Fine-tuning the Simulated Annealing method for Traveling Salesperson and the Thomson Problem

(Faster optimization through the use of appropriate temperature values and Markov Chain length)

Maarten Both, 10801871, Wenlin Chen, 12649678,

***Abstract*—In this study, Simulated Annealing is implemented to find optimal solutions to both the Traveling Salesperson problem and the Thomson problem. For a specific TSP problem of 280 locations, SA was able to converge to a local optimum and for the Thomson problem with a range of particles from $N = 1$ to $N = 200$ the global optimum was found. In this paper we discuss the different parameter settings and their influence on the convergence rate towards the optimal solution. Both the use of different cooling schedules and the influence of the Markov Chain length are investigated. Some remarks are made on the computational costs of SA and the Markov Chain length that is used for each iteration. Using a longer Markov Chain length in the SA algorithm leads to better optimal solutions, but may not be feasible because of the increase in computation time. Thelocally optimal solution for the TSP problem that we found had a distance 2722.36 units. The Marko**

***Index Terms*—Simulated Annealing, Optimization, Traveling Salesperson Problem, Thomson Problem**

I. Introduction

SIMULATED ANNEALING (SA) is a search algorithm that is heavily inspired by the way in which nature works. It makes use of a method that is based on the way crystal structures are generated in metals, by heating the metal to a high temperature and subsequently cooling it down slowly. This process minimizes the free energy between the atoms, which induces the crystal structure in the metal. Instead of the in SA the desired cost function is minimized such that the optimal solution is found. The fact that 'simulated annealing' shows up in google scholar for almost two million search results, suggests that it is a method that is widely used in the field of research. In this study, the focus is more on the method of SA itself and the parameters involved, rather than on the specific problem implementation. We use SA to find the optimal solution in a 280 city TSP and to solve the Thomson Problem for multiple number of particles. For both these problems we elaborate on how we chose our parameter settings, which involve both the Markov Chain length and the temperature parameters of SA. While the implementation of SA in other types of search problems will lead to different results, the arguments that we use to determine our parameter settings will still hold. Therefore this study might especially be of help for inexperienced users of SA.

II. Concept and Method

A. Optimization problems

For any problem, optimization is to find the best solution from a set of possible solutions , also called the search space. This comes across with a search for the solution with the smallest cost function (eg, shortest distance). Usually there are lots of feasible configurations for a solution, using algorithms for iterating between solutions and generating new ones can help a lot. The complexity of the search space is problem specific and should thus always be investigated when an optimization algorithm is implemented.

1) Greedy Algorithm: A Greedy Algorithm is an intuitive and common method in optimization. It compares and selects the best solution for the next step based on the current state, and hereafter it can not return to the worse solutions. For smooth cost functions, this principle of optimization will usually lead to the global optimum. However sometimes the Greedy Algorithm can not return the "best" optimal compared to a Dynamic Algorithm, since the previous steps eliminate exploration of different regions in the search space, which makes it impossible to test all the solutions.

For the travelling salesperson problem, the Greedy Algorithm is helpful. In the current city, the next stop should be the nearest city. But to find the global optimal, some improvement like adding stochasticity to algorithm is needed.

2) 2-opt Algorithm: The 2-opt algorithm, which is first proposed to solve TSP problem, is a kind of local search algorithm. The idea behind it is to reorder the part of routine which crosses itself, in such a way that is no longer does.

In this study we use this method such that two elements i_{th} and k_{th} are chosen randomly from the sequence containing the traveling order of cities. The route within these two elements are swapped to generate a new solution. For example, there is a list of seven letters below, now we randomly select a segment and reverse this part.

$$\begin{aligned} &A, B, C, \{\underbrace{D, E, F}_{reversed}\}, G, H, \\ \Longrightarrow\ &A, B, C, \{F, E, D\}, G, H. \end{aligned} \tag{1}$$

B. Homogeneous Markov chain

A homogeneous Markov chain means it is time-independent that the transition matrix of probability is same after each step.

In TSP problem, a new solution of travel routine, under the same temperature in a Markov Chain, is generated independently with the last step and transition possibility of each solution is always same.

