Application of Simulated Annealing on the Traveling Salesman Problem

Coen Lenting *(11332484, coen.lenting@student.uva.nl)*
Sjoerd Terpstra *(11251980, sjoerd.terpstra@student.uva.nl)*

18 December 2020

Abstract

The aim for this assignment was to solve the Travelling Salesman Problem (TSP) using Simulated Annealing (SA). This is a method based on Markov chains, where newly generated states depend only on the previous state. This method outperforms more simple methods based on the accept-reject methods, because of the dependency between states. For a small scale 51 nodes (eil51.tsp, from the TSPLIB library) problem, a thorough parameter search was conducted, where the optimal parameters for the starting temperature t_0 and Markov chain multiplier ζ were selected for three cooling schedules, each based either on a logarithmic, linear or quadratic function. The selected parameters were used to find the shortest path for the 280 nodes (a280.tsp) problem. The simulations with the linear and quadratic cooling schedules resulted in the identical shortest paths and the best average estimate for the shortest path. The simulations with logarithmic cooling resulted in a relatively worse estimation of the shortest path, but was significantly quicker in computational time. Depending on the objective of the problem, it either being the absolute shortest path length or the best estimation for a certain budget, the most suitable cooling schedule can be chosen.

1 Introduction

One of the most famous problems in Computational Science is the Traveling Salesman Problem (TSP). The objective is to find the shortest route such that the salesman visits each city exactly once and returns to its starting position. This type of problem is prevalent in a variety of applications. It is important for navigation purposes to find the shortest route or to find the least distance of electricity cables in microchips to limit the communication time between components (Grötschel, Jünger, and Reinelt, 1991).

The TSP is a NP-hard problem, which means that there does not exist an algorithm to solve it in polynomial time. However, there exists a variety of algorithms to tackle this problem and find solutions which are local or global optima (Antosiewicz, Koloch, and Kamiński, 2013). One way to find the minimum route is to use a Simulated Annealing (SA) algorithm, which finds its origins in statistical physics (Sharp and Matschinsky, 2015).

We applied the SA algorithm to the TSP's eil51.tsp and a280.tsp. Both are part of the TSPLIB library (Reinelt, 1991). Three different cooling schedules were used, based either on a logarithmic, linear or quadratic function. To tune the parameters of the algorithms, convergence plots were made to determine the range of good parameters for the initial temperature cooling schedules and the length of the Markov chains in the SA algorithm.

2 Theory & Methods

2.1 Markov Chain

A Markov chain is a stochastic model where each newly generated state only depends on the previously generated state, which is very broadly applied to predict traffic flows, communication networks, genetic issues and queues (Baudoin, 2010). When considering a large Markov chain, the overall system will be memoryless as there is only correlation between two subsequent states. The fact that the Markov chain retains memorylessness allows it to be a very efficient stochastic model when small regions in the search domain are of interest, since only the information on the current location is considered when generating a possible new state. This model performs exceptionally better compared to a complete random accept-reject method, which does not 'know' what the actual region of interest is when generating new states. A Markov chain is characterised by three properties: irreducibility, a-periodicity and time-reversibility.

A Markov chain is irreducible when for each pair of states there exists a finite amount of intermediate steps to transition from one to the other. Formally spoken, when considering a collection of random variables

$X_0, X_1, ..., X_n$, interpreted as the state of the system at a time n, the possibility of transitioning from a state i to a state j is nonzero (Ross, 2013). This property is important when constructing a problem. When a Markov chain is not irreducible, portions of the solution space can not be explored, which can bottleneck the best estimated solution. For an irreducible Markov chain π_j can be proposed as the long-run proportion of time that a variable is in a state j. This quantity can be combined with the transition probability matrix $P_{i,j}$ to transition from a state i to state j or vice versa to define the equality

$$\pi_j = \pi_i P_{i,j}, \ j = 1, ..., N. \tag{1}$$

These π_j's are often called the stationary probabilities of the Markov chain. The ability to construct these stationary probabilities, allows the use of the law of large numbers for the Markov chain, meaning that the ensemble average can still be estimated by considering a long Markov chain.

