Universiteit van Amsterdam

Stochastic Simulation

Assignment 3: Simulated Annealing - Traveling Salesman Problem

Caterina Buranelli (13169408)[a]
Ignas Krikštaponis (13250868)[b]

[a] caterina.buranelli@student.uva.nl
[b] ignas.krikstaponis@student.uva.nl

18 Dec. 2020

UNIVERSITY
OF AMSTERDAM

Abstract

In this paper the Travelling Salesman Problem is investigated through the optimisation method known as simulated annealing. Three datasets of different sizes were used each for a specific purpose: the smallest, of 51 cities, was utilised for exploratory analysis; with the middle-sized problem of 280 cities, experiments were carried out to examine how the convergence of the solution depended both on the cooling schedule and on the length of the Markov chain; the analysis was also carried out for the smallest dataset. It was found that among geometric, linear or adaptive cooling schedules, the adaptive was the one which guaranteed faster convergence; the chain length was found to have a constant behaviour of convergence for the chain lengths of 100 and over for the dataset of 51 cities and from 250 and over for the dataset of 280 cities. Finally, the best set of parameters was used to find an optimal solution for the biggest dataset of 442 cities. Solutions from literature were set as benchmarks.

1 Introduction

The Traveling Salesman Problem, or TSP for short, is the optimisation problem that will be investigated in this paper. The traveling salesman problem is stated as follows: find the shortest way to visit all the cities and return to the starting point, given a list of cities with the traveling distance between each pair. The task is to simply find the order in which to visit all the cities once; the order is referred to as a city tour or circuit (David L. et al. [2011]).

Due to its simple definition and complexity the TSP is one of the most intensively studied problems in computational mathematics. It has seen application in the areas of logistics, genetics, manufacturing, telecommunications, and neuroscience (David L. et al. [2011]).

Many different ways of optimising the TSP were proposed over the years. These include such techniques as genetic algorithms, neural networks, particle swarm optimisation, dynamic programming; just to name a few (Greco [2008]). In this paper a technique that takes inspiration from nature, simulated annealing (SA), will be investigated. The main objectives of this paper are as follows:

- investigate different cooling schemes
- investigate how Markov chain length influences the solution convergence
- investigate optimal algorithm parameters

2 Methods

Simulated annealing is an optimization technique that takes inspiration from the natural creation of crystals, items with a perfect lattice structure. The idea is firstly to heat the material to a high temperature, allowing the atoms to explore lots of possible configurations, secondly to cool the system at a low speed, until the equilibrium state is reached. Simulated annealing is the most common approach in solving the TSP (Wang et al. [2019]). For N cities, the amount of feasible solutions to be investigated is $N!$, a substantial number even for small values of N.

The probability p_i that the system is in a certain state is given by the Boltzmann distribution, adapted to the optimization problem: $p_i \approx e^{-\frac{h(x_i)}{T}}$, where $h(x_i)$ is the function to be minimized and T is the temperature. For high values of T, the probability of a certain state is uniform, therefore all the configurations are equally accessible from the solver; as T approaches 0, only configurations with minimal $h(X)$ have a positive probability to be explored, and the search will eventually converge to one global minimum.

2.1 Markov chain and Metropolis algorithm

Markov chains are sequences of configurations associated to a transition matrix, which defines the probability of transition from state i to state j; each state depends on the previous state only and the Markov chain evolves until a stationary state. In simulated annealing, for each value of the temperature a Markov chain is created. The length of the chain determines the number of configurations generated for each temperature value and it is a fixed parameter that will be investigated. Longer chains should converge more easily, but would increase the computational effort.

The Boltzmann distribution is used for sampling the configurations, through the Metropolis algorithm (Metropolis et al. [1953]).

2.2 Initial value and cooling schedules

The choice of the initial value for the temperature t_0 is a balancing act between exploring - high value - and exploiting - low value. The choice of it goes along with the stopping criteria. In the paper, the best temperature initial value to be used in the experiments will be explored among three values.

The decreasing law followed by the temperature is called cooling schedule and it tunes the amount of time the system spends at high and low temperature. After the initial value is given, the temperature takes values t_k for $k = 1, 2, \ldots, K$ where K is said to be the stopping criteria.

