Optimization by Simulated Annealing - A Review

Jon Raunkjær Søndergaard, Bjørn Hübschmann Mølvig, and Mads Ehrhorn Kjær
Niels Bohr Insitute
(Dated: March 6, 2019)

INTRODUCTION

Simulated annealing (SA) is a meta-heuristic optimization algorithm proposed in the article *Optimization by Simulated Annealing* (OBSA) [1] especially applicable in optimization problems with many local extrema. The algorithm draws inspiration from statistical mechanics, where the probability of finding a system in a certain state is proportional to its Boltzmann factor

$$P_i = \frac{1}{Z}\exp\left(-\frac{E_i}{k_\mathrm{B}T}\right), \tag{1}$$

where E_i is the energy of the system, T is the temperature and Z is the partition function. In particular, annealing refers to the process of first heating up a material to make it workable, then letting it cool slowly to improve strength; SA combines the Metropolis algorithm and this notion of a physical system to locate a global extremum in a configuration space.

As mentioned SA is a metaheuristic, and only requires the formulation of a cost function to be applicable, e.g. the Hamiltonian in statistical mechanics or the log-likelihood in fitting problems. The authors demonstrate how the algorithm can be used to search for optimal ways to wire computer-chips, where the minimization of the amount of wire and the amount of kinks is wanted. Furthermore it is demonstrated, how near-optimal solutions to the Travelling Salesman Problem (TSP) can be found.

THE SIMULATED ANNEALING ALGORITHM

SA as described in OBSA uses Metropolis-sampling to iteratively minimize a cost function (here E like the energy function in statistical mechanics), which describes the state of the system. In each step of the algorithm, a change is made to the system's configuration, thereby changing the value of the cost function. The change is accepted with probability

$$P_\mathrm{transition} = \begin{cases} 1, & \text{if } E_\mathrm{new} < E_\mathrm{old} \\ \exp(-\Delta E/T), & \text{if } E_\mathrm{new} \geq E_\mathrm{old}. \end{cases} \tag{2}$$

Changes, which leave the system in a worse state therefore have a non-zero probability of being accepted. This feature is what enables the simulated annealing heuristic to make its way out of local extrema to find the global extremum.

Throughout the annealing process, the temperature T is lowered. In the early stages, the temperature is high and therefore changes to the system, which increase E, are more likely to happen. By slowly cooling the system, the neighbourhood in the parameter space containing the extremum of E is zoomed in on, hopefully navigating away from local extrema.

To use SA, one must assign a number of problem specific hyperparameters related to

- the initial and final temperatures T_0 and T_f,
- the number of different temperatures N_T,
- the number of iterations at each temperature N_ite and
- the functional form of the temperature-decrease $f(T)$ (which might be associated with even more hyperparameters).

The annealing process must proceed long enough for the system to reach a steady state at each temperature. The authors suggest a linear or a small power relation between N_ite and the number of free parameters in the system.

APPLICATIONS

In the article the SA algorithm is used to optimize the physical design of a computer, partitioned into three design stages: The optimal way for circuits to be distributed on two chips, the optimal placement of 98 chips on a module and the optimal wiring of the chips. In each case, the authors provide cost-functions, where attributes like the number of kinks on the wire is wanted minimized.

In each stage of the design SA is used to optimize the placement of the components without compromising the optimization done in previous steps. For each of the stages, the Metropolis algorithm is used with decreasing temperatures to optimize the design.

These examples demonstrate how SA can be used to optimize real but complex problems, which do not have a clear analytical solution.