An irreducible Markov chain is defined to be aperiodic when for some state $n \geq 0$ and for some state j

$$P\{X_n = j | X_0 = j\} > 0 \tag{2}$$

and

$$P\{X_{n+1} = j | X_0 = j\} > 0. \tag{3}$$

Lastly, a Markov chain is said to be time-reversible when for all $i \neq j$ $\pi_i P_{i,j} = \pi_j, P_{j,i}$. Under this condition it can be shown that when an initial state is chosen, going forward or backward in time will both result in a Markov chain with transition probability matrix $P_{i,j}$.

2.2 Traveling Salesman Problem

The TSP is a classic constraint optimization problem. The objective is to find the global minimum of the solution landscape or at least to find a satisfying local minimum. In this problem a salesman has to visit N cities, such that he visits each cities exactly once. The salesman has to finish in the same city they started from. The goal is to reduce the total distance travelled to a minimum. In Figure 1 an example configuration is shown for a possible path that satisfies the above conditions.

It is assumed that the problem is symmetric i.e. the distance from city A to B is the same as the distance from B to A. The second assumption is that there is a connection between every city, thus the problem is irreducible. We assume that all cities lie in a 2D-plane and that the triangle inequality holds. This means that we can express the distance between two cities i and j as an Euclidean distance

$$d(i,j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \ i \neq j. \tag{4}$$

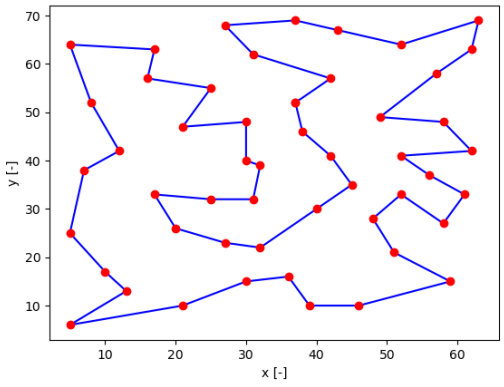


Figure 1: Example route for the TSP problem eil51.tsp

The total distance of a route S with length N can then be expressed as

$$d_{tot}(S) = d(S_N, S_0) + \sum_{i=0}^{N-2} d(S_i, S_{i+1}). \tag{5}$$

Equation 5 is the objective function that has to be minimized.

Each city is represented by a node and chained together in an ordered list. Each node is connected with the previous and next node in the list. So, each city connects with exactly two other cities.

2.3 Simulated Annealing

The SA algorithm was inspired by a physical phenomenon: the cooling of crystals in a lattice. When the temperature of the crystal lowers slow enough the molecules will arrange themselves in such a way that the total energy of the lattice is minimized. When the temperature is low enough the molecules will settle in the lattice so that the energy reaches a global minimum. This can be described by the Boltzmann distribution

$$P(i) = \frac{1}{Z} e^{-\epsilon_i / kt}, \tag{6}$$

with $P(i)$ the probability of the system to be in state i with associated energy ϵ_i, k is the Boltzmann constant and t the temperature (McQuarrie, 2000). The function is normalized by the partition function Z, which is the sum of each individual state (N possible states in total)

$$Z = \sum_{i=0}^{N-1} e^{-\epsilon_i / kt}. \tag{7}$$

SA is based on this same principle of slowly cooling of the system to reach a global minimum. The states S_i are part of a Markov chain; each next state relies only

on the previous state. One begins with an initial state of the system S_0 and a (high) starting temperature t_0. Modifications are applied to the system for each state in the Markov chain. The temperature is then gradually decreased after each iteration over the Markov chain. The probability of the states are calculated with a modified Boltzmann distribution of Equation 6:

$$P_{boltzmann} = exp\left(-\frac{(d_{tot}(S_{new}) - d_{tot}(S_{current})}{t}\right). \tag{8}$$

We omit the normalization factor Z and the Boltzmann constant k. The energy is calculated as the difference in cost function of two states.

Algorithm 1: Simulated Annealing

```
S_current ← shuffle(S_0);
while improving over last 1000 iterations do
    for i to markov_length do
        Generate S_new;
        if d_tot(S_new) < d_tot(S_current) then
            S_current ← 2-opt(S_new);
        else
            P ← P_boltzmann;
            r random in [0, 1);
            if r < p then
                S_current ← S_new;
            end
        end
    end
    t ← decrease t;
end
return S_current;
```

In algorithm 1 the procedure is written out in detail. The first step is to initialize with a starting solution S_0. This is generated by randomly shuffling the input list of cities. Then the body of the while loop is performed, until no better solution are found after 1000 iterations. First we loop over the whole Markov chain. The length of the chain is defined as

$$markov\_length = \zeta \cdot N, \tag{9}$$

with ζ the Markov length multiplier, a constant parameter.