A quick analysis of different cooling schedules was carried out, e. g. by considering the logarithmic or the additive rule, but accordingly to the best results in terms of time and performance, the following three were chosen to be investigated further in the experiments:

- Linear: $t_{k+1} = t_k - \beta$; a β equals to $\frac{t_0}{\mathrm{K}}$ grants that the values are equidistant. (Zhan et al. [2016])
- Geometric: $t_{k+1} = t_k \alpha$ where $0 < \alpha < 1$ is the cooling factor. (Triki et al. [2005])
- Adaptive: $t_{k+1} = (1 + \frac{f(s_i) - f^*}{f(s_i)})t_k$, where f^* denotes the best result found until that moment and $f(s_i)$ denotes the current solution; this method is different from the previous ones as the next temperature value depends on the record of the sequence and not on the past temperature. (Martín and Sierra [2008])

2.3 Mutation methods

The temperature tunes the probability that a new configuration is accepted as the best one; the process that creates the new configuration is called *mutation*, an idea that comes from the Genetic Algorithm branch. The mutation methods used in the paper are the following (Zhan et al. [2016]):

- **Move** A randomly chosen node is moved to a randomly chosen position.
- **Swap** Two randomly chosen nodes swap positions.
- **Reverse** Two randomly chosen nodes swap positions and the nodes in between are reversed.
- **Shuffle** Nodes between two randomly chosen positions are randomly shuffled.

In the exploratory analysis the performance of the algorithm is investigated when only one of the above methods is applied; these results were compared to the efficiency of a greedy *hybrid* algorithm: it updates the path configuration for each of the mutation methods and keeps the one with the minimum of the objective function. Simulated annealing algorithm with a greedy hybrid mutation mechanism is displayed in Algorithm 1.

Algorithm 1: Simulated annealing for TSP with hybrid mutation

```
Input: Stopping criteria $stop_c$, counter $c$, Initial temperature $t_0$,
 chain length $chain$
initialise random route $s_0$;
$pathLength_0$ = evaluate($s_0$);
while $c < stop_c$ do
    $stop_{chain} = 0$;
    while $stop_{chain} < chain$ do
        $s_{list}$ =hybdridMutate($s_0$);
        $s$ = chooseBest($s_{list}$);
        $pathLength$ = evaluate($s$);
        if $pathLength < pathLength_0$ then
            $s_0 = s$;
            $pathLength_0 = pathLength$
        else
            $p = e^{(pathLength_0 - pathLength)/t_0}$;
            if $random(0, 1) < p$ then
                $s_0 = s$;
                $pathLength_0 = pathLength$
        $stop_{chain} = stop_{chain} + 1$
    $t_0$ =updateTemperature($t_0$);
    $c = c + 1$
```

3 Results

The exploratory analysis was performed by using a dataset of 51 cities. This first preliminary exploration of parameters (t_0, α, $K\dots$) and methods (cooling schedules and mutation) led to a detailed analysis of the Travelling Salesman Problem, with a dataset of 280 cities. The experiments were meant to investigate two main features of the simulated annealing algorithm: the cooling method and the Markov chain's length; the results were compared with the smaller dataset. Finally, the results will be applied to solve the TSP for a configuration of 442 cities.

3.1 Exploratory analysis

The dataset `eil-51` has been widely studied in literature and solution by (Skorobohatyj [1995]) was selected as a benchmark; for this reason and due to its small dimension, it was used to carry out the first analysis of the parameters.

Stopping criteria There are several ways to terminate the process of the algorithm and the convergence depends partially on the chosen criteria. This feature was not investigated deeply in the paper, but a quick exploratory analysis led to the choice of the following stopping criteria: the algorithm would end after the temperature had undertaken K values, where 1000 was an educated choice for K. This option was preferred to the more common idea of stopping the algorithm once the temperature had reached a final value t_{end}; that is because, due to how the cooling schedules are built, the temperature would have spent too much time at low values, stuck in the same local optimum and therefore without providing better results.

