Lab11 Report

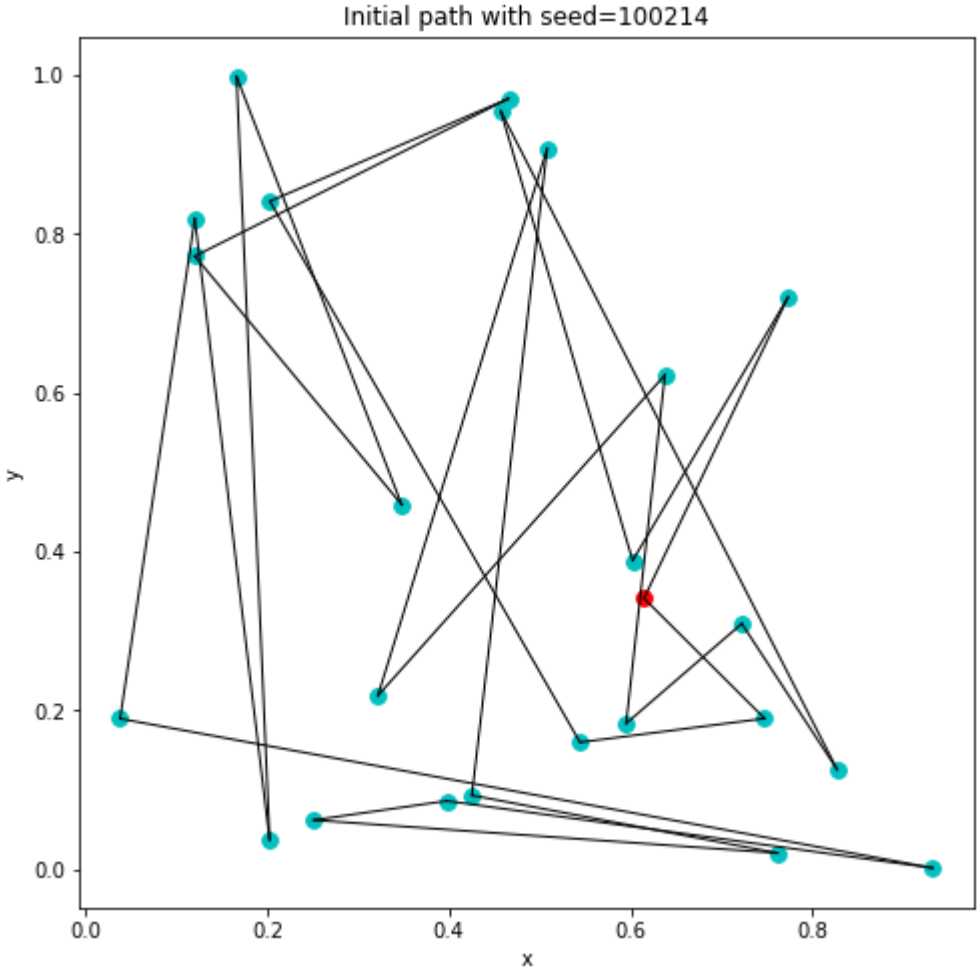
Zirui Wan (Question 3), Rundong Zhou (Question 1, 2)

December 7, 2020

1 Question 1, Simulated Annealing Optimization

1.1 Q1a, Travelling salesman problem

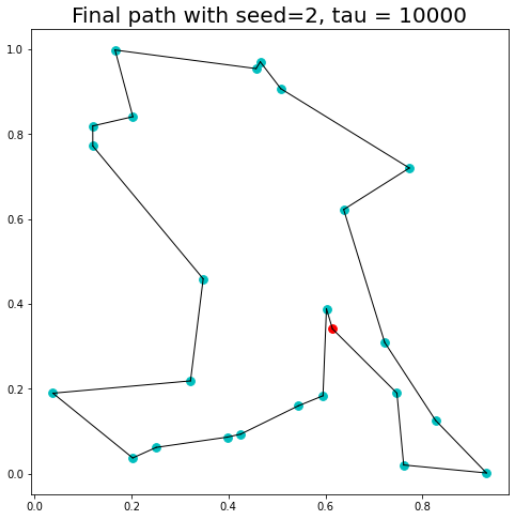
I chose 100214 as my seed to generate the map, and you can tell from the plot, the initial path is very chaos and not cost-efficient at all. The red dot is where the salesman starts his journey.
The initial distance travelled is 12.813208263562485.

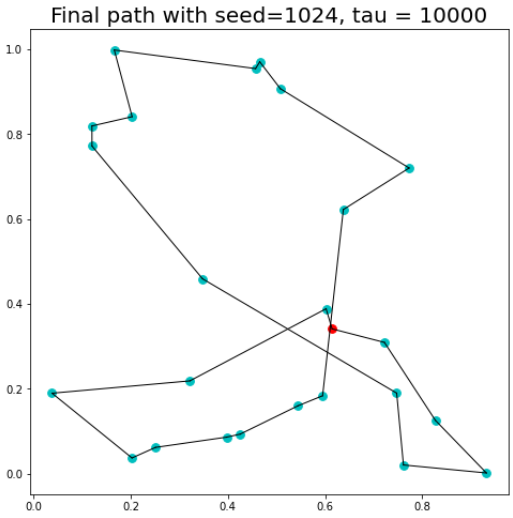


Now I arrange the seed with the fashion of powers of 2 and obtain the following data set with $\tau = 10^4$.

Seed	Optimized distance D
2	4.232287012102024
4	4.232326880805929
8	3.9259137269305
16	3.9259535956344043
32	4.645014039968084
64	4.582507727020363
128	4.252921786665023
256	4.239757144818262
512	4.531841705578827
512	4.531841705578827
1024	4.621959268330134

The average optimized distance D with default time constant $\tau = 10^4$ is 4.319 and the standard deviation is 0.254.
We can tell this is a pretty consistent set of data. Taking different path does not vary the result very much. The following plots are two selected optimized paths with $\tau = 10^4$.

