Review Article

Multiobjective Simulated Annealing: Principles and Algorithm Variants

Khalil Amine [ID]

MOAD6 Research Team, MASI Laboratory, E3S Research Centre, Mohammadia School of Engineering,
Mohammed V University of Rabat, Morocco

Correspondence should be addressed to Khalil Amine; kh.amine.sbai@gmail.com

Simulated annealing is a stochastic local search method, initially introduced for global combinatorial mono-objective optimisation problems, allowing gradual convergence to a near-optimal solution. An extended version for multiobjective optimisation has been introduced to allow a construction of near-Pareto optimal solutions by means of an archive that catches nondominated solutions while exploring the feasible domain. Although simulated annealing provides a balance between the exploration and the exploitation, multiobjective optimisation problems require a special design to achieve this balance due to many factors including the number of objective functions. Accordingly, many variants of multiobjective simulated annealing have been introduced in the literature. This paper reviews the state of the art of simulated annealing algorithm with a focus upon multiobjective optimisation field.

1. Introduction

Simulated annealing is a probabilistic local search method for global combinatorial optimisation problems that allows gradual convergence to a near-optimal solution. It consists of a sequence of moves from a current solution to a better one according to certain transition rules while accepting occasionally some uphill solutions in order to guarantee a diversity in the domain exploration and to avoid getting caught at local optima. The process is managed by a certain cooling schedule that controls the number of iterations. Simulated annealing has shown robustness and success for many applications and has received significant attention since its introduction. Consequently, multiobjective simulated annealing has been introduced with the aim to allow a construction of near-Pareto optimal solutions by means of an archive in which nondominated solutions are gradually gathered while exploring the feasible domain. Although simulated annealing provides a balance between the exploration and the exploitation, multiobjective optimisation problems require a special design to achieve this balance due to many factors including the number of objective functions. Accordingly, many variants of multiobjective simulated annealing have been introduced in the literature.

This paper reviews and discusses simulated annealing algorithm for multiobjective optimisation and delineates some algorithm variants.

2. Overview on Mono-Objective Simulated Annealing

Simulated annealing is a meta-heuristic that dates back to the works of Kirkpatrick et al. [1] and Černý [2] having shown that the Metropolis algorithm [3] (an algorithm of statistical physics that consists in constructing a sequence of Markov chains for sampling from a probability distribution. The algorithm is often used under an extended version called Metropolis-Hastings algorithm.) can be used to obtain optimal solution of optimisation problems by considering the energetic state as objective function and thermodynamic equilibrium states as local optima. Simulated annealing mimics the metallurgical process of careful annealing that consists in cooling a heated metal or alloy until reaching the most solid state called the ground state. Simulated annealing is considered as an extension of the hill climbing algorithm which consists of a sequence of transitions across solutions while improving a certain energetic objective function at
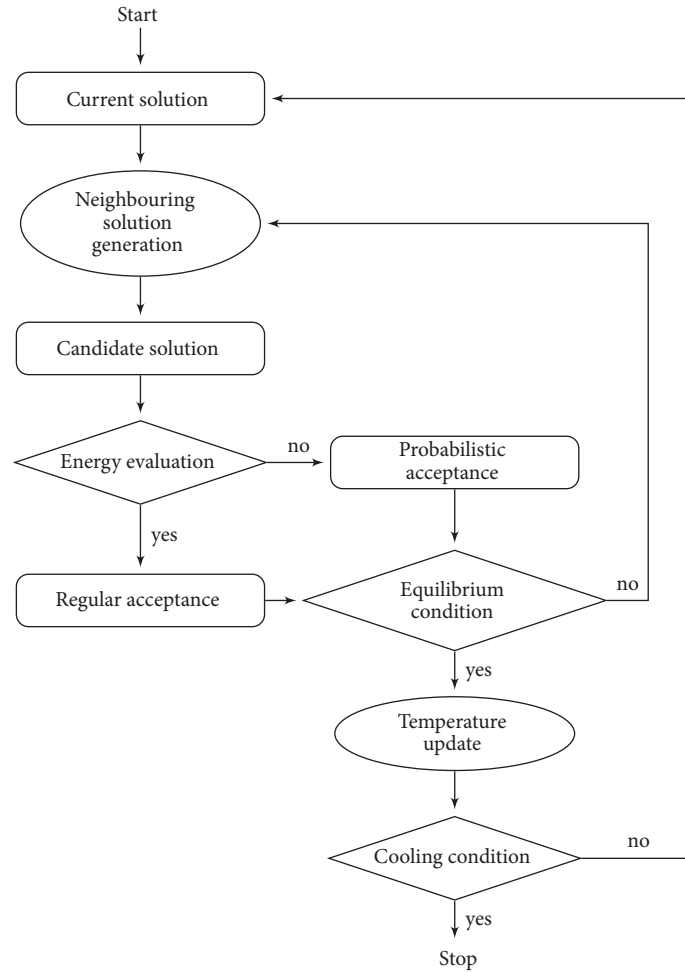
FIGURE 1: Flowchart of the mono-objective simulated annealing algorithm.