C. Simulated Annealing with Metropolis Algorithm

The Metropolis algorithm is a sampling algorithm that uses Markov Chains to sample the most optimal solution from a

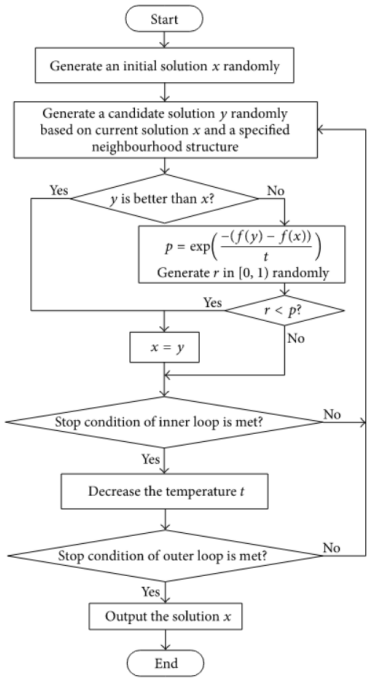

Fig. 1. A flowchart of the Simulated annealing optimization algorithm. The inner loop comes across with the Markov Chain length, the length of the outer loop is determined by the simulated annealing cooling schedule (1).

probability distribution that is very hard to sample from. In Simulated Annealing (SA), this sampling is being exploited even further to generate an optimal solution with respect to some cost function. This is visualized in the flowchart of figure 1, with the Metropolis algorithm as the inner loop, such that new candidate solutions y are being generated and y is accepted to be t new best solution x only if y is better than the optimal solution that we found prior to this point. Whether y is actually better than x is determined by making use of the Boltzmann distribution

$$e^{-(\frac{cost(y)-cost(x))}{t}} \tag{2}$$

and taking a random sample r between 0 and 1. If this random sample is smaller than the Boltzmann probability, y is accepted as the new optimal solution x. During SA the temperature is subsequently decreased as to make it less likely to accept the new solution as the best. This makes it possible to begin by exploring the entire sample space at the start, and increasingly zooming in on the global optimum.

D. Traveling salesperson problem (TSP)

The traveling salesperson problem is one of the first optimization problems for which a computer was used to provide a solution. It is a simple problem to describe: given a set of locations, try to find the shortest route to visit all of these locations once. However, the solution to the problem is very hard to find, since the search space scales with $(n-1)!/2$ routes for n locations. In 1965, computers were able to find locally optimal solutions for n=145 (2). In this study we look for the optimal solution for a set of 280 locations, which comes across with checking 279!/2 ($\hat{=}3 \cdot 10^{562}$) possible routes. This is obviously a very large number, and thus it is an impossible task to check all of the possibilities in a reasonable amount of time.

Simulated annealing has provided better optimal solutions than other optimization algorithms. But the downside to SA is that it has a slower convergence rate (3). Therefore we study the convergence rate of SA, and see if the parameters of SA can be adjusted in such a way that the convergence rate is minimized.

E. Thomson problem

The Thomson problem is another classic example of an optimization problem: given a fixed amount of point particles N located on a charged sphere, try to find the configuration that minimizes the free energy in the system. Again the description of the problem is simple, while the search space is incredibly large. Just as with the TSP, simulated annealing has proven to be a good method for providing an optimal solutions this problem, at least locally (4). An interesting note is that the Thompson configuration optimization is basically the same thing that is exploited in the process of annealing (from which SA draws its inspiration).

For a higher number of particles N, the number of degrees of freedom increases as well. This leads to a higher total energy inside the system and also a more complex search space. There are certain 'magic numbers' of N for which the optimal configuration has an energy state that is particularly lower than expected. In the range of N from 1 to 200, the configuration of $N = 192$ particles is the most magical (4).

III. Exploration and Results

A. TSP

Sometimes a deterministic algorithm, like Greedy Algorithm, sometimes is unable to find the global minimum because it could be stuck in the valley of a local minimum, then adding stochasticity into deterministic model can help the current state jump out. This phenomenon is vividly demonstrated in figure 2, When the deterministic algorithm approaches a local optimal, a perturbation can generate a new solution, even though it has a cost function that is less than the optimal solution, in order to jump out of the valley.

Here, Simulated Annealing and metropolis Algorithm have been implemented to solve TSP problem. Combined with the simulation process in figure 1, the specific simulation process for TSP are: **first**, generate an initial solution using Greedy Algorithm, **Then**, do 2-Opt Algorithm to get a new solution and decide whether accepting it or not with Metropolis Algorithm. After repeating this process for Markov Chin-length times, decrease the temperature according to cooling schedule. **At last**, repeat the above 2 steps until meet the stop condition (stop temperature and stop iterations).

