Simulated Annealing

Chapter · September 2010

2 authors:

A.G. Nikolaev
University at Buffalo, State University of New York
62 PUBLICATIONS **1,366** CITATIONS

SEE PROFILE

Sheldon H Jacobson
University of Illinois Urbana-Champaign
271 PUBLICATIONS **7,562** CITATIONS

SEE PROFILE

Chapter 1
Simulated Annealing

Alexander G. Nikolaev and Sheldon H. Jacobson

Abstract Simulated annealing is a well-studied local search metaheuristic used to address discrete and, to a lesser extent, continuous optimization problems. The key feature of simulated annealing is that it provides a mechanism to escape local optima by allowing hill-climbing moves (i.e., moves which worsen the objective function value) in hopes of finding a global optimum. A brief history of simulated annealing is presented, including a review of its application to discrete, continuous, and multi-objective optimization problems. Asymptotic convergence and finite-time performance theory for simulated annealing are reviewed. Other local search algorithms are discussed in terms of their relationship to simulated annealing. The chapter also presents practical guidelines for the implementation of simulated annealing in terms of cooling schedules, neighborhood functions, and appropriate applications.

1.1 Background Survey

Simulated annealing is a local search algorithm (metaheuristic) capable of escaping from local optima. Its ease of implementation and convergence properties and its use of hill-climbing moves to escape local optima have made it a popular technique over the past two decades. It is typically used to address discrete and, to a lesser extent, continuous optimization problems. Survey articles that provide a good overview of simulated annealing's theoretical development and domains of application include [46, 55, 75, 90, 120, 144]. Aarts and Korst [1] and van Laarhoven

Alexander G. Nikolaev
Industrial and Systems Engineering, University at Buffalo, Buffalo, NY 14260-2050
e-mail: anikolae@buffalo.edu

Sheldon H. Jacobson
Department of Computer Science, University of Illinois, Urbana, IL, USA 61801-2302
e-mail: shj@illinois.edu

and Aarts [155] devote entire books to the subject. Aarts and Lenstra [2] dedicate a chapter to simulated annealing in their book on local search algorithms for discrete optimization problems.

1.1.1 History and Motivation

Simulated annealing is so named because of its analogy to the process of physical annealing with solids, in which a crystalline solid is heated and then allowed to cool very slowly until it achieves its most regular possible crystal lattice configuration (i.e., its minimum lattice energy state), and thus is free of crystal defects. If the cooling schedule is sufficiently slow, the final configuration results in a solid with such superior structural integrity. Simulated annealing establishes the connection between this type of thermodynamic behavior and the search for global minima for a discrete optimization problem. Furthermore, it provides an algorithmic means for exploiting such a connection.

At each iteration of a simulated annealing algorithm applied to a discrete optimization problem, the values for two solutions (the current solution and a newly selected solution) are compared. Improving solutions are always accepted, while a fraction of non-improving (inferior) solutions are accepted in the hope of escaping local optima in search of global optima. The probability of accepting non-improving solutions depends on a temperature parameter, which is typically non-increasing with each iteration of the algorithm.

The key algorithmic feature of simulated annealing is that it provides a means to escape local optima by allowing *hill-climbing* moves (i.e., moves which worsen the objective function value). As the temperature parameter is decreased to zero, hill-climbing moves occur less frequently, and the solution distribution associated with the inhomogeneous Markov chain that models the behavior of the algorithm converges to a form in which all the probability is concentrated on the set of globally optimal solutions (provided that the algorithm is convergent; otherwise the algorithm will converge to a local optimum, which may or may not be globally optimal).

1.1.2 Definition of Terms

To describe the specific features of a simulated annealing algorithm for discrete optimization problems, several definitions are needed. Let Ω be the *solution space* (i.e., the set of all possible solutions). Let $f : \Omega \to \Re$ be an *objective function* defined on the solution space. The goal is to find a global minimum, ω^* (i.e., $\omega^* \in \Omega$ such that $f(\omega^*) \leq (\omega)$ for all $\omega \in \Omega$). The objective function must be bounded to ensure that ω^* exists. Define $N(\omega)$ to be the *neighborhood function* for $\omega \in \Omega$. Therefore, associated with every solution, $\omega \in \Omega$, are neighboring solutions, $N(\omega)$, that can be reached in a single iteration of a local search algorithm.

Simulated annealing starts with an initial solution $\omega \in \Omega$. A neighboring solution $\omega' \in N(\omega)$ is then generated (either randomly or using some pre-specified rule). Simulated annealing is based on the Metropolis acceptance criterion [101], which models how a thermodynamic system moves from the current solution (state) $\omega \in \Omega$ to a candidate solution $\omega' \in N(\omega)$, in which the energy content is being minimized. The candidate solution, ω', is accepted as the current solution based on the acceptance probability

$$P\{\text{Accept } \omega' \text{ as next solution}\} = \begin{cases} \exp[-(f(\omega') - f(\omega))/t_k] & \text{if } f(\omega') - f(\omega) > 0 \\ 1 & \text{if } f(\omega') - f(\omega) \le 0. \end{cases}$$
(1.1)

Define t_k as the temperature parameter at (outer loop) iteration k, such that

$$t_k > 0 \text{ for all } k \text{ and } \lim_{k \to \infty} t_k = 0.$$
(1.2)

This acceptance probability is the basic element of the search mechanism in simulated annealing. If the temperature is reduced sufficiently slowly, then the system can reach an equilibrium (steady state) at each iteration k. Let $f(\omega)$ and $f(\omega')$ denote the energies (objective function values) associated with solutions $\omega \in \Omega$ and $\omega' \in N(\omega)$, respectively. This equilibrium follows the Boltzmann distribution, which can be described as the probability of the system being in state $\omega \in \Omega$ with energy $f(\omega)$ at temperature T such that

$$P\{\text{System is in state } \omega \text{ at temperature } T\} = \frac{\exp(-f(\omega)/t_k)}{\sum_{\omega'' \in \Omega} \exp(-f(\omega'')/t_k)}.$$
(1.3)

If the probability of generating a candidate solution ω' from the neighbors of solution $\omega \in \Omega$ is $g_k(\omega, \omega')$, where

$$\sum_{\omega' \in N(\omega)} g_k(\omega, \omega') = 1, \text{ for all } \omega \in \Omega, \ k = 1, 2, ...,$$
(1.4)

then a non-negative square stochastic matrix $\mathbf{P}_k$ can be defined with transition probabilities

$$P_k(\omega, \omega') = \begin{cases} g_k(\omega, \omega') \exp(-\Delta_{\omega, \omega'}/t_k) & \omega' \in N(\omega), \quad \omega' \ne \omega \\ 0 & \omega' \notin N(\omega), \quad \omega' \ne \omega \\ 1 - \sum_{\omega'' \in N(\omega), \omega'' \ne \omega} \mathbf{P}_k(\omega, \omega'') & \omega' = \omega \end{cases}$$
(1.5)

for all solutions $\omega \in \Omega$ and all iterations $k = 1, 2, ...$ and with $\Delta_{\omega, \omega'} \equiv f(\omega') - f(\omega)$. These transition probabilities define a sequence of solutions generated from an inhomogeneous Markov chain [120]. Note that boldface type indicates matrix/vector notation, and all vectors are row vectors.

1.1.3 Statement of Algorithm

Simulated annealing is outlined in pseudo-code (see [46]).

Select an initial solution $\omega \in \Omega$
Select the temperature change counter $k = 0$
Select a temperature cooling schedule, t_k
Select an initial temperature $T = t_0 \geq 0$
Select a repetition schedule, M_k, that defines the number of iterations executed at each temperature, t_k
Repeat
    Set repetition counter $m = 0$
    Repeat
        Generate a solution $\omega' \in N(\omega)$
        Calculate $\Delta_{\omega,\omega'} = f(\omega') - f(\omega)$
        If $\Delta_{\omega,\omega'} \leq 0$, then $\omega \leftarrow \omega'$
        If $\Delta_{\omega,\omega'} > 0$, then $\omega \leftarrow \omega'$ with probability $exp(-\Delta_{\omega,\omega'}/t_k)$
        $m \leftarrow m+1$
    Until $m = M_k$
    $k \leftarrow k+1$
Until stopping criterion is met

This simulated annealing formulation results in $M_0 + M_1 + \cdots + M_k$ total iterations being executed, where k corresponds to the value for t_k at which some stopping criterion is met (for example, a pre-specified total number of iterations has been executed or a solution of a certain quality has been found). In addition, if $M_k = 1$ for all k, then the temperature changes at each iteration.

1.1.4 Discrete Versus Continuous Problems

The majority of the theoretical developments and application work with simulated annealing has been for discrete optimization problems. However, simulated annealing has also been used as a tool to address problems in the continuous domain. There is considerable interest in using simulated annealing for global optimization over regions containing several local and global minima (due an inherent non-linearity of objective functions). Fabian [48] studies the performance of simulated annealing methods for finding a global minimum of a given objective function. Bohachevsky et al. [15] propose a generalized simulated annealing algorithm for function optimization for use in statistical applications, and Locatelli [96] presents a proof of convergence for the algorithm. Optimization of continuous functions involves finding a candidate solution by picking a direction from the current (incumbent) solution and a step size to take in this direction and evaluating the function at the new (candidate) location. If the function value of this candidate location is an improvement over the function value of the incumbent location, then the candidate

becomes the incumbent. This migration through local minima in search of a global minimum continues until the global minimum is found or some termination criteria are reached. Belisle [12] presents a special simulated annealing algorithm for global optimization, which uses a heuristically motivated cooling schedule. This algorithm is easy to implement and provides a reasonable alternative to existing methods.

Belisle et al. [13] discuss convergence properties of simulated annealing algorithms applied to continuous functions and apply these results to hit-and-run algorithms used in global optimization. The presented convergence properties are consistent with those presented in [72] and provide a good contrast between convergence in probability and (the stronger) almost sure convergence. This work is further extended in [166] to an improved hit-and-run algorithm used for global optimization.

Fleischer and Jacobson [57] propose cybernetic optimization by simulated annealing as a method of parallel processing that accelerates the convergence of simulated annealing to the global optima. This theory is extended by Fleischer [56] into the continuous domain by applying probabilistic feedback control to the generation of candidate solutions. The probabilistic feedback control method of generating candidate solutions effectively accelerates convergence to a global optimum using parallel simulated annealing on continuous variable problems.

Locatelli [94] presents convergence properties for a class of simulated annealing algorithms for continuous global optimization by removing the restriction that the next candidate point must be generated according to a probability distribution whose support is the whole feasible set. A study on simulated annealing algorithms for globally minimizing functions of multiple continuous variables is conducted by Siarry et al. [131]. The study focuses on how high dimensionality can be addressed using variables discretization and addresses the design and implementation issues for several complementary stopping criteria. Convergence results and criteria for simulated annealing applied to continuous global optimization problems are also provided in [163] and [95]. A general-purpose simulated annealing algorithm that solves mixed integer linear programs is introduced by Kiatsupaibul and Smith [88]. The simulated annealing algorithm is constructed using a Markov chain sampling algorithm to generate uniformly distributed points on an arbitrary bounded region of a high-dimensional integer lattice. They show that the algorithm converges in probability to a global optimum. Romeijn et al. [119] study a simulated annealing algorithm that uses a reflection generator for mixed integer/continuous global optimization problems. Locatelli [96] establishes the convergence of simulated annealing algorithms for continuous global optimization and an upper bound for the expected first hitting time, i.e., the expected number of iterations before reaching the global optimum value within accuracy ε.

1.1.5 Single-objective Versus Multi-objective Problems

Originally used as an optimization tool for combinatorial optimization problems, simulated annealing has recently been adapted to address multi-objective problems

(see [144]). Its framework is easy to implement and simulated annealing-based algorithms are capable of producing a Pareto set of solutions in a single run with very little computational cost. Additionally, its performance is not influenced by the shape of the Pareto set, which is a concern for mathematical programming techniques.

The first multi-objective version of simulated annealing has been proposed by Serafini [128, 129]. The method closely follows the guidelines of regular single-objective simulated annealing and uses a modification of the solution acceptance criteria in the original algorithm. Various alternative criteria have been investigated, with the objective to increase the probability of accepting non-dominated solutions. A special selection rule produced by the combination of several criteria has been proposed in order to concentrate the search almost exclusively on the non-dominated solutions. This idea has also been used by Ulungu and Teghem [152] and Serafini [130], with the latter utilizing a target-vector approach to solve a bi-objective optimization problem. Ulungu et al. [154] propose a complete MOSA algorithm, where a weighted aggregating function is used to evaluate and compare the obtained solutions. The MOSA algorithm works with only one current solution but keeps a record of the population of non-dominated solutions found during the search. A further improved, interactive version of MOSA is presented in [153] and is referred to as the UMOSA method. Suppapitnerm and Parks [145] propose a different simulated annealing-based approach proposed to tackle multi-objective problems (SMOSA method). At each iteration, the algorithm does the search based on one solution and the annealing process adjusts the temperature adaptively, using the objective function value of the obtained solution in each of the multiple objectives. An archive is created to store all the non-dominated solutions.

The idea of introducing the "new-acceptance" probability formulation based on an annealing schedule with multiple temperatures (one for each objective) has also been proposed. The acceptance probability of a "new solution" depends on whether or not it is added to the set of potentially Pareto-optimal solutions. If it is added to the Pareto set, then it is accepted as the current solution with probability equal to 1. Otherwise, a multi-objective acceptance rule is used. Czyzak et al. [36] and Czyzak and Jaszkiewicz [37] propose another way to adapt simulated annealing to a multi-objective context, which combines the ideas of unicriterion simulated annealing and genetic algorithms to provide efficient solutions for a multi-criteria shortest path problem. A classical neighborhood search has been used to explore the population of solutions, with the weight for each objective function adjusted in each iteration. This technique increases the probability of escaping local optima, in a way similar to multi-objective tabu search.