each iteration until reaching the global optimum. Simulated annealing has introduced an occasional acceptance mechanism of uphill solutions in order to explore the feasible domain intensively and to avoid getting caught at local optima. The process is managed by a certain cooling schedule that controls the temperature variation during the algorithm process. Figure 1 presents the generic simulated annealing algorithm flowchart.

The generic simulated annealing algorithm consists of two nested loops. Given a current solution and a fixed temperature, the inner loop consists, at each iteration, in generating a candidate neighbouring solution that will undergo an energy evaluation to decide whether to accept it as current. Occasionally, some nonimproving solutions are accepted according to a certain probabilistic rule. The loop is controlled by an equilibrium condition that limits the number of iterations at every temperature level. It refers to a thermal equilibrium when the algorithm is not expected to find more improving solutions within the current explored neighbourhood. The outer loop consists in decreasing the temperature level according to a certain cooling scheme. The temperature is supposed to be iteratively decreased until reaching a cooling condition that often corresponds to a final

temperature due. The algorithm involves therefore two main considerations, namely the neighbourhood structure and the cooling schedule.

Neighbourhood structure refers to a set of conceptual considerations regarding both the representation and the generation mechanism of neighbouring solutions. It is supposed to be proper to the problem under consideration and to provide a restricted number of possible transitions. For ease handling, solutions of optimisation problems are usually considered in a certain indirect representation that consists of a finite number of components within a certain configuration. Thus, two solutions are neighbours if their configurations are similar; that is, the two configurations differ in few components only or may have close energy values. Neighbourhood structure consists rather in defining the candidate solutions that the algorithm may move to at each iteration. Even often underestimated in early studies [4, 5], the neighbourhood structure choice has a significant impact on the neighbourhood size and even on the simulated annealing performance. Moreover, it may affect the concept of local optimality itself [6, 7], thereby identifying a local optimum for every neighbourhood. Accordingly, the algorithm

TABLE 1: Summary of classical cooling schemes.

Denotation	General functional form	Iterative formula	Authors
Linear	$T_k = T_0 - k\beta$	$T_k = T_{k-1} - \beta$	Strenski and Kirkpatrick [9]
Geometric	$T_k = \alpha^k T_0$	$T_k = \alpha T_{k-1}$	Kirkpatrick et al. [1]
Logarithmic	$T_k = \dfrac{T_0}{\ln(k+1)}$	$T_k = \dfrac{\ln(k)}{\ln(k+1)} T_{k-1}$	Geman and Geman [10]
Hybrid	$T_k = \begin{cases} \dfrac{T_0}{1+k} & \text{if } k \leq \beta \\ \alpha^k T_0 & \text{if } k > \beta \end{cases}$	$T_k = \begin{cases} \dfrac{k}{k+1} T_{k-1} & \text{if } k \leq \beta \\ \alpha T_{k-1} & \text{if } k > \beta \end{cases}$	Sekihara et al. [11]
Exponential	$T_k = \dfrac{T_0}{1 + k\beta T_0}$	$T_k = \dfrac{T_{k-1}}{1 + \beta T_{k-1}}$	Lundy and Mees [12]
Adaptive		$T_k = \phi(S_c) T_{k-1}$	Ingber [13]