1) Parameter setting: According to the cooling schedule here, α decides the speed of decreasing of temperature. And $alpha$ should be small enough as same as the Quenching process, since the faster the temperature decreases, the more

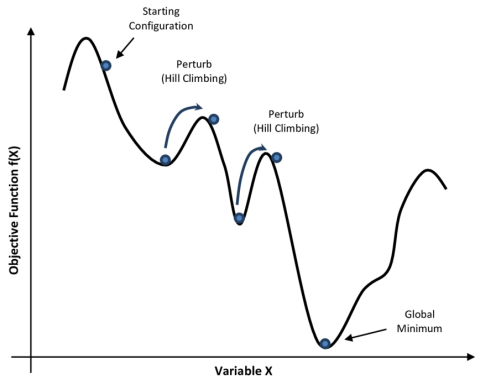

Fig. 2. General simulated annealing process. Before arriving the global optimal, there are chances to be stuck in local optimal. Then perturbation for hill climbing is important.

Parameter	Symbol
Initial temperature	T_0
Cooling schedule $T_k = T0 \cdot \alpha^k$	α
End temperature	T_n
Stop iterations	N
Markov Chain length	MC

TABLE I
PARAMETERS AND THEIR SYMBOLS IN SIMULATION.

irregular the final particle arrangement; and the the faster the temperature decreases, the easier to miss the optimal solution. So, set a small α to guarantee a large probability to find the optimal solution.

Both the initial temperature T_0 and end temperature T_n determine how many iterations the simulated annealing simulation will do. And according to the Metropolis Algorithm, the acceptance rate equals to $Min(random(0,1), e^{\frac{-\delta T}{T}})$. When T is larger, the acceptance rate for a "worse" solution is larger. Then it will take more time to "cool down" to the optimal solution. Limited by the computing power and under such a small cooling step α, the initial temperature is already limited to be not too large. The end temperature is determined by the iteration times, which could be limited by specific research condition.

???? why increase MC could improve optimal result?

In the following part, the influence of using parameter settings of different cooling processes is being explored.

2) Convergence of local minimum: From the beginning, the parameters are initialized in the table II, according to the principle of the last section.

Parameter	Symbol	setting
Initial temperature	T_0	10
Cooling schedule $T_k = T0 \cdot \alpha^k$	α	0.999
End temperature	T_n	1e-13
Stop iterations	N	30000
Markov Chain length	MC	10

TABLE II
PARAMETER SETTING USED TO FIND LOCAL OPTIMAL IN FIG 4.

To approach the local minimum that is the most globally optimal, 30000 iterations are allowed.

As shown in fig 3, the optimal decreases to 2800 approximately at the end of iterations. Given the best solution, which

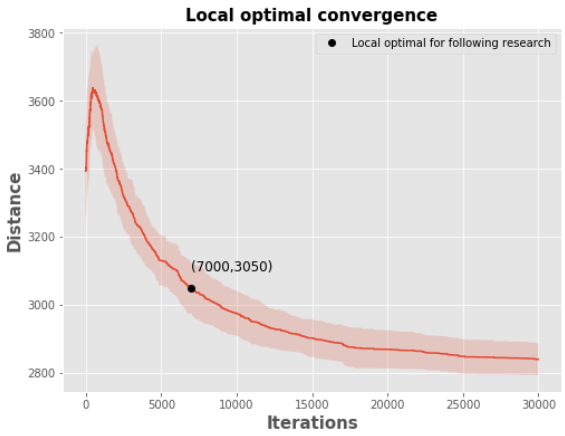

Fig. 3. Change of distance of TSP problem under simulated annealing process with parameter setting in table II.

has a shortest distance of 2586.77, there is difference between the experimental result and ideal result. This phenomenon is caused by the finite iteration times N, cooling schedule and Markov Chain length MC.

Limited by the computing power, multiple simulations with 30000 iterations (each iteration does MC-length 2-Opt steps) per simulation are difficult to be implemented on various parameter setting. The computing complexity can reach $O(C \cdot 10^7)$***, C is a constant decided by the MC length, simulation times, iteration times and configuration.***

Since then, to decrease the computing complexity, for the following simulation, the maximum iterations is chosen to be ***10000***. And the optimal now is ***3050***, which has been arrived after 7000 iterations in fig 3.

use this new "optimal", the convergence, which represents the speed to arrive optimal solution, of different configurations is studied in the following parts.

3) Convergence of different cooling schedule: In the fig 3, simulated based on a basis cooling schedule (schedule 1). To explore the influence of different cooling schedules on simulation which aims to find the global optimal, other four kinds of cooling schedules are implemented. The principle of them are shown in table III.

Cooling schedule	T_k	Requirement
schedule 1	$T_0 \cdot \alpha^k$	$\alpha < 1$
schedule 2	$\frac{T_0}{1+\alpha k^2}$	$\alpha > 0$
schedule 3	$\frac{T_0}{1+log(1+k)}$	
schedule 4	$T_n + (T_0 - T_n)(\frac{N-k}{N})^2$	
schedule 5	$T_n + (T_0 - T_n)(\frac{N-k}{N})$	

TABLE III
FIVE VARIOUS COOLING SCHEDULES AND FORMULAS.

