Chapter 2
Simulated Annealing

Patrick Siarry

2.1 Introduction

The complex structure of the configuration space of a hard optimization problem has inspired people to draw analogies with physical phenomena, which led three researchers at IBM—Kirkpatrick, Gelatt, and Vecchi—to propose in 1982, and to publish in 1983, a new iterative method, the simulated annealing technique [23], which can avoid local minima. A similar method, developed independently at the same time by Cerny [7], was published in 1985.

Since its discovery, the simulated annealing method has proved its effectiveness in various fields, such as the design of electronic circuits, image processing, the collection of household garbage, and the organization of the data-processing network of the French Loto Lottery. On the other hand, it has been found too greedy to solve certain combinatorial optimization problems, which could be solved better by specific heuristics, or completely incapable of solving them.

This chapter starts by initially explaining the principle of the method, with the help of an example of the problem of the layout of an electronic circuit. This is followed by a simplified description of some theoretical approaches to simulated annealing, which underlines its strong points (conditional guaranteed convergence towards a global optimum) and its weak points (tuning of the parameters, which can be delicate in practice). Then various techniques for parallelization of the method are discussed. This is followed by the presentation of some applications. In conclusion, we recapitulate the advantages and the most significant drawbacks of simulated annealing. We put forward some simple practical suggestions, intended for users who are planning to develop their first application based on simulated annealing. In

P. Siarry (✉)
Laboratoire Images, Signaux et Systèmes Intelligents (LiSSi, E.A. 3956),
Université Paris-Est Créteil Val-de-Marne, 122 rue Paul Armangot,
94400 Vitry-sur-Seine, France
e-mail: siarry@u-pec.fr

P. Siarry (ed.), *Metaheuristics*, DOI 10.1007/978-3-319-45403-0_2

Sect. 2.8, we recapitulate the main results of the modeling of simulated annealing based on Markov chains.

This chapter presents, in part, a summary of the review book on the simulated annealing technique [42], which we published at the beginning of 1989; this presentation is augmented by mentioning more recent developments [31, 40]. The references mentioned in the text were selected either because they played a significant role or because they illustrate a specific point in the discussion. A much more exhaustive bibliography—although old—can be found in [37, 42, 47, 50] and in the article [8] on the subject. Interested readers are also recommended to read the detailed presentations of simulated annealing in the article [29] and in Chap. 3 of [31].

2.2 Presentation of the Method

2.2.1 Analogy Between an Optimization Problem and Some Physical Phenomena

The idea of simulated annealing can be illustrated by a picture inspired by the problem of the layout and routing of electronic circuits: let us assume that a relatively inexperienced electronics specialist has randomly spread the components out on a plane, and connections have been made between them without worrying about technological constraints.

It is clear that the solution obtained is an unacceptable one. The purpose of developing a layout-routing program is to transform this disordered situation to an ordered electronic circuit diagram, where all connections are rectilinear, and the components are aligned and placed so as to minimize the length of the connections. In other words, this program must carry out a disorder–order transformation which, starting from a "liquid of components," leads to an ordered "solid."

However, such a transformation occurs spontaneously in nature if the temperature of a system is gradually lowered; there are computer-based digital simulation techniques available which show the behavior of sets of particles interacting in a way that depends on the temperature. In order to apply these techniques to optimization problems, an analogy can be established which is presented in Table 2.1.

Table 2.1 Analogy between an optimization problem and a physical system

Optimization problem	*Physical system*
Objective function	Free energy
Parameters of the problem	"Coordinates" of the particles
Find a "good" configuration (or even an optimal configuration)	Find the low-energy states

To lead a physical system to a low-energy state, physicists generally use an annealing technique: we will examine how this method of treatment of materials (real annealing) is helpful in dealing with an optimization problem (simulated annealing).

2.2.2 Real Annealing and Simulated Annealing

To modify the state of a material, physicists have an adjustable parameter: the temperature. To be specific, annealing is a strategy where an optimum state can be approached by controlling the temperature. To gain a deeper understanding, let us consider the example of the growth of a monocrystal. The annealing technique consists in heating the material beforehand to impart high energy to it. Then the material is cooled slowly, in a series of stages at particular temperatures, each of sufficient duration; if the decrease in temperature is too fast, it may cause defects which can be eliminated by local reheating. This strategy of a controlled decrease in the temperature leads to a crystallized solid state, which is a stable state, corresponding to an absolute minimum of energy. The opposite technique is that of quenching, which consists in lowering the temperature of the material very quickly: this can lead to an amorphous structure, a metastable state that corresponds to a local minimum of energy. In the annealing technique, the cooling of the material causes a disorder–order transformation, while the quenching technique results in solidifying a disordered state.

The idea of using an annealing technique in order to deal with optimization problems gave rise to the simulated annealing technique. This consists in introducing a control parameter in to the optimization process, which plays the role of the temperature. The "temperature" of the system to be optimized must have the same effect as the temperature of a physical system: it must condition the number of accessible states and lead towards the optimal state if the temperature is lowered gradually in a slow and well-controlled manner (as in the annealing technique), and towards a local minimum if the temperature is lowered abruptly (as in the quenching technique).

To conclude, we have to describe an algorithm in such a way that will enable us to implement annealing on a computer.

2.2.3 Simulated Annealing Algorithm

The algorithm is based on two results from statistical physics.

On one hand, when thermodynamic equilibrium is reached at a given temperature, the probability that a physical system will have a given energy E is proportional to the Boltzmann factor: $e^{\frac{-E}{k_B T}}$, where k_B denotes the Boltzmann constant. Then, the distribution of the energy states is the Boltzmann distribution at the temperature considered.

On the other hand, to simulate the evolution of a physical system towards its thermodynamic equilibrium at a given temperature, the Metropolis algorithm [25]

can be utilized: starting from a given configuration (in our case, an initial layout for all the components), the system is subjected to an elementary modification (for example, a component is relocated or two components are exchanged); if this transformation causes a decrease in the objective function (or "energy") of the system, it is accepted; in contrast, if it causes an increase ΔE in the objective function, it may also be accepted, but only with a probability $e^{-\Delta E/T}$. (In practice, this condition is realized in the following manner: a real number is drawn at random, ranging between 0 and 1, and a configuration causing a degradation by ΔE in the objective function is accepted if the random number drawn is less than or equal to $e^{-\Delta E/T}$.) By repeatedly following this Metropolis rule of acceptance, a sequence of configurations is generated, which constitutes a Markov chain (in the sense that each configuration depends on only that one which immediately precedes it). With this formalism in place, it is possible to show that, when the chain is of infinite length (in practice, of "sufficient" length), the system can reach (in practice, can approach) thermodynamic equilibrium at the temperature considered: in other words, this leads us to a Boltzmann distribution of the energy states at this temperature.

Hence the role given to the temperature by the Metropolis rule is well understood. At high temperature, $e^{-\Delta E/T}$ is close to 1, and therefore the majority of the moves are accepted and the algorithm becomes equivalent to a simple random walk in the configuration space. At low temperature, $e^{-\Delta E/T}$ is close to 0, and therefore the majority of the moves that increase the energy are rejected. Hence the algorithm reminds us of a classical iterative improvement. At an intermediate temperature, the algorithm intermittently allows transformations that degrade the objective function: hence it leaves a chance for the system to be pulled out of a local minimum.

Once thermodynamic equilibrium is reached at a given temperature, the temperature is lowered "slightly," and a new Markov chain is implemented in this new temperature stage (if the temperature is lowered too quickly, the evolution towards a new thermodynamic equilibrium is slowed down: the theory of the method establishes a narrow correlation between the rate of decrease in the temperature and the minimum duration of the temperature stage). By comparing the successive Boltzmann distributions obtained at the end of the various temperature stages, a gradual increase in the weight of the low-energy configurations can be noted: when the temperature tends towards zero, the algorithm converges towards the absolute minimum of energy. In practice, the process is terminated when the system is "solidified" (which means that either the temperature has reached zero or no more moves causing an increase in energy have been accepted during the stage).

2.3 Theoretical Approaches

The simulated annealing algorithm was implemented in many theoretical studies for the following two reasons: on one hand, it was a new algorithm, for which it was necessary to establish the conditions for convergence; and on the other hand, the method contains many parameters and has many variants, whose effect or influence

on the mechanism needed to be properly understood if one wished to implement the method to maximum effect.

These approaches, especially those which appeared during the initial years of its formulation, are presented in detail in the book [42]. Here, we focus on emphasizing on the principal aspects treated in the literature. The theoretical convergence of simulated annealing is analyzed first. Then those factors which are influential in the operation of the algorithm are analyzed in detail: the structure of the configuration space, the acceptance rules, and the annealing program.

2.3.1 Theoretical Convergence of Simulated Annealing

Many mathematicians have invested effort in research into the convergence of the simulated annealing (see in particular [1, 16, 17]) and some of them have even endeavored to develop a general model for the analysis of stochastic methods of global optimization (notably [32, 33]). The main outcome of these theoretical studies is that under certain conditions (discussed later), simulated annealing probably converges towards a global optimum, in the sense that it is possible to obtain a solution arbitrarily close to this optimum with a probability arbitrarily close to unity. This result is, in itself, significant because it distinguishes simulated annealing from other metaheuristic competitors, whose convergence is not guaranteed.