Suman [141–143] proposes different simulated annealing-based approaches to tackle constrained multi-objective optimization problems. In [142], a comparative analysis of five simulated annealing algorithms is conducted for the system reliability optimization problem. The goal of these methods is to generate a set of solutions that are a good approximation to the entire set of efficient (non-dominated or Pareto-optimal) solutions over a fixed time period.

Villalobos-Arias et al. [159, 160] study asymptotic convergence of simulated annealing algorithms for multi-objective optimization problems in comparison with other algorithms such as an Artificial Immune System and a General Evolutionary Algorithm. Tekinalp and Karsli [146] present a simulated annealing algorithm for continuous multi-objective optimization that has an adaptive cooling schedule and uses a population of fitness functions to accurately generate the Pareto front. Whenever an improvement with a fitness function is encountered, the trial point is accepted and the temperature parameters associated with the improving fitness functions are cooled down. In addition to well-known linear fitness functions, special elliptic and ellipsoidal fitness functions, suitable for the generation on non-convex fronts, are used.

1.2 Convergence Results

1.2.1 Asymptotic Performance

Asymptotic convergence results for simulated annealing have typically taken one of two directions: the algorithm has been modeled either as a sequence of homogeneous Markov chains or as a single inhomogeneous Markov chain.

1.2.1.1 Homogeneous Markov Chain Approach

The homogeneous Markov chain approach [3, 49, 70, 84, 85, 97, 104, 123] assumes that each temperature t_k is held constant for a sufficient number of iterations m such that the stochastic matrix $\mathbf{P}_k$ can reach its stationary (steady-state) distribution π_k. Note that in the interest of simplifying notation, the inner loop index m is suppressed. However, the index k should be interpreted as the double index k, m, where a sequence of $m = 1, 2, ..., M_k$ simulated annealing iterations occur for each fixed k. The existence of a stationary distribution at each iteration k follows from Theorem 1. (Note: To ensure that Theorem 1 is consistent with the simulated annealing algorithm depicted in Section 1.1.3, without loss of generality, let t_k be a function only of each outer loop iteration k, and let the respective number of inner loop iterations M_k and outer loop iterations k each approach infinity).

Theorem 1 *Let $\mathbf{P}_k(\omega, \omega')$ be the probability of moving from solution ω to solution ω' in one inner iteration at outer loop k, and let $\mathbf{P}_k^{(m)}(\omega, \omega')$ be the probability of going from solution ω to solution ω' in m inner loops. If the Markov chain associated with $\mathbf{P}_k^{(m)}(\omega, \omega')$ is irreducible and aperiodic with finitely many solutions, then $\lim_{m \to \infty} \mathbf{P}_k^{(m)}(\omega, \omega') = \pi_k(\omega')$ exists for all $\omega, \omega' \in \Omega$ and iterations k. Furthermore, $\pi_k(\omega')$ is the unique strictly positive solution of*

$$\pi_k(\omega') = \sum_{\omega \in \Omega} \pi_k(\omega)\mathbf{P}_k(\omega, \omega'), \text{ for all } \omega' \in \Omega, \tag{1.6}$$

and

$$\sum_{\omega \in \Omega} \pi_k(\omega) = 1. \tag{1.7}$$

Proof See [33].

The key requirements for the existence of the stationary distributions and for the convergence of the sequence of π_k vectors include the following:

1. transition matrix irreducibility (for every finite outer loop k, the transition matrix can assign a path of non-zero probability between any two solutions $\omega, \omega' \in \Omega$),
2. aperiodicity (starting at solution $\omega' \in \Omega$), it is possible to return to ω, ω' with period 1; see [78],
3. A non-zero stationary probability distribution, as the number of outer loops k approaches infinity.

Note that all simulated annealing proofs of convergence in the literature based on homogeneous Markov chain theory, either explicitly or implicitly, use the sufficient condition of reversibility (also called detailed balance; see [122]) defined as

$$\pi_k(\omega)\mathbf{P}_k(\omega, \omega') = \pi_k(\omega')\mathbf{P}_k(\omega', \omega), \text{ for all } \omega, \omega' \in \Omega, \text{ and all iterations } k. \tag{1.8}$$

Reversibility is a sufficient condition for a unique solution to exist for Equations (1.6) and (1.7) at each outer loop iteration k. A necessary condition for reversibility is multiplicativity. That is, for any three solutions $\omega, \omega', \omega'' \in \Omega$ such that $f(\omega) \leq f(\omega') \leq f(\omega'')$ and for all iterations k,

$$\kappa_k(\Delta_{\omega,\omega''}) = \kappa_k(\Delta_{\omega,\omega'})\kappa_k(\Delta_{\omega',\omega''}), \tag{1.9}$$

where $\kappa_k(\Delta_{\omega,\omega'})$ is the probability of accepting the transition from solution ω to solution ω' at outer loop iteration k. Reversibility is enforced by assuming conditions of symmetry on the solution generation probabilities g_k and either by directly expressing the acceptance probability using an exponential form or by requiring the multiplicative condition in Equation (1.9).

The homogeneous Markov chain proofs of convergence in the literature (implicitly or explicitly) require the condition in Equation (1.9) to hold for the acceptance function and then address the sufficient conditions on the solution generation matrix $\mathbf{P}_k$. For example, the original homogeneous proofs of convergence [3, 97] require the multiplicative condition for the acceptance function, and then assume that the solution generation matrix is symmetric and constant for all outer loop iterations k. Rossier et al. [123] partition the solution space into blocks composed of neighboring solutions of equal objective function value and then require that only the solution generation probabilities be symmetric between these blocks. Rossier et al. then express the acceptance function as a ratio of the stationary distribution probabilities. Faigle and Schrader [51] and Faigle and Kern [49] use a graph theoretic approach to relax the solution generation matrix symmetry condition.

However, they require that the solution acceptance probability function satisfies Equation (1.9).

Granville et al. [70] propose a simulated annealing procedure for filtering binary images, where the acceptance function is based on the probability of the current solution, instead of the change in objective function value. The probability function that Granville et al. [70] present for accepting a candidate solution at (outer loop) iteration k is based on the ratio of the stationary probability of the incumbent solution from iteration $k-1$ versus the stationary probability of an initial solution (which is based on a maximum likelihood estimate). The acceptance probability is

$$\xi_k = q\pi_0(\omega)/\pi_{k-1}^{\phi(k)}(\omega') \tag{1.10}$$

where $q = \inf_{\omega \in \Omega} \pi(\omega)/\sup_{\omega' \in \Omega} \pi(\omega')$ (q must also be estimated), and $\phi(k)$ is a slowly increasing function. Therefore, the probability of a solution transition does not consider the objective function value of the candidate solution. Granville et al. [70] provide a proof of asymptotic convergence of this approach, but note that the proof methodology does not show that the set of globally optimal solutions are asymptotically uniformly distributed.

Simulated annealing and the homogeneous convergence theory are based on the work of Metropolis et al. [101], which addresses problems in equilibrium statistical mechanics [74]. To see this relationship, consider a system in thermal equilibrium with its surroundings, in solution (state) S with energy $F(S)$. The probability density in phase space of the point representing S is proportional to

$$\exp(-F(S)/bT), \tag{1.11}$$

where b is the Boltzmann constant, and T is the absolute temperature of the surroundings. Therefore the proportion of time that the system spends in solution S is proportional to Equation (1.11) (see [74]), hence the equilibrium probability density for all $S \in \Omega$ is

$$\pi_s = \frac{\exp(-F(S)/bT)}{\int \exp(-F(S)/bT)\mathrm{d}S}. \tag{1.12}$$

The expectation of any valid solution function $f(S)$ is thus

$$E[f] = \frac{\int f(S)\exp(-F(S)/bT)\mathrm{d}S}{\int \exp(-F(S)/bT)\mathrm{d}S}. \tag{1.13}$$

Unfortunately, for many solution functions, Equation (1.13) cannot be evaluated analytically. Hammersley and Handscomb [74] note that one could theoretically use naive Monte Carlo techniques to estimate the value of the two integrals in Equation (1.13). However, this often fails in practice since the exponential factor means that a significant portion of the integrals is concentrated in a very small region of the solution space Ω. This problem can be overcome using importance sampling (see [18], Chapter 2), by generating solutions with probability density Equation (1.12). This approach would also seem to fail, because of the integral in

the denominator of Equation (1.12). However, Metropolis et al. [101] solve this problem by first discretizing the solution space, such that the integrals in Equations (1.12) and (1.13) are replaced by summations over the set of discrete solutions $\omega' \in \Omega$, and then by constructing an irreducible, aperiodic Markov chain with transition probabilities $\mathbf{P}(\omega, \omega')$ such that

$$\pi(\omega') = \sum_{\omega \in \Omega} \pi(\omega)\mathbf{P}(\omega, \omega') \ \text{ for all } \omega' \in \Omega, \tag{1.14}$$

where

$$\pi(\omega') = \frac{\exp(-F(\omega')/bT)}{\sum_{\omega \in \Omega} \exp(-F(\omega)/bT)\mathrm{d}S} \ \text{ for all } \omega' \in \Omega. \tag{1.15}$$

Note that to compute the equilibrium distribution π, the denominator of Equation (1.13) (a normalizing constant) does not need to be calculated. Instead, the ratios $\pi(\omega')/\pi(\omega)$ need only be computed and a transition matrix $\mathbf{P}$ defined that satisfies Equation (1.14). Hammersley and Handscomb [74] show that Metropolis et al. [101] accomplish this by defining $\mathbf{P}$ as the product of symmetric solution generation probabilities $g(\omega, \omega')$, and the equilibrium ratios $\pi(\omega')/\pi(\omega)$,

$$\mathbf{P}(\omega, \omega') = \begin{cases} g(\omega, \omega')\pi(\omega')/\pi(\omega) & \text{if } \pi(\omega')/\pi(\omega) < 1, \quad \omega' \neq \omega \\ g(\omega, \omega') & \text{if } \pi(\omega')/\pi(\omega) \geq 1, \quad \omega' \neq \omega \\ g(\omega, \omega') + \Delta & \text{if } \omega' = \omega \end{cases} \tag{1.16}$$

with $\Delta = \sum_{\omega'' \in \Omega, \pi(\omega'') < \pi(\omega)} g(\omega, \omega'')(1 - (\pi(\omega'')/\pi(\omega)))$, where

$$g(\omega, \omega') \geq 0, \ \sum_{\omega' \in \Omega} g(\omega, \omega') = 1, \text{ and } g(\omega, \omega') = g(\omega', \omega) \qquad \text{for all } \omega, \omega' \in \Omega. \tag{1.17}$$

The use of stationary probability ratios to define the solution acceptance probabilities, combined with symmetric solution generation probabilities, enables Metropolis et al. [101] to use the reversibility condition in Equation (1.8) to show that Equations (1.16) and (1.17) satisfy Equation (1.14).

Homogeneous proofs of convergence for simulated annealing become more difficult to establish when the reversibility condition is not satisfied. Note that the existence of a unique stationary distribution for each outer loop iteration k is easily shown by specifying that each transition matrix $\mathbf{P}_k$ be irreducible and aperiodic. On the other hand, it becomes very difficult to derive an explicit closed-form expression for each stationary distribution π_k that remains analytically tractable as the problem's solution space becomes large. One can no longer use Equation (1.8) to describe each stationary distribution, since, in general, the multiplicative condition is not met. Instead, one must directly solve the system of equations formed with Equations (1.6) and (1.7). For example, in [38], Davis attempts to obtain a closed-form expression for π_k by using Cramer's rule and rewriting Equation (1.6) and (1.7) as

$$\pi_k(\mathbf{I} - \mathbf{P}_k) = 0 \tag{1.18}$$

and

$$\pi_k \mathbf{e}^T = 1, \tag{1.19}$$

respectively, where boldface type indicates vector/matrix notation, $\mathbf{I}$ is the identity matrix, and $\mathbf{e}^T$ is a column vector of ones. Note that the $card(\Omega) \times card(\Omega)$ transition matrix $\mathbf{P}_k$ associated with Equation (1.18) is of rank $card(\Omega) - 1$ [33]. Therefore, by deleting any one equation from Equation (1.18) and substituting Equation (1.19), the result is the set of $card(\Omega)$ linearly independent equations

$$\pi_k(\mathbf{I} - \mathbf{P}_k)^{[i]} = \mathbf{e}_i, \tag{1.20}$$

where the square matrix $(\mathbf{I} - \mathbf{P}_k)^{[i]}$ is obtained by substituting the ith column of matrix $(\mathbf{I} - \mathbf{P}_k)$ with a column vector of ones. The vector $\mathbf{e}_i$ is a row vector of zeroes, except for a one in the ith position. Since $(\mathbf{I} - \mathbf{P}_k)^{[i]}$ is of full rank, then its determinant (written as $\det((\mathbf{I} - \mathbf{P}_k)^{[i]})$) is non-zero. Define $(\mathbf{I} - \mathbf{P}_k)^\omega$ to be the same matrix as $(\mathbf{I} - \mathbf{P}_k)$ except that the elements of the ωth row of $(\mathbf{I} - \mathbf{P}_k)$ are replaced by the vector $\mathbf{e}_\omega$. Therefore, for all iterations k,

$$\pi_k(\omega) = \frac{\det((\mathbf{I} - \mathbf{P}_k)^{[i]^\omega})}{\det((\mathbf{I} - \mathbf{P}_k)^{[i]})}, \text{ for all } \omega \in \Omega. \tag{1.21}$$

In [38], an attempt is made to solve Equation (1.21) for each $\omega \in \Omega$ via a multivariate Taylor series expansion of each determinant, but the method failed to produce a closed-form analytical expression.

