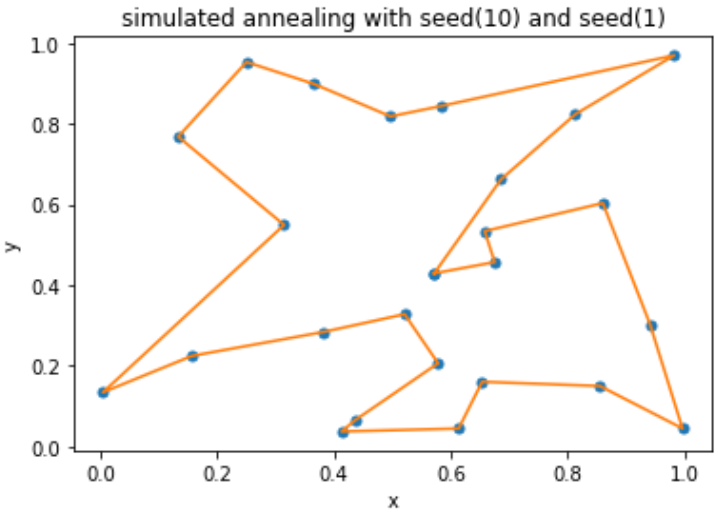
December 4^{th}, 2020

1 Examples of simulated annealing optimization

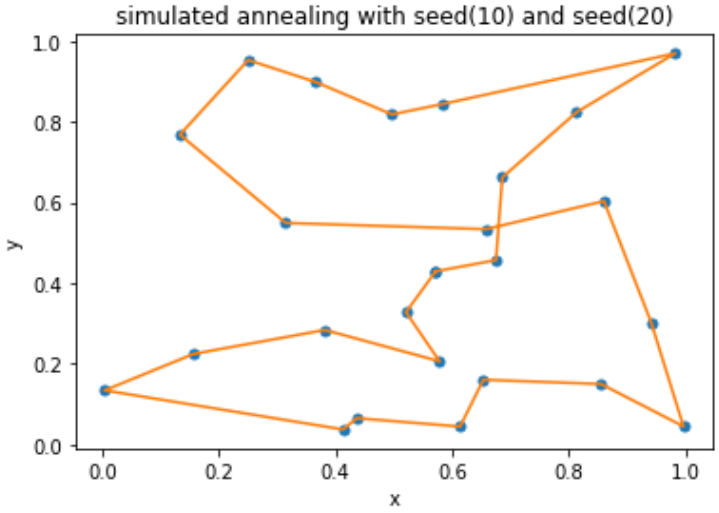
In this section we will use the simulated annealing optimization on a set of points to find the shortest route to take to connect all the points and then we will use simulated annealing to find the global minimum of functions f(x,y)=$x^2 - cos(4\pi x) + (y-1)^2$ and f(x,y)=$cos(x) + cos(\sqrt{2}x) + cos(\sqrt{3}x) + (y-1)^2$.

1.a

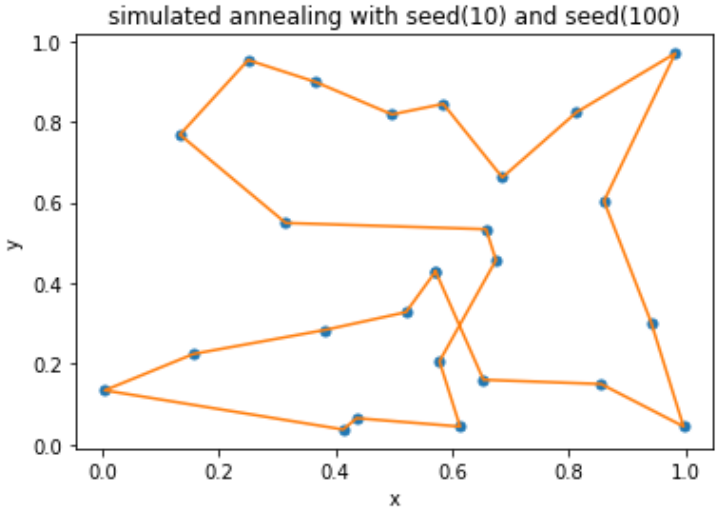
first we will use the simulated annealing optimization on a set of points a few times with different seeds to see how it affects the distance of the path (D):



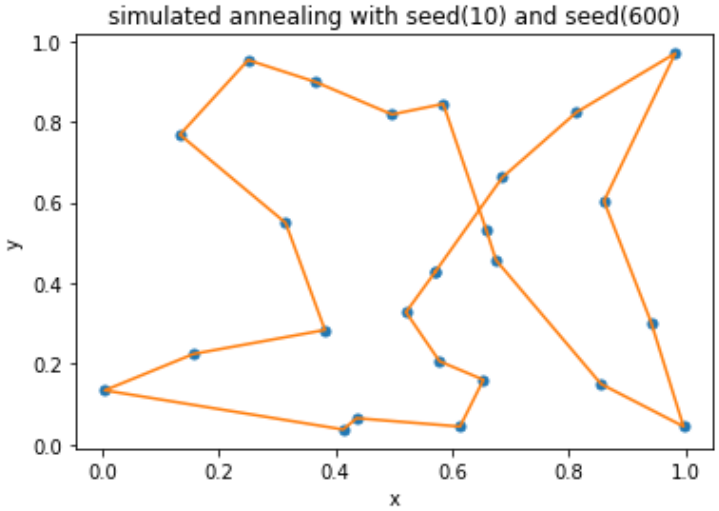
with seed(10) and seed(1), D= 5.103949568026902



with seed(10) and seed(20), D= 5.173159206572483



with seed(10) and seed(100), D= 5.295353352525823



with seed(10) and seed(600), D= 5.233831352044736

as we change the seed the route and the D changes as well. There doesn't seem to be a pattern between the seed number and D but D seems to always be between 5.0 and 5.8 no matter which seed we choose.