Each time a new solution is created by permuting $S_{current}$ with a 2-opt swap (see next section). If the cost function is lower for the new state, than this state is accepted. If not, the value of the Boltzmann distribution (Equation 8) is calculated and a random number r is thrown. If $r < p$ than the new state is still accepted, even though it has a worse cost function. The purpose of this step is to explore the solution space when the temperature is still high. The Boltzmann distribution has a higher value with a high temperature, so in the beginning the algorithm is exploratory of nature. When the temperature decreases, the algorithm focuses more on the promising areas. Finally, after the for-loop is completed, the temperature is updated and we go back to the beginning of the while loop. After the while-loop has terminated the $S_{current}$ is returned, which should be the optimal solution.

2.3.1 2-opt swap

The elementary edit used in our SA is the 2-opt swap. First, two random nodes are taken from $[0, N)$. These nodes severe their connection to the main chain. This created a second chain between both nodes. The nodes reattach themselves back again to the main chain at the original side of the other one of the nodes. The effect is that the cities between the two nodes have reversed their visiting order.

2.3.2 Cooling schedules

As previously mentioned, it is important to have a good cooling schedule in order for the algorithm to work. The temperature function should continuously decrease from a high starting temperature to a lower temperature. One of the challenges is to determine the starting temperature t_0. We propose three different cooling schedules which are based on either of three basic functions: a logarithmic (Equation 10), a linear (Equation 11) and a quadratic (Equation 12) function.

$$Logarithmic : t_{log} = \frac{t_0}{1 + \ln(1 + i)} \tag{10}$$

$$Linear : t_{linear} = \frac{t_0}{1 + \alpha i},\ \beta = 0.01 \tag{11}$$

$$Quadratic : t_{quadratic} = \frac{t_0}{1 + \beta i^2}\ \alpha = 0.000002 \tag{12}$$

The progression of the temperature for the different cooling schedules as a function of the number of iterations is shown in Figure 2 for a starting temperature of $t_0 = 10$.

3 Results

3.1 Experiments

In order to find the best solution to the TSP, the parameters ζ and t_0 had to be optimized. These were carefully selected, because the optimal values of the parameters are highly dependent on the problem at hand and the cooling schedule. The selection was done by computing the convergence of the shortest path as a function of t_0 for three fixed values of ζ and by

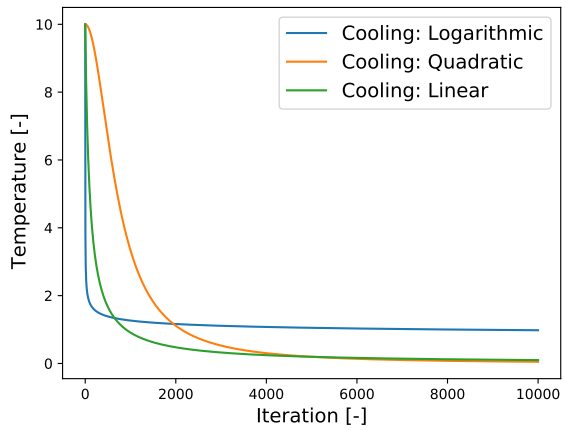


Figure 2: The progression of the temperature for the different cooling schedules with starting temperature t_0. Logarithmic seems to cool down the quickest, followed by linear and lastly quadratic, staying the hottest up until 2000 iterations, after which linear and quadratic approach $t = 0$. The logarithmic function has also a limit of $t = 0$ in infinity. However, the approach to $t = 0$ is slower, because it is a logarithmic function.

computing the convergence of the shortest path as a function of ζ for three fixed values of t_0. The optimal values for the simulations were taken from this plots at their respective minima whereas possible. This tuning process was conducted on the eil5q.tsp data set with 51 nodes, as parameter tuning on the larger problem would require a significantly larger computational cost. In order to translate the optimal parameter value for the Markov Chain length between the differently sized problems, the Markov Chain multiplier was introduced as ζ (see section **2.3 Simulated Annealing**). The corresponding Markov Chain length was then computed by multiplying ζ by the amount of cities.