Overall, the difficulty of explicitly expressing the stationary distributions for large solution spaces, combined with bounding the transition matrix condition number for large k, suggests that it is very difficult to prove asymptotic convergence of the simulated annealing algorithm by treating Equations (1.5) and (1.6) as a linear algebra problem.

Lundy and Mees [97] note that for each fixed outer loop iteration k, convergence to the solution equilibrium probability distribution vector π_k (in terms of the Euclidean distance between $P_k^{(m)}$ and π_k, as $m \to +\infty$) is geometric since the solution space is finite, and the convergence factor is given by the second largest eigenvalue of the transition matrix $\mathbf{P}_k$. This result is based on a standard convergence theorem for irreducible, aperiodic homogeneous Markov chains (see [33]). Note that a large solution space precludes practical calculation of this eigenvalue. Lundy and Mees [97] conjecture that when the temperature t_k is near zero, the second largest eigenvalue will be close to one for problems with local optima, and thus convergence to the equilibrium distribution will be very slow (recall that the dominant eigenvalue for $\mathbf{P}_k$ is 1, with algebraic multiplicity 1 [78]). Lundy and Mees [97] use the conjecture to justify why simulated annealing should be initiated with a relatively high temperature. For an overview of current methods for assessing non-asymptotic rates of convergence for general homogeneous Markov chains, see [121].

The assumption of stationarity for each outer loop iteration k limits practical application of homogeneous Markov chain theory. Romeo and Sangiovanni-Vincentelli [120] show that if equilibrium (for a Markov chain that satisfies the

reversibility condition) is reached in a finite number of steps, then it can be achieved in one step. Thus, Romeo and Sangiovanni-Vincentelli [120] conjecture that there is essentially no hope for the most used versions of simulated annealing to reach equilibrium in a finite number of iterations.

1.2.1.2 Inhomogeneous Markov Chain Approach

The second convergence approach for simulated annealing is based on inhomogeneous Markov chain theory [10, 65, 104]. In this approach, the Markov chain need not reach a stationary distribution (e.g., the simulated annealing inner loop need not be infinitely long) for each outer loop k. On the other hand, an infinite sequence of (outer loop) iterations k must still be examined, with the condition that the temperature parameter t_k cool sufficiently slowly. The proof given by Mitra et al. [104] is based on satisfying the inhomogeneous Markov chain conditions of weak and strong ergodicity [78, 127]. The proof requires four conditions:

1. The inhomogeneous simulated annealing Markov chain must be weakly ergodic (i.e., dependence on the initial solution vanishes in the limit).
2. An eigenvector π_k with eigenvalue 1 must exist such that Equations (1.6) and (1.7) hold for every iteration k.
3. The Markov chain must be strongly ergodic (i.e., the Markov chain must be weakly ergodic and the sequence of eigenvectors π_k must converge to a limiting form), i.e.,

$$\sum_{k=0}^{\infty} \|\pi_k - \pi_{k+1}\| < +\infty. \tag{1.22}$$

4. The sequence of eigenvectors must converge to a form where all probability mass is concentrated on the set of globally optimal solutions ω^*. Therefore,

$$\lim_{k \to \infty} \pi_k = \pi^{\text{opt}}, \tag{1.23}$$

where π^{opt} is the equilibrium distribution with only global optima having probabilities greater than 0. (Note that weak and strong ergodicity are equivalent for homogeneous Markov chain theory.)

Mitra et al. [104] satisfy condition 1 (weak ergodicity) by first forming a lower bound on the probability of reaching any solution from any local minimum and then showing that this bound does not approach zero too quickly. For example, they define the lower bound for the simulated annealing transition probabilities in Equation (1.5) as

$$P^{(m)}(\omega, \omega') \geq w^m \exp(-m\Delta_L/t_{km-1}), \tag{1.24}$$

for any integer k greater than or equal to some fixed integer k_0, where m is the number of transitions needed to reach any solution from any solution of non-maximal objective function value, $w > 0$ is a lower bound on the one-step solution generation probabilities, Δ_L is the maximum one-step increase in objective function value

between any two solutions, and t_{km-1} is a temperature at iteration $km-1$. Mitra et al. [104] show that the Markov chain is weakly ergodic if for any fixed integer k_0

$$\sum_{k=k_0}^{\infty} \exp(-m\Delta_L/t_{km-1}) = +\infty. \tag{1.25}$$

Therefore, weak ergodicity is obtained if the temperature t_k is reduced sufficiently slowly to zero such that Equation (1.25) is satisfied. In general, the (infinite) sequence of temperatures $\{t_k\}$, $k = 1, 2, ...$, must satisfy

$$t_k \geq \frac{\beta}{\log(k)}, \tag{1.26}$$

where $\lim_{k \to \infty} t_k = 0$, β is a problem-dependent constant and k is the number of iterations. Mitra et al. [104] show that conditions (1.2), (1.3), and (1.4) are satisfied by using the homogeneous Markov chain theory developed for the transition probabilities Equation (1.5), provided that the solution generation function is symmetric.

Romeo and Sangiovanni-Vincentelli [120] note that while the logarithmic cooling schedule in Equation (1.26) is a sufficient convergence condition, there are only a few values for β which make the logarithmic rule also necessary. Furthermore, there exists a unique choice for β which makes the logarithmic rule both necessary and sufficient for the convergence of simulated annealing to the set of global optima. In [72], Hajek was the first to show that the logarithmic cooling schedule (Equation (1.26)) is both necessary and sufficient, by developing a tight lower bound for β, namely the depth of the deepest local minimum which is not a global minimum, under a weak reversibility assumption (note that Hajek requires the depth of global optima to be infinitely large). Hajek defines a Markov chain to be weakly reversible if, for any pair of solutions $\omega, \omega' \in \Omega$ and for any non-negative real number h, ω is reachable at height h from ω' if and only if ω' is reachable at height h from ω. Note that Hajek [72] does not attempt to satisfy the conditions of weak and strong ergodicity, but rather uses a more general probabilistic approach to develop a lower bound on the probability of escaping local, but not global, optima. Connors and Kumar [35] substantiate the necessary and sufficient conditions in Hajek [72] using the orders of recurrence,

$$B_i \equiv \sup \left\{ x \geq 0 \colon \sum_{k=0}^{\infty} \exp(-x/t_k)\pi_k(\omega) = +\infty \right\} \text{ for all } i \in \Omega. \tag{1.27}$$

Connors and Kumar [35] show that these orders of recurrence quantify the asymptotic behavior of each solution's probability in the solution distribution. The key result is that the simulated annealing inhomogeneous Markov chain converges in a Cesaro sense to the set of solutions having the largest recurrence orders. Borkar [16] improves this convergence result by using a convergence/oscillation dichotomy result for martingales. Tsitsiklis [150] uses bounds and estimates for singularly perturbed, approximately stationary Markov chains to develop a convergence theory

that subsumes the condition of weak reversibility in [72]. Note that Tsitsiklis [150] defines $N(h) \subset \Omega$ as the set of all local minima (in terms of objective function value) of depth $h+1$ or more. Therefore β is the smallest h such that all local (but not global) minima have depth h or less. Tsitsiklis conjectures that without some form of reversibility, there does not exist any h such that the global optima are contained in the set of local optima. Note that in [16, 28, 30, 35, 72, 104], the multiplicative condition (1.9) is required (either explicitly or implicitly) for the proofs of convergence.

Anily and Federgruen [10] use perturbation analysis techniques (e.g., see [102]) to prove convergence of a particular stochastic hill-climbing algorithm by bounding the deviations of the sequence of stationary distributions of the particular hill-climbing algorithm against the sequence of known stationary distributions corresponding to a simulated annealing algorithm. In general, this convergence proof approach is only useful for a restrictive class of simulated annealing algorithms, since the transition matrix condition number grows exponentially as the number of iterations k becomes large.

Anily and Federgruen [10] also present a proof of convergence for simulated annealing with general acceptance probability functions. Using inhomogeneous Markov chain theory, they prove convergence under the following necessary and sufficient conditions:

1. The acceptance probability function must, for any iteration k, allow any hill-climbing transition to occur with positive probability.
2. The acceptance probability function must be bounded and asymptotically monotone, with limit zero for hill-climbing solution transitions.
3. In the limit, the stationary probability distribution must have 0 probability mass for every non-globally optimal solution.
4. The probability of escaping from any locally (but not globally) optimal solution must not approach 0 too quickly.

Anily and Federgruen [10] use condition (3) to relax the acceptance function multiplicative condition (1.9). However, in practice, condition (3) would be very difficult to check without assuming that Equation (1.9) holds. Condition (4) provides the necessary condition for the rate that the probability of hill-climbing transitions approaches 0. Condition (4) is expressed quantitatively as follows: let t_k be defined by Equation (1.2) and define the minimum one-step acceptance probability as

$$\underline{a}_k = \min_{\omega \in \Omega, \omega' \in N(\omega)} a_{t_k}(\omega, \omega'). \tag{1.28}$$

Define the set of local optima $L \subset \Omega$ such that $\omega \in L$ implies that $f(\omega) \leq f(\omega')$ for all $\omega' \in N(\omega)$, and let

$$\overline{a}_k = \min_{\omega \in L, \omega' \in N(\omega) \setminus L} a_{t_k}(\omega, \omega'). \tag{1.29}$$

Finally, let any solution $\omega' \in \Omega$ be reachable from any solution $\omega \in \Omega$ in q transitions or less. If (non-globally) locally optimal solutions exist,

$$\sum_{k=1}^{\infty} (\underline{a}_k)^q = +\infty, \tag{1.30}$$

and conditions (1), (2), and (3) hold, then the simulated annealing algorithm will asymptotically converge to the set of global optima with probability 1. However, if (non-globally) locally optimal solutions exist and

$$\sum_{k=1}^{\infty} \overline{a}_k < +\infty, \tag{1.31}$$

then the simulated annealing algorithm will not always converge to the set of global optima with probability 1. Johnson and Jacobson [85] relax the sufficient conditions found in [10] by using a path argument between global optima and local (but not global) optima.

Yao and Li [165] and Yao [164] also discuss simulated annealing algorithms with general acceptance probabilities, though their primary contribution is with respect to general neighborhood generation distributions. In [126], Schuur provides a description of acceptance functions ensuring the convergence of the associated simulated annealing algorithm to the set of global optima.

The inhomogeneous proof concept is stronger than the homogeneous approach in that it provides necessary conditions for the rate of convergence, but its asymptotic nature suggests that practical implementation may not be feasible. Romeo and Sangiovanni-Vincentelli [120] note that "there is no reason to believe that truncating the logarithmic temperature sequence would yield a good configuration, since the tail of the sequence is the essential ingredient in the proof." In addition, the logarithmic cooling schedule dictates a very slow rate of convergence. Therefore, most recent work has focused on methods of improving simulated annealing's finite-time behavior and modifying or blending the algorithm with other search methods such as genetic algorithms [92], tabu search [66], or both [59].

1.2.2 Finite-Time Performance

Over the past decade, a growing body of work has been devoted to finite-time behavior of simulated annealing. Jacobson and Yucesan [82] present necessary and sufficient (asymptotic) convergence conditions for generalized hill climbing algorithms that include simulated annealing as a special case. They also introduce new performance measures that can be used to evaluate and compare both convergent and nonconvergent generalized hill-climbing algorithms with random restart local search [79]. Such a comparison provides insights into both asymptotic and finite-time performance of discrete optimization algorithms. For example, they use the global visit probability to evaluate the performance of simulated annealing using random restart local search as a benchmark. These results suggest that random restart local search may outperform simulated annealing provided that a sufficiently large number of

restarts are executed. In [60], Fox notes that a similar result that compares random restart local search with simulated annealing can be true but only if both the number of accepted and rejected moves are counted. A clever example is also provided in [60] to illustrate this point, which illustrates that comparing random restart local search and simulating annealing may not be prudent. In [59] and [61], Fox presents modifications of simulated annealing that circumvent the counting issue described in [60], hence yielding superior performing simulated annealing algorithm implementations. The primary value of using simulated annealing may therefore be for finite-time executions that obtain near-optimal solutions reasonably quickly. This, in turn, suggests that studying the finite-time behavior of simulated annealing is equally important as its asymptotic convergence.

Chiang and Chow [29] and Mazza [100] investigate the statistical properties of the first visit time to a global optimum, which provides insight into asymptotic properties of the algorithm as the outer loop counter $k \rightarrow +\infty$. In [20], Catoni investigates optimizing a finite-horizon cooling schedule to maximize the number of visits to a global optimum after a finite number of iterations. In [44], Desai focuses on finite-time performance by incorporating size-asymptotic information supplied by certain eigenvalues associated with the transition matrix. Desai [44] also provides some quantitative and qualitative information about the performance of simulated annealing after a finite number of steps, by observing the quality of solutions is related to the number of steps that the algorithm has taken.

Srichander [133] examines the finite-time performance of simulated annealing using spectral decomposition of matrices. He proposes that an annealing schedule on the temperature is not necessary for the final solution of the simulated annealing algorithm to converge to the global minimum with probability 1. Srichander shows that initiating the simulated annealing algorithm with high initial temperatures produces an inefficient algorithm in the sense that the number of function evaluations required to obtain a global minima is very large. A modified simulated annealing algorithm is presented with a low initial temperature and an iterative schedule on the size of the neighborhood sets that leads to a more efficient algorithm. The new algorithm is applied to a real-world example and computational performance is reported.