The table notations are summarised as follows:

T_0 is the initial temperature value.

T_k is the temperature value for the kth iteration.

α and β are two constants such that $\alpha \in [0,1]$ and $\beta > 0$.

ϕ is a function depending on the candidate solution S_c that takes a value less than 1 if S_c improves the objective function in comparison to the best solution found so far and greater than 1 otherwise.

purpose is to hopefully reach, at each temperature level, the local optimum before reaching the equilibrium condition.

The cooling schedule controls the temperature decrease in the course of the algorithm. It consists of four entities, namely, an initial temperature, a final temperature, an equilibrium condition, and a cooling scheme that consists of a decreasing sequence of temperature. Whilst the final temperature is supposed to be preset to zero or around, the choice of the other parameters is rather challenging and strongly related to the case under investigation; no specific rule does exist for all situations and preliminary experiments are often required. The equilibrium condition is a parameter that limits the number of iterations at every temperature level. It could be merely preset to a certain upper bound on that number, plausibly proportional to the current solution neighbourhood cardinality. In order to utterly mimic the careful annealing, thereby receiving satisfactory solutions, any parameters choice should ensure that the process starts at a sufficiently high temperature value and terminates at sufficiently low temperature value according to a sufficiently slow cooling scheme.

The choice of an appropriate cooling scheme is crucial in order to ensure success of the algorithm. A wide range of cooling schemes have been introduced in the literature that are either monotonic or adaptive. A cooling scheme is a function of iteration index that depends more often on the previous temperature value and the initial temperature. Monotonic schemes consist of static decrease of temperature at every iteration independently of the quality of the found solutions nor the current explored neighbourhood structure, whereas adaptive cooling schemes involve a mechanism of decreasing the temperature in accordance to the quality of transitions. Therefore, more satisfactory moves occur more the decrease hop is larger. This would imply some reannealing occurrences allowing more diversification and intensification. A summary of classical schemes is presented in Table 1. Nevertheless, Triki et al. [8] have demonstrated based on an empirical study that almost all classical cooling schemes are equivalent; that is, they can be tuned to result in similar decrease of the temperature during the course of the algorithm.

During the course of the algorithm, two kinds of acceptance are involved, namely, a regular acceptance corresponding to the case when an improving solution is found, and a probabilistic acceptance corresponding to the case when a nonimproving solution is found that would imply diversification and escape from local optima. Given a candidate solution S_c, the acceptance of the candidate move to S_c is expressed by means of the Metropolis rule given by the following formula:

$$P(\text{Accept } S_c) = \min\left(1; \exp\left(-\frac{\Delta E}{kT}\right)\right) \tag{1}$$

where ΔE is the move energetic variation and T is the current temperature. k is called Boltzmann constant and often assumed to be equal to 1. This rule tolerates frequent acceptance occurrences of uphill solutions at high temperature levels. That is, almost all candidate solutions would be accepted if the temperature is high, whereas only improving solutions would be accepted if the temperature is low. Furthermore, given that the exponential function is increasing, the acceptance rule favours small deteriorations than large ones. The algorithm is supposed to stop when one of the two following cases is due: the temperature level attains the final temperature or either the process results on a solution for which no improving solution could be found within a reasonable number of annealing cases. A criterion responsible for dealing with both cases is called a cooling condition.