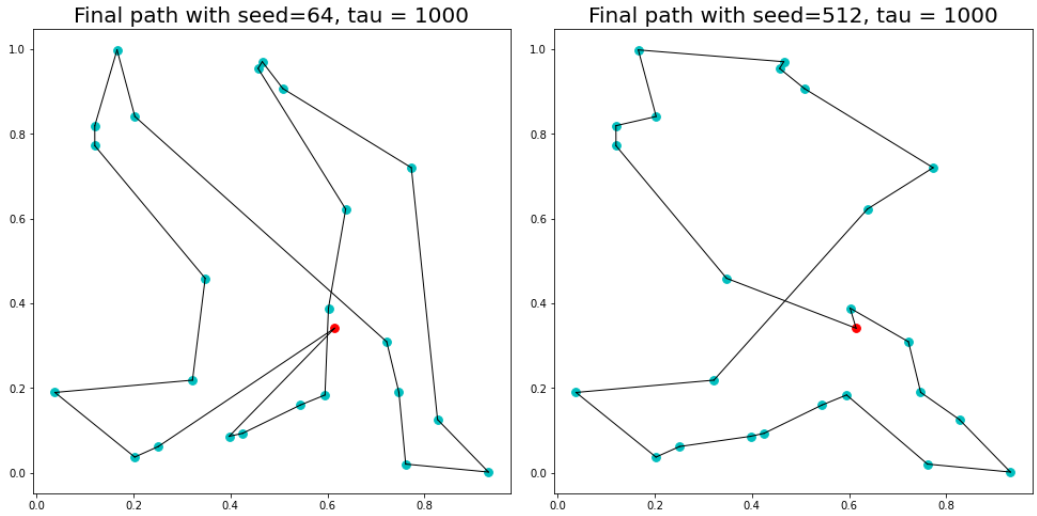




Now use this list of seeds for different paths, I alter the time constant τ.
For $\tau = 10^3$, we have:

Seed	Optimized distance D
2	4.725738274262915
4	4.520452063043142
8	4.556944958978509
16	5.282289890317243
32	4.451429632653655
64	5.806974570037571
128	4.220650411249977
256	4.595977321562188
512	4.2832405807723015
1024	5.221949073965177

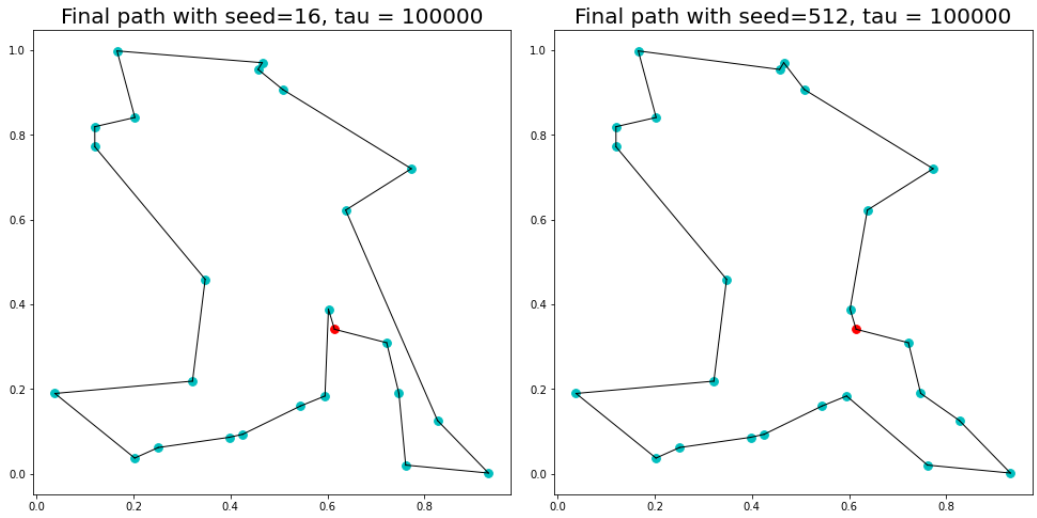
The average optimized distance is 4.767 and the standard deviation is 0.483.
The following plots are two selected optimized paths with $\tau = 10^3$.



Now for $\tau = 10^5$, we have:

Seed	Optimized distance D
2	4.187146752580037
4	4.247309030785868
8	4.2039120957300256
16	4.26116254904442
32	4.099966589323794
64	4.49063447480239
128	4.319550873376077
256	3.9259535956344056
512	3.925913726930501
1024	4.395663790342732

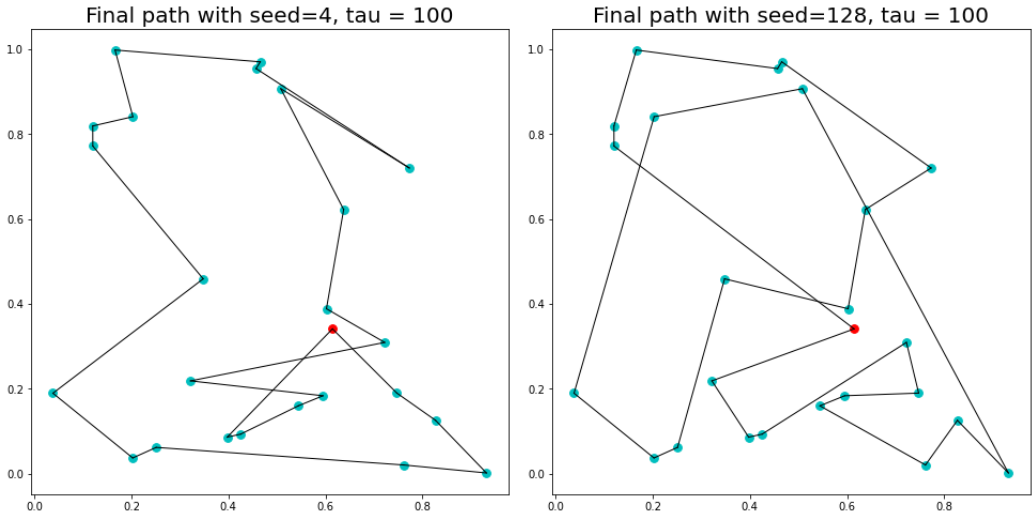
The average optimized distance is 4.206 and the standard deviation is 0.174.
The following plots are two selected optimized paths with $\tau = 10^5$.