However, the establishment of the "conditions of convergence" is not unanimously accepted. Some of these conditions, such as those proposed by Aarts and Van Laarhoven [1], are based on the assumption of decreasing the temperature in stages. This property enables one to represent the optimization process in the form of completely connected homogeneous Markov chains, whose asymptotic behavior can be described simply. It has also been shown that convergence is guaranteed provided that, on one hand, reversibility is respected (the opposite of any allowed change must also be allowed) and, on the other hand, connectivity of the configuration space is also maintained (any state of the system can be reached starting from any other state with the help of a finite number of elementary changes). This formalization has two advantages:

- it enables us to legitimize the lowering of the temperature in stages, which improves the convergence speed of the algorithm;
- it enables us to establish that a "good"-quality solution (located significantly close to the global optimum) can be obtained by simulated annealing in a polynomial time for certain NP-hard problems [1].

Some other authors, in particular Hajek et al. [16, 17], were interested in the convergence of simulated annealing within the more general framework of the theory of inhomogeneous Markov chains. In this case, the asymptotic behavior was the more sensitive aspect of the study. The main result of this work was the following: the algorithm converges towards a global optimum with a probability of unity if, as the

time t tends towards infinity, the temperature $T(t)$ does not decrease more quickly than the expression $C/\log(t)$, where C is a constant related to the depth of the "energy wells" of the problem. It should be stressed that the results of this theoretical work, at present, are not sufficiently general and unambiguous to be used as a guide to an experimental approach when one is confronted with a new problem. For example, the logarithmic law of decrease of the temperature recommended by Hajek is not used in practice for two major reasons: on one hand, it is generally impossible to evaluate the depth of the energy wells of the problem, and, on the other hand, this law leads to an unfavorable increase in computing time.

We now continue this analysis with careful, individual examination of the various components of the algorithm.

2.3.2 Configuration Space

The configuration space plays a fundamental role in the effectiveness of the simulated annealing technique in solving complex optimization problems. It is equipped with a "topology," originating from the concept of proximity between two configurations: the "distance" between two configurations represents the minimum number of elementary changes required to pass from one configuration to the other. Moreover, there is an energy associated with each configuration, so that the configuration space is characterized by an "energy landscape." All of the difficulties of the optimization problem lie in the fact that the energy landscape comprises of a large number of valleys of varying depth, possibly relatively close to each other, which correspond to local minima of energy.

It is clear that the shape of this landscape is not specific to the problem under study, but depends to a large extent on the choice of the cost function and the choice of the elementary changes. However, the required final solution, i.e., the global minimum (or one of the global minima of comparable energy), must depend primarily on the nature of the problem considered, and not (or very little) on these choices. We have shown, with the help of an example problem of placement of building blocks, considered specifically for this purpose, that an apparently sensitive problem can be greatly simplified either by widening the allowable configuration space or by choosing a better adapted topology [42].

Several authors have endeavored to establish general analytical relations between certain properties of the configuration space and the convergence of simulated annealing. In particular, some of their work was directed towards an analysis of the energy landscapes, and they sought to develop a link between "ultrametricity" and simulated annealing [22, 30, 44]: the simulated annealing method would be more effective for those optimization problems whose low local minima (i.e., the required solutions) formed an ultrametric set. Thereafter, Sorkin [45] showed that certain fractal properties of the energy landscape induce polynomial convergence of simulated annealing; Sorkin explained this on the basis of the effectiveness of the method in the field of electronic circuit layouts. In addition, Azencott [3] utilized the "theory of cycles"

(originally developed in the context of dynamic systems) to establish general explicit relations between the geometry of the energy landscape and the expected performance of simulated annealing. This work led to the proposal of the "method of distortions" for the objective function, which significantly improved the quality of the solutions for certain difficult problems [11]. However, all these approaches to simulated annealing are still in a nascent stage, and their results have not yet been generalized.

Lastly, another aspect of immediate practical interest relates to the adaptation of simulated annealing to the solution of continuous optimization problems [9, 39]. Here, we stress only the transformations necessary to make the step from "combinatorial simulated annealing" to "continuous simulated annealing." In fact, the method was originally developed for application in the domain of combinatorial optimization problems, where the free parameters can take discrete values only. In the majority of these types of problems encountered in practice, the topology is almost always considered as data for the problem: for example, in the traveling salesman problem, the permutation of two cities has a natural tendency to generate round-trip routes close to a given round-trip route. The same thing occurs in the problem of placement of components when the exchange of two blocks is considered. On the other hand, when the objective is to optimize a function of continuous variables, the topology has to be updated. This gives rise to the concept of "adaptive topology": here, the length of the elementary steps is not imposed by the problem anymore. This choice must instead be dictated by a compromise between two extreme situations: if the step is too small, the program explores only a limited region of the configuration space; the cost function is then improved very often, but by a negligible amount. In contrast, if the step is too large, the test results are accepted only seldom, and they are almost independent of each other. From the point of mathematical interest, it is necessary to mention the work of Miclo [26], which was directed towards the convergence of simulated annealing in the continuous case.

2.3.3 Rules of Acceptance

The principle of simulated annealing requires that one accepts, occasionally and under the control of the "temperature," an increase in the energy of the current state, which enables it to be pulled out of a local minimum. The rule of acceptance generally used is the Metropolis rule described in Sect. 2.2.3. This possesses the advantage that it originates directly from statistical physics. There are, however, several variations of this rule [42], which can be more effective from the point of view of computing time.

Another aspect arises from examination of the following problem: at low temperature, the rate of acceptance of the algorithm becomes very small, and hence the method is ineffective. This is a well-known problem encountered in simulated annealing, which can be solved by substituting the traditional Metropolis rule with an accelerated alternative, called the "thermostat" [42], as soon as the rate of acceptance falls too low. In practice, this methodology is rarely employed.

2.3.4 Program of Annealing

The convergence speed of the simulated annealing methodology depends primarily on two factors: the configuration space and the program of annealing. With regard to the configuration space, readers have already been exposed to the effects of topology on convergence and the shape of the energy landscape. Let us discuss the influence of the "program of annealing": this addresses the problem of controlling the "temperature" as well as the possibility of a system reaching a solution as quickly as possible. The program of annealing must specify the following values of the control parameters for the temperature:

- the initial temperature;
- the length of the homogeneous Markov chains, i.e., the criterion for changing to the next temperature stage;
- the law of decrease of the temperature;
- the criterion for program termination.

In the absence of general theoretical results which can be readily exploited, the user has to resort to empirical adjustment of these parameters. For certain problems, the task is complicated even further by the great sensitivity of the result (and the computing time) to this adjustment. This aspect—which unites simulated annealing with other metaheuristics—is an indisputable disadvantage of this method.

To elaborate on the subject a little more, we shall look at the characteristic of the program of annealing that has drawn most attention: the law of decrease of the temperature. The geometrical law of decrease, $T_{k+1} = \alpha \cdot T_k$, $\alpha =$ constant, is a widely accepted one, because of its simplicity. An alternative solution, potentially more effective, is an adaptive law of the form $T_{k+1} = \alpha(T_k) \cdot T_k$, but it is then necessary to exercise a choice from among several laws suggested in the literature. One can show, however, that several traditional adaptive laws, which have quite different origins and mathematical expressions, are in practice equivalent (see Fig. 2.1), and can be expressed in the following generic form:

$$T_{k+1} = \left(1 - T_k \cdot \frac{\Delta\left(T_k\right)}{\sigma^2\left(T_k\right)}\right) \cdot T_k$$

where

$$\sigma^2\left(T_k\right) = \left\langle f_{T_k}^2 \right\rangle - \left\langle f_{T_k} \right\rangle^2,$$

f denotes the objective function, and $\Delta\left(T_k\right)$ depends on the adaptive law selected. The simplest adjustment, $\Delta\left(T_k\right) =$ constant, can then be made effective, although it does not correspond to any of the traditional laws.

Owing to our inability to synthesize the results (both theoretical and experimental) presented in the literature, which show some disparities, the reader is referred to Sect. 2.7, where we propose a suitable tuning algorithm for the four parameters of the program of annealing, which can often be useful at least to start with.

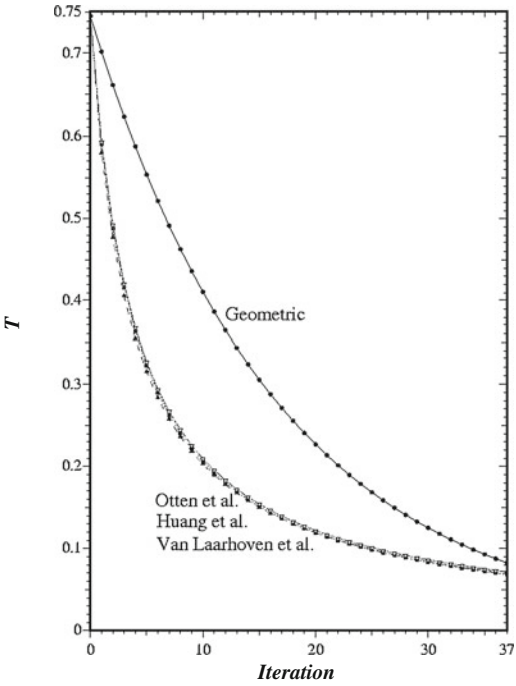


Fig. 2.1 Lowering of the temperature according to the number of stages for the geometrical law and several traditional laws

Those readers who are interested in the mathematical modeling of simulated annealing are advised to refer to Sect. 2.8: the principal results produced by the Markov formalism are described there.