Fleischer and Jacobson [58] use a reverse approach to establish theoretical relationships between the finite-time performance of an algorithm and the characteristics of problem instances. They observe that the configuration space created by an instance of a discrete optimization problem determines the efficiency of simulated annealing when applied to that problem. The entropy of the Markov chain embodying simulated annealing is introduced as a measure that captures the entire topology of the configuration space associated with the particular instance of the discrete optimization problem. By observing the expected value of the final state in a simulated annealing algorithm as it relates to the entropy value of the underlying Markov chain, they present measures of performance that determine how well the simulated annealing algorithm performs in finite time. Their computational results suggest that superior finite time performance of a simulated annealing algorithm are associated with higher entropy measures.

Nolte and Schrader [111] give a proof of the convergence of simulated annealing by applying results about rapidly mixing Markov chains. With this proof technique, it is possible to obtain better bounds for the finite-time behavior of simulated annealing than previously known.

To evaluate the expected run-time required by a simulated annealing algorithm to reach solution of a pre-specified quality, Wood et al. [161] present an approach to model and analyze a generic stochastic global optimization algorithm using a sequence of stochastic processes, culminating in a backtracking adaptive search process. The theory developed for this backtracking adaptive search procedure is then used to analyze the classic simulated annealing algorithm.

In [118], Rajasekaran presents an analysis of simulated annealing that provides a time bound for convergence with very high probability. Convergence of simulated annealing in the limit then follows as a corollary to the established finite-time performance results.

1.3 Relationship to Other Local Search Algorithms

The hill-climbing strategy inherent in simulated annealing has lead to the formulation of other such algorithms (e.g., threshold accepting, the noising method). Moreover, though different in how they traverse the solution space, both tabu search and genetic algorithms share with simulated annealing the objective of using local information to find global optima over solution spaces with multiple local optima.

1.3.1 Threshold Accepting

Questioning the very need for a randomized acceptance function, Dueck and Scheuer [45] and, independently, Moscato and Fontanari [106] propose the threshold accepting algorithm, where the acceptance probability function is

$$a_k(\Delta_{\omega,\omega'}) = \begin{cases} 1 & \text{if } Q_k \geq \Delta_{\omega,\omega'} \\ 0 & \text{otherwise} \end{cases},$$

with Q_k defined as the threshold value at iteration k. Q_k is typically set to be a deterministic, non-increasing step function in k. Dueck and Scheuer [45] report computational results that suggest dramatic improvements in traveling salesman problem solution quality and algorithm run-time over basic simulated annealing. Moscato and Fontanari [106] report more conservative results—they conjecture that simulated annealing's probabilistic acceptance function does not play a major role in the search for near-optimal solutions.

Althofer and Koschnick [8] develop a convergence theory for threshold accepting based on the concept that simulated annealing belongs to the convex hull of

threshold accepting. The idea presented in [8] is that (for a finite Q_k threshold sequence) there can exist only finitely many threshold accepting transition matrices, but simulated annealing can have infinitely many transition matrices because of the real-valued nature of the temperature at each iteration. However, every simulated annealing transition matrix for a given problem can be represented as a convex combination of the finitely many threshold accepting transition matrices. Althofer and Koschnick [8] are unable to prove that threshold accepting will asymptotically reach a global minimum, but it does prove the existence of threshold schedules that provide convergence to within an ε-neighborhood of the optimal solutions. Jacobson and Yucesan [81] prove that if the threshold value approaches 0 as k approaches infinity, then the algorithm does not converge in probability to the set of globally optimal solutions.

Hu et al. [77] modify threshold accepting to include a non-monotonic, self-tuning threshold schedule in the hope of improving the algorithm's finite-time performance. Hu et al. allow the threshold Q_k to change dynamically (either up or down), based on the perceived likelihood of being near a local minimum. These changes are accomplished using a principle they call *dwindling expectation*—when the algorithm fails to move to neighboring solutions, the threshold Q_k is gradually increased, in the hope of eventually escaping a local optimum. Conversely, when solution transitions are successful, the threshold is reduced, in order to explore local optima. The experimental results based on two traveling salesman problems presented in [77] showed that the proposed algorithm outperformed previous hill-climbing methods in terms of finding good solutions earlier in the optimization process.

Threshold accepting's advantages over simulated annealing lie in its ease of implementation and its generally faster execution time, due to the reduced computational effort in avoiding acceptance probability computations and the generation of random numbers [106]. However, compared to simulated annealing, relatively few threshold accepting applications are reported in the literature [93, 110, 125].

1.3.2 Noising Method

Charon and Hudry [23] advocate a simple descent algorithm called the *noising method*. The algorithm first perturbs the solution space by adding random noise to the problem's objective function values. The noise is gradually reduced to 0 during the algorithm's execution, allowing the original problem structure to reappear. Charon and Hudry provide computational results, but do not prove that the algorithm will asymptotically converge to the set of globally optimal solutions. Charon and Hudry [24] show how the noising method is a generalization of simulated annealing and threshold accepting.

Storer et al. [136] propose an optimization strategy for sequencing problems, by integrating fast, problem-specific heuristics with local search. Its key contribution is to base the definition of the search neighborhood on a heuristic problem pair (h, p), where h is a fast, known, problem-specific heuristic and p represents the problem data. By perturbing the heuristic, the problem, or both, a neighborhood of solutions

is developed. This neighborhood then forms the basis for local search. The hope is that the perturbations will cluster good solutions close together, thus making it easier to perform local search.

1.3.3 Tabu Search

Tabu search [66] is a general framework for a variety of iterative local search strategies for discrete optimization. Tabu search uses the concept of *memory* by controlling the algorithm's execution via a dynamic list of forbidden moves. This allows the tabu search algorithm to intensify or diversify its search of a given problem's solution space in an effort to avoid entrapment in local optima. See [67] for a discussion on the convergence of tabu search algorithms.

Given that simulated annealing is completely memoryless (i.e., simulated annealing disregards all historical information gathered during the algorithm's execution), tabu search provides an alternative mechanism to hill-climb and escape local optima. Faigle and Kern [50] propose a particular tabu search algorithm called *probabilistic tabu search* as a meta-heuristic to help guide simulated annealing. Probabilistic tabu search attempts to capitalize on both the asymptotic optimality of simulated annealing and the memory feature of tabu search. In probabilistic tabu search, the probabilities of generating and accepting each candidate solution are set as functions of both a temperature parameter (as in simulated annealing) and information gained in previous iterations (as in tabu search). Faigle and Kern [50] are then able to prove asymptotic convergence of their particular tabu search algorithm by using methods developed for simulated annealing [49, 52]. Note that the results in [50] build upon work by Glover [67] where probabilistic tabu search was first introduced and contrasted with simulated annealing.

Vaughan and Jacobson [158] develop a framework, termed tabu guided generalized hill climbing, that uses a tabu release parameter that probabilistically accepts solutions currently on the tabu list. The presented algorithms are modeled as a set of stationary Markov chains, where the tabu list is fixed for each outer loop iteration. This framework provides practitioners with guidelines for developing tabu search strategies to use in conjunction with generalized hill-climbing algorithms that preserve some of the algorithms' known performance properties. Sufficient conditions are obtained that indicate how to design iterations for problem-specific tabu search strategies.

1.3.4 Genetic Algorithms

Genetic algorithms [92] emulate the evolutionary behavior of biological systems. They generate a sequence of populations of candidate solutions to the underlying optimization problem by using a set of genetically inspired stochastic solution transition operators to transform each population of candidate solutions into a

descendent population. The three most popular transition operators are reproduction, cross-over, and mutation [38]. Davis and Principe [39] and Rudolph [124] attempt to use homogeneous finite Markov chain techniques to prove convergence of genetic algorithms [21], but are unable to develop a theory comparable in scope to that of simulated annealing.

Zolfaghari and Liang [167] undertake a comparative study of simulated annealing, genetic algorithms, and tabu search for solving binary (considering only machines and part types) machine-grouping problems of varying types (involving machine/part types, processing times, lot sizes, and machine capacities). To test the performance of the three metaheuristics, two binary performance indices and two generalized performance indices are used for binary and comprehensive machine/part grouping problems, respectively. The comparisons are made in terms of solution quality, search convergence behavior, and pre-search effort. The results indicate that simulated annealing outperforms both genetic algorithm and tabu search, particularly for large problems.

In [107], Muhlenbein presents a theoretical analysis of genetic algorithms based on population genetics. He counters the popular notion that models that mimic natural phenomenon are superior to other models. The article argues that evolutionary algorithms can be inspired by nature, but do not necessarily have to copy a natural phenomenon. He addresses the behavior of transition operators and designs new genetic operators that are not necessarily related to events in nature, yet still perform well in practice.

One criticism of simulated annealing is the slow speed at which it converges. In [41], Delport combines simulated annealing with evolutionary algorithms to improve performance in terms of speed and solution quality. The benefit of this hybrid system of simulated annealing and evolutionary selection is due to the adjustments in the cooling schedule based on fast recognition of the thermal equilibrium in terms of selection intensity, which results in much faster convergence of the algorithm.

Sullivan and Jacobson [139, 140] link genetic algorithms with simulated annealing using generalized hill-climbing algorithms [80]. They first link genetic algorithms to ordinal hill-climbing algorithms, which can then be used, through its formulation within the generalized hill-climbing algorithm framework, to form a bridge with simulated annealing. Though genetic algorithms have proven to be effective for addressing intractable discrete optimization problems and can be classified as a type of hill-climbing approach, its link with generalized hill-climbing algorithms (through the ordinal hill-climbing formulation) provides a means to establish well-defined relationships with other generalized hill-climbing algorithms (like simulated annealing and threshold accepting). They also present two formulations of genetic algorithms that provide a first step toward developing a bridge between genetic algorithms and other local search strategies like simulated annealing.

1.3.5 Generalized Hill-Climbing Algorithms

Generalized hill-climbing algorithms (GHC) (see [80]) provide a framework for modeling local search algorithms used to address intractable discrete optimization

problems. All generalized hill-climbing algorithms have the same basic structure, but can be tailored to a specific instance of a problem by changing the hill-climbing random variable (which is used to accept or reject inferior solutions) and neighborhood functions. Generalized hill-climbing algorithms are described in pseudo-code form:

Select an initial solution $\omega \in \Omega$
Set the outer loop counter bound K and the inner loop counter bounds M_k, $k = 1, 2, ..., K$
Define a set of hill-climbing (random) variables $R_k : \Omega \times \Omega \rightarrow (-\infty, +\infty)$, $k = 1, 2, ..., K$
Set the iteration indices $k = m = 1$
Repeat while $k \leq K$
    Repeat while $m \leq M_k$
        Generate a solution $\omega' \in N(\omega)$
        Calculate $\Delta_{\omega, \omega'} = f(\omega') - f(\omega)$
        If $R_k(\omega, \omega') \geq \Delta_{\omega, \omega'}$, then $\omega \leftarrow \omega'$
        $m \leftarrow m + 1$
    Until $m = M_k$
    $m \leftarrow 1, k \leftarrow k + 1$
Until $k = K$

Note that the outer and inner loop bounds, K and M_k, $k = 1, 2, ..., K$, respectively, may all be fixed, or K can be fixed with the M_k, $k = 1, 2, ..., K$, defined as random variables whose values are determined by the solution at the end of each set of inner loop iterations satisfying some property (e.g., the solution is a local optima).

Generalized hill-climbing algorithms can be viewed as sampling procedures over the solution space Ω. The key distinction between different generalized hill-climbing algorithm formulations is in how the sampling is performed. For example, simulated annealing produces biased samples, guided by the neighborhood function, the objective function, and the temperature parameter. More specifically, simulated annealing can be described as a generalized hill-climbing algorithm by setting the hill-climbing random variable, $R_k(\omega, \omega') = -t_k ln(u_k)$, $\omega \in \Omega$, $\omega' \in N(\omega)$, $k = 1, 2, ..., K$, with the $\{u_k\}$ independently and identically distributed $U(0,1)$ random variables. To formulate Monte Carlo search as a generalized hill-climbing algorithm, set $R_k(\omega, \omega') = +\infty$, $\omega \in \Omega$, $\omega' \in N(\omega)$, $k = 1, 2, ..., K$. Deterministic local search, which accepts only neighbors of improving (lower) objective function value, can be expressed as a generalized hill-climbing algorithm with $R_k(\omega, \omega') = 0$, $\omega \in \Omega$, $\omega' \in N(\omega)$, $k = 1, 2, ..., K$. Other algorithms that can be described using the generalized hill-climbing framework include threshold accepting, some simple forms of tabu search, and Weibull accepting. For detailed discussions of these algorithms and a description of how they fit into the generalized hill-climbing algorithm framework, see [80, 85, 139].

Measures for assessing the finite-time performance of generalized hill-climbing algorithms have been developed, including the expected number of iterations to visit a predetermined objective function value level. Jacobson et al. [83] introduce the cyclical simulated annealing algorithm and describe a procedure to estimate

lower and upper bounds for the expected number of iterations to visit a near-optimal solution for this algorithm. Computational results with four traveling salesman problem instances are reported.

1.4 Practical Guidelines

Implementation issues for simulated annealing can follow one of two paths—that of specifying problem-specific choices (objective function and neighborhood) and that of specifying generic choices (generation and acceptance probability functions and the cooling schedule) (see [46]). These choices often depend on the domain of each specific problem. The principal shortcoming of simulated annealing is that it often requires extensive computer time. Implementation modifications generally strive to retain simulated annealing's asymptotic convergence character, but at reduced computer run-time. The methods discussed here are mostly heuristic.