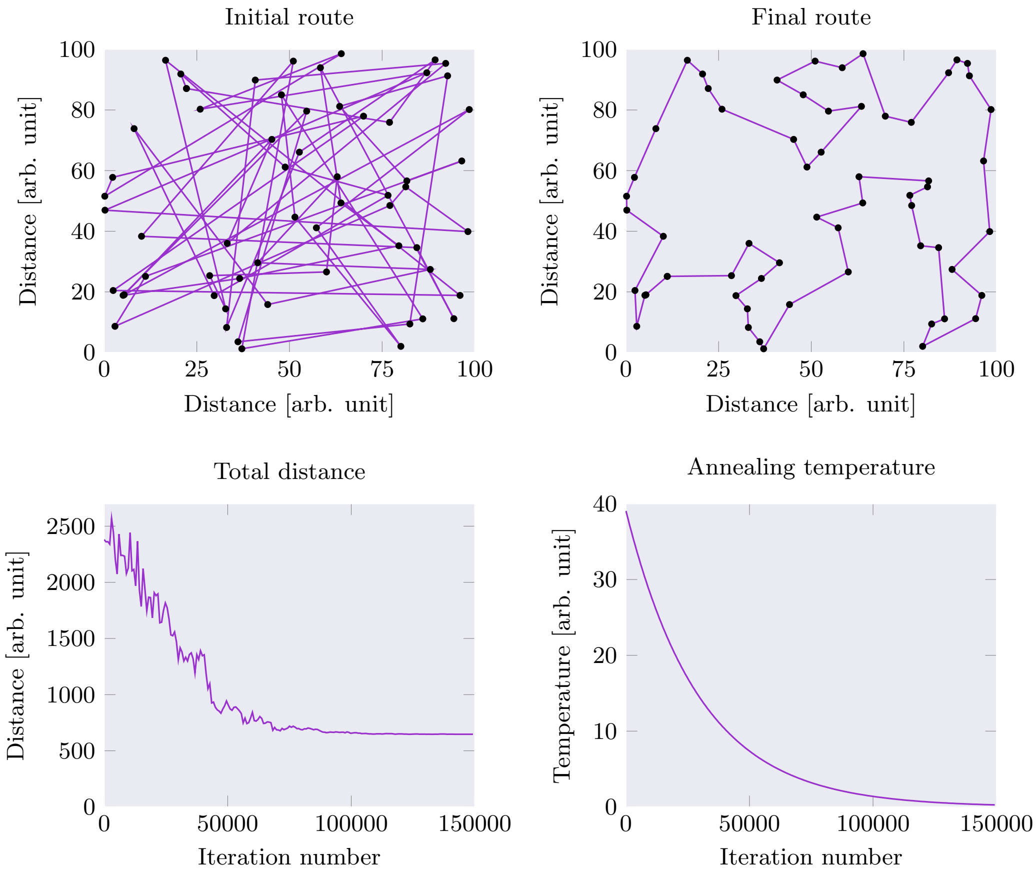

FIG. 1. A solution to TSP (with relevant plots) generated using SA with an exponentially decaying temperature. Upper left: Initial, random route. Upper right: Near-optimal route after a total of 150.000 steps. Lower left: Length of the route as a function of iteration number. Lower right: Temperature as a function of iteration number.

THE TRAVELLING SALESMAN - A DEMONSTRATION

TSP is a heavily studied NP-hard problem with a brute-force solution of time complexity $\mathcal{O}(n!)$[2]. The problem can be stated as: Given a list of cities, what is the shortest route that visits each city once before returning home? The problem is often used in testing optimization algorithms, which is also the case for the article being discussed here.

The cost-function to be minimized in TSP is simply the length of the route. To find a solution to the problem using SA, an ordered list of cities is given, each city is visited in consecutive order and the total length of the route is determined. In each step of the algorithm, two cities are swapped and the new total length is evaluated. The swap is accepted with a probability evaluated by Eq. (2), and after the desired amount of iterations, a candidate solution is attained.

In Figure 1 a near-optimal solution generated with our own implementation for 60 cities is shown along with the initial state, the time evolution of the temperature and the time evolution of the cost function. The cities were generated as pairs of uniformly distributed numbers between 0 and 100. An exponentially decreasing temperature function of the form

$$f(T) = T_0 \exp(-\alpha t) \tag{3}$$

was used with $\alpha = 0.998$, $T_0 = 40$, 2500 time steps and 60 swaps were attempted at each temperature. The lower left plot in Figure 1 shows how swaps increasing the cost-function by large amounts are only accepted early in the annealing process. Furthermore it shows how the cost function fluctuations become increasingly smaller as the annealing process evolves.

[1] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220:671–680, 1983.

[2] Travelling salesman problem. `https://en.wikipedia.org/wiki/Travelling_salesman_problem`. Accessed 4 Mar. 2019.