We can tell from the graphs that with $\tau = 10^5$, the paths are very nicely arranged.
Since with $\tau = 10^5$, the program already takes about 10 minutes to run, I cannot go with higher τ, so the last time constant will be $\tau = 10^2$:

Seed	Optimized distance D
2	6.24229282410968
4	5.49232490466663
8	5.256744963003345
16	6.7146730027994215
32	6.086201342329616
64	5.266452120889646
128	6.677799831441298
256	6.344238911514615
512	5.872578254197915
1024	5.671782912004698

The average optimized distance is 5.963 and the standard deviation is 0.511.
The following plots are two selected optimized paths with $\tau = 10^2$.



We can tell that these paths are pretty chaotic.
With four different time constants, we can form the following table:

Time constant τ	Average optimized distance	Standard deviation D
10^2	5.963	0.511
10^3	4.767	0.483
10^4	4.319	0.254
10^5	4.206	0.174

We see that with larger time constant τ, the optimized distance and standard deviation are both improved. But consider the run time, $\tau = 10^4$ would be the most cost-time efficient selection.

1.2 Q1b, Find the global minimum of a complex function with simulated annealing

1.2.1 Optimize the function $f(x, y) = x^2 - \cos(4\pi x) + (y-1)^2$

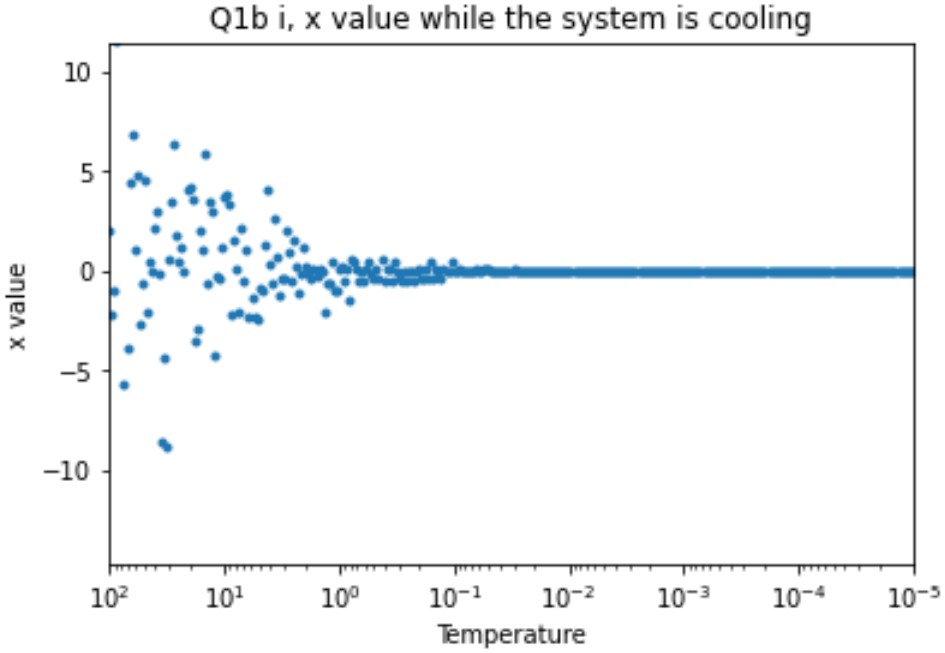
With several attempts, the following are the optimized parameters I find:

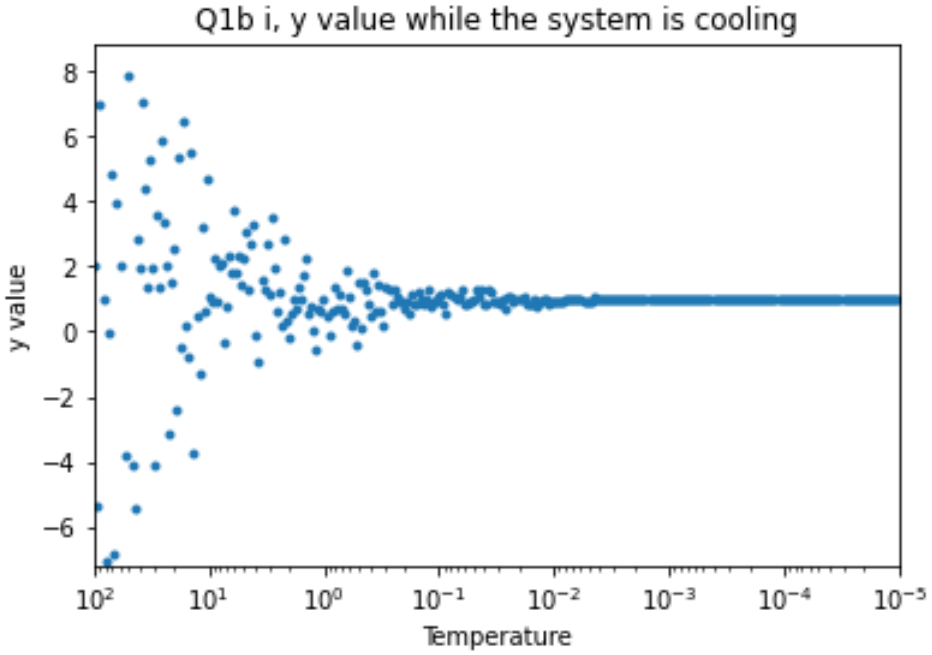
- $T_{max} = 100$
- $T_{min} = 10^{-5}$
- $\tau = 10^4$

I start the simulated annealing from $(x, y) = (2, 2)$. And the optimized x, y values are:

```
Optimized x value is: 0.001324614020030701
Optimized y value is: 1.008100099433876
```

I record x and y values every 500 iterations, and plot them against the temperature, I arrange the temperature axis in log form.





From the plots, we can tell that the x and y values vary a lot at the beginning

of the annealing, then they soon converge to the optimized values. Also, the value obtained from the simulated annealing is very close to the correct answer.