2.4 Parallelization of the Simulated Annealing Algorithm

Often, the computing time becomes a critical factor in the economic evaluation of the utility of a simulated annealing technique for applications to real industrial problems. A promising research direction to reduce this time is the parallelization of the algorithm, which consists in simultaneously carrying out several of the calculations necessary for its realization. This step can be considered in the context of the significant activity that has been developing around the algorithms and architectures of parallel computation for quite some time now. This may appear paradoxical, because of the sequential structure of the algorithm. Nevertheless, several types of parallelization have been considered to date. A book [3] completely devoted to this topic has been published; it describes simultaneously the rigorous mathematical results available and the results, of simulations executed on parallel or sequential computers. To provide a concrete idea, we shall describe the idea behind two principal modes of parallelization, which are independent of the problem being dealt

with and were suggested very soon after the invention of simulated annealing. The distinction between these two modes remains relevant today, as has been shown in the recent status of the state of the art described by Delamarre and Virot [11].

The first type of parallelization [2] consists in implementing several Markov chain computations in parallel, by using K elementary processors. To implement this, the algorithm is decomposed into K elementary processes, constituting K Markov chains. Let L be the length of these Markov chains, assumed constant, each chain is divided into K subchains of length L/K. The first processor executes the first chain at the initial temperature, and implements the first L/K elements of this chain (i.e., the first subchain); then it calculates the temperature of the following Markov chain, starting from the states already obtained. The second elementary processor then begins executing the second Markov chain at this temperature, starting from the final configuration of the first subchain of the first chain. During this time, the first processor begins the second subchain of the first chain. This process continues for the K elementary processors. It has been shown that this mode of parallelization—described in more detail in [42]—allows one to divide the total computing time by a factor K, if K is small compared with the total number of Markov chains carried out. However, the procedure has a major disadvantage: its convergence towards an optimum is not guaranteed. The formalism of Markov chains enables one to establish that the convergence of simulated annealing is assured provided that the distribution of the states, at the end of each Markov chain is close to the stationary distribution. In the case of the algorithm described here, however, this closeness is not established at the end of each subchain, and the larger the number K of processors in parallel, the larger is the deviation from closeness.

The second type of parallelization [24, 35] consists in carrying out the computation in parallel for several states of the same Markov chain while keeping in mind the following condition: at low temperature, the number of elementary transformations rejected becomes very important; it is thus possible to assume that these moves are produced by independent elementary processes, which may likely be implemented in parallel. Then the computing time can be divided by approximately the number of processes. One strategy consists in subdividing the algorithm into K elementary processes, each of which is responsible for calculating the energy variations corresponding to one or more elementary moves, and for carrying out the corresponding Metropolis tests. Two operating modes are considered:

- At "high temperature," a process corresponds to only one elementary move. Each time K elementary processes are implemented in parallel, one can randomly choose a transition from among those which have been accepted, and the memory, containing the best solution known, is updated with the new configuration.
- At "low temperature," the accepted moves become very rare: less than one transition is accepted for K moves carried out. Each process then consists in calculating the energy variations corresponding to a succession of disturbances until one of them is accepted. As soon as any of the processes succeeds, the memory is updated.

These two operating modes can ensure behavior, and in particular convergence, which is strictly identical to that of sequential algorithms. This type of parallelization

has been tested by experimenting on the optimization problem of the placement of connected blocks [35]. We estimated the amount of computing time saved in two cases: the placement of presumed point blocks in predetermined sites and the placement of real blocks on a plane. With five elementary processes in parallel, the saving in computing time was between 60 and 80 %, depending on the program of annealing used. This work was then continued, in the thesis work of Roussel-Ragot [34] by considering a theoretical model, which was validated by programming the simulated annealing using a network of "transputers."

In addition to these two principal types of parallelization of simulated annealing, which should be applicable to any optimization problem, other methodologies have been proposed to deal with specific problems. Some of these problems are problems of placement of electronic components, problems in image processing and problems of meshing of areas (for the finite element method). In each of these three cases, information is distributed in a plane or in space, and each processor can be entrusted with the task of optimizing the data pertaining to a geographical area by simulated annealing; here information is exchanged periodically between neighboring processors.

Another step to reduce the cost of synchronization between processors has been planned: the algorithms known as "asynchronous algorithms" are designed to calculate the energy variations starting from partially out-of-date data. However, it seems very complex and sensitive to control the admissible error, except for certain particular problems [12].

As an example, let us describe the asynchronous parallelization technique suggested by Casotto et al. [6] to deal with the problem of the placement of electronic components. The method consists in distributing the components to be placed into K independent groups, assigned to K respective processors. Each processor applies the simulated annealing technique to seek the optimal site for the components that belong to its group. The processors function in parallel, and in an asynchronous manner with respect to each other. All of them have access to a common memory, which contains the current state of the circuit plan. When a processor plans to exchange the position of a component in its group with that of a component in another group belonging to another processor, it temporarily blocks the activity of that processor. It is clear that the asynchronous working of the processors involves errors, in particular in the calculation of the overlap between the blocks, and thus in the evaluation of the cost function. In fact, when a given processor needs to evaluate the cost of a move (translation or permutation), it will search in the memory for the current positions of all the components in the circuit. However, the information collected is partly erroneous, since certain components are in the course of displacement because of the activities of other processors. In order to limit these errors, the method is supplemented by the following two processes. On one hand, the distribution of the components between the processors is in itself an object of optimization by the simulated annealing technique, which is performed simultaneously with the optimization process already described: in this manner, membership of the components geographically close to the same group can be favored. In addition, the maximum amplitude of the moves carried out by the components is reduced as the temperature decreases. Consequently, when the

temperature decreases, the moves relate mainly to nearby components, which thus generally belong to the same group. In this process, the interactions between the groups can be reduced, thus reducing the frequency of the errors mentioned above. This technique of parallelization of simulated annealing was validated using several examples of real circuits: the algorithm functioned approximately six times faster with eight processors than with only one, the results being of comparable quality to those of the sequential algorithm.

2.5 Some Applications

The majority of the preceding theoretical approaches are based on asymptotic behaviors which impose several restrictive assumptions, very often causing excessive increases in computing times. This is why, to solve real industrial problems under reasonable conditions, it is often essential to adopt an experimental approach, which may frequently result in crossing the barriers recommended by the theory. The simulated annealing method has proved to be interesting for solving many optimization problems, both NP-hard and not. Some examples of these problems are presented here.

2.5.1 Benchmark Problems of Combinatorial Optimization

The effectiveness of the method was initially tested on some "benchmark problems" of combinatorial optimization. In this type of problem, the practical purpose is secondary: the initial objective is to develop the optimization method and to compare its performance with that of other methods. We will detail only one example: that of the traveling salesman problem.

The reason for the choice of this problem is that it is very simple to formulate and, at the same time, very difficult to solve: the largest problems for which the optimum has been found and proved comprise a few thousands of cities. To illustrate the disorder–order transformation that occurs in the simulated annealing technique as the temperature goes down, we present in Fig. 2.2 four intermediate configurations obtained by Eric Taillard, in the case of 13 206 towns and villages in Switzerland.

Bonomi and Lutton also considered very high-dimensional examples, with between 1000 and 10 000 cities [4]. They showed that, to avoid a prohibitive computing time, the domain containing the cities can be deconstructed into areas, and the moves for the route of the traveler can be forced so that they are limited to being between cities located in contiguous areas. Figure 2.3 shows the effectiveness of this algorithm for a problem comprising 10 000 cities: the length of this route does not exceed that of the optimal route by more than 2 % (the length of the shortest route can be estimated a priori when the number of cities is large). Bonomi and Lutton compared simulated annealing with traditional techniques of optimization

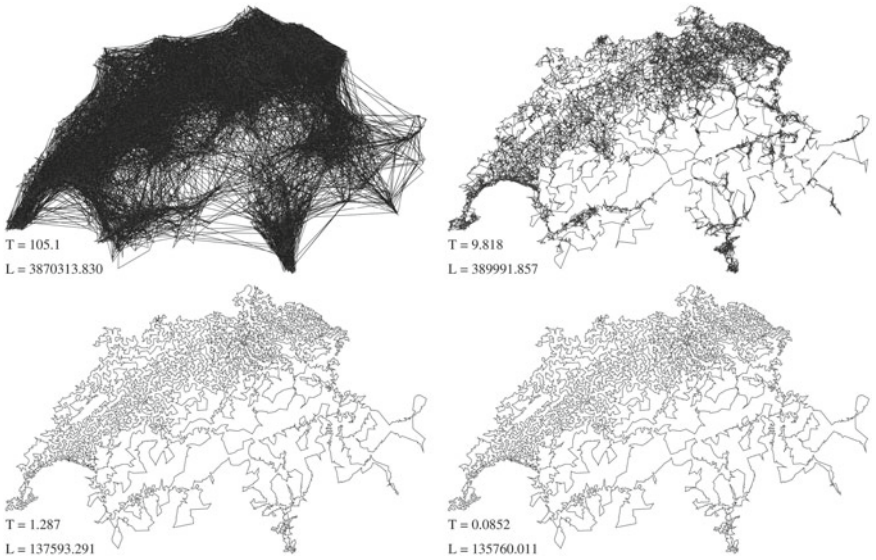


Fig. 2.2 The traveling salesman problem (13 206 cities): the better known configurations (length L) at the end of four temperature stages (T)