As discussed above, limiting by computing power, parameter setting in this part is shown in table IV.

In fig 4, different cooling schedule have different speeds of reaching the "optimal", which value is plotted as the dotted

Parameter	Symbol	setting
Initial temperature	T_0	10
Cooling schedule 1 $T_k = T0 \cdot \alpha^k$	α	0.999
Cooling schedule 2 $\frac{T_0}{1+\alpha k^2}$	α	0.5
End temperature	T_n	1e-8
Stop iterations	N	20000
Markov Chain length	MC	10

TABLE IV
ADJUSTED PARAMETER SETTING LIMITED BY COMPUTING POWER OF FIG 4.

line. Compared with the basic cooling schedule 1, it seems that cooling schedule 2 and 3 have the higher speed with gradually downward trend; schedule 4 and 5 have lower speed with greater ups and downs.

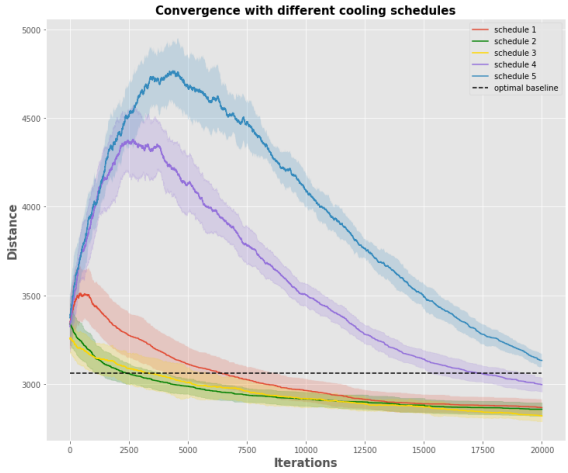

Fig. 4. Change distance of TSP problem under simulated annealing process with different cooling schedules. Parameter setting is in table IV

After multiple simulations, the smallest iterations of each schedule to arrive at the optimal is plotted in the boxplot of figure 5. Since the optimal can not be reached under cooling schedule 5, there is no box of this cooling method.

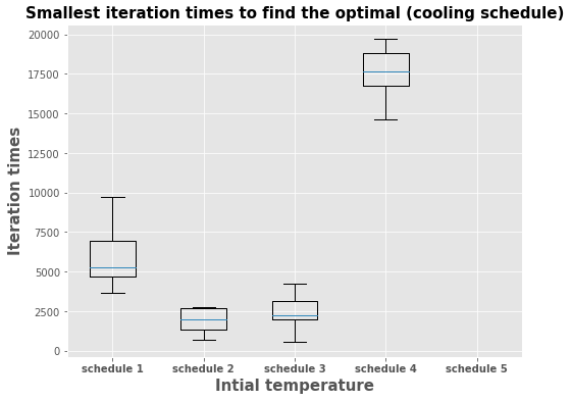

Fig. 5. The smallest iterations of different cooling schedule to converge to the optimal 3050.

It is clear that cooling schedule 2 has the faster convergence to its optimal solution; schedule 5 has quite slower convergence in all schedules. The full ranking of Convergence is as below.

$$\text{Convergence: } schedule2 > schedule3, \\ > schedule1 > schedule4 > schedule5. \tag{3}$$

Actually, the drop rate of temperature affects the convergence of cost. According to the equation 2 of Metropolis Algorithm, the higher temperature is, the more possible to accept a "worse" solution. So that, the cost will continue to rise until it reaches its peak, when the temperature decreases to a small value.

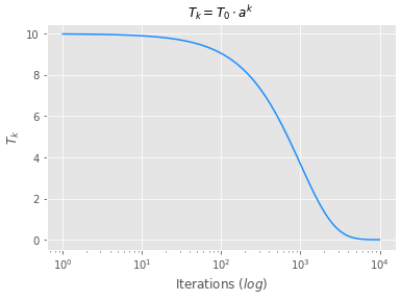

Fig. 6. Temperature drop process of cooling schedule 1.