1.4.1 Problem-Specific Choices

1.4.1.1 Objective Functions

One problem-specific choice involves the objective function specification. In [135], Stern recommends a heuristic temperature-dependent penalty function as a substitute for the actual objective function for problems where low cost solutions have neighbors of much higher cost or in cases of degeneracy (i.e., large neighborhoods of solutions of equal, but high costs). The original objective function surfaces, as the penalty and the temperature are gradually reduced to 0. This technique is similar to the noising method presented by Charon and Hudrey in [23], where the penalty function is described as noise and is reduced at each outer loop iteration of the algorithm. One speed-up technique is to evaluate only the difference in objective functions, $\Delta_{\omega,\omega'}$, instead of calculating both $f(\omega)$ and $f(\omega')$. In [148], Tovey suggests several methods of approximating $\Delta_{\omega,\omega'}$ by using surrogate functions (that are faster to evaluate than $\Delta_{\omega,\omega'}$, but not as accurate) probabilistically for cases when evaluation of $\Delta_{\omega,\omega'}$ is expensive; this technique is referred to as the *surrogate function swindle*.

Straub et al. [137] improve the performance of simulated annealing on problems in chemical physics by using the continuous normal density distribution instead of the molecular dynamics of single point particles to describe the potential energy landscape. Ma and Straub [98] report that using this distribution has the effect of smoothing the energy landscape by reducing both the number and depth of local minima.

Yan and Mukai [162] consider the case when a closed-form formula for the objective function is not available. They use a probabilistic simulation (termed the

stochastic ruler method) to generate a sample objective function value for an input solution and then accept the solution if the sample objective function value falls within a predetermined bound. They also provide a proof of asymptotic convergence by extrapolating the convergence proofs for simulated annealing and analyze the rate of convergence.

1.4.1.2 Neighborhoods

A key problem-specific choice concerns the neighborhood function definition. The efficiency of simulated annealing is highly influenced by the neighborhood function used [105]. The choice of neighborhood serves to enforce a topology. In [46], Eglese reports that "a neighborhood structure which imposes a 'smooth' topology where the local minima are shallow is preferred to a 'bumpy' topology where there are many deep local minima." Solla et al. [132] and Fleischer and Jacobson [58] report similar conclusions. This also supports the result in [72] that shows that asymptotic convergence to the set of global optima depends on the depth of the local minima.

Another factor to consider when choosing neighborhood functions is the neighborhood size. No theoretical results are available, other than the necessity of reachability (in a finite number of steps) from any solution to any other solution. Cheh et al. [25] report that small neighborhoods are best, while Ogbu and Smith [113] provide evidence that larger neighborhoods result in better simulated annealing performance. Goldstein and Waterman [68] conjecture that if the neighborhood size is small compared to the total solution space cardinality, then the Markov chain cannot move around the solution space fast enough to find the minimum in a reasonable time. On the other hand, a very large neighborhood has the algorithm merely sampling randomly from a large portion of the solution space and thus is unable to focus on specific areas of the solution space. It is reasonable to believe that neighborhood size is heavily problem specific. For example, problems where the smoothness of its solution space topology is moderately insensitive to different neighborhood definitions may benefit from larger neighborhood sizes.

Concepts from information theory are used in [54] and [58] to show that the neighborhood structure can affect the *information rate* or total uncertainty associated with simulated annealing. In [54], Fleischer shows that simulated annealing tends to perform better as the entropy level of the associated Markov chain increases and thus conjectures that an entropy measure could be useful for predicting when simulated annealing would perform well on a given problem. However, efficient ways of estimating the entropy are needed to transform this result into a practical tool.

Triki et al. [151] present an empirical study on the efficiency of the simulated annealing algorithm, as impacted by the landscape and the choice of the neighborhood function. The experiments they conducted follow the observation that it is possible to compute the exact probability for the algorithm to reach any point in the landscape, provided that the number of solutions and the number of neighbors per solution are sufficiently small. A developed computational tool allows to study the

influence of the tuning of all the main parameters of simulated annealing, as well as theoretical concepts such as thermodynamic equilibrium and optimal temperature decrement rules.

Bouffard and Ferland [17] propose a method to improve the simulated annealing algorithm with a variable neighborhood search to solve the resource-constrained scheduling problem. The method is compared numerically with other neighborhood search techniques: threshold accepting methods and tabu search. Furthermore, these techniques are combined with multi-start diversification strategies. The numerical results indicate that using a variable neighborhood search technique indeed improves the performance.

Another issue on neighborhood function definition addresses the solution space itself. Chardaire et al. [22] propose a method for addressing $0 - 1$ optimization problems, in which the solution space is progressively reduced by fixing the value of *strongly persistent* variables (which have the same value in all optimal solutions). They isolate the persistent variables during simulated annealing's execution by periodically estimating the expectation of the random variable (a vector of binary elements) that describes the current solution and fixing the value of those elements in the random variable that meet threshold criteria.

1.4.2 Generic Choices

1.4.2.1 Generation Probability Functions

Generation probability functions are usually chosen as uniform distributions with probabilities proportional to the size of the neighborhood. The generation probability function is typically not temperature dependent. In [59], Fox suggests that instead of blindly generating neighbors uniformly, adopt an intelligent generation mechanism that modifies the neighborhood and its probability distribution to accommodate search intensification or diversification, in the same spirit of the tabu search metaheuristic. Fox also notes that simulated annealing convergence theory does not preclude this idea. In [148], Tovey suggests an approach with a similar effect, called the *neighborhood prejudice swindle*.

1.4.2.2 Acceptance Probability Functions

The literature reports considerable experimentation with acceptance probability functions for hill-climbing transitions. The most popular is the exponential form (1.1). Ogbu and Smith [113] consider replacing the basic simulated annealing acceptance function $a_k(\Delta_{\omega,\omega'})$ with a geometrically decreasing form that is independent of the change in objective function value. They adopt a *probabilistic-exhaustive* heuristic technique in which randomly chosen neighbors of a solution are examined and all solutions that are accepted are noted, but only the last solution accepted

becomes the new incumbent. The hope is that this scheme will explore a broader area of the problem solution space. Their acceptance probability function is defined for all solutions $\omega, \omega' \in \Omega$ and for $k = 1, 2, ..., K$ as

$$a_k(\Delta_{\omega,\omega'}) = a_k = \begin{cases} a_1 x^{k-1} & \text{if } f(\omega') > f(\omega) \\ 1 & \text{otherwise} \end{cases},$$

where a_1 is the initial acceptance probability value, $x \in (0,1)$ is a reducing factor, and K is the number of stages (equivalent to a temperature cooling schedule). They also experiment with this method (and a neighborhood of large cardinality) on a permutation flow shop problem and report that its approach found comparable solutions to the basic simulated annealing algorithm in one-third the computation time.

1.4.2.3 Cooling Schedules

The simulated annealing cooling schedule is fully defined by an initial temperature, a schedule for reducing/changing the temperature, and a stopping criterion. Romeo and Sangiovanni-Vincentelli [120] note that an effective cooling schedule is essential to reducing the amount of time required by the algorithm to find an optimal solution. Therefore, much of the literature on cooling schedules (see [19, 34, 62, 112]) is devoted to this efficiency issue.

Homogeneous simulated annealing convergence theory has been used to design effective cooling schedules. Romeo and Sangiovanni-Vincentelli [120] suggest the following procedure for designing a cooling schedule:

1. Start with an initial temperature t_0 for which a good approximation of the stationary distribution π_{t_0} is quickly reached.
2. Reduce t_0 by an amount $\delta(t)$ small enough such that π_{t_0} is a good starting point to approximate $\pi_{t_0 - \delta(t)}$.
3. Fix the temperature at a constant value during the iterations needed for the solution distribution to approximate $\pi_{t_0 - \delta(t)}$.

Repeat the above process of cooling and iterating until no further improvement seems possible.

Generally, the initial temperature is set such that the acceptance ratio of bad moves is equal to a certain value. In [14], Ben-Ameur proposes an algorithm to compute a temperature which is compatible with a given acceptance ratio. The acceptance probability function is shown to be convex for low temperatures and concave for high temperatures, and a lower bound is provided for the number of required temperature changes based on a geometric cooling schedule.

Cooling schedules are grouped into two classes: static schedules, which must be completely specified before the algorithm begins; and *adaptive* schedules, which adjust the temperature's rate of decrease from information obtained during the algorithm's execution. Cooling schedules are almost always heuristic; they seek to balance moderate execution time with simulated annealing's dependence on asymptotic behavior.

Strenski and Kirkpatrick [138] present an exact (non-heuristic) characterization of finite-length annealing schedules. They consider extremely small problems that represent features (local optima and smooth/hilly topologies) and determine probability distribution of outcomes of annealing process in a finite number of iterations to gain insights into some popular assumptions and intuition behind cooling schedules. Their experiments suggest that optimal cooling schedules are not monotone decreasing in temperature. They also show that for the test problem (a white noise surface), geometric and linear cooling schedules perform better than inverse logarithmic cooling schedules, when sufficient computing effort is allowed. Moreover, their experiments do not show measurable performance differences between linear and geometric cooling schedules. They also observe that geometric cooling schedules are not greatly affected by excessively high initial temperatures. The results presented suggest that even the most robust adaptive cooling schedule "produces annealing trajectories which are never in equilibrium" [138]. However, they also conclude that the transition acceptance rate is not sensitive to the degree of closeness to the equilibrium distribution.

Christoph and Hoffmann [31] also attempt to characterize optimal cooling schedules. They derive a relationship between a finite sequence of optimal temperature values (i.e., outer loops) and the number of iterations (i.e., inner loops) at each respective temperature for several small test problems to reach optimality (i.e., the minimal mean final energy). They find that this *scaling behavior* is of the form

$$x_m = a_m v^{-b_m}, \qquad (1.32)$$

where a and b are scaling coefficients, $x_m = \exp(-1/t_k)$ is referred to as the temperature, v is the number of inner loop iterations at temperature x_m, and m is the number of outer loops at which the temperature x_m is reduced. The proposed approach is to solve for the coefficients a and b based on known temperature and iteration parameter values for an optimal schedule based on several replications of the algorithm using $(m \times v)$ iterations for each replication, and then use Equation (1.32) to interpolate the optimal cooling schedule for intermediate iterations. They however do not make any suggestions on how to efficiently solve for the necessary optimal cooling schedules for a (typically large) problem instance.

Romeo and Sangiovanni-Vincentelli [120] present a theoretical framework for evaluating the performance of the simulated annealing algorithm. They discuss annealing schedules in terms of initial temperature $T = t_0$, the number of inner loops for each value of t_k, the decrease rate of the temperature (i.e., cooling schedule), and the criteria for stopping the algorithm. They conclude that the theoretical results obtained thus far have not been able to explain why simulated annealing is so successful even when a diverse collection of static cooling schedule heuristics is used. Many heuristic methods are available in the literature to find optimal cooling schedules, but the effectiveness of these schedules can only be compared through experimentation. They conjecture that the neighborhood and the corresponding topology of the objective function are responsible for the behavior of the algorithm.

Cohn and Fielding [34] conduct a detailed analysis of various cooling schedules and how they affect the performance of simulated annealing. Convergent simulated

annealing algorithms are often too slow in practice, whereas a number of non-convergent algorithms may be preferred for good finite-time performance. They analyze various cooling schedules and present cases where repeated independent runs using a non-convergent cooling schedule provide acceptable results in practice. They provide examples of when it is both practically and theoretically justified to use a very high, fixed temperature, or even fast cooling schedules which have a small probability of reaching global minima and apply these cooling schedules to traveling salesman problems of various sizes. Fielding [53] computationally studies fixed temperature cooling schedules for the traveling salesman problem, the quadratic assignment problem and the graph partitioning problem and demonstrates that a fixed temperature cooling schedule can yield superior results in locating optimal and near-optimal solutions. Orosz and Jacobson [115, 116] present finite-time performance measures for simulated annealing with fixed temperature cooling schedules. They illustrate their measures using randomly generated instances of the traveling salesman problem.

Another approach to increasing the speed of simulated annealing is to implement a two-staged simulated annealing algorithm. In two-staged simulated annealing algorithms, a fast heuristic is used to replace simulated annealing at higher temperatures, with a traditional simulated annealing algorithm implemented at lower temperatures to improve on the fast heuristic solution. In addition to implementing an intelligent cooling schedule, finding the initial temperature t_0 to initialize the traditional simulated annealing algorithm is important to the success of the two-staged algorithm. Varanelli and Cohoon [157] propose a method for determining an initial temperature t_0 for two-staged simulated annealing algorithms using traditional cooling schedules. They note that if t_0 is too low at the beginning of the traditional simulated annealing phase, the algorithm can get trapped in an inferior solution, while if the initial temperature t_0 is too high, the algorithm can waste too many iterations (hence computing time) by accepting too many hill-climbing moves.

Azizi and Zolfaghari [11] propose two variations of simulated annealing, tested on the minimum makespan job shop scheduling problems. In the conventional simulated annealing, the temperature declines monotonically, providing the search with a higher transition probability in the beginning of the search and lower probability toward the end of the search. In [11], an adaptive temperature control scheme with a tabu list is used that changes temperature based on the number of consecutive improving moves, resulting in an improved algorithm performance.

In [69], a sample adaptive simulated annealing algorithm, inspired by the idea of Metropolis algorithm, is constructed on a finite state space. The algorithm can be viewed as a substitute of the annealing of iterative stochastic schemes. In [108], the optimal cooling schedule for simulated annealing is formulated to derive a differential equation for the time-dependent temperature $T(t)$. Based on this equation, the long-term behavior of $T(t)$, entropy production, and the Kullback–Leibler entropy are studied. For some simple examples, such as a many-level system and the small scale traveling salesman problem, the explicit time dependence of the temperature is obtained.