Initial temperature value In order to choose the initial temperature value from $t_0 = 15, 45$ and 75, ten simulations with the following features were performed: Markov chain's length equal to 500, geometric as cooling schedule with $\alpha = 0.99$, stopping criteria $K = 1000$ and *Move* as updating function. The boxplots of the minimal found distance for each t_0 are found in Figure 1. For $t_0 = 15, 75$ the minimum value is around 430, while for $t_0 = 45$ is 436, slightly bigger. As the benchmark solution is around 429, all the three initial values perform well; on the other hand, for $t_0 = 45$ the variance of the results is smaller than the other (90.5, 49.5, 77 respectively for 15, 45, 75), a factor as important as the minimum value. For this reason, $t_0 = 45$ was chosen for the further analysis.

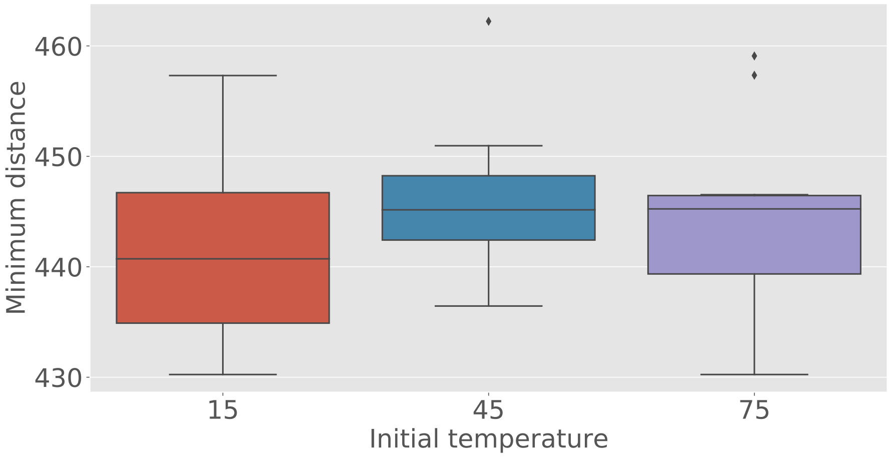


Figure 1: Box plot of found minimum distances of the 51 city tour for different initial temperatures

Mutation method Ten simulations for the four mutation methods and the hybrid method were performed with the following features: $t_0 = 45$, 500 as Markov chain's length, stopping criteria $K = 1000$ and the geometric cooling schedule, with $\alpha = 0.99$. The minimum found distance results are shown as box-plots in Figure 2. The worst mutation method is the most random, *Shuffle*, which provided the highest optimal values with the biggest variance. *Move* and *Swap* have similar variance, but *Move* yields lower optimal values. As both *Reverse* and *Hybrid* have small variances and small optimal values, a t-test was performed in order to see if there were differences: hybrid resulted as the best (p-value< 0.0001).

3.2 Experiments

With the information gained from the exploratory analysis, the dataset `a280` was used to analyse the best cooling schedule and the influence of the Markov chain's length on the convergence of the algorithm. Ten simulations were performed for chain's length equals to $\begin{pmatrix}10 & 100 & 250 & 500 & 750 & 1000 & 2000\end{pmatrix}$, each with linear, geometric and adaptive cooling function. The other features were set coherently with previous findings: $t_0 = 45$, hybrid mutation function and $K = 1000$ as stopping criteria.

In order to investigate how the size of the problem influences the results, the same number of simulations with the same settings were performed for the 51-city dataset.

The results for the 280-city dataset are plotted in the second column of Figure 3.

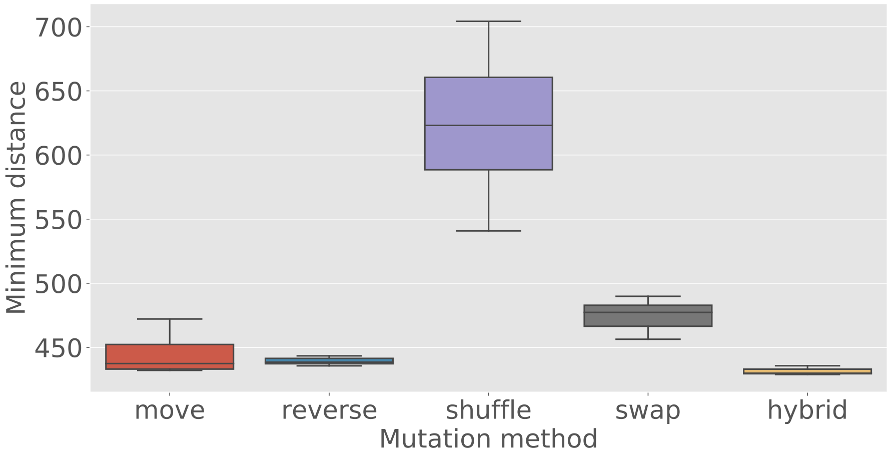


Figure 2: Box plot of found minimum distances of the 51 city tour for different mutation methods