1.2.2 Optimize the function $f(x, y) = \cos x + \cos(\sqrt{2}x) + \cos(\sqrt{3}x) + (y-1)^2$

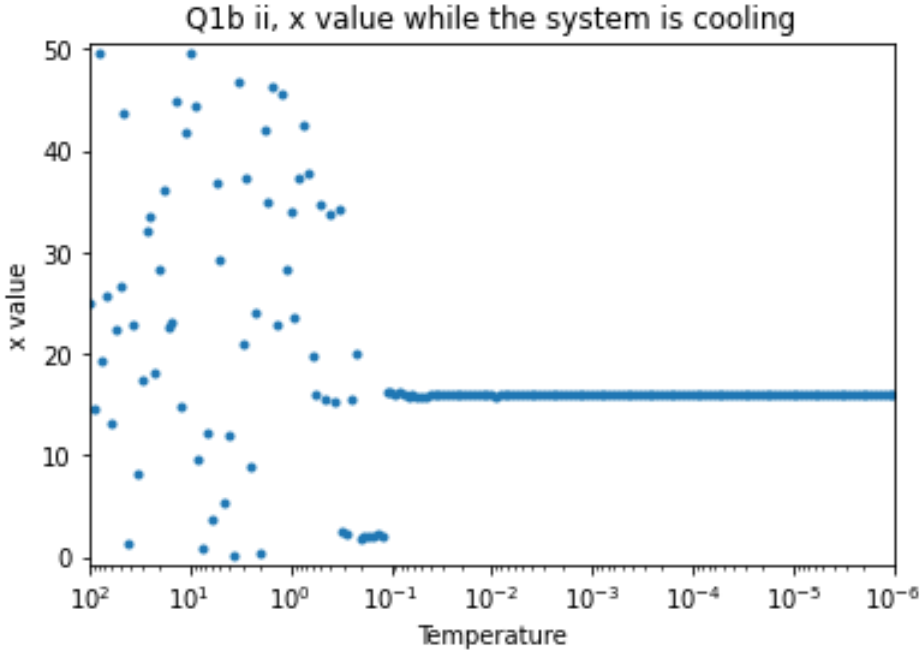
This time, I choose the parameter to be:

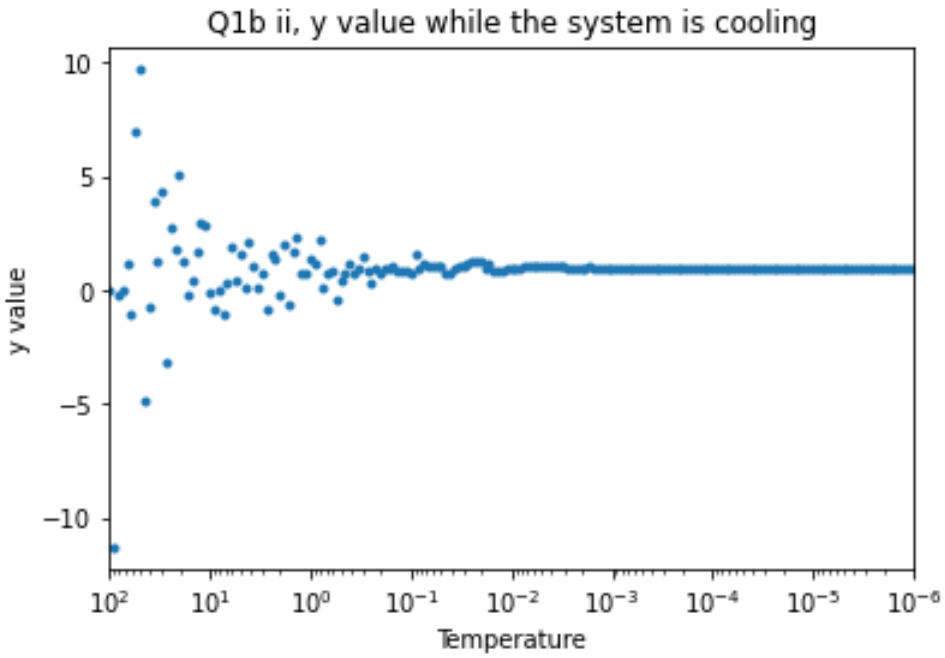
- $T_{max} = 100$
- $T_{min} = 10^{-6}$
- $\tau = 10^4$

I start the simulated annealing from $(x, y) = (25, 0)$, which is the middle point of the domain. The result is:

```
Optimized x value is: 15.936919786168444
Optimized y value is: 1.0063088671509135
```

I record x and y values every 1000 iterations, and plot them against the temperature, I arrange the temperature axis in log form.



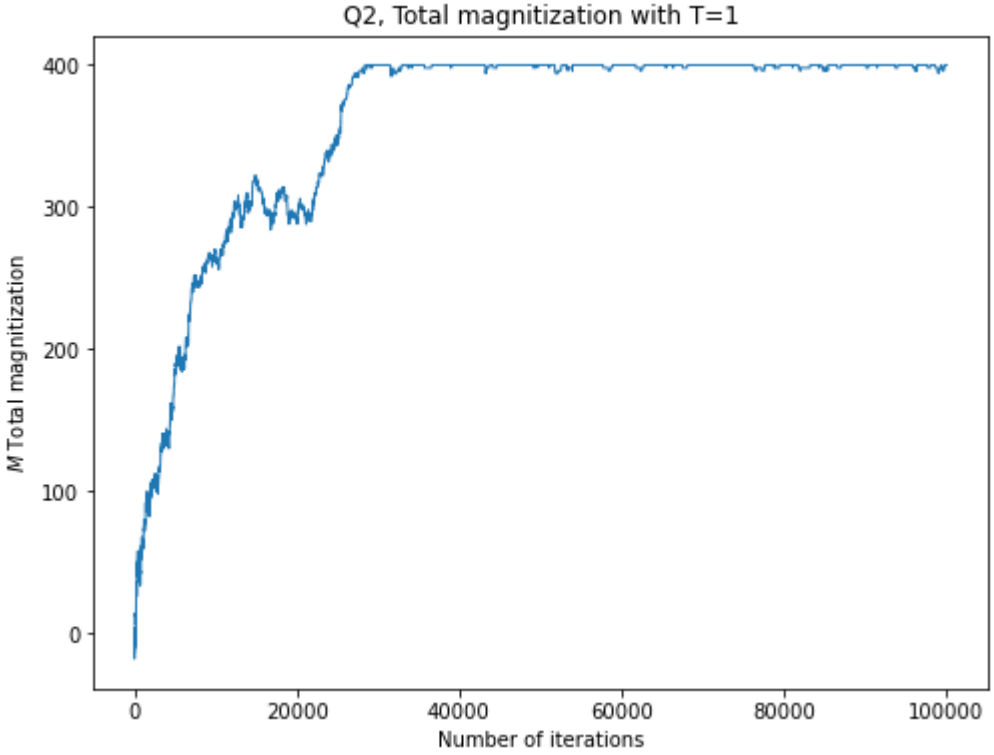


The plots show a similar pattern as the previous question. And the optimized values are close to the correct answer.