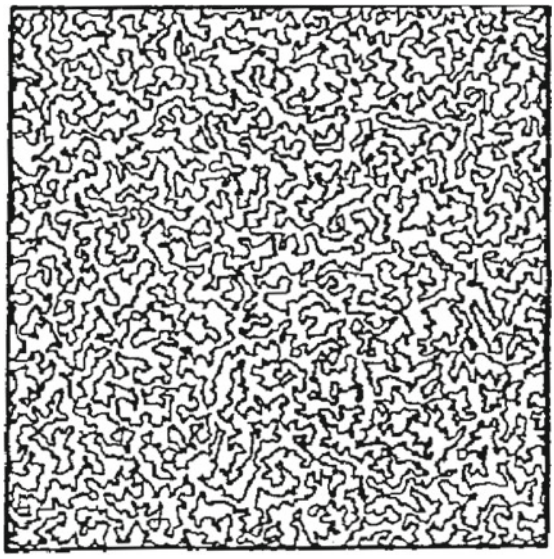


Fig. 2.3 The traveling salesman problem: solution, by simulated annealing for a case of 10 000 cities

for the traveling salesman problem: simulated annealing was slower for small-dimensional problems (N lower than 100) but, on the other hand, it was far more powerful for higher-dimensional problems (N higher than 800). The traveling salesman

problem has been extensively studied to illustrate and establish several experimental and theoretical developments in the simulated annealing method [42].

Many other benchmark problems of combinatorial optimization have also been solved using simulated annealing [29, 42]: in particular, the problems of the "partitioning of a graph," the "minimal coupling of points," and"quadratic assignment." Comparison with the best known algorithms leads to different results, varying according to the problems and the authors. Thus the studies by Johnson et al. [19–21], which were devoted to a systematic comparison of several benchmark problems, conclude that the only benchmark problem for which the results favor simulated annealing is that of the partitioning of a graph. For some problems, promising results were only obtained with the simulated annealing method for high-dimensional examples (a few hundreds of variables), and at the cost of a high computing time. Therefore, if simulated annealing has the merit to be adapted simply to a great diversity of problems, it cannot claim very much to supplement the specific algorithms that already exist for these problems.

We now present the applications of simulated annealing to practical problems. The first significant application of industrial interest was developed in the field of electronic circuit design; this industrial sector still remains the domain in which the greatest number of publications describing applications of simulated annealing have been produced. Two applications in the area of electronics are discussed in detail in the following two subsections. This is followed by discussions of other applications in some other fields.

2.5.2 Layout of Electronic Circuits

The first application of the simulated annealing method to practical problems was developed in the field of the layout and routing of electronic circuits [23, 41, 49]. Numerous studies have now been reported on this subject in several publications and, in particular, two books have been completely devoted to this problem [37, 50]. Detailed bibliographies, concerning the work carried out in the initial period from 1982 to 1988 can be found in the books [37, 42, 47, 50].

The search for an optimal layout is generally carried out in two stages. The first consists in calculating an initial placement quickly, by a constructive method: the components are placed one after another, in order of decreasing connectivity. Then an algorithm for iterative improvement is employed that gradually transforms, by elementary moves (e.g., exchange of components, and operations of rotation or symmetry), the initial layout configuration. The algorithms for iterative improvement of the layout differ according to the rule adopted for the succession of elementary moves. Simulated annealing can be used in this second stage.

Our interest was in a unit of 25 identical blocks to be placed on predetermined sites, which were the nodes of a planar square network. The list of connections was such that, in the optimal configurations, each block would be connected only to its closer neighbors (see Fig. 2.4a): an a priori knowledge of the global minima of the

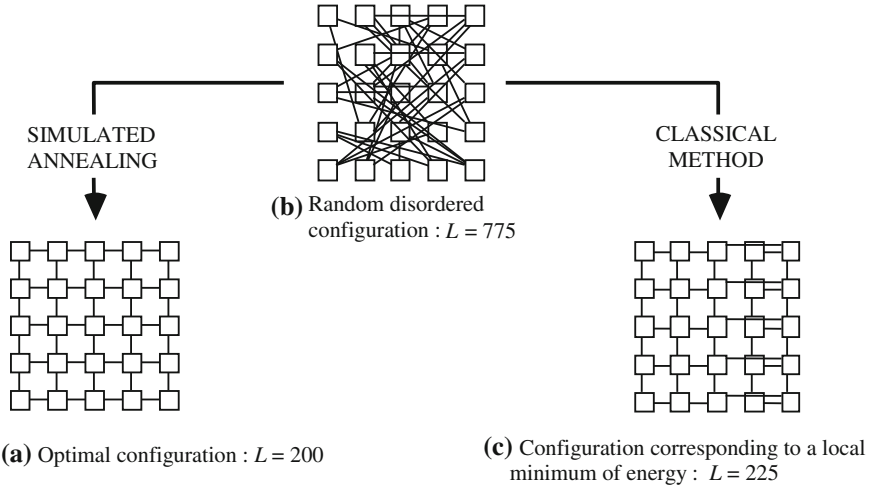


Fig. 2.4 The traditional method getting trapped in a local minimum of energy

problem then made it easier to study the influence of the principal parameters of the method on its convergence speed. The cost function was the overall Manhattan length (i.e., the length of L-type) of the connections. The only allowed elementary move was the permutation of two blocks. A detailed explanation for this benchmark problem on layout design—which is a form of "quadratic assignment" problem—can be found in [38, 43]. Here, the discussion will be limited to the presentation of two examples of applications. First of all, to appreciate the effectiveness of the method, we started with a completely disordered initial configuration (Fig. 2.4b), and an initial "elevated" temperature (in the sense that at this temperature 90 % of the moves are accepted). In this example, the temperature profile was that of a geometrical decrease, of ratio 0.9. A global optimum of the problem could be obtained after 12 000 moves, whereas the total number of possible configurations is about 10^{25}.

To illustrate the advantages of the simulated annealing technique, we applied the traditional method of iterative improvement (simulated annealing at zero temperature), with the same initial configuration (see Fig. 2.4b), and allowed the same number of permutations as during the preceding test. It was observed that the traditional method got trapped in a local minimum (Fig. 2.4c); it is clear that shifting from this configuration to the optimal configuration as shown in Fig. 2.4a would require several stages (at least five), the majority of which correspond to an increase in energy, which is inadmissible in the traditional method. This particular problem of placement made it possible to develop empirically a program of "adaptive" annealing, which could achieve a gain in computing time by a factor of 2; the lowering of the temperature was carried out according to the law $T_{k+1} = D_k \cdot T_k$, where:

$$D_k = \min\left(D_0, \frac{E_k}{\langle E_k \rangle}\right)$$

Here, $D_0 = 0.5$ to 0.9, E_k is the minimum energy of the configurations accepted during stage k, and $\langle E_k \rangle$ is the average energy of the configurations accepted during stage k. (At high temperature, $D_k = E_k / \langle E_k \rangle$ is small, and hence the temperature is lowered quickly; at low temperature, $D_k = D_0$, which corresponds to slow cooling).

Then we considered a more complex problem consisting of positioning components of different sizes, with the objective of simultaneous minimization of the length of the necessary connections and of the surface area of the circuit used. In this case, the translation of a block is a new means of iterative transformation of the layout. Here we can observe that the blocks can overlap with each other, which is allowed temporarily, but must generally be excluded from the final layout. This new constraint can be accommodated within the cost function of the problem by introducing a new factor called the "overlapping surface" between the blocks. Calculating this surface area can become very cumbersome when the circuit comprises many blocks. For this reason the circuit was divided into several planar areas, whose size was such that a block could overlap only with those blocks located in the same area or in a very close area. The lists of the blocks belonging to each area were updated after each move, using a chaining method. Moreover, to avoid leading to a circuit obstruction such as an impossible routing, a fictitious increase in the dimensions of each block was introduced. The calculation of the length of the connections consisted in determining, for each equipotential, the barycenter of the terminations, and then adding the distances of L-type of the barycenter with each termination. Lastly, the topology of the problem was adaptive, which can be described in the following manner: when the temperature decreases, the maximum amplitude of the translations decreases, and exchanges are considered more between neighboring blocks only.