For shorter Markov chains, from 10 to 500, the convergence of the process is similar when the cooling system is geometric or adaptive; for a chain of length 10, geometric converges first. As the chain becomes longer, the difference between geometric and adaptive increases, the latter provides faster convergence. Linear cooling schedule performs the worst: the curve starts as the other two, but then decreases slowly, reaches convergence at the very end of the process and only for Markov chain of length 750 or higher. Moreover, linear schedule has big oscillations, that denote high inaccuracy.

For the 51-city dataset, the three schedules have a behaviour similar to the one described above. But while for 280 cities the difference between adaptive and geometric is unclear for short chains, with 51 cities this discrepancy is big for chain lengths 100 and above. Moreover, now both geometrical and linear have frequent oscillations. These results are plotted in the first column of Figure 3.

In Figure 4 the mean best value for each chain's length and the 95% confidence interval is plotted. When connecting 51 cities (left- hand-side graph), the convergence decreases when passing from 10 to 100: for linear the change is abrupt, for geometric and adaptive it is moderate. After 100 the decrease slows down; it becomes almost constant when the chain is longer than 500. When the cooling schedule is adaptive, the convergence is less conditioned by the chain length, than the other schedules. When considering 280 cities (right-hand-side graph) the optimal distance decreases sharply from 10 to 100 for all three schedules. For chains longer than 250 the difference in finding the best value is minimal while using geometric and adaptive, but it