3.1.1 Different cooling schedules

The comparison of the performance of the cooling schedules was done by considering how fast the cooling schedules reach their minimum compared to each other and which reaches the lowest minimum. Moreover, other characteristic differences can be observed between the cooling schedules. This can tell something about the efficiency of the different methods, as sometimes a quicker, less optimal method may be preferred compared to the absolute best, but computationally expensive solution. The simulations were ran for a ζ and t_0 of 20 for 30 runs. The parameters for this analysis were not tuned for maximal performance, but they were selected as a neutral point to compare the methods at.

3.1.2 Convergence by varying t_0

In order to have an estimation of the best parameters for the specifics of the problem, a parameter search was done by executing the simulated annealing for multiple values of t_0. The simulations were first ran with a very broad range of values for t_0 (between 1 and 80) and a ζ of $\{5, 10, 15\}$. From this result a range of values for t_0 was selected where a higher resolution parameter search was conducted for the same three values for ζ. This resulted in a graph with the correlation between the convergence of the total distance and t_0.

3.1.3 Convergence by varying Markov Chain multiplier ζ

The convergence of the total distance as a function of ζ was constructed similarly to the method used to compute the convergence of the total distance as a function of t_0, with the roles of t_0 and the ζ interchanged. The values for t_0 used to do the parameter search were selected depending on the result from the parameter search on the t_0 spectrum. This search was able to give a good indication of the range of the optimal parameters for the problem.

3.1.4 Solving the TSP a280.tsp

The estimated optimal parameters for the 51 nodes problem were then used to solve the a280 system for each of the cooling schedules. This problem consists of 280 cities. The simulation was done over 100 runs to acquire a good estimate of the performance of the algorithm.

3.2 Results

3.2.1 Different cooling schedules

The result for the simulations on the different cooling schedules is shown in Figure 3. A clear distinction in shape between the schedules is observable. For these parameters logarithmic cooling approaches a low mean distance very quickly in comparison to the linear and quadratic cooling and also terminates first. Quadratic cooling terminates the last, making it the most computationally expensive cooling schedule for these general parameters. What can also be observed is that quadratic cooling has the largest variance in the early to mid stage during the cooling process. This can be explained by considering Figure 2, where it was observed that quadratic cooling resulted in the hottest temperature up until around 2000 iterations. A higher temperature allows for more exploration in the starting phase, resulting in a large variability.

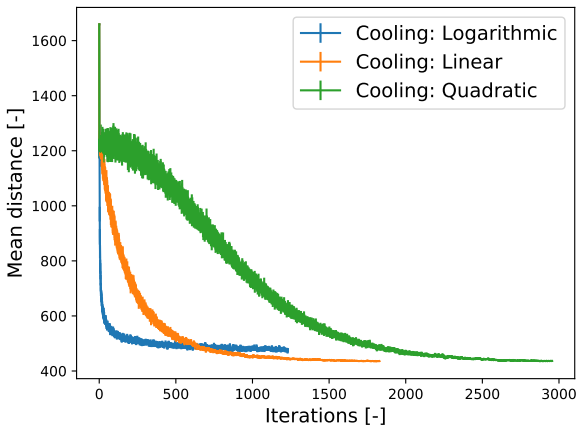


Figure 3: The convergence of the mean total distance, with the corresponding confidence interval, as a function of the number of iterations before termination for the different cooling schedules. ζ and t_0 are set to 20 and the results are averaged over 30 runs. For the specified parameters, the logarithmic cooling seems to converge much faster than linear and quadratic, whereas the quadratics termination happens at the highest amount of iterations.

3.2.2 Convergence by varying t0

The results from the high resolution parameter search for t_0 and fixed values for ζ, $\{5, 10$ and $15\}$, for the different cooling schemes are shown in Figure 4, where sub figure a) represents the logarithmic cooling, b) linear cooling and c) quadratic cooling. In the first sub figure a distinct minimum is observed around $t_0 = 11$. For the linear and quadratic cooling, no clear minimum is observed for the chosen parameters.