2 Question 2, Ising model

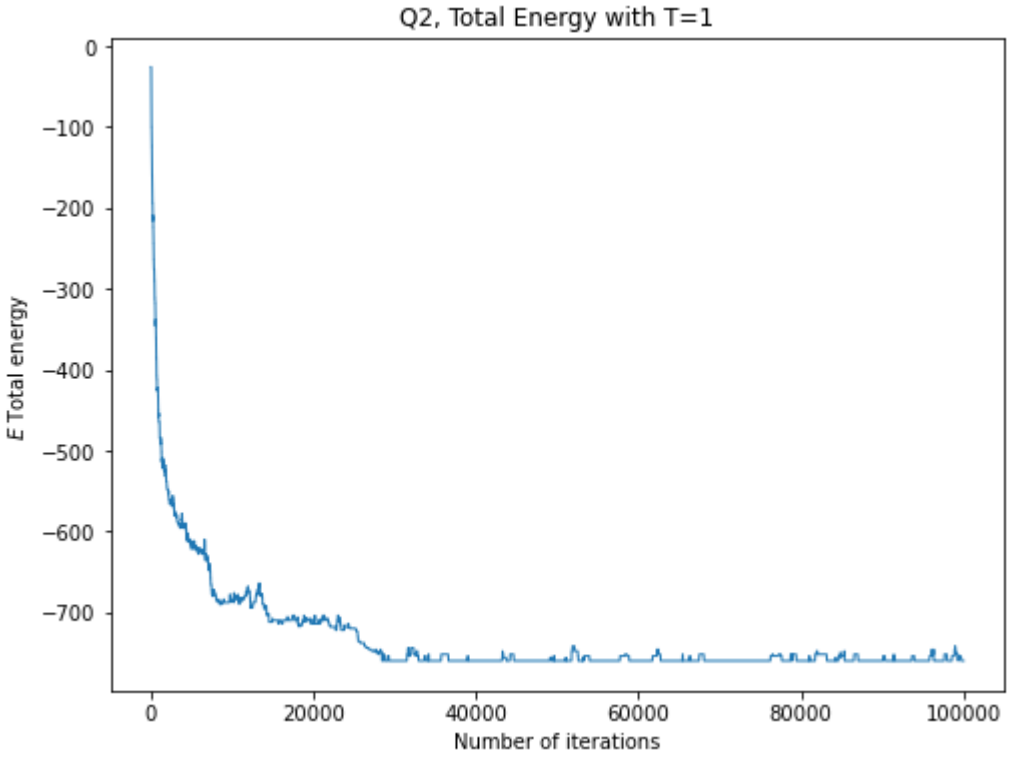
2.1 Q2c, Plot of the total magnetization

So here I present the plot of the total magnetization of one instance generated by the program:



From the plot, we can tell that over time, the system develops a spontaneous magnetization. Towards the end, the whole system is magnetized in the same direction, and the total magnetization is 400.

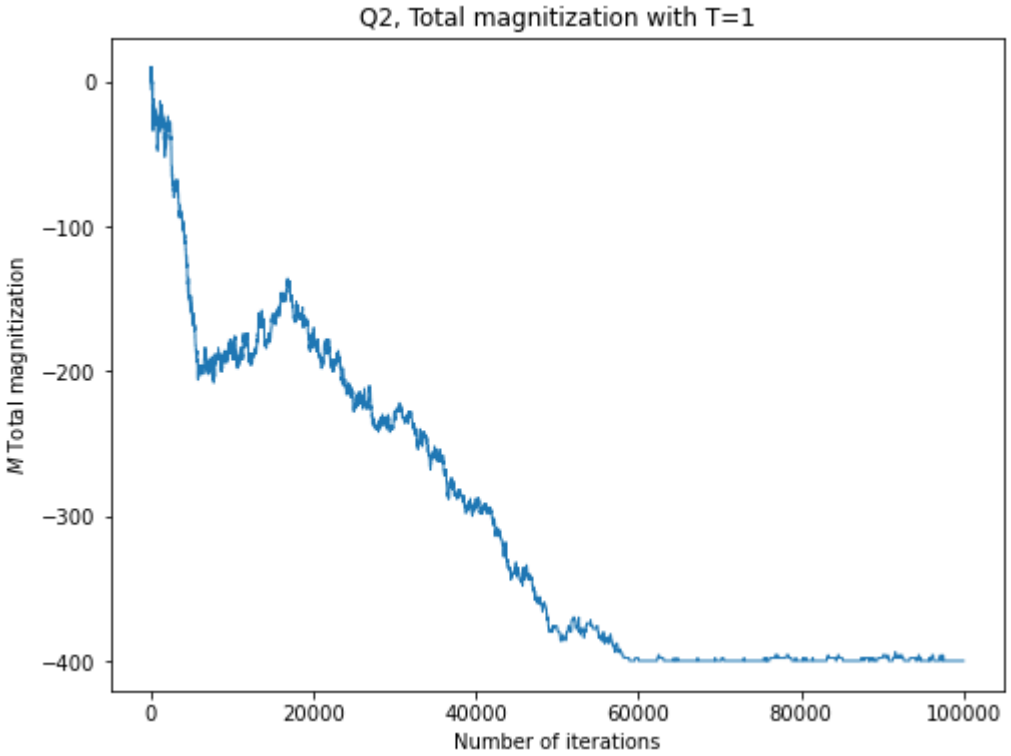
For fun, I also plot the total energy of the system:



We can see, the system tends to keep itself at the lowest possible energy.