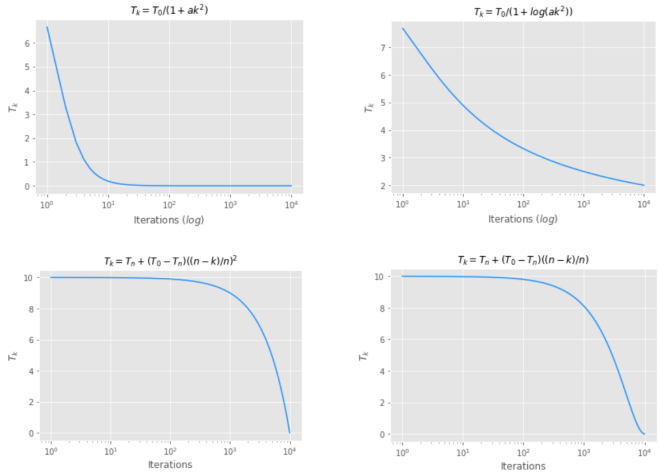

Fig. 7. Temperature drop process of cooling schedule 2, 3, 4,and 5.

from the fig 4 and 7, the ranking of drop rate of five schedule is the same as the ranking of convergence. **A conclusion can be arrived that The rate of convergence is proportional to the rate of temperature drop.**

$$\text{Temperature drop rate : } schedule2 > \\ schedule3 > schedule1 > schedule4 > schedule5. \tag{4}$$

In detail, the temperature of schedule 2 and 3 keeps decreasing, so that there is no peak in the cost curve. And the convergence of schedule 2 is faster than schedule 3 because of a higher drop rate of temperature in schedule 2. As far as the other three schedules, the peak of each cost curve in fig 4 is approximately close to the iteration times in fig 6 and 7 where the temperature starts to drop sharply.

4) Convergence of different initial temperature: Further more, the effect of different initial temperature on the convergence is discussed here. We used five different initial temperatures and control other parameters like in table IV; we use simulated annealing under the basic cooling schedule 1 with 10000 iterations.

From fig 8, the red curve (with highest T_0) goes up to the highest peak before cooling down thus it has the slowest convergence. The green curve (with lowest T_0) goes down gradually with the fastest convergence.

This phenomenon is similar to the process of various cooling schedules. High initial temperatures comes across with a high acceptance rate of "worse solutions", which leads to a higher peak before the curve going down.

The number of iterations it took for each setting to find the optimal (3050) in 50 simulations are plotted in the boxplot of 9.

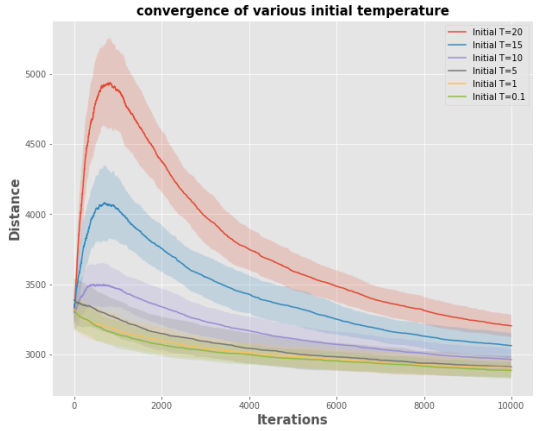

Fig. 8. Change of distance of TSP problem with different initial temperature. use cooling schedule 1, $\alpha = 0.999$, $T_n = 1e-8$, $N = 10000$.

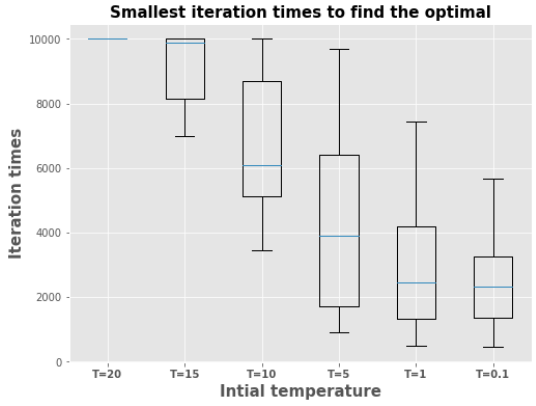

Fig. 9. The smallest iterations to converge to optimal with different initial temperature.

It is obvious that the convergence speed is inversely proportional to initial temperature.

5) Convergence of different end temperature: Both cooling schedule and initial time affect the convergence to optimal, the effect of end temperature is worth of exploration. Set 6 different end temperature from $1e-8$ to 1, the simulation result is shown in fig 10.

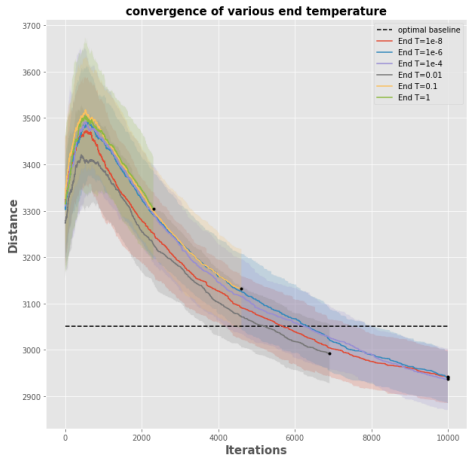

Fig. 10. Change of distance of TSP problem with different end temperature. use cooling schedule 1, $\alpha = 0.999$, $T_0 = 10$, $N = 10000$.