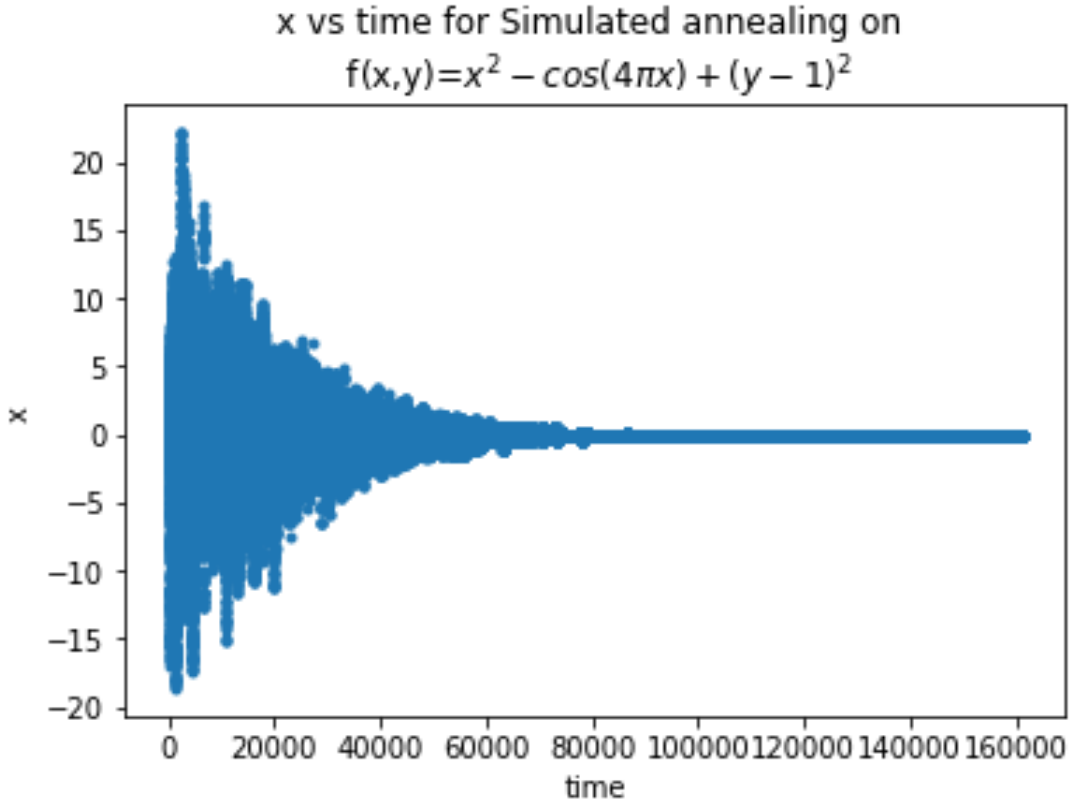
The cooling rate is T=$T_0 * e^{-t/\tau}$. We found that as we decrease the cooling rate (increase τ) D gets smaller and as we increase the cooling rate (decrease τ) D gets bigger. This means that a slower cooling rate gives us a more accurate minimum distance path.

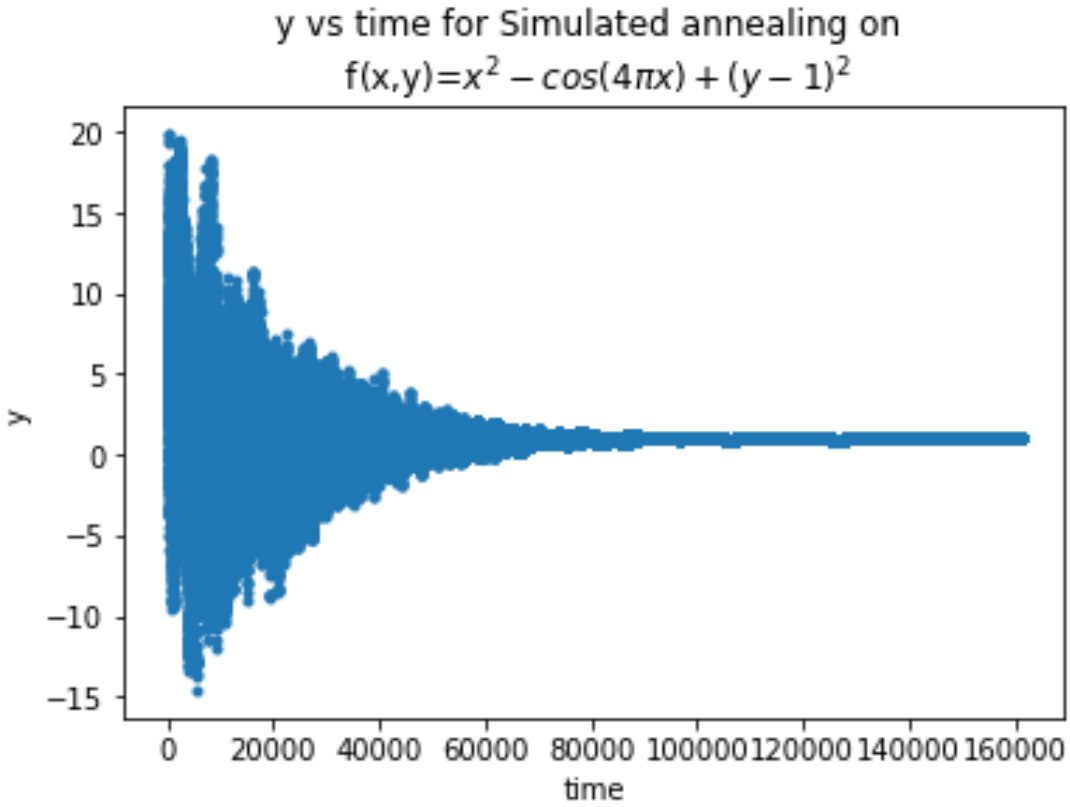
1.b

Now we will use simulated annealing to find the global minimum of f(x,y)=$x^2 - cos(4\pi x) + (y-1)^2$ and f(x,y)=$cos(x) + cos(\sqrt{2}x) + cos(\sqrt{3}x) + (y-1)^2$ and plot the x vs time and y vs time of the annealing function

1.b.i

first we will find the global maximum of f(x,y)=$x^2 - cos(4\pi x) + (y-1)^2$:
The max temperature in the simulated annealing process is 100, the minimum temperature is 10^{-4}, and the cooling rate is set to T=$100 * e^{-time/10^4}$ (time is the x axis in the x vs time and y vs time graphs).