2.2 Q2d, Run the program several time and observe the sign of the magnetization

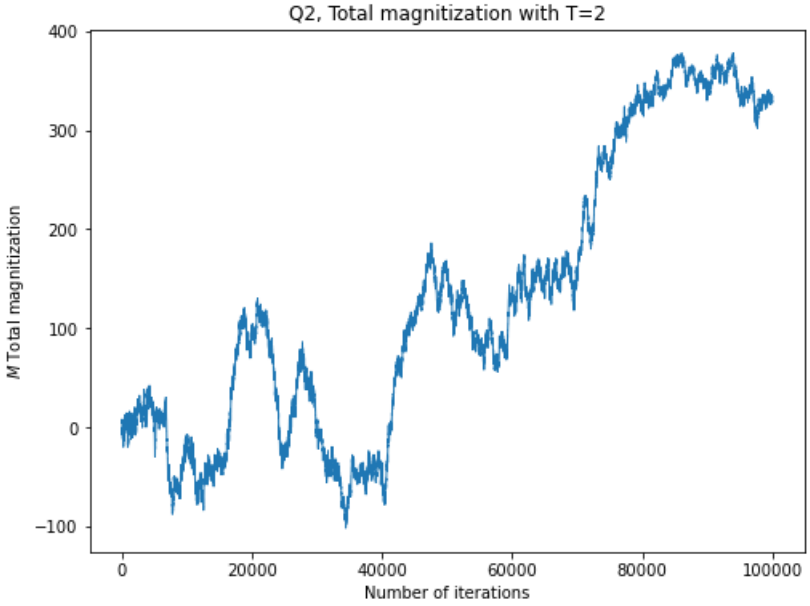
So I run the program several time, and sometimes, the total magnetization tends to develop towards the opposite direction, like the following plot:

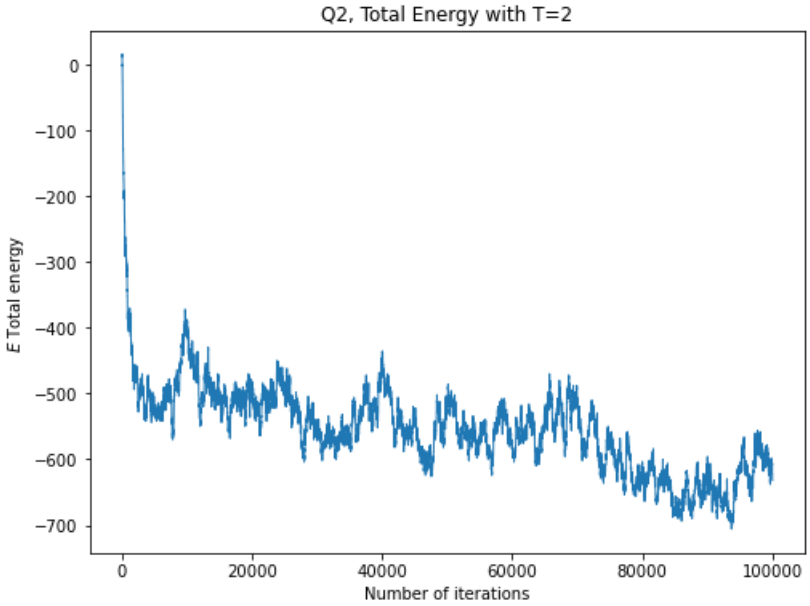


Since the program is a random process, it is natural for the total magnetization to develop towards different directions. But it will always end up at one end, positive or negative, due to the nature of ferromagnetic. The system tries to stay the the lowest possible energy state.
An interesting observation is that the sign development seems depend on the initial total magnetization. If the initial sign is positive, the system is more likely to develop positively and vice versa. But since it's all random, it is also possible for a positive initial condition to end up negative.

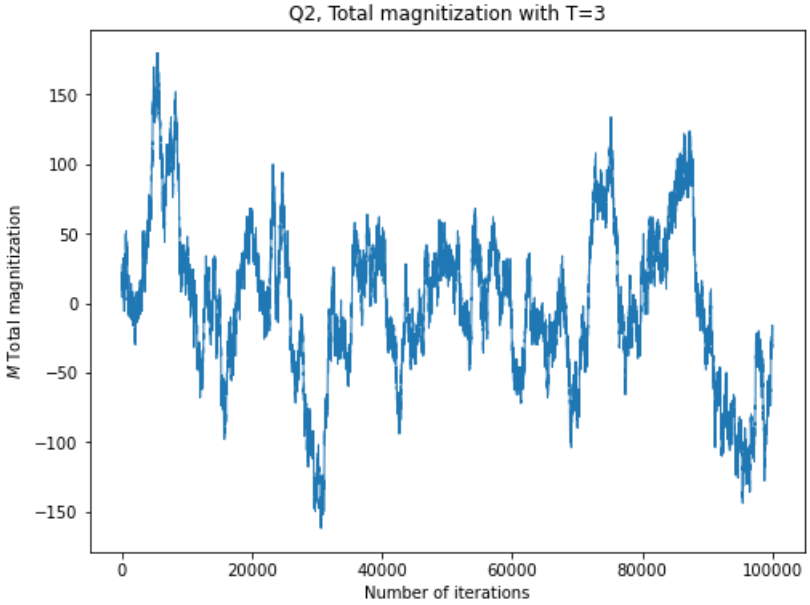
2.3 Q2e, Vary the system temperature and Animations

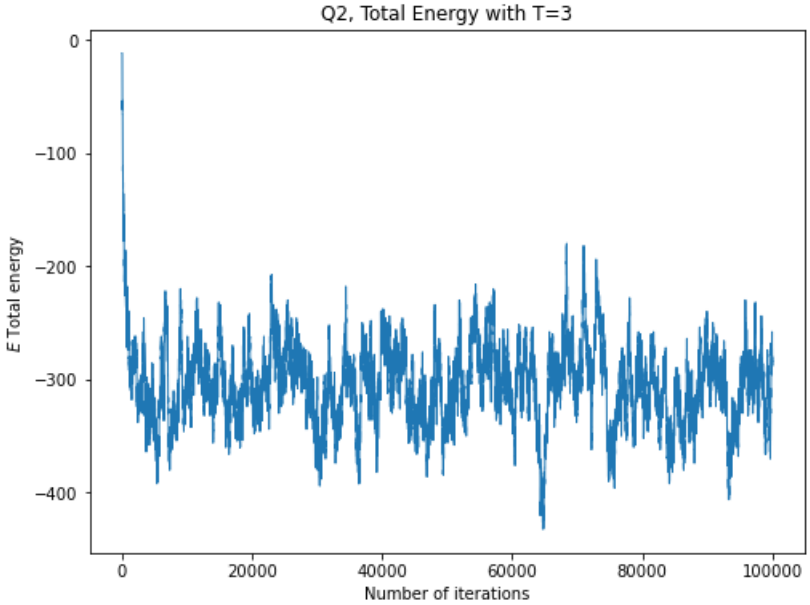
So I run the program at $T = 2$ and 3. For animations, please refer to the file 'Q2_T=1.gif', 'Q2_T=2.gif', and 'Q2_T=3.gif'
I plot the total magnetization and energy for both $T = 2$ and 3.
For T=2:





For T=3:





From the plots, we can tell that if we increase the temperature, the system will struggle at developing a uniform magnetization. At T=2, there is a trend, but it fails to keep at max magnetization as it does at T=1. At T=3, we cannot even see a trend of magnetization.