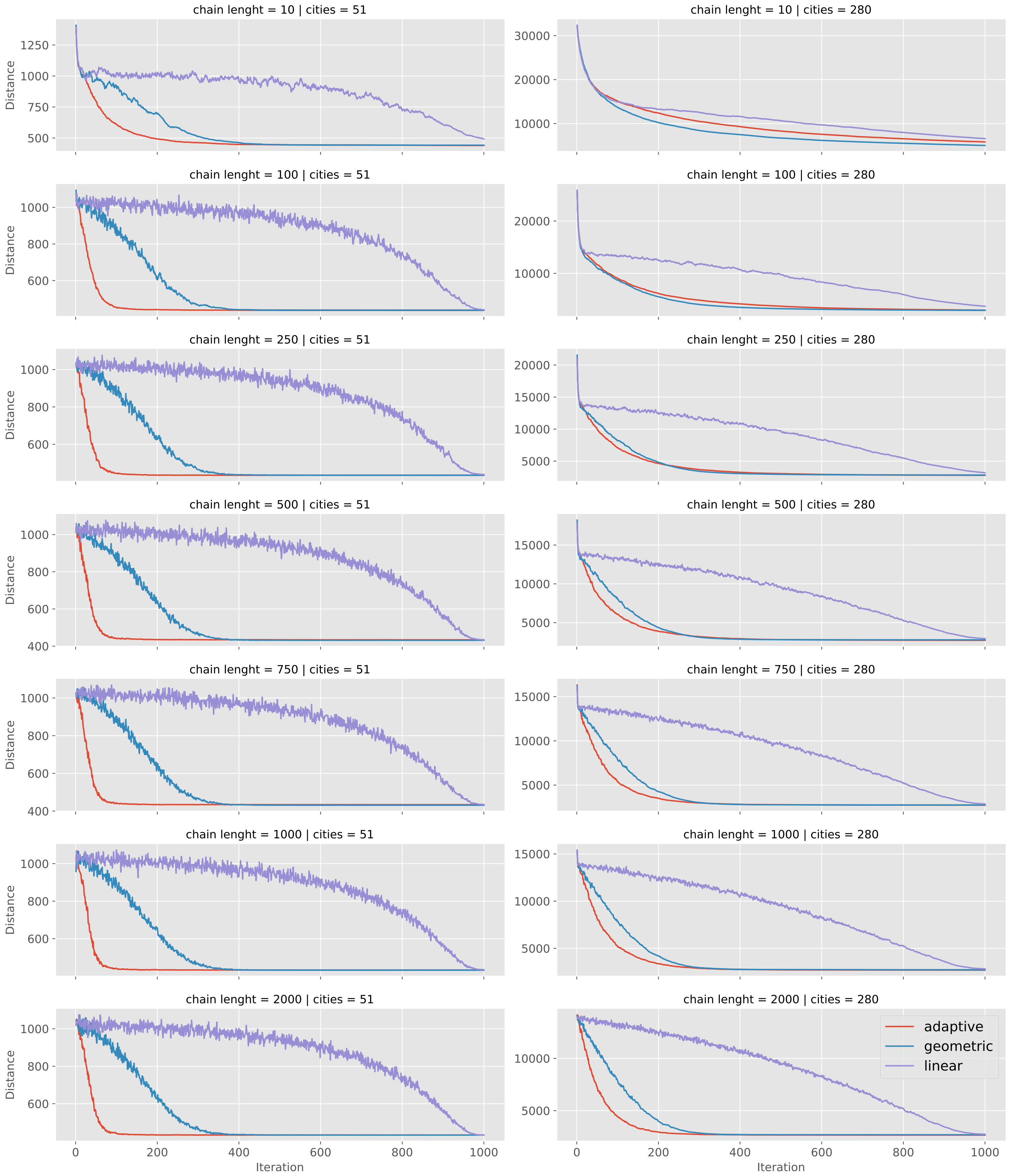


Figure 3: Simulated annealing convergence.

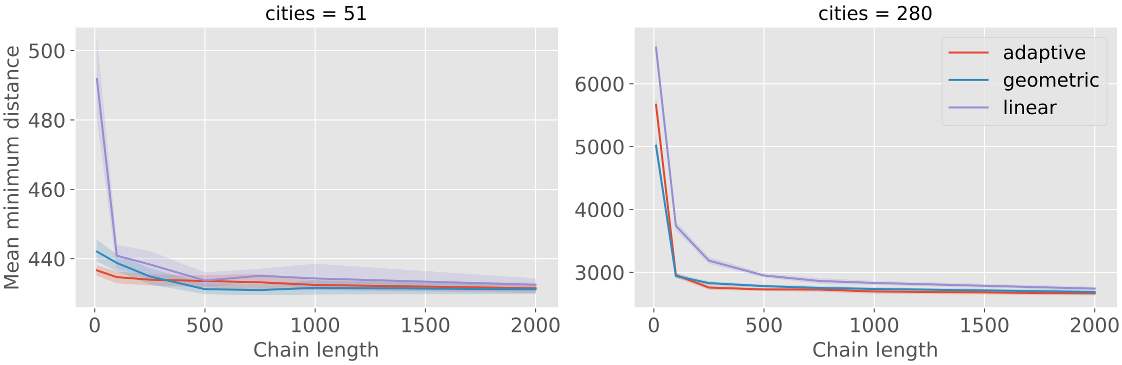


Figure 4: Progression of best distance found with different Markov chain length.

is still declining for linear.

Temperature In Figure 5 the behaviour of the temperature for each step of the Markov chain and for different cooling schedules is plotted. For geometric and linear the temperature decreases following a *fixed* rule, therefore the curve is smooth, as it is the curve of a continuous function. On the other hand, the behaviour of the curve for adaptive changes for different chain's length: when the chain length is 10, there is less probability of finding a new best solution, therefore the curve decreases slowly. As the chain's length increases, also the probability of improving at each step increases and the curves decrease faster. NB: the adaptive curve is dependent on the specific best value of the run, which is subject to oscillations due to stochasticity. In the plot, the results of just one simulation is reported as an example and not as a general conclusion, for it is not a mean of several simulations.

Mutation method In Figure 6 the percentage of the updating function chosen at every single step of the iterations is plotted. *Reverse* is the most used mutation: around 40%, but for the smaller dataset at the beginning the percentage is even 50%; this is coherent with what was found in the exploratory analysis. *Shuffle* is the method accepted the least in the beginning, but its acceptance grows with the last iterations, while the curves of *Move* and *Swap* decrease; this change corresponds to low values of the temperature, as we can see from the dashed line in the plot. The size of the dataset does not considerably influence the behaviour of the curves, which is, instead, influenced by the temperature.