1.4.3 Domains—Types of Problems with Examples

Over the past decade, simulated annealing has developed into a popular optimization tool. It has been used to address numerous discrete optimization problems as well as continuous variable problems. Several application articles and surveys have been published on simulated annealing. Johnson et al. [86, 87] present a series of articles on simulated annealing applied to certain well-studied discrete optimization problems. The first in the series of articles uses the graph partitioning problem to illustrate simulated annealing and highlight the effectiveness of several modifications to the basic simulated annealing algorithm. The second in the series focuses on applying lessons learned from the first article to the graph coloring and number partitioning problems. Local optimization techniques were previously thought to be unacceptable approaches to these two problems. In [87], it is also observed that for long run lengths, simulated annealing outperforms the traditional techniques used to solve graph coloring problems. However, simulated annealing did not compare well with traditional techniques on the number partitioning problem except for small problem instances. The third article in the series (not yet published) uses simulated annealing to approach the well-known traveling salesman problem.

Koulamas et al. [90] focus on simulated annealing applied to applications in production/operations management and operations research. They discuss traditional problems such as single machine, flow shop and job shop scheduling, lot sizing, and traveling salesman problems as well as non-traditional problems to include graph coloring and number partitioning. They conclude that simulated annealing is an effective tool for solving many problems in operations research and that the degree of accuracy that the algorithm achieves can be controlled by the practitioner, in terms of number of iterations and neighborhood functions (i.e., an increased number of iterations (outer loops) combined with increased number of searches at each iteration (inner loops) can result in solutions with a higher probability of converging to the optimal solution). In [55], Fleischer discusses simulated annealing from a historical and evolutionary point of view in terms of solving difficult optimization problems. Fleischer also summarizes ongoing research and presents an application of simulated annealing to graph problems.

Steinhofel et al. [134] also address flow shop scheduling problems by applying logarithmic cooling schedules of simulated annealing-based algorithms to flow shop scheduling. In the considered problem setting, the objective is to minimize the overall completion time (called the makespan). A lower bound is derived for the number of steps that are needed to approach an optimum solution with a certain probability, based on the maximum escape depth Γ from local minima of the underlying energy landscape.

The simulated annealing algorithm has proved to be a good technique for solving difficult discrete optimization problems. In engineering optimization, simulated annealing has emerged as an alternative tool to address problems that are difficult to solve by conventional mathematical programming techniques. The algorithm's major disadvantage is that solving a complex system problem may be an extremely

slow, albeit convergent process, using much more processor time than conventional algorithms. Consequently, simulated annealing has not been widely embraced as an optimization algorithm for engineering problems. Attempts have been made to improve the performance of the algorithm either by reducing the annealing length or changing the generation and the acceptance mechanisms. However, these faster schemes, in general, do not inherit the property of escaping local minima. A more efficient way to reduce the processor time and make simulated annealing a more attractive alternative for engineering problems is to add parallelism (see [73]). However, the implementation and efficiency of parallel simulated annealing algorithms are typically problem dependent. Leite et al. [91] consider the evaluation of parallel schemes for engineering problems where the solution spaces may be very complex and highly constrained, with function evaluations varying from medium to high cost. In addition, they provide guidelines for selecting appropriate schemes for engineering problems. They also present an engineering problem with relatively low fitness evaluation cost and strong time constraints to demonstrate the lower bounds of applicability of parallel schemes.

Many signal processing applications create optimization problems with multimodal and non-smooth cost functions. Gradient methods are ineffective in these situations because of multiple local minima and the requirement to compute gradients. Chen and Luk [27] propose an adaptive simulated annealing algorithm as a viable optimization tool for addressing such difficult non-linear optimization problems. The adaptive simulated annealing algorithm maintains the advantages of simulated annealing, but converges faster. Chen and Luk demonstrate the effectiveness of adaptive simulated annealing with three signal processing applications: maximum likelihood joint channel and data estimation, infinite-impulse-response filter design, and evaluation of minimum symbol-error-rate decision feedback equalizer. They conclude that the adaptive simulated annealing algorithm is a powerful global optimization tool for solving such signal processing problems.

Abramson et al. [4] describe the use of simulated annealing for solving the school timetabling problem. They use the scheduling problem to highlight the performance of six different cooling schedules: the basic geometric cooling schedule, a scheme that uses multiple cooling rates, geometric reheating, enhanced geometric reheating, non-monotonic cooling, and reheating as a function of cost. The basic geometric cooling schedule found in [156] is used as the baseline schedule for comparison purposes. Experimental results suggest that using multiple cooling rates for a given problem yields better quality solutions in less time than the solutions produced by a single cooling schedule. The conclusion in [4] is that the cooling scheme that uses the phase transition temperature (i.e., when sub-parts of the combinatorial optimization problem are solved) in combination with the best solution to date produces the best results.

Emden-Weinert and Proksch [47] present a study of a simulated annealing algorithm for the flight pairing subproblem in crew scheduling, which models the matching of flight segments as a preliminary phase to crew rostering. It is revealed that the algorithm run-time can be decreased and solution quality can be improved by using a problem-specific initial solution, relaxing constraints, combining simulated

annealing with a problem-specific local improvement heuristic, and conducting multiple independent runs.

There is no question that simulated annealing can demand significant computational time to reach global minima. Recent attempts to use parallel computing schemes to speed up simulated annealing have provided promising results. Chu et al. [32] present a new, efficient, and highly general-purpose parallel optimization method based on simulated annealing that does not depend on the structure of the optimization problem being addressed. Their algorithm is used to analyze a network of interacting genes that control embryonic development and other fundamental biological processes. They use a two-stage procedure which monitors and pools performance statistics obtained simultaneously from all processors and then mixes states at intervals to maintain a Boltzmann-like distribution of costs. They demonstrate that their parallel simulated annealing approach leads to nearly optimal parallel efficiency for certain optimization problems. In particular, the approach is appropriate when the relative effort required to compute the cost function is large compared to the relative communication effort between parallel machines for pooling statistics and mixing states.

Chen et al. [26] implement five variants of simulated annealing algorithm from sequential-to-parallel forms on high-performance computers and applied them to a set of standard function optimization problems in order to test their performances. The experimental results indicate that the traditional approach to parallelizing simulated annealing, namely executing algorithm runs simultaneously on multiple communicating processors, does not enjoy much success in solving hard problem instances. Divide-and-conquer decomposition strategy used to traverse the search space sometimes might find the global optimum function value, but frequently results in high computing times as the problem size increases. A hybrid version of a genetic algorithm combined with simulated annealing has proven to be most efficient.

Alrefaei and Andradottir [7] present a modified simulated annealing algorithm with a constant temperature to address discrete optimization problems and use two approaches to estimate an optimal solution to the problem. One approach estimates the optimal solution based on the state most visited versus the state last visited, while the other approach uses the best average estimated objective function value to estimate the optimal solution. Both approaches are guaranteed to converge almost surely to the set of global optimal solutions under mild conditions. They compare the performance of the modified simulated annealing algorithm to other forms of simulated annealing used to solve discrete optimization problems.

Creating effective neighborhood functions or neighborhood generation mechanisms is a critical element in designing efficient and effective simulated annealing algorithms for discrete optimization problems. Tian et al. [147] investigate the application of simulated annealing to discrete optimization problems with a permutation property, such as the traveling salesman problem, the flow shop scheduling problem, and the quadratic assignment problems. They focus on the neighborhood function of the discrete optimization problem and, in particular, the generation mechanism for the algorithm used to address the problem. They introduce six types of perturbation

schemes for generating random permutation solutions and prove that each scheme satisfies asymptotic convergence requirements. The results of the experimental evaluations on the traveling salesman problem, the flow shop scheduling problem, and the quadratic assignment problem suggest that the efficiencies of the perturbation schemes are different for each problem type and solution space. Tian et al. conclude that with the proper perturbation scheme, simulated annealing can produce efficient solutions to different discrete optimization problems that possess a permutation property.

In [5], Ahmed proposes a modification of the simulated annealing algorithm for solving discrete stochastic optimization problems where the objective function is stochastic and can be evaluated only through Monte Carlo simulation (i.e., the objective function cannot be computed exactly or such an evaluation is computationally expensive). In this modification, the temperature is held constant, and the Metropolis criterion depends on whether the objective function values indicate a statistically significant difference at each iteration. The obtained algorithm compares favorably with three previously proposed methods (see [6, 63, 71]).

Research also continues on the application of simulated annealing to optimization of continuous functions. Continuous global optimization is defined as the problem of finding points on a bounded subset of $\Re^n$ where some real-valued function f assumes its optimal (maximal or minimal) value. The application of simulated annealing to continuous optimization generally falls into two classes. The first approach closely follows the original idea presented by Kirkpatrick et al. [89], where the algorithm mimics the physical annealing process. The second approach describes the annealing process with Langevin equations, where the global minimum is found by solving a set of stochastic differential equations (see [9]). Gemen and Hwang [64] prove that continuous optimization algorithms based on Langevin equations converge to the global optima. Dekkers and Aarts [40] propose a third stochastic approach to address global optimization based on simulated annealing, which is similar to the formulation of simulated annealing applied to discrete optimization problems. They extend the mathematical formulation of simulated annealing to continuous optimization problems and prove asymptotic convergence to the set of global optima based on the equilibrium distribution of Markov chains. They also discuss an implementation of the proposed algorithm and compare its performance with other well-known algorithms on a standard set of test functions from the literature.

Tsallis and Stariolo [149] discuss and illustrate a new stochastic algorithm (generalized simulated annealing) for computationally finding the global minimum of a given (not necessarily convex) energy/cost function defined on a continuous multidimensional space. This algorithm covers, as particular cases, the so-called classical (Boltzmann machine) and fast (Cauchy machine) simulated annealings and turns out to be quicker than both. This method, which has been widely used in many fields as a global optimization tool, is composed of three parts: visiting distribution, accepting rule, and cooling schedule. The most complicated of these is the visiting distribution. Although Tsallis and Stariolo [149] did provide a heuristic algorithm to generate a random number for the visiting distribution, empirical simulations

have shown that it is inappropriate. Deng et al. [43] propose an alternative method of generating random numbers based on the results from [99]. Nishimori and Inoue [109] prove the weak ergodicity of the inhomogeneous Markov process generated by the generalized transition probability of Tsallis and Stariolo [149] under power-law decay of the temperature. Del Moral and Miclo [42] study the convergence of the generalized simulated annealing with time-inhomogeneous communication cost functions. This study is based on the use of log-Sobolev inequalities and semigroup techniques in the spirit of a previous article by one of the authors.

1.5 Summary

Simulated annealing optimization algorithms have been well studied in the literature. These algorithms can be used to optimize single as well as multi-objective optimization problems. They have been applied in various fields like process system engineering, operations research, and smart materials. Recent work on simulated annealing primarily involves ad hoc techniques, adaptive cooling schedules, and the development of hybrid algorithms.

The popularity and flexibility of simulated annealing has spawned several new annealing algorithms. Pepper et al. [117] introduce demon algorithms and test them on the traveling salesman problem. Ohlmann et al. [114] introduce another variant of simulated annealing termed compressed annealing. They incorporate the concepts of pressure and volume, in addition to temperature, to address discrete optimization problems with relaxed constraints. They also introduce a primal/dual metaheuristic by simultaneously adjusting temperature and pressure in the algorithm.

In [76], Herault presents rescaled simulated annealing, which is designed for combinatorial problems where the available computational effort is limited. This generalization performs the rescaling of the energies of the states that are candidates for a transition, before applying the Metropolis criterion. A direct consequence of this rescaling is an acceleration of the algorithm's convergence, by avoiding dives and escaping from high-energy local minima. Mingjun and Huanwen [103] propose chaos simulated annealing with chaotic initialization and chaotic sequences replacing the Gaussian distribution. These features improve the rate of convergence and are efficient and easy to implement.

Once a very popular approach to solving hard combinatorial problems, simulated annealing is now taking a backseat and giving way to new algorithms and heuristics designed to better exploit the unique properties and features of problems. However, simulated annealing continues to be widely used, given its simplicity and ease of implementation. Moreover, its simple structure is often incorporated and blended with other metaheuristics. It also remains one of the most analyzed metaheuristics, which underlines the importance and usefulness of its basic idea.