3.2.3 Convergence by varying the Markov Chain Length

The results from the high resolution parameter search for ζ for the different cooling schemes are shown in Figure 5, where subfigure a) represents the logarithmic cooling, b) linear cooling and c) quadratic cooling. The values selected for t_0 for these convergence tests were $\{6, 12$ and $18\}$ for logarithmic cooling, $\{40, 50$ and $60\}$ for linear cooling and $\{35, 45$ and $55\}$ for quadratic cooling. For the quadratic and linear cooling there is a clear dependence between the best found solution and the parameter ζ. This corresponds with the general notion that more computations will yield a better solution when solving an iterative process, of course also coming at a computational cost. This clear dependence is less so observable for logarithmic cooling, where a slight minimum is observed around a ζ of 20.

3.2.4 Solving the TSP a280.tsp

The results from the TSP with 280 nodes for the different cooling schedules are shown in Table 1. The parameters selected for the simulations are; Logarithmic ($t_0 = 11$ and $\zeta = 50$), Linear ($t_0 = 50$ and $\zeta = 40$ and Quadratic ($t_0 = 40$ and $\zeta = 50$). From the table it can be observed that both the linear and quadratic cooling were able to find a minimum of 2586.77. This optimal solution is drawn in Figure 6. The solutions from the logarithmic cooling were able to come close to this minimum, but on average performed slightly worse than the linear and quadratic cooling. However, the simulations with logarithmic cooling were much quicker than linear and quadratic cooling, with quadratic cooling having the longest execution time. Thus, when computational cost has a higher priority than finding the lowest minimum, it might be beneficial to use logarithmic cooling over linear and quadratic cooling.

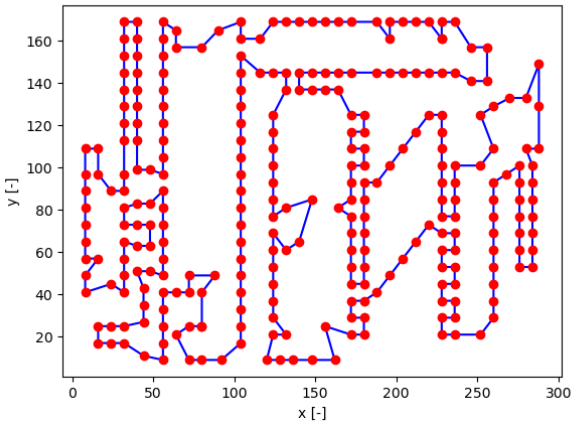
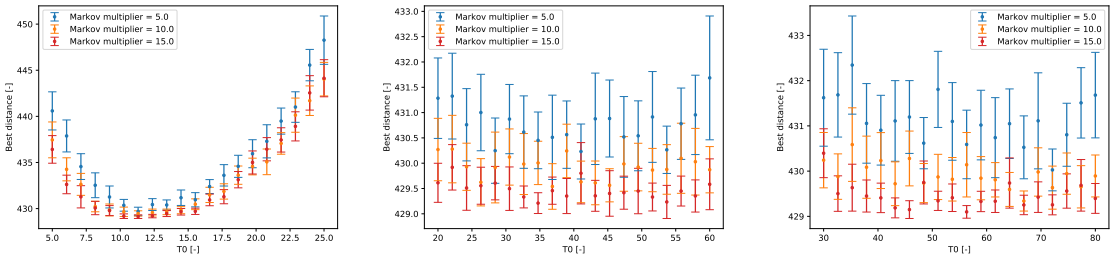


Figure 6: The shortest path found for the a280.tsp tour with a total distance of 2586.77. This solution was obtained with both the linear ($t_0 = 50$ and $\zeta = 40$) and the quadratic ($t_0 = 40$ and $\zeta = 50$) cooling schedules (Equation 11 and Equation 12).

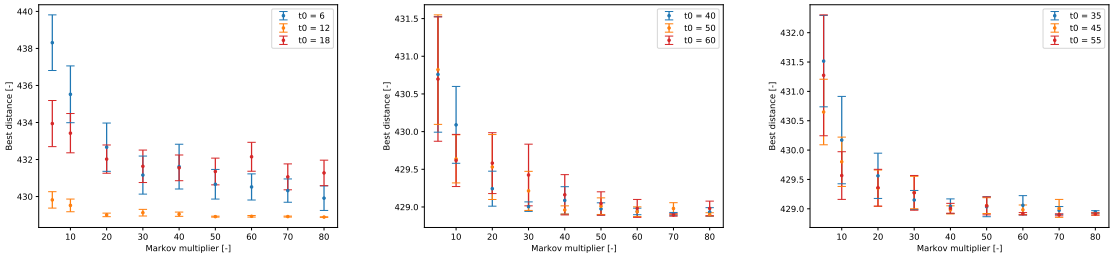
Table 1: The average and shortest found distance of the SA algorithm for the three different cooling schedules (logarithmic, linear & quadratic). The average distance is shown with a 95% confidence interval over 100 runs.