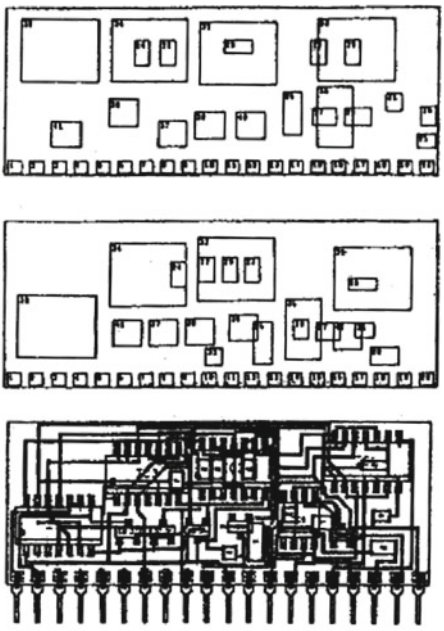
With the simulated annealing algorithm, it was possible to optimize industrial circuits, in particular some used in hybrid technology, in collaboration with the Thomson D.C.H. (Department of Hybrid Circuits) company. As an example, we present in Fig. 2.5, the result of the optimization of a circuit layout comprising 41 components and 27 equipotentials: the automated layout design procedure leads to a gain of 18 % in the connection lengths compared with the initial manual layout.

This study showed that the flexibility of the method enables it to take into account not only the rules of drawing, which translate the standards of technology, but also the rules of know-how, which are intended to facilitate routing. In particular, the rules of drawing impose a minimum distance between two components, whereas the rules of know-how recommend a larger distance, allowing the passage of connections. To balance these two types of constraints, the calculation of the area of overlap between the blocks, on a two-to-two basis, was undertaken according to the formula

$$S = S_{\mathrm{r}} + a \cdot S_{\mathrm{v}},$$

where S_{r} is the "real" overlapping area, S_{v} is the "virtual" overlapping area, and a is a weight factor (typically: 0.1).

These areas S_{r} and S_{v} were calculated by increasing the dimensions of the components fictitiously, with a larger increase in S_{v}. This induces some kind of an "intelligent" behavior, similar to that of an expert system. We notice from Fig. 2.5



- *Top: initial manual layout; length of connections: 9532,*
- *Middle: final layout, optimized by annealing; length of connections* 7861,
- *Bottom: manual routing using the optimized layout.*

Fig. 2.5 Optimization by simulated annealing of the design of an electronic circuit layout comprising 41 components

a characteristic of hybrid technology which was easily incorporated into the program: the resistances, shown by a conducting link, can be placed under the diodes or integrated circuits.

The observations noted by the majority of authors concerning the application of the simulated annealing technique to the layout design problem agree with our observations: the method is very simple to implement, it can be adapted easily to various evolving technological standards, and the final result is of good quality, but it is sometimes obtained at the cost of a significant computing time.

2.5.3 Search for an Equivalent Schema in Electronics

We now present an application which mixes the combinatorial and the continuous aspects: automatic identification of the "optimal" structure of a linear circuit pattern. The objective was to automatically determine a model which includes the least

possible number of elementary components, while ensuring a "faithful" reproduction of experimental data. This activity, in collaboration with the Institute of Fundamental Electronics (IEF, CNRS URA 22, Orsay), began with the integration, in a single software package, of a simulation program for linear circuits (implemented at the IEF) and a simulated annealing-based optimization program developed by us. We initially validated this tool by characterizing models of real components, with a complex structure, described using their distribution parameters S. Comparison with a commercial software package (developed using the gradient method) in use at the time of the IEF showed that simulated annealing was particularly useful if the orders of magnitude of the parameters of the model were completely unknown: obviously the models under consideration were of this nature, since even their structure was to be determined. We developed an alternative simulated annealing method, called logarithmic simulated annealing [9], which allows an effective exploration of the space of variations of the parameters when this space is very wide (more than 10 decades per parameter). Then the problem of structure optimization was approached by the examination—in the case of a passive circuit—of the progressive simplification of a general "exhaustive" model: we proposed a method which could be successfully employed to automate all the simplification stages [10]. This technique is based on the progressive elimination of the parameters according to their statistical behavior during the process of optimization by simulated annealing.

We present here, with the help of illustrations, an example of a search for an equivalent schema for a monolithic microwave integrated circuit (MMIC) inductance, in the frequency range from 100 MHz to 20 GHz. On the basis of an initial "exhaustive" model with 12 parameters, as shown in Fig. 2.6, and allowing each parameter to move over 16 decades, we obtained the equivalent schema shown in Figure 2.7 (the final values of the six remaining parameters are beyond the scope of our present interest: they are specified in [10]). The layouts in the Nyquist plane of the four S parameters of the quadrupole shown in Fig. 2.7 coincided nearly perfectly with the experimental results for the MMIC inductance, and this was true over the entire frequency range [10].

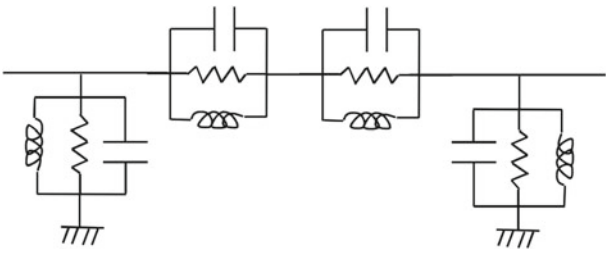


Fig. 2.6 Initial structure with 12 elements

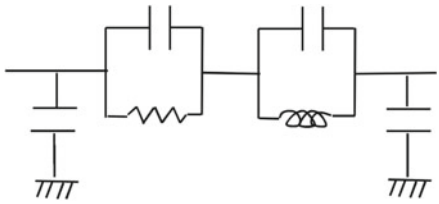


Fig. 2.7 Optimal structure with six elements

2.5.4 Practical Applications in Various Fields

An important field of application of simulated annealing is image processing: here the main problem is to restore images, mainly in three-dimensional forms, using a computer, starting from incomplete or irregular data. There are numerous practical applications in several domains, such as robotics, medicine (e.g., tomography), and geology (e.g., prospecting). The restoration of an image using an iterative method involves, under normal circumstances, the treatment of a large number of variables. Hence it calls for the development of a suitable method which can limit the computing time of the operation. Based on the local features of the information contained in an image, several authors have proposed numerous structures and algorithms specifically to address the problem of carrying out calculations in parallel. Empirically, it appears that the simulated annealing method should be particularly well suited for this task. A rigorous theoretical justification of this property can be obtained starting from the concepts of Markovian fields [14], which provide a convenient and coherent model of the local structure of the information in an image. This concept has been explained in detail in [42]. The "Bayesian approach" to the problem of optimal restoration of an image, starting from a scrambled image, consists in determining the image which has "the maximum a posteriori probability." It has been shown that this problem can ultimately be expressed as a well-known minimization problem of an objective function, comprising a very large number of parameters, for example the light intensities of all the "pixels" of an image in case of an image in black and white. Consequently, the problem can be considered as a typical problem for simulated annealing. The iterative application of this technique consists in updating the image by modifying the intensities of all of the pixels in turn, in a prespecified order. This procedure leads to a significant consumption of computing time: indeed, the number of complete sweeps of the image necessary to obtain a good restoration is typically about 300 to 1000. But, as the calculation of the energy variation is purely local in nature, several methods have been proposed to update the image by simultaneously treating a large number of pixels, using specialized elementary processors. The formalism of Markovian fields has made it possible to treat by simulated annealing several crucial tasks in the automated analysis of images: restoration of scrambled images, image segmentation, image identification. Apart from this formalism, other problems in the image-processing domain have also been solved by annealing: for example, the method has been utilized to determine the geological structure of the basement, starting from the results of seismic experiments.

To finish, we will mention some specific problems, in very diverse fields, where simulated annealing has been employed successfully: organization of the data-processing network for Loto (this required ten thousand playing machines to be connected to host computers), optimization of the collection of household garbage in Grenoble, timetabling problems (one problem was, for example, to perform the optimal planning of rest days in a hospital), and optimization of structures (in a project on constructing a 17-floor building for an insurance company, it was necessary to distribute the activities among the various parts so that the work output from 2000 employees could be maximized). Several applications of simulated annealing to scheduling problems can be found, in particular, in [5, 18, 27, 48]. The adequacy of the method for this type of problem has also been discussed. For example, Van Laarhoven et al. [48] showed that the computing time involved was unsatisfactory. Moreover, in [13], Fleury underlined several characteristics of scheduling problems which make them unsuitable for simulated annealing and he recommended a different stochastic method for this problem inspired by simulated annealing and tabu search: the "kangaroo method."

2.6 Advantages and Disadvantages of the Method

From the preceding discussion, the principal characteristics of the method can be established. Firstly, the *advantages*: it is observed that the simulated annealing technique generally achieves a good-quality solution (i.e., an absolute minimum or good relative minimum for the objective function). Moreover, it is a general method: it is applicable, to all problems which can potentially employ iterative optimization techniques, and it is easy to implement, under the condition that after each transformation the corresponding change in the objective function can be evaluated directly and quickly (often the computing time becomes excessive if complete re–computation of the objective function cannot be avoided after each transformation). Lastly, it offers great flexibility, as one easily can build new constraints into the program afterwards.

Now, let us discuss the disadvantages. Users are sometimes repelled by the involvement of a great many parameters (initial temperature, rate of decrease of the temperature, length of the temperature stages, termination criterion for the program). Although the standard values published for these parameters generally allow effective operation of the method, the essentially empirical nature of them can never guarantee suitability for a large variety of problems. The second defect of the method—which depends on the preceding one—is the computing time involved, which is excessive in certain applications.

