A Comparison of Cooling Schedules for Simulated Annealing

José Fernando Díaz Martín
University of Deusto, Spain

Jesús M. Riaño Sierra
University of Deusto, Spain

INTRODUCTION

Simulated annealing is one of the most important **metaheuristics** or general-purpose algorithms of **combinatorial optimization**, whose properties of convergence towards high quality solutions are well known, although with a high computational cost. Due to that, it has been produced a quite number of research works on the convergence speed of the algorithm, especially on the treatment of the temperature parameter, which is known as **cooling schedule** or strategy. In this article we make a comparative study of the performance of simulated annealing using the most important cooling strategies (Kirkpatrick, S., Gelatt, C.D. & Vecchi, M.P., 1983), (Dowsland, K.A., 2001), (Luke, B.T., 1995), (Locatelli, M., 2000). Two classical problems of combinatorial optimization are used in the practical analysis of the algorithm: the **travelling salesman problem** and the **quadratic assignment problem**.

BACKGROUND

The main aim of **combinatorial optimization** is the analysis and the algorithmic solving of constrained optimization problems with discrete variables. Problems that require algorithms of non-polynomial time complexity with respect to the problem size, called NP-complete problems, are the most important ones.

The general solving techniques of this type of problems belong to three different, but related, research fields. First, we can mention heuristic search algorithms, such as the deterministic algorithms of *local search* (Johnson, D.S., Papadimitriou, C.H. & Yannakakis, M., 1985), (Aarts, E.H.L. & Lenstra, J., 1997), the stochastic algorithm of *simulated annealing* (Kirkpatrick, S., Gelatt, C.D. & Vecchi, M.P., 1983), and the *taboo search* (Glover, F., 1986). A second

kind of solving techniques are algorithms inspired in genetics and the evolution theory, such as *genetic and evolutionary algorithms* (Holland, J.H., 1973), (Goldberg, D.E., 1989), and *memetic algorithms* (Moscato, P., 1999). Finally, due to the collective computation properties of some neural models, the area of artificial neural networks has contributed a third approach, although possibly not so relevant as the former ones, to the combinatorial optimization problem solving with the *Hopfield nets* (Hopfield, J.J. & Tank, D., 1985), the *Boltzmann machine* (Aarts, E.H.L. & Korst, J., 1989), and the *self-organizing map* (Kohonen, T., 1988).

Simulated Annealing Algorithm

The **simulated annealing** is a stochastic variant of the **local search** that incorporates a stochastic criterion of acceptance of worse quality solutions, in order to prevent the algorithm from being prematurely trapped in local optima. This acceptance criterion is based on the **Metropolis algorithm** for simulation of physical systems subject to a heat source (Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. & Teller, E., 1953).

Algorithm (simulated annealing). Be a combinatorial optimization problem (X,S,f,R), with generator function of random k-neighbour feasible solutions $g : S \times [0,1[\to S$. Supposing, without loss of generality, that f must be minimized, the **simulated annealing** algorithm can be described in the following way:

1. Set an initial random feasible solution as current solution, $s_i = s_0$.
2. Set initial temperature or control parameter $T = T_0$.
3. Obtain a new solution that differs from the current one in the value of k variables using the generator

function of random k-neighbour feasible solutions, $s_j = g(s_i, random[0,1[)$.

4. If the new solution s_j is better than the current one, $f(s_j) < f(s_i)$, then s_j is set as current solution, $s_i = s_j$. Otherwise, if

$$e^{\frac{f(s_i) - f(s_j)}{T}} > random[0,1[$$

s_j is equally accepted as current solution, $s_i = s_j$.

5. If the number of executed state transitions (steps 3 and 4) for the current value of temperature is equal to L, then the temperature T is decreased.

6. If there are some k-neighbour feasible solutions near to the current one that have not been processed yet, steps 3 to 5 must be repeated. The algorithm ends in case the set of k-neighbour solutions near to the current one has been processed completely with a probability close to 1 without obtaining any improvement in the quality of the solutions.