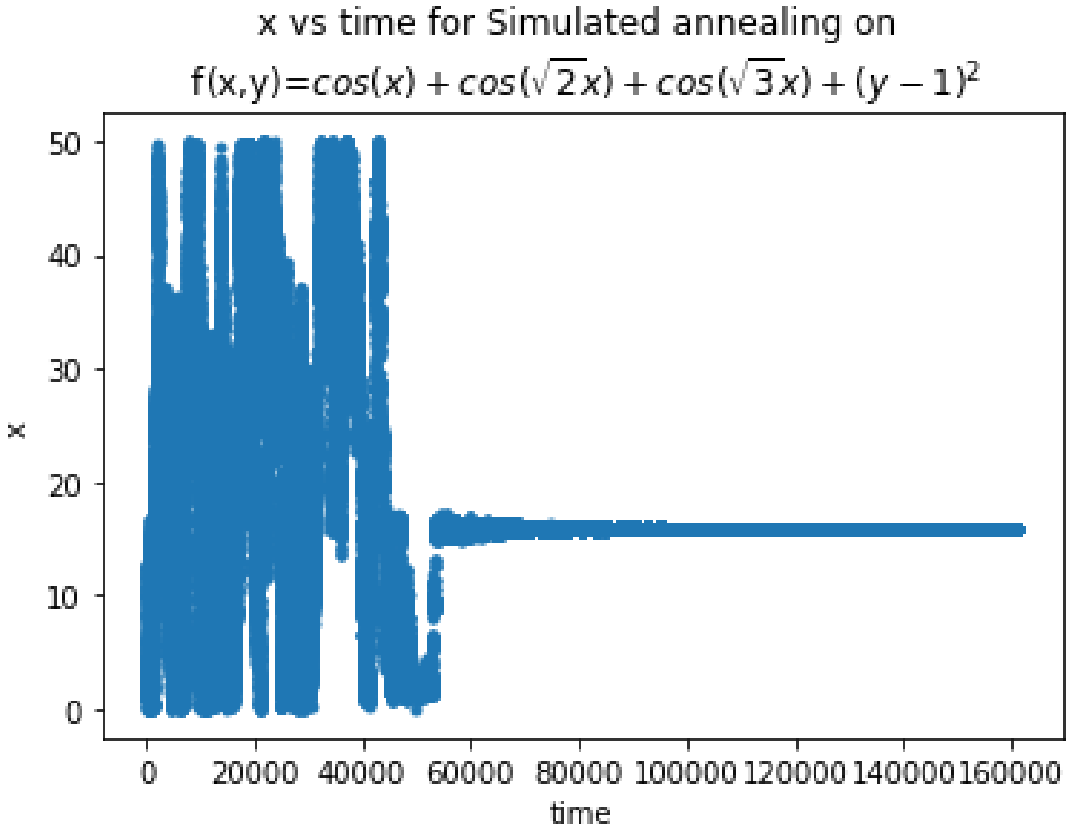

x vs time for Simulated annealing on
f(x,y)=$x^2 - cos(4\pi x) + (y-1)^2$
x
time
20
15
10
5
0
−5
−10
−15
−20
0
20000
40000
60000
80000
100000
120000
140000
160000

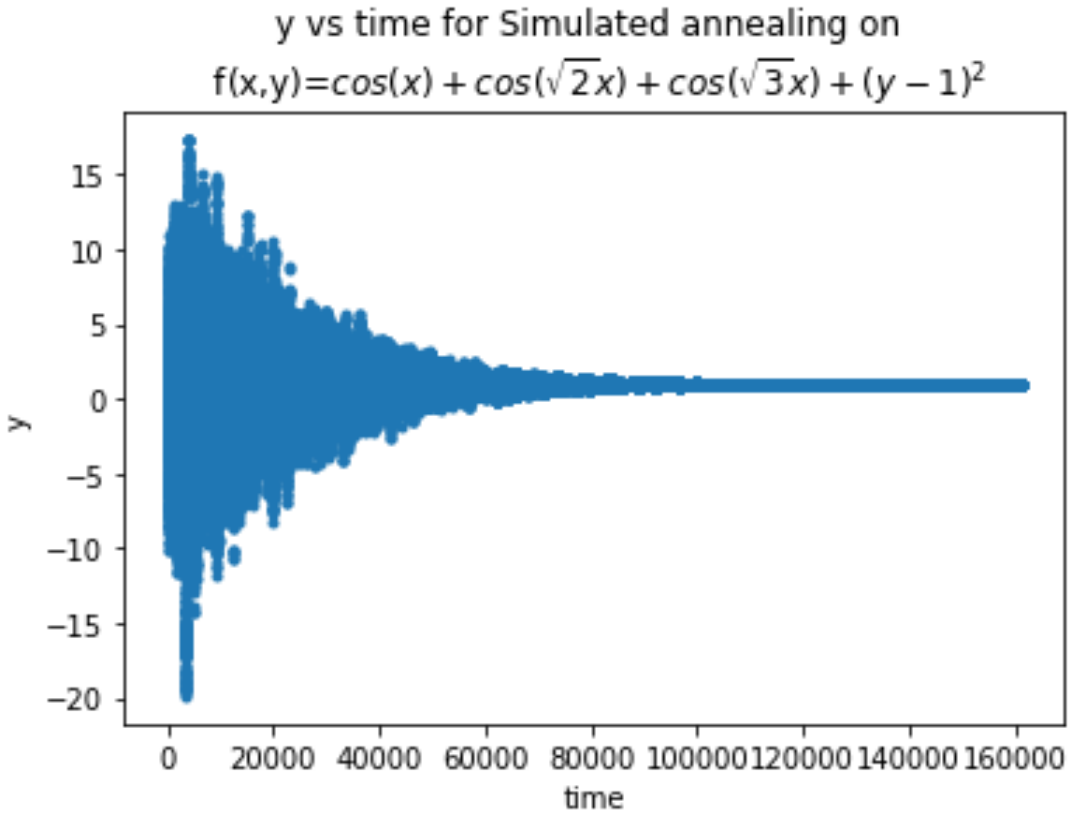


As shown in the y vs time and x vs time graphs above, x and y converge to x= -8.89526771e-04 and y=1.01567812 as time increases. This means that the global maximum of f(x,y)=$x^2 - cos(4\pi x) + (y-1)^2$ is about (-8.89526771e-04, 1.01567812) $\approx$ (0,1).