In general, there is not too much difference between each curve except the amount of iterations due to the similar temperature reduction. Black dots represent the end of each curve. It can be seen that for larger end temperature, more iterations can be taken. This rule is intuitive from the cooling schedule formula $T_n = T_0 \cdot \alpha^N$.

The relationship between end temperature and stop iterations is studied in fig 10. These purple dots represent the black dots of six different settings in the figure above.

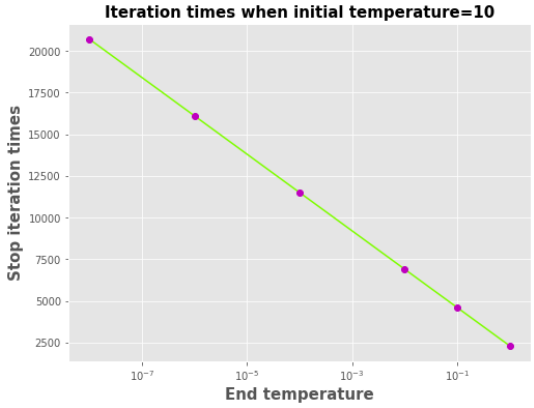

Fig. 11. The smallest iterations to converge to optimal with different end temperature.

Initial temperature, cooling schedule and end temperature determine how many iterations can be implemented in one simulation.

6) Convergence of different length of Markov Chain: The length of the Markov Chain is equal to the length of the inner loop in the flowchart of figure 1. A longer Markov Chain length results in a longer search before the temperature is decreased. **For each iteration of the simulated annealing process, the best solution that we expect to find would be better when a longer Markov Chain is being used.** It is thus

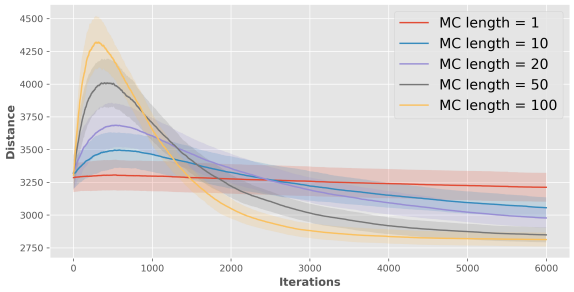

Fig. 12. Convergence to the final solution of the simulated annealing algorithm using different Markov Chain lengths in the range of 1 to 100. As is clear, a longer Markov Chain length leads to a better optimal solution. Another aspect that is clear from this graph is that using longer Markov Chains leads to more exploration, which causes the large bump in the beginning (which is due to the search in a region of larger distance routes).

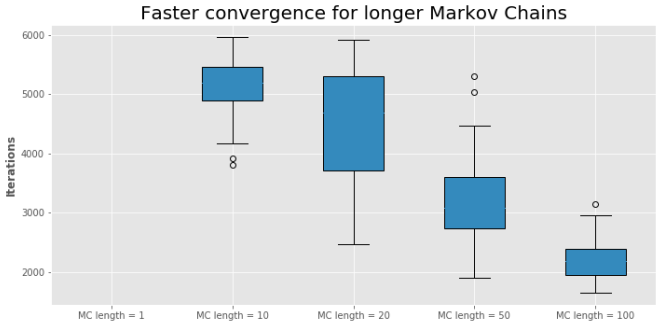

Fig. 13. This boxplot shows the decreasing trend of the convergence rate with increasing Markov Chain length. To measure the convergence, the number of iterations it took the simulation to find a solution with a baseline distance of 3050 was measured. The values in each box are drawn from 100 simulations. The simulations with a Markov Chain length of 1 never reached the baseline, which is why there is no box for this setting. Notice that there is overlap between all the boxplots except for the ones with a Markov Chain length of 10 and 100. This means that there is only a significant difference between these two. However the decreasing trend is still clearly visible.

logical to assume that the convergence towards the optimal solution takes less iterations for a longer Markov Chain length. This is indeed the result that we find looking at both figure 12 and 13.

7) Improvement of optimal solution: After all the exploration of each parameters, adjustment of setting can help simulation converge to a better local or global optimal compared with fig 3. Now, the "global" optimal converges to 2722.36. The adjusted parameter setting is in table V.