The most important feature of the simulated annealing algorithm is that, besides accepting transitions that imply an improvement in the solution cost, it also allows to accept a decreasing number of transitions that mean a quality loss of the solution.

The simulated annealing algorithm converges asymptotically towards the set of global optimal solutions of the problem. E. Aarts and J. Korst provide a complete proof in their book *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing* (1989). Essentially, the convergence condition towards global optimum sets that temperature T of the system must be decreased logarithmically according to the equation:

$$T_k = \frac{T_0}{1 + Log(1 + k)} \tag{1}$$

where k = 0,1,...,n indicates the temperature cycle. However, this function of system cooling requires a prohibitive computing time, so it is necessary to consider faster methods of temperature decrease.

Cooling Schedules

A practical simulated annealing implementation requires generating a finite sequence of decreasing values of temperature T, and a finite number L of state transitions for each temperature value. To achieve this aim, a ***cooling schedule*** must be specified.

The following cooling schedule, frequently used in the literature, was proposed by Kirkpatrick, Gelatt and Vecchi (1983), and it consists of three parameters:

- *Initial temperature*, T_0. The initial value of temperature must be high enough so that any new solution generated in a state transition should be accepted with a certain probability close to 1.
- *Temperature decrease function*. Generally, an exponential decrease function is used, such as $T_k = T_0 \cdot \alpha^k$, where α is a constant smaller than the unit. Usual values of α fluctuate between 0.8 and 0.99.
- *Number of state transitions*, L, for each temperature value. Intuitively, the number of transitions for each temperature must be high enough so that, if no solution changes were accepted, the whole set of k-neighbour feasible solutions near to the current one could be gone round with a probability close to 1.

The initial temperature, T_0, and the number of state transitions, L, can be easily obtained. On the other hand, the temperature decrease function has been studied in numerous research works (Laarhoven, P.J.M. Van & Aarts, E.H.L., 1987), (Dowsland, K.A., 2001), (Luke, B.T., 2005), (Locatelli, M., 2000).

MAIN FOCUS OF THE CHAPTER

In this section nine different cooling schedules used in the comparison of the simulated annealing algorithm are described. They all consist of, at least, three parameters: initial temperature T_0, temperature decrease function, and number of state transitions L for each temperature.

Multiplicative Monotonic Cooling

In the multiplicative monotonic cooling, the system temperature T at cycle *k* is computed multiplying the initial temperature T_0 by a factor that decreases with respect to cycle *k*. Four variants are considered:

C

Related Content

War and Peace: Ethical Challenges and Risks in Military Robotics
Racquel D. Brown-Gastonand Anshu Saxena Arora (2021). *International Journal of Intelligent Information Technologies (pp. 1-12).*
www.irma-international.org/article/war-and-peace/286621

Personalized Learning Experiences: AI Applications in the Metaverse for Education
Ankush Joshi, Vikash Kumar, Gesu Thakurand Naman Chauhan (2025). *Navigating AI and the Metaverse in Scientific Research (pp. 413-434).*
www.irma-international.org/chapter/personalized-learning-experiences/377304

AI Techniques
P. Selvakumar, Pamarthi Satyanarayana, Yuvraj Lahoti, Lalitha Saipogu, Mohit Sharmaand T. C. Manjunath (2025). *AI Methods for Environmental Protection and Resource Conservation (pp. 107-138).*
www.irma-international.org/chapter/ai-techniques/379032

Blockchain-Based Intelligent Risk Management Decision Support System for Supply Chain Financing
Xinjian Zhang, Yuting Zhang, Xuefeng Liuand Ruping Wang (2025). *International Journal of Intelligent Information Technologies (pp. 1-24).*
www.irma-international.org/article/blockchain-based-intelligent-risk-management-decision-support-system-for-supply-chain-financing/369153

A Scalable Algorithm for Constructing Frequent Pattern Tree
Zailani Abdullah, Tutut Herawan, A. Noraziahand Mustafa Mat Deris (2014). *International Journal of Intelligent Information Technologies (pp. 42-56).*
www.irma-international.org/article/a-scalable-algorithm-for-constructing-frequent-pattern-tree/111324