	Average distance	Shortest distance
Logarithmic	2664.78 ± 7.32	2587.78
Linear	2611.69 ± 4.06	2586.77
Quadratic	2612.83 ± 3.26	2586.77



(a) (b) (c)

Figure 4: The result from the parameter search to optimize t_0 for fixed values of ζ, being $\{5, 10 \text{ and } 15\}$. a) Logarithmic cooling b) Linear cooling c) Quadratic cooling. For logarithmic cooling a distinct minimum is visible, which is not apparent for the other cooling schedules.



(a) (b) (c)

Figure 5: The result from the parameter search to optimize the Markov Chain Length for fixed values of t_0, chosen according to the results from Figure 4. a) Logarithmic cooling b) Linear cooling c) Quadratic cooling. For linear and quadratic cooling, a distinct decrease in best distance is found for a higher ζ . This is less so observable for logarithmic cooling.

4 Conclusion

The SA algorithm was able to find a possible global minimum for the linear and quadratic cooling schedules. It was also able for these cooling schedules to reliably find solutions close to this minimum. Logarithmic cooling resulted in much quicker, but less optimal solutions to the problem. When computational cost is a factor for the problem, this cooling schedule might be preferred over linear and quadratic. From the results we can conclude that the parameter search resulted in very suitable parameters to solve the TSP. The method of tuning ζ instead of the absolute Markov chain length seemed to result in a great translation from the smaller 51 node problem to the 280 node problem.

In this research we tuned the parameters by creating convergence plots and manually picking the right parameters for the problem. However, an improvement might be made by automating this process to make it easier to implement different cooling schedules and tackle different TSP problems. This can be done by using a genetic algorithm to tune the parameters t_0 and ζ. Genetic algorithms are already applied to solve the TSP in combination with SA. Lin, Kao, and Hsu, 1993 uses a combined approach of an evolutionary algorithm with SA. However, it might be an interesting approach to only apply the genetic algorithm to the parameter search before performing SA.

References

Antosiewicz, M., Koloch, G., & Kamiński, B. (2013). Choice of best possible metaheuristic algorithm for the travelling salesman problem with limited computational time: Quality, uncertainty and speed. *Journal of Theoretical and Applied Computer Science*, *7*(1), 46–55.

Baudoin, F. (2010). Stochastic processes. In P. Peterson, E. Baker, & B. McGaw (Editors), *International encyclopedia of education (third edition)* (Third Edition, Pages 451–452). Elsevier. https://doi.org/https://doi.org/10.1016/B978-0-08-044894-7.01369-5

Grötschel, M., Jünger, M., & Reinelt, G. (1991). Optimal control of plotting and drilling machines: A case study. *Zeitschrift für Operations Research*, *35*(1), 61–84. https://doi.org/10.1007/BF01415960

Lin, F.-T., Kao, C.-Y., & Hsu, C.-C. (1993). Applying the genetic approach to simulated annealing in solving some np-hard problems. *IEEE Transactions on Systems, Man, and Cybernetics*, *23*(6), 1752–1767. https://doi.org/10.1109/21.257766

McQuarrie, D. (2000). Statistical mechanics university science books. *Sausalito, CA*, 222–223.

Reinelt, G. (1991). Tsplib—a traveling salesman problem library. *ORSA Journal on Computing*, *3*(4), 376–384. https://doi.org/10.1287/ijoc.3.4.376

Ross, S. (2013). Chapter 12 - markov chain monte carlo methods. In S. Ross (Editor), *Simulation (fifth edition)* (Fifth Edition, Pages 271–276). Academic Press.

Sharp, K., & Matschinsky, F. (2015). Translation of ludwig boltzmann's paper "on the relationship between the second fundamental theorem of the mechanical theory of heat and probability calculations regarding the conditions for thermal equilibrium" sitzungberichte der kaiserlichen akademie der wissenschaften. mathematisch-naturwissen c. *Entropy*, *17*, 1971–2009. https://doi.org/https://doi.org/10.3390/e17041971