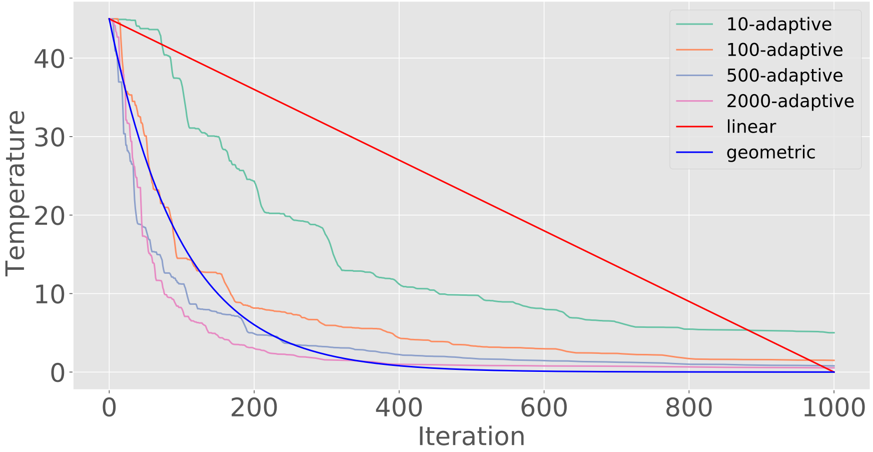


Figure 5: Temperature progression for different cooling schedules and chain lengths

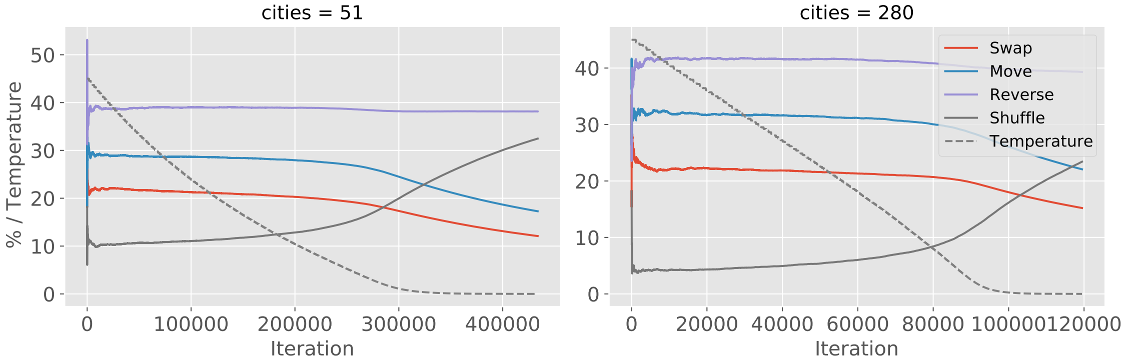
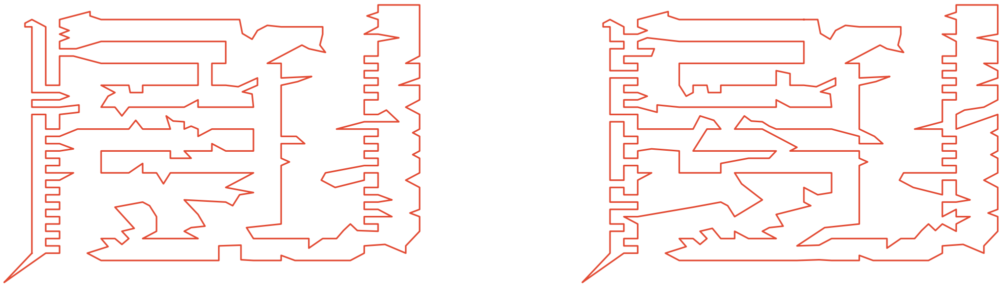


Figure 6: Percentage usage of the four mutation methods during one simulation of SA with chain length of 2000 and geometric cooling schedule.

Table 1: Best found and benchmark tour solution lengths (Skorobohatyj [1995]).