In order to reduce this computing time, we still require an extensive research effort to determine the best values of the parameters of the method beyond the generalized results published so far [39], particularly the law of decrease of the temperature. Any progress in the effectiveness of the technique and in the computing time involved is likely to be obtained by continuing the analysis of the method in three specific directions: the utilization of interactive parameter setting, parallelization of

the algorithm, and the incorporation of statistical physics-based approaches to the analysis and study of disordered media.

2.7 Simple Practical Suggestions for Beginners

- *Definition of the objective function*: some constraints can be integrated into the objective function, whereas others constitute a limitation on the form of the disturbances for the problem.
- *Choice of disturbance mechanisms* for the " current configuration": the calculation of the corresponding variation ΔE of the objective function must be *direct* and rapid.
- *Initial temperature* T_0: this may be calculated in a preliminary step using the following algorithm:
  - initiate 100 disturbances at random; evaluate the average $\langle \Delta E \rangle$ of the corresponding variations ΔE;
  - choose an initial rate of acceptance τ_0 of the "degrading perturbations" according to the assumed "quality" of the initial configuration; for example:
    - "poor" quality: $\tau_0 = 50\,\%$ (starting at high temperature),
    - "good" quality: $\tau_0 = 20\,\%$ (starting at low temperature);
  - deduce T_0 from the relation: $e^{-\langle \Delta E \rangle / T_0} = \tau_0$.
- *Metropolis acceptance rule*: this can be utilized practically in the following manner: if $\Delta E > 0$, a number r in [0, 1] is drawn randomly, and the disturbance is accepted if $r < e^{-\Delta E/T}$, where T indicates the current temperature.
- *Change to next temperature stage*: this can take place as soon as one of the following two conditions is satisfied during a temperature stage:
  - $12 \cdot N$ perturbations accepted;
  - $100 \cdot N$ perturbations attempted, N indicates the number of degrees of freedom (or parameters) of the problem.
- *Decrease of the temperature*: this can be carried out according to the geometrical law $T_{k+1} = 0.9 \cdot T_k$.
- *Program termination*: this can be activated after three successive temperature stages without any acceptances.
- *Essential verifications during the first executions of the algorithm*:
  - the generation of the real random numbers (in [0, 1]) must be very *uniform*;
  - the "quality" of the result should not vary significantly when the algorithm is implemented *several times*:
    - with different "seeds" for the generation of the random numbers,
    - with different initial configurations;

  - for each initial configuration used, the result of simulated annealing should compare favorably, theoretically, with that of the *quenching* ("disconnected" Metropolis rule).

- *An alternative version of the algorithm in order to achieve less computation time*: simulated annealing is greedy and not very effective at low temperature; hence one might be interested in utilizing the simulated annealing technique, prematurely terminated, in cascade with an algorithm of local type for specific optimization of the problem, whose role is to "refine" the optimum.

2.8 Modeling of Simulated Annealing Through the Markov Chain Formalism

Let R be the complete space of all possible configurations of the system, and let $r \in R$ be a "state vector," whose components entirely define a specified configuration (or "state"). Let the set I_R consist of the numbers assigned to each configuration of R:

$$I_R = (1, 2, \ldots, |R|)$$

where $|R|$ is the cardinality of R. Finally, let us denote by $C(r_i)$ the value of the cost function (or "energy") in the state i, where r_i is the state vector for the state, and let $M_{ij}(T)$ be the probability of a transition from the state i to the state j at a "temperature" T. In the case of the simulated annealing algorithm, the succession of states forms a Markov chain, in the sense that the probability of transition from the state i to the state j depends only on these two states, and not on the states previous to i. In other words, all the past information about the system is summarized in the current state. When the temperature T is maintained constant, the probability of a transition $M_{ij}(T)$ is constant, and the corresponding Markov chain is known as *homogeneous*. The probability of a transition $M_{ij}(T)$ from the state i to the state j can be expressed in the following form:

$$M_{ij}(T) = \begin{cases} P_{ij} \cdot A_{ij}(T) & \text{if } i \neq j \\ 1 - \Sigma_{k \neq i} P_{ik} \cdot A_{ik}(T) & \text{if } i = j \end{cases}$$

where P_{ij} is the probability of perturbation, i.e., the probability of generating the state j when one is in the state i, and $A_{ij}(T)$ is the probability of acceptance, i.e., the probability of accepting the state j when one is in the state i at a temperature T.

The first factor, P_{ij}, can be calculated easily. In fact, the system is generally perturbed by randomly choosing a movement from the allowed elementary movements.

The results of this is that

$$P_{ij} = \begin{cases} |R_i|^{-1} & \text{if } j \in I_{R_i} \\ 0 & \text{if } j \notin I_{R_i} \end{cases}$$

where R_i denotes the subset of R comprising all the configurations which can be obtained in only one movement starting from the state i, and I_{R_i} denotes the set of the numbers of these configurations. As for the second factor, $A_{ij}(T)$, this is often defined by the Metropolis rule. Aarts and Van Laarhoven [1] noted that, more generally, the simulated annealing method makes it possible to impose the following five conditions:

1. The configuration space is *connected*, i.e. two unspecified states i and j correspond to a finite number of elementary movements.
2. $\forall i, j \in I_R : P_{ij} = P_{ji}$ (reversibility).
3. $A_{ij}(T) = 1$, if $\Delta C_{ij} = C(r_j) - C(r_i) \leq 0$ (the movements which result in a reduction in energy are systematically accepted).
4. If $\Delta C_{ij} > 0$ $\begin{cases} \lim\limits_{T \to \infty} A_{ij}(T) = 1 \\ \lim\limits_{T \to 0} A_{ij}(T) = 0 \end{cases}$
   (movements which result in an increase in energy are all accepted at infinite temperature, and all refused at zero temperature).
5. $\forall i, j, k \in I_r \mid C(r_k) \geq C(r_j) \geq C(r_i) : A_{ik}(T) = A_{ij}(T) \cdot A_{jk}(T)$.

2.8.1 Asymptotic Behavior of Homogeneous Markov Chains

By using the results obtained for homogeneous Markov chains, one can establish the following properties.

2.8.1.1 Property 1

Consider a Markov process generated by a mechanism of transition which observes the five conditions stated above. This mechanism is applied n times, at constant temperature, starting from a specified initial configuration, arbitrarily chosen. When n tends towards infinity, the Markov chain obtained has one and only one equilibrium vector, called $q(T)$, which is independent of the initial configuration. This vector, which consists of $|R|$ components, is called *distribution of static probability* of the Markov chain. Its ith component, i.e., $q_i(T)$, represents the probability that the system is in the configuration i when, after an infinity of transitions, the steady state is reached.

2.8.1.2 Property 2

$q_i(T)$ is expressed by the following relation:

$$q_i(T) = \frac{A_{i_0 i}(T)}{\sum_{i=1}^{|R|} A_{i_0 i}(T)},$$

where i_0 denotes the number of an optimal configuration.

2.8.1.3 Property 3

When the temperature tends towards infinity or zero, the limiting values of $q_i(T)$ are given by $\lim_{T \to \infty} q_i(T) = |R|^{-1}$ and

$$\lim_{T \to 0} q_i(T) = \begin{cases} |R_0|^{-1} & \text{if } i \in I_{R_0} \\ 0 & \text{if } i \notin I_{R_0} \end{cases}$$

where R_0 denotes the set of the optimal configurations, i.e.,

$$R_0 = \left\{ r_i \in R \mid C(r_i) = C\left(r_{i_0}\right) \right\}$$

Property 3 results immediately from property 2 when condition 4 is used. Its interpretation is the following: for larger values of the temperature, all configurations can be obtained with the same probability. On the other hand, when the temperature tends towards zero, the system reaches an optimal configuration with a probability equal to unity. In both cases, the result is obtained at the end of a Markov chain of infinite length.

Remark If one chooses the probability of acceptance $A_{ij}(T)$ recommended by Metropolis (see [1] for a justification for this choice regardless of any analogy with physics),

$$A_{ij}(T) = \begin{cases} e^{-\Delta C_{ij}/T} & \text{if } \Delta C_{ij} > 0 \\ 1 & \text{if } \Delta C_{ij} \leq 0 \end{cases}$$

one finds in property 2 the expression for the Boltzmann distribution.

2.8.2 Choice of Annealing Parameters

We saw in the preceding subsection that the convergence of the simulated annealing algorithm is assured when the temperature tends towards zero. A Markov chain of

infinite length undoubtedly ends in the optimal result if it is built at a sufficiently low (though nonzero) temperature. But this result is not of any practical utility because, in this case, the equilibrium is approached very slowly. The Markov chain formalism makes it possible to examine theoretically the convergence speed of the algorithm. One can show that this speed is improved when one starts from a high temperature and this temperature is then decreased in stages. This procedure requires the use of an annealing program, which defines the optimal values of the parameters of the descent in temperature. We will examine four principal parameters of the annealing program:

- the initial temperature;
- the length of the homogeneous Markov chains, i.e., the criterion for changing between temperature stages;
- the law of decrease of the temperature;
- the criterion for program termination.