1.b.ii

Now we will use simulated annealing to find the minimum of f(x,y)=$cos(x) + cos(\sqrt{2}x) + cos(\sqrt{3}x) + (y-1)^2$ with the same max temperature, minimum temperature, and cooling rate as (1bi):


x vs time for Simulated annealing on
f(x,y)=cos(x) + cos(√2x) + cos(√3x) + (y − 1)²
x
time
0
10
20
30
40
50
0
20000
40000
60000
80000
100000
120000
140000
160000



As shown in the y vs time and x vs time graphs above, x and y converge to x= 15.95267043 and y=0.99567439 as time increases. This means that the global maximum of f(x,y)=$x^2 - cos(4\pi x) + (y-1)^2$ is about (15.95267043, 0.99567439) $\approx$ (16,1).

2 Ising model

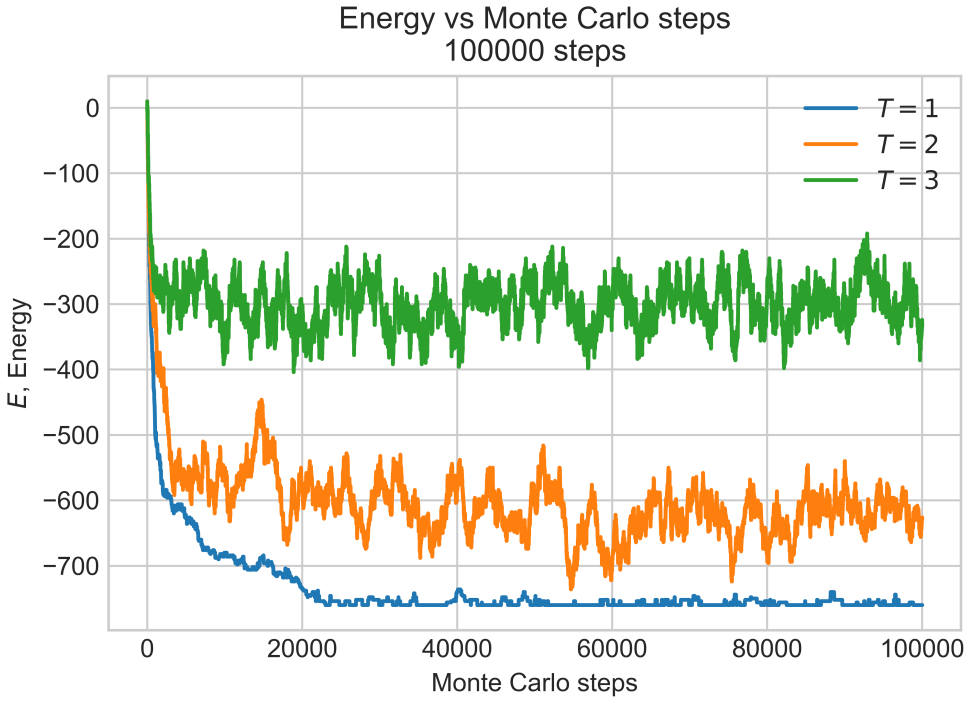
In this exercise, Ising model is going to be shown in three different temperature values after each of them looped 100000 times under the Markov chain Monte Carlo simulation with Metropolis probability. Magnet is modelled by up/down spins at each grid of the 20x20 lattice. Under the Metropolis probability, a random spin is chosen as the victim at each loop to be checked if a sign flip (spin down to up, up to down) going to be accepted.

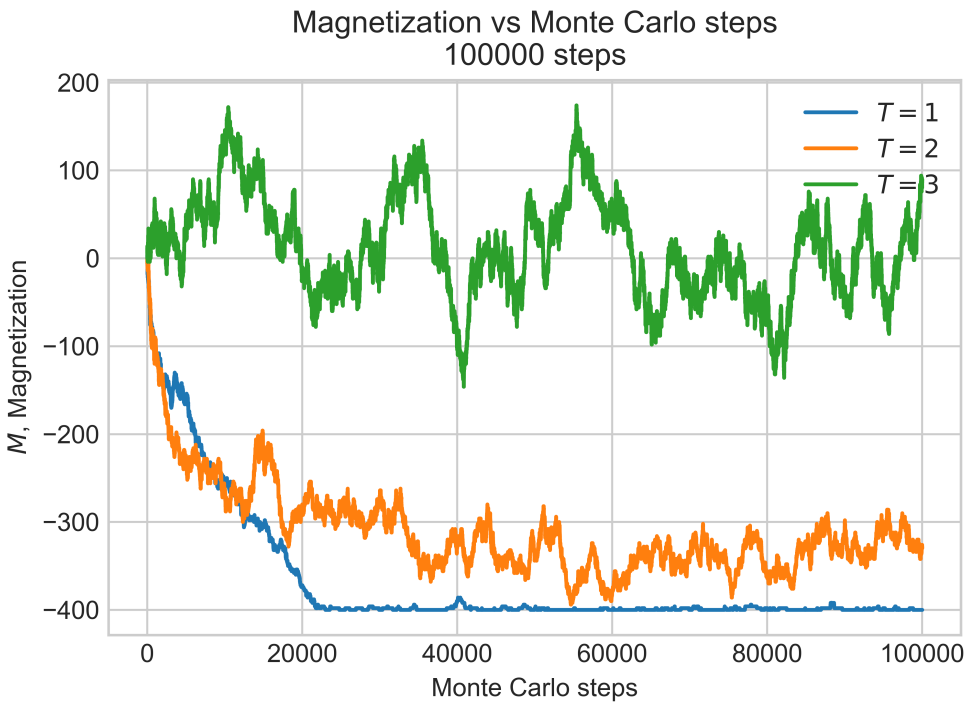
$$E = -J \sum_{<ij>} s_i s_j$$

and

$$M = \sum_i s_i$$

where E is energy and M is magnetization. In the code, `J=1.0, kB=1.0`. Below are the figures that show the final result of the simulation with three different `T` values: `1, 2, 3`:





It took 19.220 sec to do the loop for all three temperature values with 100,000 Monte Carlo steps. After running the program several times, it was seen that the Energy vs Steps graph remained roughly the same with decreasing stability with increasing temperature. Energy having a decreasing trend in to negative values was expected since looking at the formula, the s_i and s_j values are expected to align at the stability, which gives a deep energy value. But the Magnetization vs Steps graph was seen to be changing the most. In some executions of the program Magnetization graph reflected a positive plateau for `T=1`, but in this execution we see a negative plateau. `T=2` seems to be fluctuating a lot compared to `T=1` over an area of stability that is around $M = -350$. `T=3` is the graph that is fluctuating the most and does not seem to have a stability region. It is visible that the increasing temperature makes the magnet model less stable.

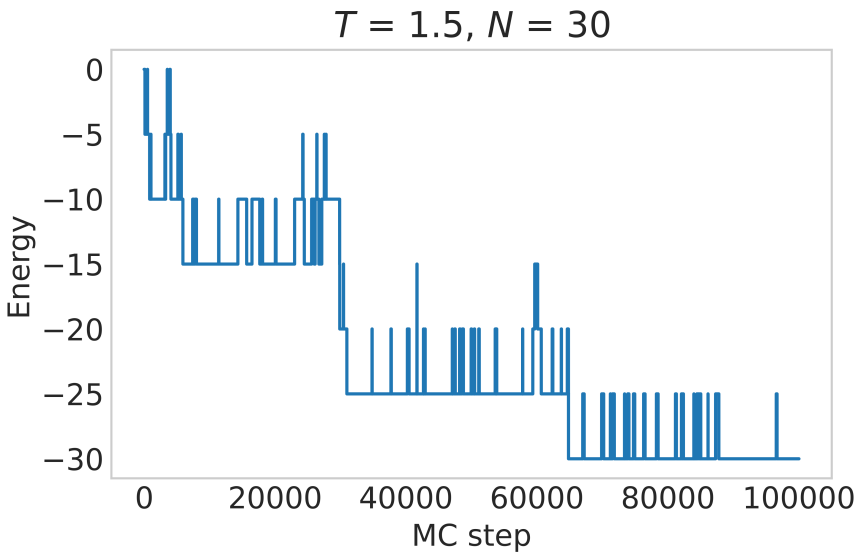
3 Protein folding

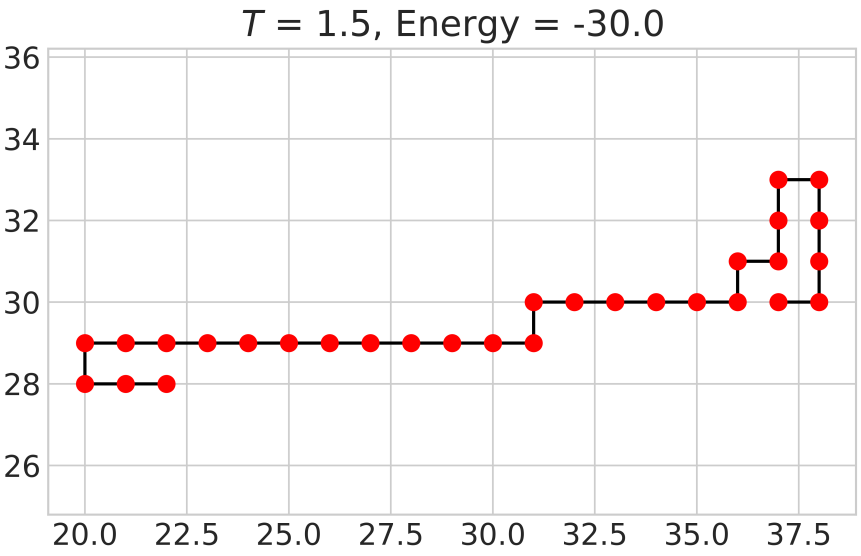
Monte Carlo Simulation and the Markov Chain/Metropolis method for Protein folding will be investigated in this exercise.

3.a

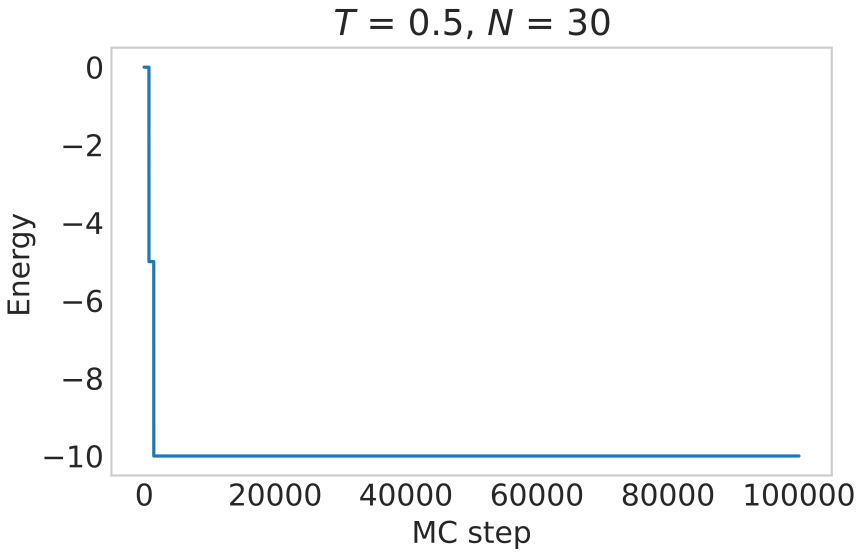
Program is run using `N = 30`, `T = 1.5`, ϵ `= -5`, `n` $= 10^5$ and then with `T = 0.5` and `T = 5`. Below are the figures:

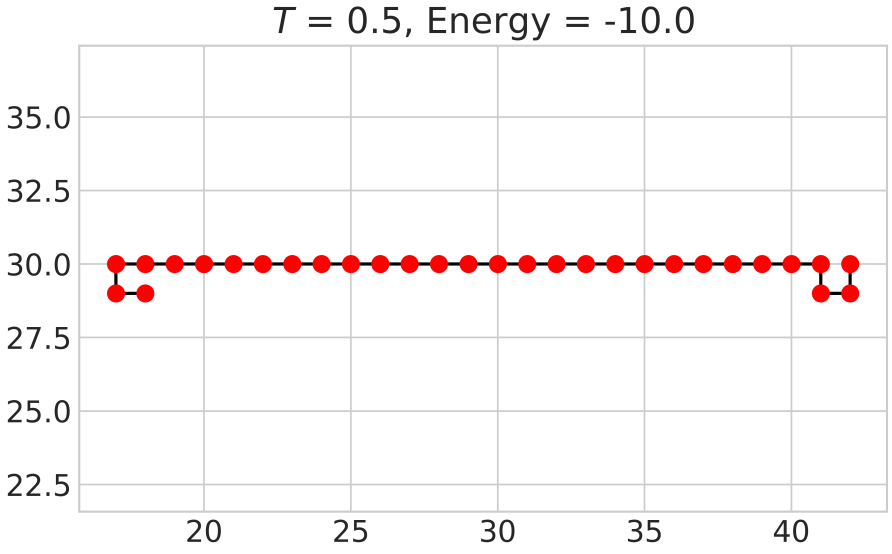
T=1.5, Energy averaged over last half of simulations is: -27.98:



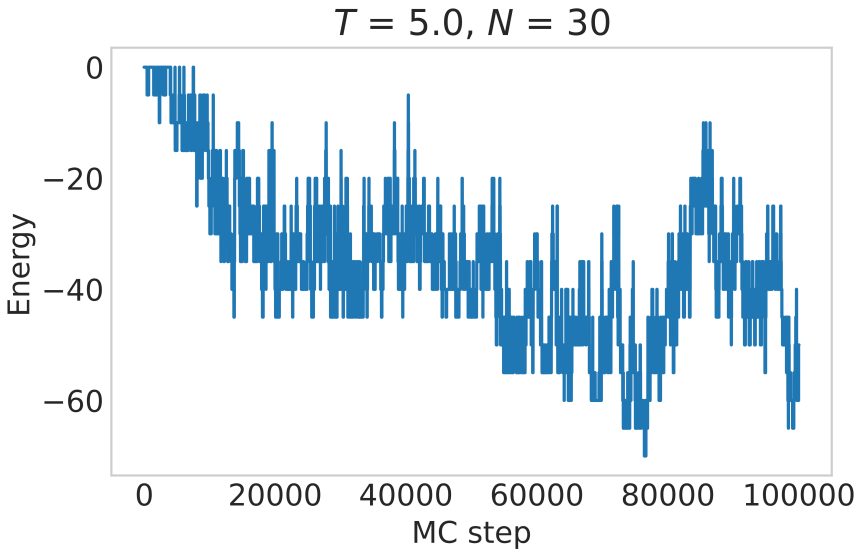


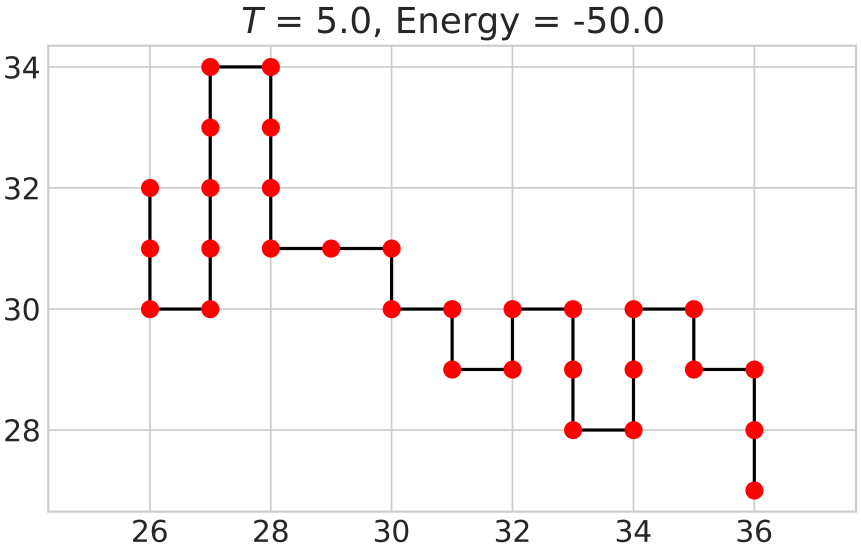
T=0.5, Energy averaged over last half of simulations is: -10.00:





T=5, Energy averaged over last half of simulations is: -43.97:



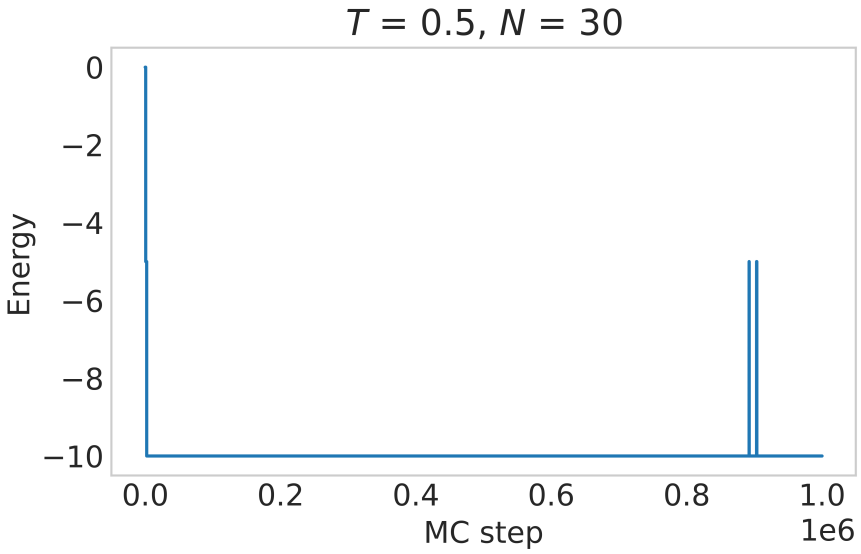


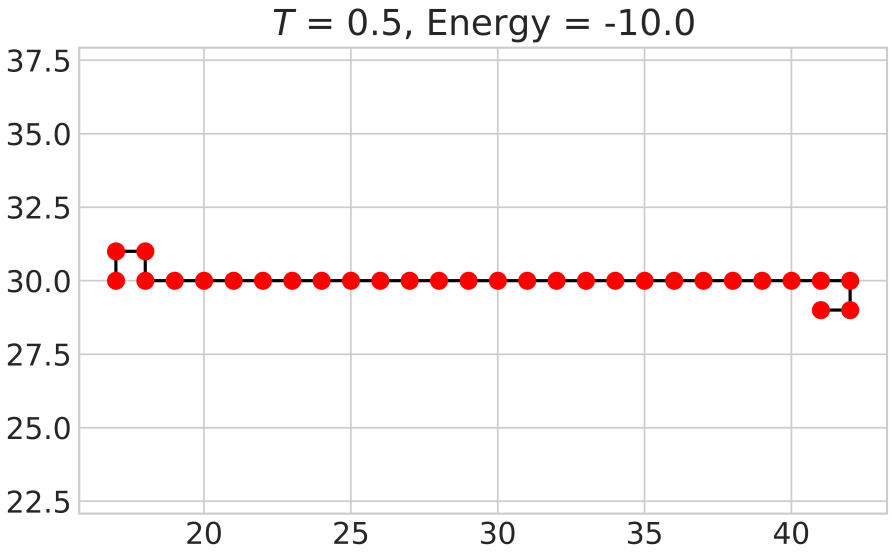
As can be seen, as temperature increases there is more folding. This is the because of the condition we have: as T increases, it gets more probable to make the move because the Boltzmann factor $(exp(-\frac{\Delta E}{T}))$ will allow more room for the folding. And as expected more folding means more negative energy. It is interesting to juxtapose the energy graphs as lower temperature graphs have a much less violent behaviour. With lower temperature, protein does not change substantially.

3.b

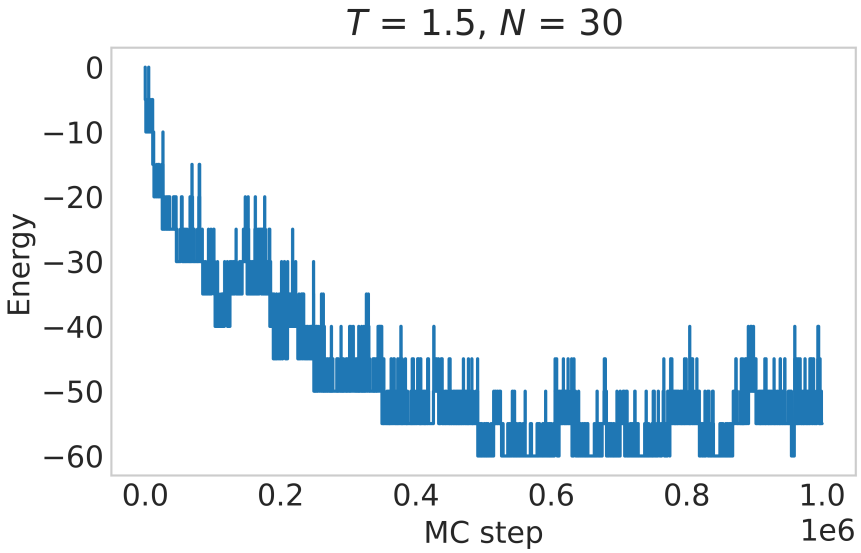
Now `n=1,000,000` and `T=0.5, 1.5`. Results are below:

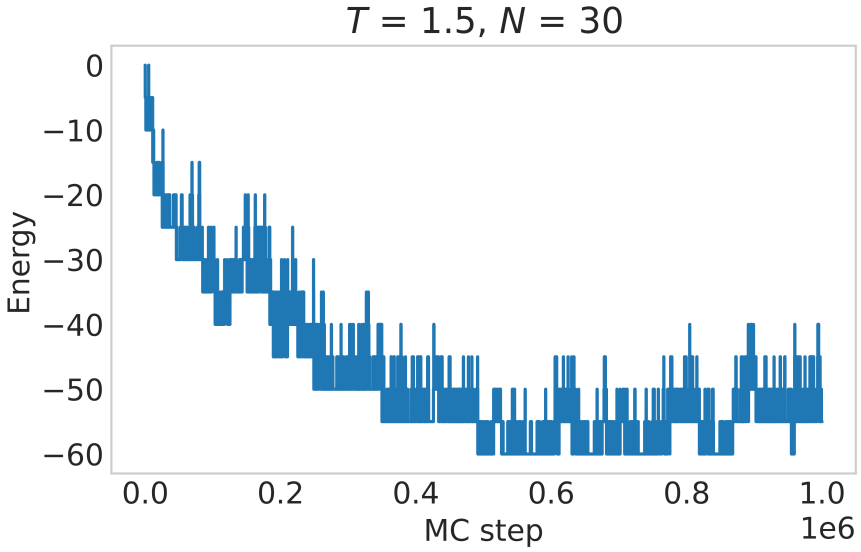
`T=0.5, Energy averaged over last half of simulations is: -10.00`