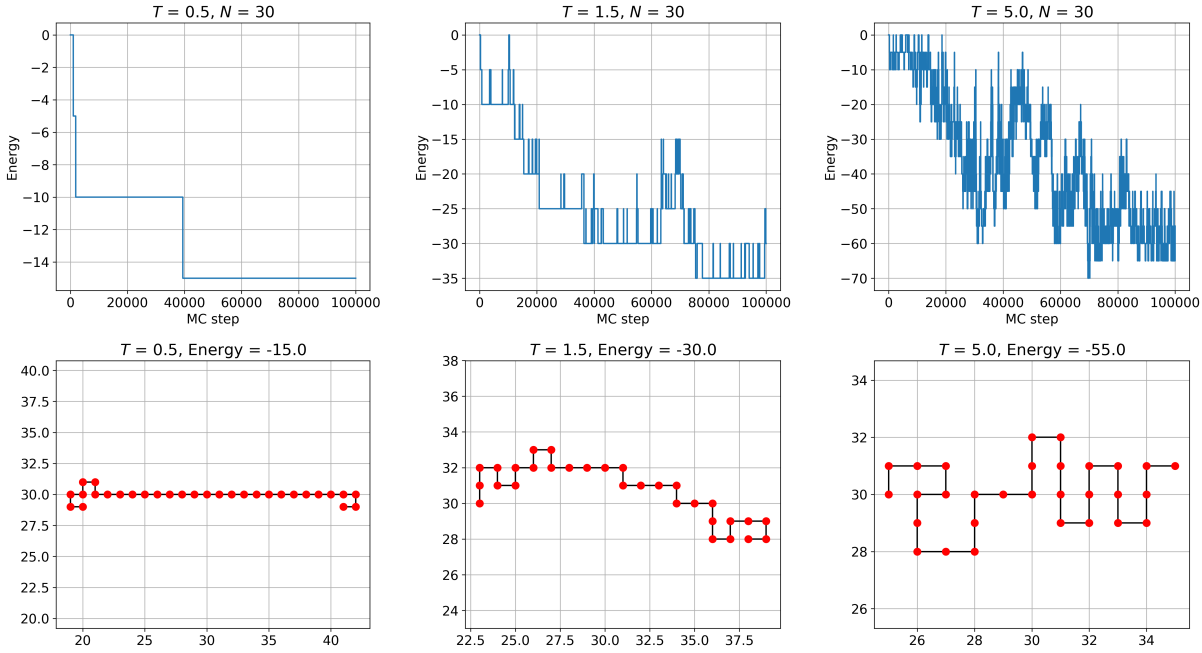
At the energy plots, we can see that the lowest energy the system can reach is

increased at T=2 and 3. Also the plots become very wavy when the temperature is increased.
If you look into the animation, you can see that at T=1, the system converge to a uniform magnetization quickly, and in the end, all the particles are magnetized in the same direction.
At T=2, the system struggles a little bit, but it shows a trend to develop a uniform magnetization.
At T=3, the system is quite chaotic, and no sign of unification can be seen.
This can be confirmed by a common knowledge, if we heat a magnet, it will lose its magnetization and become neutral.

3 Q3

We are asked to simulate protein folding in a 2D lattice in this question.

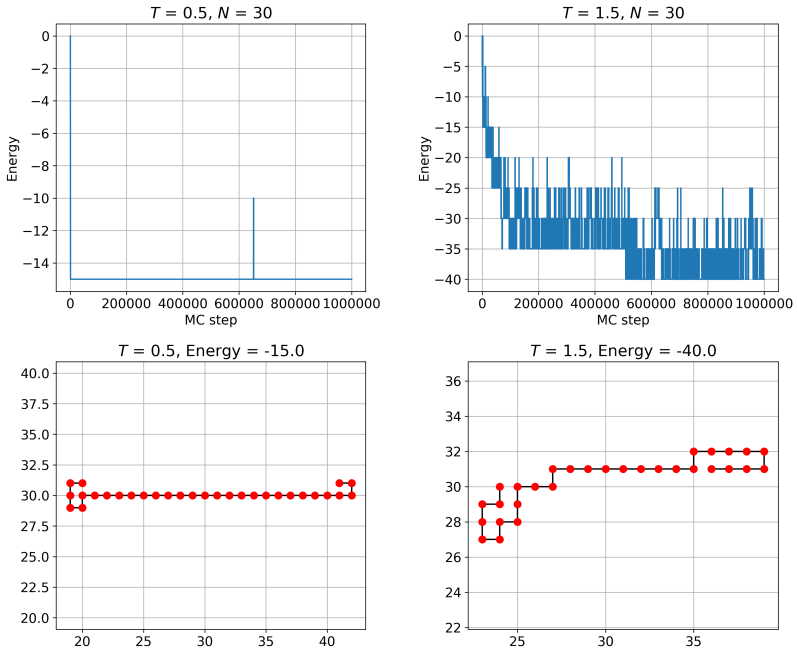
3.1 Q3 (a)



Outputs for the 3 cases with T=0.5, T=1.5 and T=5.0 are shown above. We can conclude that, as the temperature increases:

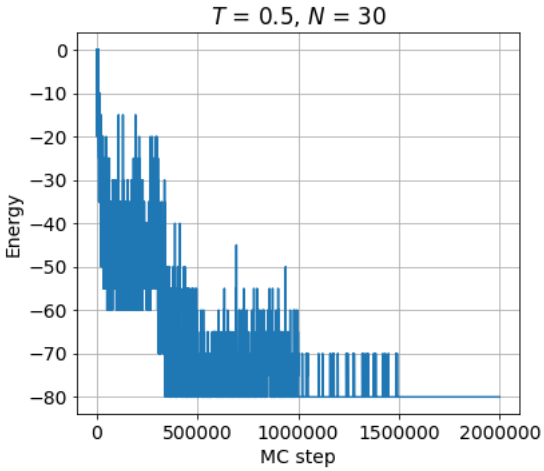
1. the protein tends to fold more
2. the lowest energy of the system decreases, as there are more adjacent unconnected monomers
3. it takes more Monte Carlo steps for the system to stabilize

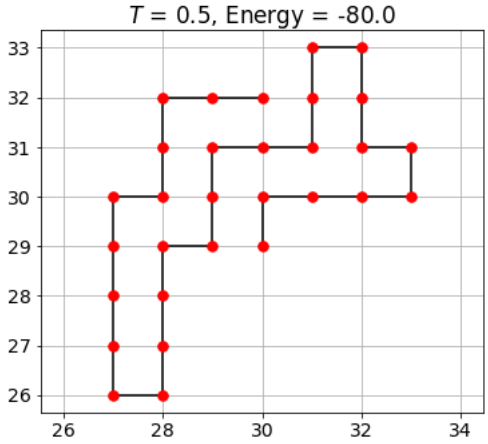
3.2 Q3 (b)



Outputs for the T=0.5 and T=1.5 cases are shown above. The typical energy of the protein is -15.0 for T=0.5 and -40 for T=1.5. This actually makes sense. Because at higher temperature, the monomers would have higher kinetic energy. And the protein is then able to do more folding. As a result, there would be more adjacent unconnected monomer pairs, and the potential energy of the protein system becomes lower.

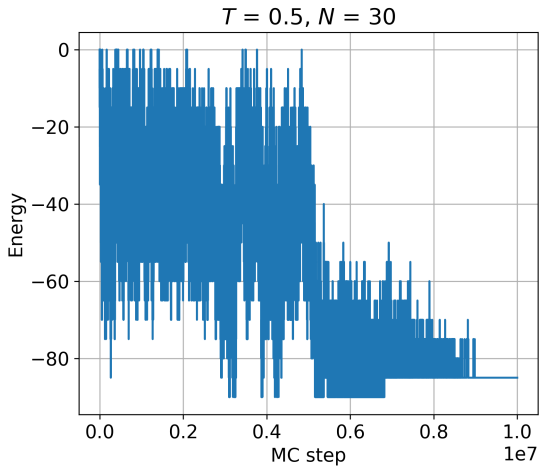
3.3 Q3 (c)

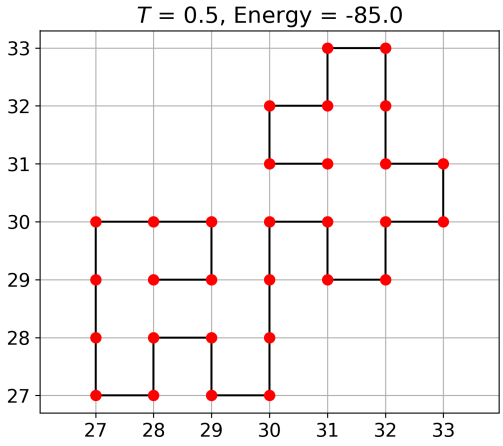




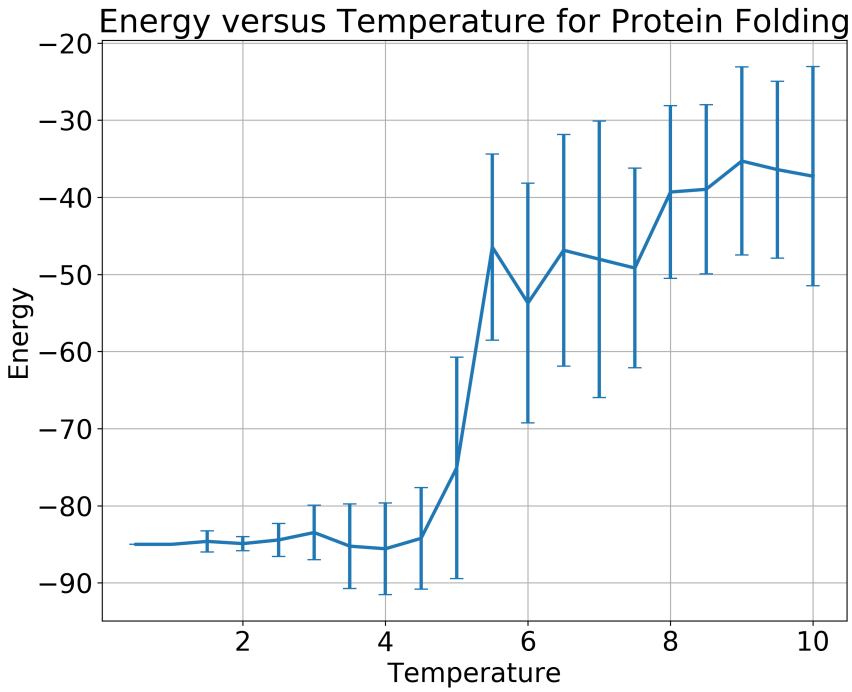
The output for the protein folding with simulated annealing is shown above. Compared to the one without simulated annealing in Q3 (b), the typical energy decreases from -15.0 to -80. This makes sense because simulated annealing is helpful for finding the global minimum.

3.4 Q3 (d)





The energy convergence and the final protein are shown above. Although the simulated annealing is even slower now, the typical energy of the protein is not too different from before. This means that the global minimum of the protein energy is just around -80 -90.



The energy versus temperature plot is shown above. You can see that clearly there is a phase transition. Around T=5.0, the protein energy suddenly jumps from -85 to -45. This means that T=5.0 is the threshold for this protein to transition from a low energy regime to a higher energy regime. Also, the error bars in the higher energy regime are much larger than those in low energy regime. This also makes physical sense because protein in higher energy regime must be less stable.