Cities	Best found	Benchmark
51	428.872	429.983
280	2600.728	2586.770
442	51634.703	50783.548

3.3 Solving the problem


(a) Benchmark solution

(b) Found best solution

Figure 7: 442-city problem solutions (Skorobohatyj [1995]).

Taking into consideration the findings of the previous section and the size of the 442 cities problem being substantially bigger than the one of 280 cities; to find the solution to the 442 cities set we chose to run the algorithm with adaptive cooling schedule, chain length of 2000, $K = 1000$ and $t_0 = 45$. The best found tour from 5 simulations as well as the tour of the benchmark solution can be seen in Figure 7. The best found distance results with benchmark solutions can be found in Table 1. The proposed algorithm managed to find a distance that is shorter for the smallest, 51 cities, tour. For the remaining two data sets it did not converge to the benchmark solution, however the distance found was only $\approx 1\%$ bigger for 280 cities and $\approx 1.5\%$ bigger for 442 cities.

4 Discussion

As the TSP is a problem whose complexity grows exponentially with the number of cities, it is reasonable to test and experiment with sets of parameters in smaller datasets, first through an exploratory analysis and then with formal and accurate experiments, finally applying the best set of parameter to the bigger and conclusive dataset. This is the path that was followed in this paper; the experiments led to the following conclusions.

Among geometric, linear and adaptive, the latter was the cooling schedule that performed the best, as it converged the fastest to the optimal value. However, this was not the case for all Markov chain lengths. For shorter chain lengths for the bigger problem, adaptive cooling schedule converged slower. This can be explained by temperature progression - adaptive schedule progressed slower than geometric. This is due to shorter chains failing to explore enough to find superior solutions, thus not triggering temperature reduction.

The Markov chain's length was found to influence the convergence of the algorithm accordingly to the size of the dataset. For the smallest dataset, the algorithm performed almost the same for chain lengths of 100 and over; for the middle-size dataset, the convergence starts becoming constant for longer chains: from 500 and over.

Reverse was found to be the most utilised tour mutation method. Its usage was constant throughout the simulation progression. The usage of remaining three update methods started changing when temperature reached low values, rendering *Shuffle* as the most useful of the three for exploitation period of the simulation.

In terms of solving the biggest problem (442 cities), the algorithm managed to produce a route that is 1% longer than the benchmark solution. Ideally, as simulated annealing is problem specific, an analysis of the best parameter settings should be carried out for specific problems. Intuitively bigger problems would require higher initial temperature setting, longer chain or higher stopping criteria. Other cooling schedules could have been explored here too; such as exponential additive cooling (Martín and Sierra [2008]).

References

Applegate David L., Bixby Robert E., Chvátal Vašek, and Cook William J. *The Traveling Salesman Problem : A Computational Study.* Princeton University Press, 2011. ISBN 9780691129938. URL `https://search-ebscohost-com.proxy.uba.uva.nl:2443/login.aspx?direct=true&db=e000xww&AN=390512&site=ehost-live&scope=site`.

Greco. *Traveling Salesman Problem.* InTech, 2008. ISBN 953-7619-10-9.

L. Wang, R. Cai, M. Lin, and Y. Zhong. Enhanced list-based simulated annealing algorithm for large-scale traveling salesman problem. *IEEE Access*, 7:144366–144380, 2019. doi: 10.1109/ACCESS.2019.2945570.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953. doi: 10.1063/1.1699114. URL `https://doi.org/10.1063/1.1699114`.

Shi-hua Zhan, Juan Lin, Ze-jun Zhang, and Yi-wen Zhong. List-based simulated annealing algorithm for traveling salesman problem. *Intell. Neuroscience*, 2016:8, March 2016. ISSN 1687-5265. doi: 10.1155/2016/1712630. URL `https://doi.org/10.1155/2016/1712630`.

E. Triki, Y. Collette, and Patrick Siarry. A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. *European Journal of Operational Research*, 166:77–92, 02 2005. doi: 10.1016/j.ejor.2004.03.035.

José Martín and Jesús Sierra. *A Comparison of Cooling Schedules for Simulated Annealing.* 01 2008. doi: 10.4018/9781599048499.ch053.

Georg Skorobohatyj, 1995. URL `http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsp/`.