Simulated annealing has shown significant success for many applications and has received significant attention since its introduction. Moreover, it is problem-dependent and the choice of appropriate parameters is rather a challenging task. More detailed review can be found in [14–19].

3. Multiobjective Simulated Annealing

3.1. Multiobjective Optimisation and Metaheuristics. Multiobjective optimisation, also called multicriteria or multiattribute optimisation, deals with problems involving more than one objective function to be simultaneously optimised. The objective functions under consideration are often in contradiction (or conflicting) in such a way no improvement on any objective function is possible without any deterioration in any one of the others—otherwise the problem is said to be trivial and is reduced to a mono-objective optimisation problem. Given m objective functions $E_i : \mathbb{R}^n \longrightarrow \mathbb{R}$ for $i = 1, \ldots, m$, a multiobjective optimisation problem (often said to be bi-objective if $m = 2$ and many-objective if $m \geq 4$) can generally be formulated as

$$\text{minimise} \quad (E_1(x), \ldots, E_m(x))$$
$$\text{subject to} \quad x \in D \tag{MOP}$$

where $D \subset \mathbb{R}^n$, called the feasible domain, is a set of equalities and inequalities of the decision vector variable x. The optimality concept in multiobjective optimisation is based on the dominance binary relation which serves as a comparison function between feasible solutions. A solution is said to be dominating another one if the former improves at least one objective function value compared to the latter without any deterioration in the other objective function values. Given x and y two feasible solutions of (MOP), that is $x, y \in D$, the following alternatives are possible:

(i) $x \prec y$; that is, x dominates y, if $E_i(x) \leq E_i(x)$ for all $i = 1, \ldots, m$ and $(E_1(x), \ldots, E_m(x)) \neq (E_1(y), \ldots, E_m(y))$

(ii) $x \sim y$; that is, x and y are equivalent, if $E_i(x) = E_i(y)$ for all $i = 1, \ldots, m$

(iii) $x \, ? \, y$; that is, x and y are noncomparable, if there exist i and j such that $E_i(x) < E_i(y)$ and $E_j(x) > E_j(y)$

A nondominated solution is defined then as one for which no feasible solution exists that dominates it—it corresponds to the best trade-off between all considered objective functions. Compared to mono-objective optimisation, the optimality is typically attained by means of a set of nondominated solutions, called Pareto set, rather than a single solution. Pareto set is usually referred to by means of its image, called Pareto front or Pareto frontier, which serves as a graphical representation of dimension $m - 1$ of optimality in the objective space, that is, the space constituted by objective functions. Each Pareto optimal solution is said to refer to a preference structure or either a utility function. Furthermore, the Pareto solutions are said to be noncomparable; that is, no preference can be stated between any pair of them. Despite the mathematical significance of this concept of optimality, a wide range of investigations have been deemed in economics giving rise to many advances in the utility theory that deal with preference structures in accordance with the users (customers) satisfaction. Nevertheless, finding all the optimal solutions for a multiobjective optimisation problem is rather

challenging and difficult, at least in comparison with mono-objective optimisation case.

A variety of techniques have been addressed in the literature to deal with multiobjective optimisation that can be classified into three approaches: a priori, a posteriori, and progressive. An a priori approach consists of a transformation of the mathematical multiobjective model into a mono-objective one. Many methods have been introduced from this perspective, namely, goal programming, aggregation or scalarisation, and lexicographical, which return typically one (Pareto) optimal solution of the original model. An a posteriori approach consists of a construction of the Pareto set. Yet, such a construction is usually difficult and computationally consuming. Thus, approximation algorithms are plausibly suggested in this case. They consist in constructing a subset of the Pareto set, or otherwise a set of efficiently nondominated (or near-Pareto optimal) solutions, that is, feasible solutions that are not dominated by any solution that can be alternatively computed in reasonable computational cost. The construction of the actual or an approximating Pareto set is sometimes referred to as Pareto optimisation. As to progressive approach, also called interactive approach, it is a hybridisation of the two aforementioned approaches where the decision-maker provides a kind of guidance to the algorithm execution. It consists actually of an a priori approach that involves an a posteriori partial result learning and preference defining as long as the algorithm progresses. More details about progressive optimisation can be found in [20, 21].