`T=1.5, Energy averaged over last half of simulations is: -56.55`

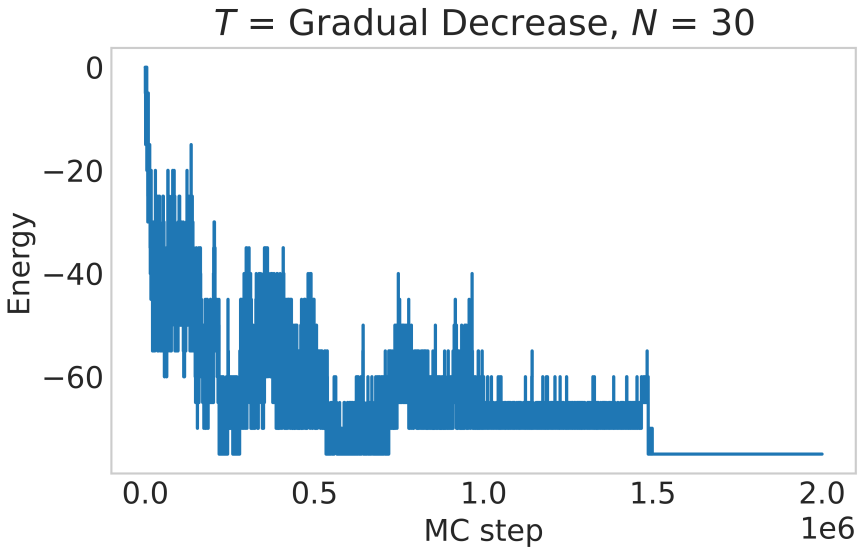


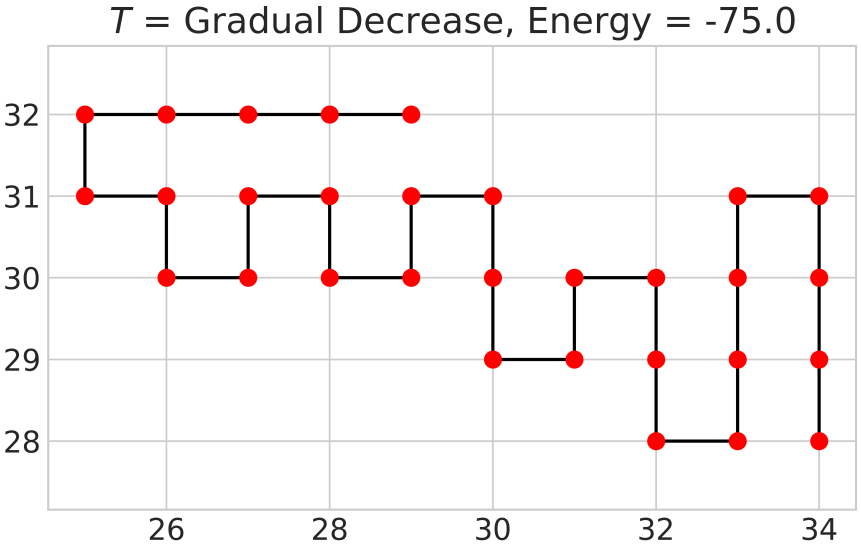


For the `T=0.5` case, increasing the number of steps (`n`) did not have a noticeable effect. Even the averaged energy value is still same. So there is no point in increasing the number of steps to seek more evolved folding in low temperature. For the `T=1.5` case, there is more evolution. The averaged energy nearly doubled and there is a noticeable increase in the folding. So allowed the step size, protein folds much more in higher temperature.

3.c

Now temperature is decreased by `Tsteps = 4`, and `Tf=0.5` and `n=2*10^6`. Below are the figures:



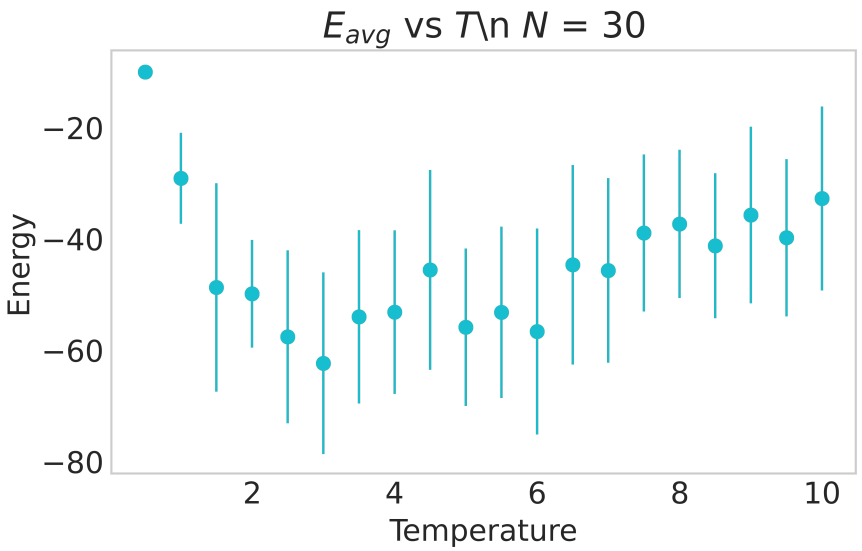


Energy: `Energy averaged over last quarter of simulations is: -75.00`
This energy value is clearly way different than $E = -10.0$ we had before, this is because protein was allowed to be folded in higher temperatures before hitting

$T = 0.5$, so more folding means more energy accumulation. This can be also observed on the energy graph as well, in higher temperatures there were violent behaviour compared to the last quarter of the values (coincides to $T = 0.5$).

3.d

With starting temperature of 10 until Here is the figure:



Looking at the sharp jumps in that average energy values make against the temperature axis, it can be inferred that there is evidence for phase change. At $T = 4.5$, the jump in the values is clearly observable.