Parameter	Symbol	setting
Initial temperature	T_0	20
Cooling schedule $T_k = \frac{T0}{\alpha^k}$	α	0.995
End temperature	T_n	1e-21
Stop iterations	N	10000
Markov Chain length	MC	5000

TABLE V

The optimal route for our specific set of cities was given to us, and is plotted in the figure 14 on the left plot. Our SA provided a different (locally) optimal solution, as can be seen on the right plot, which is rather similar to the one on the left.

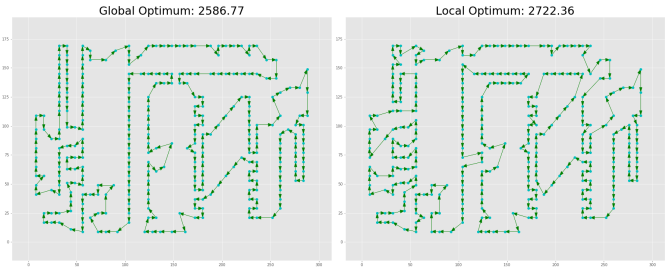

Fig. 14. The left plot depicts the route that was provided to us as the global optimum, with a distance of 2586.77. The best local optimum that we found through the use of Simulated Annealing has a distance of 2722.36.

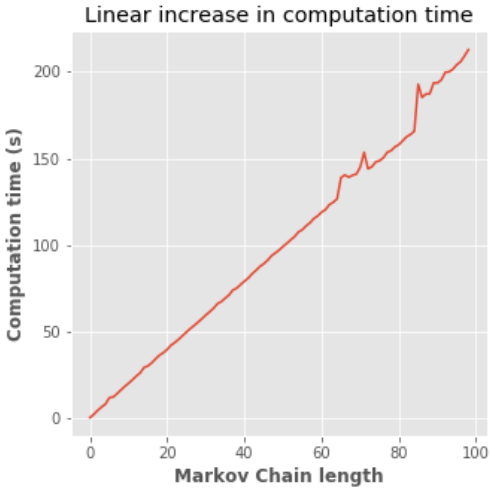

Fig. 15. This plot shows the linear increase in computation time with respect to the increase in Markov Chain length. The relation of finding a better local optimum and the increase in the Markov Chain length is sublinear as can be drawn from figure 12. This increase in computational complexity limits us to only find local optima, instead of the global optimum for our specific TSP problem. The changes in the plot around a Markov Chain length of 60 are due to some processes that started running simultaneously on the computer that performed the computations.

B. Thomson problem

We implemented SA to find the optimal configuration of the Thomson problem for $N = 11, 12$ and 192, depicted in figure 16. From the simulations that we ran for the Thomson problem, the same relation was found that longer Markov Chain length leads to a faster convergence, as can be seen in figure 17. However, the Thomson problem for $N = 100$ all proved to have a search space of lower complexity than the TSP problem. Increasing the Markov Chain length

Compared to the result for TSP in 12, the Markov Chain length used for the Thomson problem in 17 is of less influence to the convergence rate towards the optimal solution. This is due to the fact that the search space of the Thomson problem is of less complexity than that of our specific TSP.

However, we are now able to approach the tuning of the Simulated Annealing algorithm in a different way. Namely, for the TSP we were committed to a problem with a fixed search space, while we are now free to choose different values of N and make the search space more complex. Instead of

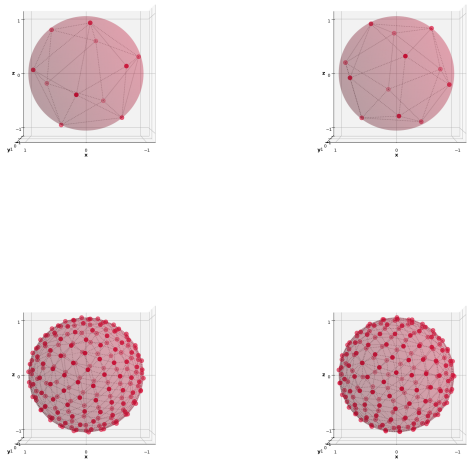

Fig. 16. Optimal configurations for the Thomson problem for the number of particles $N = 11, 12$ and $N = 191, 192$. There is a clear difference between the $N = 11$ and $N = 12$ configuration visible. This one extra particle induces an icosahedral (triangular shaped) configuration, which leads to a low total energy. The same thing happens in the case of $N = 192$, which is the most particularly low energy state in the range of N is 1 to 200.