References

1. Aarts, E.H.L., Korst, J.: Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing. Wiley, Chichester (1989)
2. Aarts, E.H.L., Lenstra, J.K.: Local Search in Combinatorial Optimization. Wiley, Chichester (1997)
3. Aarts, E.H.L., van Laarhoven, P.J.M.: Statistical cooling: A general approach to combinatorial optimization problems. Phillips J. Res. **40**, 193–226 (1985)
4. Abramson, D., Krishnamoorthy, M., Dang, H.: Simulated annealing cooling schedules for the school timetabling problem. Asia-Pac. J. Oper. Res. **16**, 1–22 (1999)
5. Ahmed, M.A.: A modification of the simulated annealing algorithm for discrete stochastic optimization. Eng. Optim. **39**(6), 701–714 (2007)
6. Alkhamis, T.M., Ahmed, M.A., Tuan, V.K.: Simulated annealing for discrete optimization with estimation. Eur. J. Oper. Res. **116**, 530–544 (1999)
7. Alrefaei, M.H., Andradottir, S.: A simulated annealing algorithm with constant temperature for discrete stochastic optimization. Manage. Sci. **45**, 748–764 (1999)
8. Althofer, I., Koschnick, K.U.: On the convergence of threshold accepting. Appl. Math. Optim. **24**, 183–195 (1991)
9. Aluffi-Pentini, F., Parisi, V., Zirilli, F.: Global optimization and stochastic differential equations. J. Optim. Theory Appl. **47**, 1–16 (1985)
10. Anily, S., Federgruen, A.: Simulated annealing methods with general acceptance probabilities. J. Appl. Probab. **24**, 657–667 (1987)
11. Azizi, N., Zolfaghari, S.: Adaptive temperature control for simulated annealing: A comparative study. Comput. Oper. Res. **31**(14), 2439–2451 (2004)
12. Belisle, C.J.P.: Convergence theorems for a class of simulated annealing algorithms on RD. J. Appl. Probab. **29**, 885–895 (1992)
13. Belisle, C.J.P., Romeijn, H.E., Smith, R.L.: Hit-and-run algorithms for generating multivariate distributions. Math. Oper. Res. **18**, 255–266 (1993)
14. Ben-Ameur, W.: Computing the initial temperature of simulated annealing. Comput. Optim. Appl. **29**, 369–385 (2004)
15. Bohachevsky, I.O., Johnson, M.E., Stein, M.L.: Generalized simulated annealing for function optimization. Technometrics **28**, 209–217 (1986)
16. Borkar, V.S.: Pathwise recurrence orders and simulated annealing. J. Appl. Probab. **29**, 472–476 (1992)
17. Bouffard, V., Ferland, J.: Improving simulated annealing with variable neighborhood search to solve resource-constrained scheduling problem. J. Sched. **10**, 375–386 (2007)
18. Bratley, P., Fox, B.L., Schrage, L.: A guide to simulation. Springer, New York, NY (1987)
19. Cardoso, M.F., Salcedo, R.L., de Azevedo, S.F.: Nonequilibrium simulated annealing: A faster approach to combinatorial minimization. Ind. Eng. Chem. Res. **33**, 1908–1918 (1994)
20. Catoni, O.: Metropolis, simulated annealing, and Iterated energy transformation algorithms: Theory and experiments. J. Complex. **12**, 595–623 (1996)
21. Cerf, R.: Asymptotic convergence of genetic algorithms. Adv. Appl. Probab. **30**, 521–550 (1998)
22. Chardaire, P., Lutton, J.L., Sutter, A.: Thermostatistical persistency: A powerful improving concept for simulated annealing algorithms. Eur. J. Oper. Res. **86**, 565–579 (1995)
23. Charon, I., Hudry, O.: The noising method - a new method for combinatorial optimization. Oper. Res. Lett. **14**, 133–137 (1993)
24. Charon, I., Hudry, O.: The Noising Methods - a generalization of some metaheuristics. Eur. J. Oper. Res. **135**, 86–101 (2001)
25. Cheh, K.M., Goldberg, J.B., Askin, R.G.: A note on the effect of neighborhood-structure in simulated annealing. Comput. Oper. Res. **18**, 537–547 (1991)
26. Chen, D., Lee, C., Park, C., Mendes, P.: Parallelizing simulated annealing algorithms based on high-performance computer. J. Global Optim. **39**, 261–289 (2007)

27. Chen, S., Luk, B.L.: Adaptive simulated annealing for optimization in signal processing applications. Signal Process. **79**, 117–128 (1999)
28. Chiang, T.S., Chow, Y.S.: On the convergence rate of annealing processes. SIAM J. Control Optim. **26**, 1455–1470 (1988)
29. Chiang, T.S., Chow, Y.Y.: A limit-theorem for a class of inhomogeneous markov-processes. Ann. Probab. **17**, 1483–1502 (1989)
30. Chiang, T.S., Chow, Y.Y.: The asymptotic-behavior of simulated annealing processes with absorption. SIAM J. Control. Optim. **32**, 1247–1265 (1994)
31. Christoph, M., Hoffmann, K.H.: Scaling behavior of optimal simulated annealing schedules. J. Phys. A - Math. Gen. **26**, 3267–3277 (1993)
32. Chu, K.W., Deng, Y.F., Reinitz, J.: Parallel simulated annealing by mixing of states. J. Comput. Phys. **148**, 646–662 (1999)
33. Cinlar, E.: Introduction to Stochastic Processes. Prentice-Hall, Englewood Cliffs, NJ (1974)
34. Cohn, H., Fielding, M.: Simulated annealing: Searching for an optimal temperature schedule. SIAM J. Optim. **9**, 779–802 (1999)
35. Connors, D.P., Kumar, P.R.: Simulated annealing type markov-chains and their order balance-equations. SIAM J. Control. Optim. **27**, 1440–1461 (1989)
36. Czyzak, P., Hapke, M., Jaszkiewicz, A.: Application of the Pareto-Simulated Annealing to the Multiple Criteria Shortest Path Problem, Technical Report, Politechnika Poznanska Instytut Informatyki, Poland (1994)
37. Czyzak, P., Jaszkiewicz, A.: Pareto simulated annealing a metaheuristic technique for multiple-objective combinatorial optimization. J. Multicriteria Decis. Anal. **7**, 34–47 (1998)
38. Davis, T.E.: Toward an extrapolation of the simulated annealing convergence theory onto the simple genetic algorithm. Doctoral Dissertation, University of Florida, Gainesville, FL (1991)
39. Davis, T.E., Principe, J.C.: A simulated annealing like convergence theory for the simple genetic algorithm. Proceedings of the Fourth International Conference on Genetic Algorithms in San Diego, CA, pp. 174–181. Morgan Kaufmann, San Francisco, CA (1991)
40. Dekkers, A., Aarts, E.: Global Optimization and Simulated Annealing, Math. Program. **50**, 367–393 (1991)
41. Delport, V.: Parallel simulated annealing and evolutionary selection for combinatorial optimisation. Electron. Lett. **34**, 758–759 (1998)
42. Del Moral, P., Miclo, L.: On the convergence and applications of generalized simulated annealing. SIAM J. Control. Optim. **37**(4), 1222–1250 (1999)
43. Deng, J., Chen, H., Chang, C., Yang, Z.: A superior random number generator for visiting distribution in GSA. Int. J. Comput. Math. **81**(1), 103–120 (2004)
44. Desai, M.P.: Some results characterizing the finite time behaviour of the simulated annealing algorithm. Sadhana-Acad. Proc. Eng. Sci. **24**, 317–337 (1999)
45. Dueck, G., Scheuer, T.: Threshold accepting - a general-purpose optimization algorithm appearing superior to simulated annealing. J. Comput. Phys. **90**, 161–175 (1990)
46. Eglese, R.W.: Simulated annealing: A tool for operational research. Eur. J. Oper. Res. **46**, 271–281 (1990)
47. Emden-Weinert, T., Proksch, M.: Best practice simulated annealing for the airline crew scheduling problem. J. Heuristics **5**, 419–436 (1999)
48. Fabian, V.: Simulated annealing simulated. Comput. Math. Appl. **33**, 81–94 (1997)
49. Faigle, U., Kern, W.: Note on the convergence of simulated annealing algorithms. SIAM J. Control. Optim. **29**, 153–159 (1991)
50. Faigle, U., Kern, W.: Some convergence results for probabilistic tabu search. ORSA J. Comput. **4**, 32–37 (1992)
51. Faigle, U., Schrader, R.: On the convergence of stationary distributions in simulated annealing algorithms. Inf. Process. Lett. **27**, 189–194 (1988)
52. Faigle, U., Schrader, R.: Simulated annealing - a case-study. Angew. Inform. **30**(6), 259–263 (1988)

53. Fielding, M.: Simulated annealing with an optimal fixed temperature. SIAM J. Optim. **11**, 289–307 (2000)

54. Fleischer, M.A.: Assessing the performance of the simulated annealing algorithm using information theory. Doctoral Dissertation, Department of Operations Research, Case Western Reserve University, Clevelend, Ohio (1993)

55. Fleischer, M.A.: Simulated annealing: Past, present, and future. In: Alexopoulos, C., Kang, K., Lilegdon, W.R., Goldsman, D., (eds.) Proceedings of the 1995 Winter Simulation Conference, pp. 155–161. IEEE Press, Arlington, Virginia (1995)

56. Fleischer, M.A.: Generalized cybernetic optimization: Solving continuous variable problems, In: Voss, S., Martello, S., Roucairol, C., Ibrahim, H., Osman, I.H., (eds.) Meta-heuristics: Advances and Trends in Local Search Paradigms for Optimization, pp. 403–418. Kluwer (1999)

57. Fleischer, M.A., Jacobson, S.H.: Cybernetic optimization by simulated annealing: An implementation of parallel processing using probabilistic feedback control, In: Osman, I.H., Kelly, J.P., (eds.) Meta-heuristics: Theory and applications, pp. 249–264. Kluwer (1996)

58. Fleischer, M.A., Jacobson, S.H.: Information theory and the finite-time behavior of the simulated annealing algorithm: Experimental results. INFORMS J. Comput. **11**, 35–43 (1999)

59. Fox, B.L.: Integrating and accelerating tabu search, simulated annealing, and genetic algorithms. Ann. Oper. Res. **41**, 47–67 (1993)

60. Fox, B.L.: Random restarting versus simulated annealing. Comput. Math. Appl. **27**, 33–35 (1994)

61. Fox, B.L.: Faster Simulated Annealing. SIAM J. Optim. **5**, 485–505 (1995)

62. Fox, B.L., Heine, G.W.: Simulated Annealing with Overrides, Technical, Department of Mathematics, University of Colorado, Denver, Colorado (1993)

63. Gelfand, S.B., Mitter, S.K.: Simulated annealing with noisy or imprecise energy measurements. J. Optim. Theory. Appl. **62**, 49–62 (1989)

64. Gemen, S., Hwang, C.R.: Diffusions for global optimization. SIAM J. Control. Optim. **24**, 1031–1043 (1986)

65. Gidas, B.: Nonstationary Markov Chains and Convergence of the Annealing Algorithm, J. Stat. Phys. **39**, 73–131 (1985)

66. Glover, F.: Tabu search for nonlinear and parametric optimization (with Links to Genetic Algorithms). Discrete Appl. Math. **49**, 231–255 (1994)

67. Glover, F., Hanafi, S.: Tabu search and finite convergence. Discrete Appl. Math. **119**(1–2), 3–36 (2002)

68. Goldstein, L., Waterman, M.: Neighborhood size in the simulated annealing algorithm. Am. J. Math. Manage. Sci. **8**, 409–423 (1988)

69. Gong, G., Liu, Y., Quin, M.: An adaptive simulated annealing algorithm. Stoch. Processes. Appl. **94**, 95–103 (2001)

70. Granville, V., Krivanek, M., Rasson, J.P.: Simulated annealing - a proof of convergence. IEEE Trans. Pattern Anal. Mach. Intell. **16**, 652–656 (1994)

71. Gutjahr, W.J., Pflug, G.C.: Simulated annealing for noisy cost functions. J. Global Optim. **8**, 1–13 (1996)

72. Hajek, B.: Cooling schedules for optimal annealing. Math. Oper. Res. **13**, 311–329 (1988)

73. Hamma, B., Viitanen, S., Torn, A.: Parallel continuous simulated annealing for global optimization. Optim. Methods. Softw. **13**, 95–116 (2000)

74. Hammersley, J.M., Handscomb, D.C.: Monte Carlo Methods, Methuen, Wiley, London, New York (1964)

75. Henderson, D., Jacobson, S.H., Johnson, A.W.: Handbook of Metaheuristics, Kluwer, Boston, MA (2003)

76. Herault, L.: Rescaled simulated annealing - accelerating convergence of simulated annealing by rescaling the state energies. J. Heuristics, **6**, 215–252 (2000)

77. Hu, T.C., Kahing, A.B., Tsao, C.W.A.: Old bachelor acceptance: A new class of non-monotone threshold accepting methods. ORSA J. Comput. **7**, 417–425 (1995)

78. Isaacson, D.L., Madsen, R.W.: Markov Chains, Theory and Applications, Wiley, New York (1976)
79. Jacobson, S.H.: Analyzing the performance of local search algorithms using generalized hill climbing algorithms. In: Hansen, P., Ribeiro C.C. (eds.) Chapter 20 in Essays and Surveys on Metaheuristics, pp. 441–467. Kluwer, Norwell, MA (2002)
80. Jacobson, S.H., Sullivan, K.A., Johnson, A.W.: Discrete manufacturing process design optimization using computer simulation and generalized hill climbing algorithms. Eng. Optim. **31**, 247–260 (1998)
81. Jacobson, S.H., Yucesan, E.: Global optimization performance measures for generalized hill climbing algorithms. J. Global Optim. **29**(2), 173–190 (2004)
82. Jacobson, S.H., Yucesan, E.: Analyzing the performance of generalized hill climbing algorithms. J. Heuristics **10**(4), 387–405 (2004)
83. Jacobson, S.H., Hall, S.N., McLay, L.A., Orosz, J.E.: Performance analysis of cyclical simulated annealing algorithms. Methodol. Comput. Appl. Probab. **7**, 183–201 (2005)
84. Johnson, A.W., Jacobson, S.H.: A class of convergent generalized hill climbing algorithms. Appl. Math. Comput. **125**(2–3), 359–373 (2002a)
85. Johnson, A.W., Jacobson, S.H.: On the convergence of generalized hill climbing algorithms. Discrete Appl. Math., **119**(1–2), 37–57 (2002b)
86. Johnson, D.S., Aragon, C.R., McGeoch, L.A., Schevon, C.: Optimization by simulated annealing - an experimental evaluation; Part 1, graph partitioning. Oper. Res., **37**, 865–892 (1989)
87. Johnson, D.S., Aragon, C.R., McGeoch, L.A., Schevon, C.: Optimization by simulated annealing - an experimental evaluation; Part 2, graph-coloring and number partitioning. Ope. Res., **39**, 378–406 (1991)
88. Kiatsupaibul, S., Smith, R.L.: A General Purpose Simulated Annealing Algorithm for Integer Linear Programming, Technical Report, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan (2000)
89. Kirkpatrick, S., Gelatt, Jr., C.D., Vecchi, M.P.: Optimization by simulated annealing. Science, **220**, 671–680 (1983)
90. Koulamas, C., Antony, S.R., Jaen, R.: A survey of simulated annealing applications to operations- research problems. OMEGA-Int. J. Manage. Sci. **22**, 41–56 (1994)
91. Leite, J.P.B., Topping, B.H.V.: Parallel simulated annealing for structural optimization. Comput. Struct. **73**, 545–564 (1999)
92. Liepins, G.E., Hilliard, M.R.: Genetic algorithms: Foundations and applications. Ann. Oper. Res. **21**, 31–58 (1989)
93. Lin, C.K.Y., Haley, K.B., Sparks, C.: A comparative study of both standard and adaptive versions of threshold accepting and simulated annealing algorithms in three scheduling problems. Eur. J. Oper. Res. **83**, 330–346 (1995)
94. Locatelli, M.: Convergence properties of simulated annealing for continuous global optimization. J. Appl. Probab. **33**, 1127–1140 (1996)
95. Locatelli, M.: Simulated annealing algorithms for continuous global optimization: Convergence conditions. J. Optim. Theory. Appl. **104**, 121–133 (2000)
96. Locatelli, M.: Convergence and first hitting time of simulated annealing algorithms for continuous global optimization. Math. Methods. Oper. Res. **54**, 171–199 (2001)
97. Lundy, M., Mees, A.: Convergence of an annealing algorithm. Math. Program. **34**, 111–124 (1986)
98. Ma, J., Straub, J.E.: Simulated annealing using the classical density distribution. J. Chem. Phy. **101**, 533–541 (1994)
99. Mantegna, R.N.: Fast, accurate algorithm for numerical simulation of Levy stable stochastic processes. Phys. Rev. E, **49**(5), 4677–4683 (1994)
100. Mazza, C.: Parallel simulated annealing, Random Struct. Algorithms, **3**, 139–148 (1992)
101. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E.: Equation of state calculations by fast computing machines. J. Chem. Phys., **21**, 1087–1092 (1953)