Due to the success of metaheuristics in mono-objective optimisation, especially for combinatorial problems where information about the problem (in terms of the feasible domain, called in this context the search space) is often accessible, metaheuristics have been widely adapted to cope with multiobjective combinatorial problems as well. Contrary to the constructive techniques, metaheuristics are solution-based allowing, therefore, a manageable compromise between the quality of solution (efficiency) and the execution time (number of iterations). Moreover, metaheuristics are expected to bring a generic and adaptable framework for a wide range of problems. Vector evaluated genetic algorithm [22] has been the first multiobjective metaheuristic proposed in the literature consisting of an adaptation of the genetic algorithm to the multiobjective optimisation case. Thenceforth many other multiobjective metaheuristics have been developed that include multiobjective Tabu search [23], multiple ant colony system for vehicle routing problems with time windows [24], multi-objective simulated annealing [25], and multiobjective scatter search [26].

Metaheuristics adaptations have been developed into two main paradigms: one run and multiruns. One-run-based techniques consist in adapting the original metaheuristic to return a set of solutions at only one execution. One of the important advantages of this paradigm is that it follows the main principle of metaheuristics consisting of dealing with a solution at each iteration, rather than a solution component, which refers in the multiobjective optimisation case to a set of solutions with a certain level of efficiency. Yet, the number of returned solutions is rather uncontrollable. Indeed, if the

potential near-optimal solutions are archived progressively, each solution returned should undergo an evaluation phase where it is compared to the previous archived potential near-optimal solutions, in order to eliminate any dominated one. If the near-optimal solutions are returned unsteadily at the algorithm termination, for example, for population-based meta-heuristics like genetic algorithms, the solutions are compared to each other at this stage.

Inspired from the a priori multiobjective approach, multiruns-based techniques consist in applying the original metaheuristic on a certain aggregation of the objective functions under consideration. At each run, the metaheuristic is expected to reach one near-optimal solution of the problem that satisfies a certain preference structure. The algorithm is assumed to be executed as many times as the required number of near-optimal solutions. Unfortunately, there is no guarantee that two different preference structures lead to two distinct near-optimal solutions. Once again, the gathered solutions are compared to each other to eliminate any dominated one. Nevertheless, the near-optimal solutions found should be diverse, that is, dispersed on the Pareto front. Das and Dennis [27] showed that evenly spread solutions are not guaranteed even with an evenly distributed set of weights. Thus, although the concept that multiruns-based techniques follow is quite simple, determining effective and pertinent preference structures in advance is not straightforward at least for a whole range of multiobjective problems.

3.2. Principles of Multiobjective Simulated Annealing.

Multiobjective simulated annealing originates from the works of Serafini [28] where many probabilistic acceptance rules have been designed and discussed with the aim at increasing the probability of accepting nondominated solutions, that is, solutions nondominated by any generated solution so far. Two alternative approaches have been proposed in consequence. Given m objective functions E_i assigned to m scalar-valued weights ω_i for $i = 1, \ldots, m$, and a candidate solution S_c, the first approach consists in accepting with certainty only improving solutions, thereby accepting with probability less than 1 any other solution. This strong acceptance rule allows deep exploration of the feasible domain and has been expressed as follows:

$$P\left(\text{Accept } S_c\right) = \prod_{i=1}^{m} \min\left(1; \exp\left(\frac{\omega_i \Delta E_i}{T}\right)\right) \quad (2)$$

The second approach consists in accepting with certainty any either dominating or noncomparable solution, allowing diversity in the exploration of the feasible domain. This weak acceptance rule is expressed in the following form:

$$P\left(\text{Accept } S_c\right) = \min\left(1; \max_{i=1,\ldots,m}\left(\exp\left(\frac{\omega_i \Delta E_i}{T}\right)\right)\right) \quad (3)$$