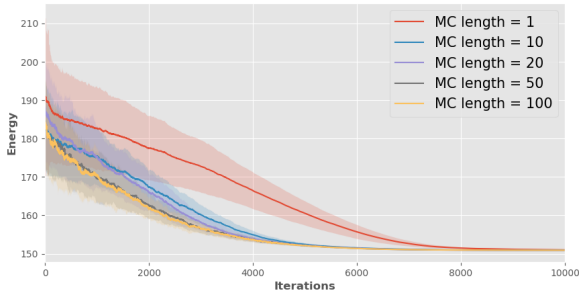

Fig. 17. The convergence rate to the optimal solution of the Thomson problem for $N = 200$ for different Markov Chain lengths from 1 to 100. As is clear, SA finds the same local optimum for each different Markov Chain length. Only the convergence rate using a Markov Chain length of 1 is significantly slower compared to the other ones.

choosing the right parameters to match our specific problem, like we did for TSP, it s now possible to research how a change in the search space is of influence to the outcome of the same SA algorithm. For a small number of particles N, it is not necessary to use a longer Markov Chain length, as is clear from the first plot in figure 18. For problems of higher complexity, using a longer Markov Chain length does lead to a faster convergence, but still the optimal solution that is found remains the same.

IV. Discussion and Conclusion

Simulated annealing is a powerful optimization method that (in theory) is able to find global optima for a wide range of problems. The parameters that are of largest influence to the outcome of SA are the Markov Chain length and the

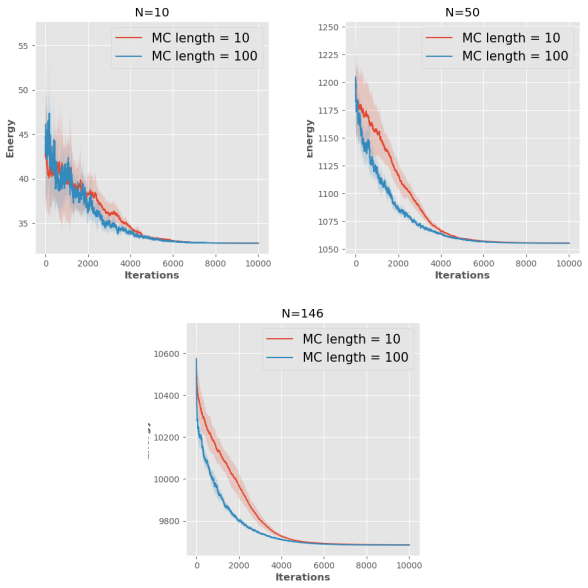

Fig. 18. Convergence rate towards optimal solution for different number of particles $N = 10, 50$ and 142 using SA with a Markov Chain length of 10 and 100. For $N = 10$ there is no significant difference between the convergence, but for a larger number of particles N (and thus a more complex search space) longer Markov Chains provide faster solutions.

cooling schedule of the temperature. Both of these should be set according to the complexity of the search space. We investigated this by running SA to solve a TSP with a Markov Chain length in the range of 1 to 100. Longer Markov Chains provide better optimal solutions, but this is a sublinear relation. Longer Markov Chains do increase the computational complexity of the SA algorithm in a linear way. Therefore, it is necessary to choose a Markov Chain length that is appropriate for the computational budget that is available. Also it is necessary to always keep into account the difference between the local optimum that was found and the global optimum (if known). For the TSP we found that the longest Markov Chain length that we used provided us with the best local optima of 2722.36 distance. However, due to the fact that we used a monotonic cooling schedule, we were not able to find the global optimum. Running more simulations with a very slow cooling schedule (schedule 2 from table III) using an even higher Markov Chain length might provide the global optimum, but this was not possible within our computational budget. Also SA was implemented to find an optimal solution to the Thomson problem. We found that the TSP problem had a much more complex search space than the Thomson configurations we tried to optimize. Using a short Markov Chain length we found the same global optimum as a longer Markov Chain length for all the configurations that we tried to optimize.

References

[1] S.-h. Zhan, S. Lin, Z.-j. Zhang, and Y. Zhong, "List-based simulated annealing algorithm for traveling salesman problem," *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–12, 03 2016.

[2] S. Lin, “Computer solutions of the traveling salesman problem,” *Bell System Technical Journal*, vol. 44, no. 10, pp. 2245–2269, 1965.

[3] X. Geng, Z. Chen, W. Yang, D. Shi, and K. Zhao, “Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search,” *Applied Soft Computing*, vol. 11, no. 4, pp. 3680 – 3689, 2011.

[4] J. Morris, D. Deaven, and K. Ho, “Genetic-algorithm energy minimization for point charges on a sphere,” *Physical Review B*, vol. 53, no. 4, p. R1740, 1996.