102. Meyer, C.D.: The condition of a finite markov chain and perturbation bounds for the limiting probabilities. SIAM J. Algebraic. Discrete Methods **1**, 273–283 (1980)
103. Mingjun, J., Huanwen, T.: Application of chaos in simulated annealing. Chaos, Solitions. Fractals **21**, 933–941 (2003)
104. Mitra, D., Romeo, F., Sangiovanni-Vincentelli, A.L.: Convergence and finite time behavior of simulated annealing. Adv. Appl. Probab. **18**, 747–771 (1986)
105. Moscato, P.: An introduction to population approaches for optimization and hierarchical objective functions: A discussion on the role of tabu search. Ann. Oper. Res. **41**, 85–121 (1993)
106. Moscato, P., Fontanari, J.F.: Convergence and finite-time behavior of simulated annealing. Adv. Appl. Probab. **18**, 747–771 (1990)
107. Muhlenbein, H.: Genetic algorithms, In: Aarts, E., Lenstra, J.K., (eds.) Local search in combinatorial optimization, pp. 137–172. Wiley, New York, NY (1997)
108. Munakata, T., Nakamura, Y.: Temperature control for simulated annealing. Phys. Rev. E Stat. Nonlin. and Soft Matter Phys. **64**(4II), 461271–461275 (2001)
109. Nishimori, H., Inoue, J.: Convergence of simulated annealing using the generalized transition probability. J. Phys. A, **31**, 5661–5672 (1998)
110. Nissen, V., Paul, H.: A modification of threshold accepting and its application to the quadratic assignment problem. OR Spektrum **17**, 205–210 (1995)
111. Nolte, A., Schrader R.: A note on finite time behavior of simulated annealing. Math. Oper. Res. **25**(3), 476–484 (2000)
112. Nourani, Y., Andresen, B.: A comparison of simulated annealing cooling strategies. J. Phys. A-Math. Gen. **31**, 8373–8385 (1998)
113. Ogbu, F.A., Smith, D.K.: The application of the simulated annealing algorithm to the solution of the N/M/Cmax flowshop problem. Comput. Oper. Res. **17**, 243–253 (1990)
114. Ohlmann, J.W., Bean, J.C., Henderson, S.G.: Convergence in probability of compressed annealing. Math. Oper. Res. **29**(4), 837–860 (2004)
115. Orosz, J.E., Jacobson, S.H.: Finite-time performance analysis of static simulated annealing algorithms. Comput. Optim. Appl. **21**, 21–53 (2002a)
116. Orosz, J.E., Jacobson, S.H.: Analysis of static simulated annealing algorithms. J. Optim. Theory. Appl. **115**(1), 165–182 (2002b)
117. Pepper, J.W., Golden, B.L., Wasil, E.A.: Solving the traveling salesman problem with annealing-based Heuristics: A computational study. IEEE Trans. Syst. Manufacturing and Cybernetics, Part A: Syst. Humans, **32**(1), 72–77 (2002)
118. Rajasekaran, S.: On simulated annealing and nested annealing. J. Global Optim. **16**, 43–56 (2000)
119. Romeijn, H.E., Zabinsky, Z.B., Graesser, D.L., Noegi, S.: New reflection generator for simulated annealing in mixed-integer/continuous global optimization. J. Optim. Theory. Appl. **101**, 403–427 (1999)
120. Romeo, F., Sangiovanni-Vincentelli, A.: A theoretical framework for simulated annealing. Algorithmica **6**, 302–345 (1991)
121. Rosenthal, J.S.: Convergence rates for markov chains. SIAM Rev. **37**, 387–405 (1995)
122. Ross, S.M.: Stochastic processes, J Wiley, New York, NY (1996)
123. Rossier, Y., Troyon, M., Liebling, T.M.: Probabilistic exchange algorithms and euclidean traveling salesman problems. OR Spektrum **8**, 151–164 (1986)
124. Rudolph, G.: Convergence analysis of cononical genetic algorithms. IEEE Trans. Neural Net. Special Issue on Evolutional Computing, **5**, 96–101 (1994)
125. Scheermesser, T., Bryngdahl, O.: Threshold accepting for constrained half-toning. Opt. Commun. **115**, 13–18 (1995)
126. Schuur, P.C.: Classification of acceptance criteria for the simulated annealing algorithm. Math. Oper. Res. **22**, 266–275 (1997)
127. Seneta, E.: Non-Negative Matrices and Markov Chains, Springer, New York, NY (1981)

128. Serafini, P.: Mathematics of Multiobjective Optimization, p. 289. CISM Courses and Lectures, Springer, Berlin (1985)
129. Serafini, P.: Simulated Annealing for Multiple Objective Optimization Problems, Proceedings of the Tenth International Conference on Multiple Criteria Decision Making, pp. 87–96, Taipei (1992)
130. Serafini, P.: Simulated Annealing for Multiple Objective Optimization Problems, Multiple Criteria Decision Making. Expand and Enrich the Domains of Thinking and Application pp. 283–292, Springer, Berlin, (1994)
131. Siarry, P., Berthiau, G., Durbin, F., Haussy, J.: Enhanced simulated annealing for globally minimizing functions of many-continuous variables. ACM Trans. Math. Softw. **23**, 209–228 (1997)
132. Solla, S.A., Sorkin, G.B., White, S.R.: Configuration space analysis for optimization problems. In: Bienenstock, E., Fogelmansoulie, F., Weisbuch, G. (eds.) Disordered Systems and Biological Organization, pp. 283–292. Springer, New York (1986)
133. Srichander, R.: Efficient schedules for simulated annealing. Eng. Optim. **24**, 161–176 (1995)
134. Steinhofel, K., Albrecht, A., Wong, C.K.: The convergence of stochastic algorithms solving flow shop scheduling. Theor. Comput. Sci. **285**, 101–117 (2002)
135. Stern, J.M.: Simulated annealing with a temperature dependent penalty function, ORSA J. Comput. **4**, 311–319 (1992)
136. Storer, R.H., Wu, S.D., Vaccari, R.: New Search Spaces for Sequencing Problems with Application to Job Shop Scheduling, Manage. Sci. **38**, 1495–1509 (1992)
137. Straub, J.E., Ma, J., Amara, P.: Simulated annealing using coarse grained classical dynamics: Smouuchowski Dynamics in the Gaussian Density Approximation. J. Chem. Phys. **103**, 1574–1581 (1995)
138. Strenski, P.N., Kirkpatrick, S.: Analysis of finite length annealing schedules. Algorithmica, **6**, 346–366 (1991)
139. Sullivan, K.A., Jacobson, S.H.: Ordinal hill climbing algorithms for discrete manufacturing process design optimization problems. Discrete Event Dyn. Syst. **10**, 307–324 (2000)
140. Sullivan, K.A., Jacobson, S.H.: A convergence analysis of generalized hill climbing algorithms. IEEE Trans. Automatic Control **46**, 1288–1293 (2001)
141. Suman, B.: Multiobjective simulated annealing a metaheuristic technique for multiobjective optimization of a constrained problem. Found. Comput. Decis. Sci., **27**, 171–191 (2002)
142. Suman, B.: Simulated annealing based multiobjective algorithm and their application for system reliability. Eng. Optim., **35**, 391–416 (2003)
143. Suman, B.: Self-stopping PDMOSA and performance measure in simulated annealing based multiobjective optimization algorithms. Comput. Chem. Eng. **29**, 1131–1147 (2005)
144. Suman, B., Kumar, P.: A survey of simulated annealing as a tool for single and multiobjective optimization. J. Oper. Res. Soc. **57**, 1143–1160 (2006)
145. Suppapitnarm, A., Parks, T.: Simulated Annealing: An Alternative Approach to True Multiobjective Optimization, Genetic and Evolutionary Computation Conference, Conference Workshop Program pp. 406–407, Orlando, FL (1999)
146. Tekinalp, O., Karsli, G.: A new multiobjective simulated annealing algorithm. J. Global Optim. **39**, 49–77 (2007)
147. Tian, P., Ma, J., Zhang, D.M.: Application of the simulated annealing algorithm to the combinatorial optimisation problem with permutation property: An investigation of generation mechanism. Eur. J. Oper. Res. **118**, 81–94 (1999)
148. Tovey, C.A.: Simulated simulated annealing. Am. J. Math. Manage. Sci., **8**, 389–407 (1988)
149. Tsallis, C., Stariolo, D.A.: Generalized simulated annealing. Physica A, **233**, 395–406 (1996)
150. Tsitsiklis, J.N.: Markov chains with rare transitions and simulated annealing. Math. Oper. Res. **14**, 70–90 (1989)
151. Triki, E., Collette, Y., Siarry, P.: A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. Eur. J. Oper. Res. **166**, 77–92 (2005)

152. Ulungu, L.E., Teghem, J.: Multiobjective combinatorial optimization problems: A survey. J. Multicriteria Decis. Anal. **3**, 83–104 (1994)
153. Ulungu, L.E., Teghem, J., Ost, C.: Interactive simulated annealing in a multiobjective framework: Application to an industrial problem. J. Oper. Res. Soc. **49**, 1044–1050 (1998)
154. Ulungu, L.E., Teghem, J., Fortemps, P.H., Tuyttens, D.: MOSA method: A tool for solving multiobjective combinatorial optimization problems. J. Multicriteria Decis. Anal., **8**, 221–236 (1999)
155. van Laarhoven, P.J.M.: Theoretical and Computational Aspects of Simulated Annealing, Centrum voor Wiskunde en Informatica, Amsterdam, Netherlands (1988)
156. van Laarhoven, P.J.M., Aarts, E.H.L.: Simulated annealing: Theory and applications, D. Reidel; Kluwer, Dordrecht, Boston, Norwell, MA (1987)
157. Varanelli, J.M., Cohoon, J.P.: A fast method for generalized starting temperature determination in homogeneous two-stage simulated annealing systems. Comput. Oper. Res. **26**, 481–503 (1999)
158. Vaughan, D., Jacobson, S.H.: Tabu guided generalized hill climbing algorithms. Methodol. Comput. Appl. Probab. **6**, 343–354 (2004)
159. Villalobos-Arias, M., Coello, C.A.C., Hernandez-Lerma, O.: Foundations of genetic algorithms. Lecture Notes in Comput. Sci. 3469, 95–111 (2005)
160. Villalobos-Arias, M., Coello, C.A.C., Hernandez-Lerma, O.: Asymptotic Convergence of a Simulated Annealing Algorithm for Multiobjective Optimization Problems, Math. Methods. Oper. Res., **64**, 353–362 (2006)
161. Wood, G.R., Alexander, D.L.J., Bulger, D.W.: J. Global Optim., **22**, 271–284 (2002)
162. Yan, D., Mukai, H.: Stochastic discrete optimization. SIAM J. Control Optim., **30**, 594–612 (1992)
163. Yang, R.L.: Convergence of the simulated annealing algorithm for continuous global optimization. J. Optim. Theory. Appl. **104**, 691–716 (2000)
164. Yao, X.: A new simulated annealing algorithm. Int. J. Comput. Math. **56**, 161–168 (1995)
165. Yao, X., Li, G.: General simulated annealing. J. Comput. Sci. Tech. **6**, 329–338 (1991)
166. Zabinsky, Z.B., Smith, R.L., McDonald, J.F., Romeijn, H.E., Kaufman, D.E.: Improving hit-and-run for global optimization. J. Global Optim. **3**, 171–192 (1993)
167. Zolfaghari, S., Liang, M.: Comparative study of simulated annealing, genetic algorithms and tabu Search for solving binary and comprehensive machine-grouping problems. Int. J. Prod. Resour. **40**(9), 2141–2158 (2002)