The two alternatives lead to quite different Markov chains for which nondominated solutions were proved to have higher stationary probability. A composite acceptance rule has been proposed consisting of the following formula:

$$\begin{aligned} P&\left(\text{Accept } S_c\right) \\ &= \alpha\prod_{i=1}^{m} \min\left(1; \exp\left(\frac{\omega_i \Delta E_i}{T}\right)\right) \\ &\quad + (1-\alpha) \min\left(1; \max_{i=1,\ldots,m}\left(\exp\left(\frac{\omega_i \Delta E_i}{T}\right)\right)\right) \end{aligned} \quad (4)$$

where $\alpha \in [0,1]$ and T/ω_i is considered as a temperature T_i related to each objective function E_i, that is, like each objective function has its own annealing scheme. Besides, each considered occurrence of weights $(\omega_i)_{i=1,\ldots,m}$ is assumed to correspond to a certain preference structure. In order to guarantee entire exploration of the Pareto set, a slow random variation of the weights has been proposed in consequence at every iteration.

3.3. MOSA Methods.

Multiobjective Simulated Annealing method (MOSA) is a class of simulated annealing extensions to multiobjective optimisation exploiting the idea of constructing an estimated Pareto front by gathering nondominated solutions found while exploring the feasible domain. An archive is considered that serves for maintenance of those efficient solutions. Accordingly, many multiobjective simulated annealing paradigms have been proposed in the literature [25, 29–34] that have in common the suggestion of considering a variation of a certain composite energy often as a linear combination of the considered objective functions. Under certain choice of acceptance probabilities, asymptotic convergence to the Pareto solutions set has been proved [35].

Ulungu et al. [25, 36] have developed a MOSA method that involves a potentially efficient solutions list that catches non-dominated solutions; it contains all the generated solutions which are not dominated by any other generated solution so far. The method uses weighted functions for measuring the quality of transitions. Each scalarising function would induce a privileged search direction towards a relative optimal solution. If that optimal solution is not reached, some near-optimal solutions would be considered. The use of a wide diversified set of weights has been proposed with the aim to cover all the efficient front. The potentially efficient solutions list is assumed to be updated after every occurrence of a new solution acceptance, at which any dominated solution is removed from the list. If no dominated solution exists, the potentially efficient solutions list increases in cardinality.

Czyżak and Jaszkiewicz [30] have proposed a Pareto Simulated Annealing (PSA) procedure using a weak acceptance criterion where any solution not dominated by the current solution could be accepted. Inspired from genetic algorithms, PSA is a population-based metaheuristic that considers at each temperature a set of (sample) generated solutions to be hopefully improved. Each solution from the sample is improved in such a way the new accepted solution should be distant from the closest solution to the former solution. This approach is performed by increasing weights of objectives according to which the closest solution is better than the

Input: A cooling schedule τ; a starting temperature $T \longleftarrow T_0$; a starting sample of generated solutions $\mathcal{S}$;
   and an initial memory $\mathcal{M} \longleftarrow \mathcal{S}$
Output: The archive $\mathcal{M}$ representing an approximation of the Pareto solutions set
 1: **repeat**
 2: **for each** $S_c \in \mathcal{S}$ **do**
 3:   **repeat**
 4:    Construct a neighbouring solution S_{new}
 5:    **if** S_{new} is not dominated by S_c **then**
 6:     Update $\mathcal{M}$ with S_{new}
 7:     Select S_{cl} (if exists) the closest solution to S_c
 8:     Update weights of objectives in accordance with S_c and S_{new} partial dominance
 9:    **else**
 10:     Accept S_{new} with certain probability
 11:    **end if**
 12:   **until** Equilibrium condition
 13: **end for**
 14: Decrease temperature T
 15: **until** Cooling condition
 16: **return** $\mathcal{M}$

ALGORITHM 1: PSA procedure.

current solution, thereby decreasing the weights of objectives according to which the closest solution is better than the current solution. The new weights combination will be used for the evaluation step in the next iteration as well as for the probabilistic acceptance. The PSA procedure is summarised in Algorithm 1.