For each of them, we will indicate first the corresponding result of the theory, which leads to an optimal result but often at the cost of a prohibitive computing time. Then we mention some values obtained by experiment.

2.8.2.1 Initial Temperature

There exists a necessary but not sufficient condition so that the optimization process does not get trapped in a local minimum. The initial temperature T_0 must be sufficiently high that, at the end of the first stage, all configurations can be obtained with the same probability. A suitable expression for T_0 which ensures a rate of acceptance close to 1 is the following:

$$T_0 = r \cdot \max_{ij} \Delta C_{ij}$$

with $r \gg 1$ (typically $r = 10$). In practice, in many combinatorial optimization problems, this rule is difficult to employ, because it is difficult to evaluate $\max_{ij} \Delta C_{ij}$ a priori. The choice of T_0 in this case has to be obtained from an experimental procedure, carried out before the process of optimization itself. During such a procedure, one calculates the evolution of the system during a limited time; one acquires some knowledge about the configuration space, from which one can determine T_0. This preliminary experiment can consist simply in calculating the average value of the variation in energy ΔC_{ij}, with the temperature maintained at zero. Aarts and Van Laarhoven [1] proposed a more sophisticated preliminary procedure: they established an iterative formula which makes it possible to adjust the value of T_0 after each perturbation so that the rate of acceptance is maintained constant. These authors indicated that this algorithm led to good results if the values of the cost function for the various system configurations were distributed in a sufficiently uniform way.

2.8.2.2 Length of the Markov Chains (or Length of the Temperature Stages); Law of Decrease of Temperature

The length of the Markov chain, which determines the length of the temperature stages, and the law of decrease of the temperature, which affects the number of stages, are two parameters of the annealing program that are very closely dependent on each other and which are most critical from the point of view of the computing time involved. An initial approach to the problem is to seek the optimal solution by fixing the length M of the Markov chains so as to reach quasi-equilibrium, i.e. to approach equilibrium to within a short distance ϵ that is fixed a priori and is characterized by the vector of the static probability distribution $q(T)$. One obtains the following condition:

$$M > K\left(|R|^2 - 3|R| + 3\right)$$

where K is a constant which depends on ϵ. In the majority of combinatorial optimization problems, the total number of configurations $|R|$ is an exponential function of the number N of variables of the system. Consequently, the above inequality leads to an exponential computing time, which has been confirmed by experimental observations in the case of a particular form of the traveling salesman problem (the cities considered occupy all the nodes of a plane square network, which makes it possible to easily calculate the exact value of the global optimum of the cost function: this a priori knowledge of the solution is very useful for analyzing the convergence of the algorithm). These experimental results also show that a considerable gain in CPU time is obtained if one is willing to deviate a little from the optimum. A deviation in the final result of only 2 % compared with the optimum makes it possible to decrease the exponential computing time to a cubic time in N.

This gave rise to the idea of performing the theoretical investigations again, seeking parameters of the annealing program that ensure a deviation from the true optimum, independently of the dimension of the problem considered. The starting postulate of the reasoning is as follows: for each homogeneous Markov chain generated during the process of optimization, the distribution of the states must be close to the static distribution (i.e., the Boltzmann distribution, if one adopts the Metropolis rule of acceptance). This situation can be implemented on the basis of a high temperature (for which one quickly reaches quasi-equilibrium, as indicated by property 3). Then it is necessary to choose the rate of decrease of the temperature such that the static distributions corresponding to two successive values of T are close together. In this way, after each change between temperature stages, the distribution of the states approaches the new static distribution quickly, so that the length of the successive chains can be kept small. Here one can see the strong interaction that exists between the length of the Markov chains and the rate of decrease of the temperature. Let T and T' be the temperatures of two unspecified successive stages and let α be the rate of decrease of the temperature $\left(T' = \alpha T < T\right)$. The condition to be satisfied can be written as

$$\left\|q(T) - q(T')\right\| < \epsilon$$

(ϵ is a positive small number).

This condition is equivalent to the following, which is easier to use:

$$\forall i \in I_R : \frac{1}{1+\delta} < \frac{q_i(T)}{q_i(T')} < 1+\delta$$

(δ is also a positive and small number, called the distance parameter). It can then be shown, with the help of some approximations, that the rate of decrease of the temperature can be written as

$$\alpha = \frac{1}{(1 + T \cdot \ln(1+\delta)/3 \cdot \sigma(T))} \tag{2.1}$$

where $\sigma(T)$ is the standard deviation of the values of the cost function for the states of the Markov chain at a temperature T.

Aarts and van Laarhoven recommend the following choice for the length of the Markov chains:

$$M = \max_{i \in I_R} |R_i| \tag{2.2}$$

where R_i is the subset of R comprising all the configurations that can be obtained in only one movement starting from the state i. The Markov chain formalism thus leads to an annealing program characterized by a constant length of the Markov chain and a variable rate of decrease of the temperature. This result, which is based on theory, differs from the usual empirical approach: in the latter case, one adopts a variable length of the temperature stages and a constant rate α of decrease of the temperature, typically ranging between 0.90 and 0.99. It is observed, however, that the parameter α is not very critical to achieving convergence of the algorithm, provided the temperature stages last long enough.

2.8.2.3 Program Termination Criterion

Quantitative information on the progress of the optimization process can be obtained from the *entropy*, which is a natural measurement of the order of the system. This is defined by the following expression:

$$S(T) = -\sum_{i=1}^{|R|} q_i(T) \cdot \ln(q_i(T))$$

It can be shown that $S(T)$ can be written in the following form:

$$S(T) = S(T_1) - \int_T^{T_1} \frac{\sigma^2(T')}{T'^3} dT'$$

and $\sigma^2(T)$ can easily be estimated numerically using the values of the cost function for the configurations obtained at the temperature T. A termination criterion can then be formulated starting from the following ratio, which measures the difference between the current configuration and the optimal configuration:

$$\frac{S(T) - S_0}{S_\infty - S_0}$$

where S_∞ and S_0 are defined by the relations

$$S_\infty = \lim_{T \to \infty} S(T) = \ln |R|$$
$$S_0 = \lim_{T \to 0} S(T) = \ln |R_0|$$

One can also detect a disorder–order transition (and consequently decide to slow down the cooling) by observing any steep increase in the following parameter, which is similar to the *specific heat*: $\sigma^2(T)/T^2$. If one wishes to perform precise numerical calculations, these criteria are applicable in practice only when the Markov chains are of sufficient length. If this is not the case, another termination criterion can be obtained starting from extrapolation to zero temperature of the smoothed average $C_l(T)$ of the values of the cost function obtained during the process of optimization:

$$\left| \frac{dC_l(T)}{dT} \cdot \frac{T}{C(T_0)} \right| < \epsilon_s \tag{2.3}$$

where ϵ_s is a positive small number, and $C(T_0)$ is the average value of the cost function at the initial temperature T_0.

Remark If one adopts the rate of decrease of the temperature and the termination criterion defined by the relations (2.1) and (2.3), respectively, Aarts and Van Laarhoven showed the existence of an upper limit, proportional to $\ln |R|$, for the total number of temperature stages. Moreover, if the length of the Markov chains is fixed in accordance with the relation (2.2), the execution time of the annealing algorithm is proportional to the following expression:

$$\max_{i \in I_R} |R_i| \cdot \ln |R|$$

But the term $\max |R_i|$ is generally a polynomial function of the number of variables of the problem. Consequently, an annealing program defined by the relations (2.1)–(2.3) allows one to solve the majority of the NP-hard problems while obtaining, in polynomial time, a result which varies by only a few percent from the global optimum, and this is true regardless of the dimension of the problem considered. The above theoretical considerations have been confirmed by the application of this annealing program to the traveling salesman and logical partitioning problems.

2.8.3 *Modeling of the Simulated Annealing Algorithm by Inhomogeneous Markov Chains*

The results which we have presented up to now are based on the assumption of a decrease of the temperature in stages (which ensures fast convergence of the simulated annealing algorithm, as we have already mentioned). This property makes it possible to represent the process of optimization in the form of a complete set of homogeneous Markov chains, whose asymptotic behavior can be described simply. We have seen that this results in a complete theoretical explanation of the operation of the algorithm, and the development of usable annealing program. Some authors have been interested in the convergence of the simulated annealing algorithm within the more general framework of the theory of inhomogeneous Markov chains. In this case, the study of the asymptotic behavior is more delicate: for example, Gidas [15] showed the possibility of the appearance of phenomena similar to phase transitions. We will be satisfied here with highlighting the main result of this work, of primarily theoretical interest: the annealing algorithm converges towards a global optimum with a probability equal to unity if, as the time t tends towards infinity, the temperature $T(t)$ does not decrease more quickly than the expression $C/\ln(t)$, where C denotes a constant that is related to the depth of the "energy well" of the problem.

2.9 Annotated Bibliography

Reference [42]	This book describes the principal theoretical approaches to simulated annealing and the applications of the method in the early years of its development (1982–1988), when the majority of the theoretical basis was established.
Reference [31]	The principal metaheuristics are described in great detail in this book. An elaborate presentation of simulated annealing is given in Chap. 3. Some applications are presented, in particular, the design of electronic circuits and the treatment of scheduling problems.
Reference [36]	In this book several metaheuristics are extensively described, including simulated annealing (in Chap. 3). The theoretical elements relating to the convergence of the method are clearly presented in detail. The book includes also a study of an application in an industrial context (that of the TimberWolf software package, in connection with the layout-routing problem). This is an invaluable resource for those undertaking academic study of the subject. Each chapter is supplemented with suitable exercises.

Reference [28]	The principal metaheuristics are also described in this book. Chapter 5 is completely devoted to simulated annealing and concludes with an application in the field of industrial production.
Reference [46]	This book is a collection of contributions from a dozen authors. Simulated annealing is not treated in detail, however.

References

1. Aarts, E.H.L., Van Laarhoven, P.J.M.: Statistical cooling: a general approach to combinatorial optimisation problems. Philips Journal of Research **40**, 193–226 (1985)
2. Aarts, E.H.L., De Bont, F.M.J., Habers J.H.A., Van Laarhoven, P.J.M.: A parallel statistical cooling algorithm. In: *Proceeding of the 3rd Annual Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science, vol. 210, pp. 87–97 (1986)
3. Azencott, R.: *Simulated Annealing: Parallelization Techniques*. Wiley-Interscience Series in Discrete Mathematics. John Wiley and Sons (1992)
4. Bonomi, E., Lutton, J.L.: The *N*-city travelling salesman problem, Statistical Mechanics and the Metropolis Algorithm. SIAM Review **26**(4), 551–568 (1984)
5. Brandimarte, P.: Neighbourhood search-based optimization algorithms for production scheduling: a survey. Computer-Integrated Manufacturing Systems **5**(2), 167–176 (1992)
6. Casotto, A., Romea, F., Sangiovanni-Vincentelli, A.: A parallel simulated annealing algorithm for the placement of macro-cells. IEEE Transactions on C.A.D. **CAD-6**(5), 838–847 (1987)
7. Cerny, V.: Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. Journal of Optimization Theory and Applications **45**(1), 41–51 (1985)
8. Collins, N.E., Eglese, R.W., Golden, B.: Simulated annealing—An annotated bibliography. American Journal of Mathematical and Management Sciences **8**, 209–307 (1988)
9. Courat, J., Raynaud, G., Mrad, I., Siarry, P.: Electronic component model minimisation based on Log Simulated Annealing. IEEE Transactions on Circuits and Systems I **41**(12), 790–795 (1994).
10. Courat, J., Raynaud, G., Siarry, P.: Extraction of the topology of equivalent circuits based on parameter statistical evolution driven by Simulated Annealing. International Journal of Electronics **79**, 47–52 (1995)
11. Delamarre, D., Virot, B.: Simulated annealing algorithm: technical improvements. Operations Research **32**(1), 43–73 (1998)
12. Durand, M., White, S.: Permissible error in parallel simulated annealing. Technical report, Institut de Recherche en Informatique et Systèmes aléatoires, Rennes (1991)
13. Fleury, G.: Application de méthodes stochastiques inspirées du recuit simulé à des problèmes d'ordonnancement. RAIRO A.P.I.I. (Automatique—Productique—Informatique industrielle) **29**(4–5), 445–470 (1995)
14. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-6**, 721–741 (1984)
15. Gidas, B.: Nonstationary Markov chains and convergence of the Annealing Algorithm. Journal of Statiscal Physics **39**, 73–131 (1985)
16. Hajek, B.: Cooling schedules for optimal annealing. Mathematics of Operations Research **13**, 311–329 (1988)
17. Hajek, B., Sasaki, G.: Simulated annealing—to cool or not. Systems and Control Letters **12**, 443–447 (1989)
18. Jeffcoat, D., Bulfin, R.: Simulated annealing for resource-constrained scheduling. European Journal of Operational Research **70**(1), 43–51 (1993)
19. Johnson, D., Aragon, C., McGeoch, L., Schevon, C.: Optimization by simulated annealing: an experimental evaluation—Part I (Graph partitioning). Operational Research **37**(6), 865–892 (1989)

20. Johnson, D., Aragon, C., McGeoch, L., Schevon, C.: Optimization by simulated annealing: an experimental evaluation—Part II (Graph coloring and number partitioning). Operational Research **39**(3), 378–406 (1991)
21. Johnson, D., Aragon, C., McGeoch, L., Schevon, C.: Optimization by simulated annealing: an experimental evaluation—Part III (The travelling salesman problem). Operational Research (1992)
22. Kirkpatrick, S., Toulouse, G.: Configuration space analysis of travelling salesman problems. Journal de Physique **46**, 1277–1292 (1985)
23. Kirkpatrick, S., Gelatt, C., Vecchi, M.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)
24. Kravitz, S., Rutenbar, R.: Placement by simulated annealing on a multiprocessor. IEEE Transactions on Computer Aided Design **CAD-6**, 534–549 (1987)
25. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E.: Equation of state calculations by fast computing machines. Journal of Chemistry Physics **21**, 1087–1090 (1953)
26. Miclo, L.: Évolution de l'énergie libre. Applications à l'étude de la convergence des algorithmes du recuit simulé. Ph.D. thesis, Université de Paris 6 (1991)
27. Musser, K., Dhingra, J., Blankenship, G.: Optimization based job shop scheduling. IEEE Transactions on Automatic Control **38**(5), 808–813 (1993)
28. Pham, D., Karaboga, D.: *Intelligent Optimisation Techniques*. Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks. Springer (2000)
29. Pirlot, M.: General local search heuristics in Combinatorial Optimization: a tutorial. Belgian Journal of Operations Research and Computer Science **32**(1–2), 7–67 (1992)
30. Rammal, R., Toulouse, G., Virasoro, M.: Ultrametricity for physicists. Reviews of Modern Physics **58**(3), 765–788 (1986)
31. Reeves, C.: *Modern Heuristic Techniques for Combinatorial Problems*. Advanced Topics in Computer Science Series. McGraw-Hill Ryerson (1995)
32. Rinnooy Kan, A., Timmer, G.: Stochastic global optimization methods—Part I: Clustering methods. Mathematical Programming **39**, 27–56 (1987)
33. Rinnooy Kan, A., Timmer, G.: Stochastic global optimization methods—part II: Multi level methods. Mathematical Programming **39**, 57–78 (1987)
34. Roussel-Ragot, P.: La méthode du recuit simulé: accélération et parallélisation. Ph.D. thesis, Université de Paris 6 (1990)
35. Roussel-Ragot, P., Siarry, P., Dreyfus, G.: La méthode du "recuit simulé" en électronique: principe et parallélisation. In: 2^e *Colloque National sur la Conception de Circuits à la Demande*, session G, Article G2, pp. 1–10. Grenoble (1986)
36. Saït, S., Youssef, H.: *Iterative computer Algorithms with Applications in Engineering*. IEEE Computer Society Press (1999)
37. Sechen, C.:*VLSI Placement and Global Routing Using Simulated Annealing*. Kluwer Academic ers (1988)
38. Siarry, P.: La méthode du recuit simulé: application à la conception de circuits électroniques. Ph.D. thesis, Université de Paris 6 (1986)
39. Siarry, P.: La méthode du recuit simulé en électronique. Adaptation et accélération. Comparaison avec d'autres méthodes d'optimisation. Application dans d'autres domaines. Habilitation à diriger les recherches en sciences physiques, Université de Paris-Sud (Orsay) (1994)
40. Siarry, P.: La méthode du recuit simulé: théorie et applications. RAIRO A.P.I.I. (Automatique—Productique—Informatique industrielle) **29**(4–5), 535–561 (1995)
41. Siarry, P., Dreyfus, G.: An application of physical methods to the computer aided design of electronic circuits. Journal de Physique Lettres **45**, L39–L48 (1984)
42. Siarry, P., Dreyfus, G.: La méthode du recuit simulé: théorie et applications. IDSET, ESPCI, Paris (1989)
43. Siarry, P., Bergonzi, L., Dreyfus, G.: Thermodynamic optimization of block placement. IEEE Transactions on Computer Aided Design **CAD-6**(2), 211–221 (1987)
44. Solla, S., Sorkin, G., White, S.: Configuration space analysis for optimization problems. In: E. Bienenstock (ed.) *Disordered Systems and Biological Organization*, pp. 283–292. Springer, New York (1986)

45. Sorkin, G.B.: Efficient simulated annealing on fractal energy landscapes. Algorithmica **6**, 367–418 (1991)
46. Teghem, J., Pirlot, M.: *Optimisation approchée en recherche opérationnelle: recherches locales, réseaux neuronaux et satisfaction de contraintes*. IC2: information, commande, communication. Hermès Science (2002)
47. Van Laarhoven, P., Aarts, E.: *Simulated annealing: theory and applications*. Reidel, Dordrecht (1987)
48. Van Laarhoven, P., Aarts, E., Lenstra, J.: Job-shop scheduling by simulated annealing. Operational Research **40**, 113–125 (1992)
49. Vecchi, M., Kirkpatrick, S.: Global wiring by simulated annealing. IEEE Transactions on C.A.D. **CAD-2**(4), 215–222 (1983)
50. Wong, D., Leong, H., Liu, C.: *Simulated Annealing for VLSI Design*. Kluwer Academic (1988)