PSA attempts to keep a kind of uniformity while improving the sample of generated solutions. That is, each sequence of improving solutions related to a certain solution from the starting sample should be moved away from the other sequences of improving solutions related to the other elements of the starting sample. This would hopefully suggest the sample cardinality conservation and thus allowing to control the number of the returned efficient solutions. Furthermore, the fact that PSA deals, at each temperature, with a sample of potential solutions, allows parallelisation. As shown in Figure 2, when PSA is parallel, all the sample elements are updated at the same time with the improving solutions of all parallel annealing sequences. Baños et al. [37] have investigated four alternative parallelisations of PSA in terms of quality of the solutions and execution time. Moreover, extensions to the fuzzy multiobjective combinatorial optimisation [38] and stochastic multiobjective combinatorial optimisation [39] have been addressed in the literature.

With the same idea of using a population of solutions as a starting archive, Li and Landa-Silva [40, 41] have proposed an adaptive evolutionary multiobjective approach combined with simulated annealing (EMOSA). It is a hybridisation of the multiobjective evolutionary algorithm based on decomposition (MOEA/D) proposed by Zhang and Li [42], and simulated annealing metaheuristic. The proposed method consists in considering a set of starting solutions that is assumed to be evenly spread and where each solution corresponds to an initial solution of a certain subproblem, that is, a problem corresponding to an aggregation objective function. Solutions evaluation is performed with respect
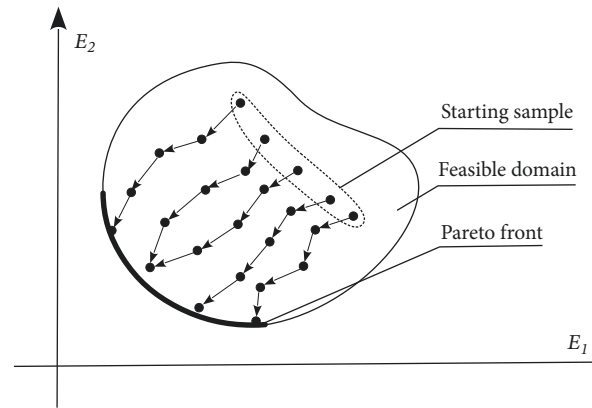

FIGURE 2: Hypothetical improvement pattern of parallel annealing chains in parallel PSA procedure.

to both weighted aggregation and Pareto dominance based on the concept of ϵ-dominance introduced by Laumanns et al. [43]: the regular acceptance of candidate solutions is performed in accordance to the improvement of the corresponding aggregation function, while the update of the nondominated solutions archive is performed using the ϵ-dominance. Besides, for each aggregation function, weights are adaptively modified at the lowest temperature in order to assure search diversify.

Suppapitnarm et al. [32] proposed an extension to a previous work conducted by Engrand [44] which consists in using a logarithmic composite objective function:

$$\sum_i \ln(E_i) = \ln\left(\prod_i E_i\right) \tag{5}$$

for a given number of objective functions E_i, and an acceptance probability formulation for a generated candidate
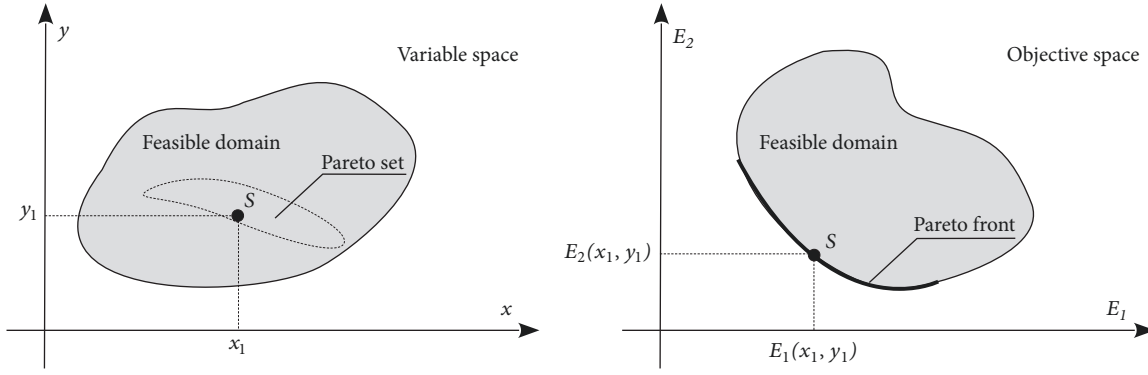
FIGURE 3: Hypothetical correspondence between variable space and objective space for a bi-objective minimisation problem.

solution S_c in reference to a given current solution S_{cur} defined as follows:

$$P\left(\text{Accept } S_c\right) = \exp\left(-\frac{1}{T}\sum_i \ln\left(\frac{E_i\left(S_c\right)}{E_i\left(S_{cur}\right)}\right)\right) \qquad (6)$$

The lack of any weight assigned to objectives, in addition to the multiplicative scheme of the considered composite objective, would suggest that all objectives have the same magnitude. This implies an acceptance issue in both regular and probabilistic modes. Indeed, any weak dominating or noncomparable solution is supposed to be accepted. However, the magnitude order of some objectives could affect the acceptance in a way that favours some objectives over others. Similarly, the probabilistic acceptance depends on the relative change of all the objectives, thereby influenced by the objectives with high magnitude order. Suppapitnarm et al. have proposed to not use any aggregation objective function for candidate solutions evaluation so that any decrease in any objective will systematically result in regular acceptance. As to the probabilistic acceptance, a probability rule that involves a temperature parameter for each objective function has been introduced as follows:

$$P\left(\text{Accept } S_c\right) = \prod_i \exp\left(-\frac{\Delta E_i}{T_i}\right) = \exp\left(\sum_i -\frac{\Delta E_i}{T_i}\right) \quad (7)$$

where ΔE_i is the energy variation of the objective function E_i between the candidate and the current solution and T_i is the current temperature assigned to E_i which serves as weight for the objective. From this perspective, some objectives could be favoured independently of their magnitude order. The method enables a return to base strategy that consists in recommencing the search periodically from an archived nondominated solution. This strategy allows diversity in the exploration of the feasible domain. Besides and in order to allow intensification in less explored regions, Suppapitnarm et al. proposed to restart the search from the most isolated solution from the archive. An isolation measurement has been introduced in addition to a monitoring strategy for the rate at which the return to base could be activated.

3.4. Domination-Based MOSA Methods.

In the above methods, the quality of solutions at hand is measured in accordance to the last accepted solution as a unique reference criteria. An alternative to this approach would consist in comparing generated solutions to the actual Pareto front as well. Yet, it is crucial to determine a certain performance indicator to characterise the closeness to the Pareto front. Thus, many performance indicators have been introduced in the literature that consist of either a certain distance to the Pareto front, or the ratio of the volume dominated by the resultant archive to that dominated by the Pareto front. Nevertheless, this approach is faced to the prevailing lack of advanced information about the actual Pareto front. Hence, algorithms performance is often investigated using test problems that are average computing cost optimisation problems with known properties. Discussions on and construction of test problems have been widely addressed in the literature within different contexts of multiobjective optimisation (e.g., [45–50]).

In concept, solutions are mainly configurations of decision variables; however, they could be considered as configurations of objective functions as well. This gives rise to two different, but equivalent, representations of the feasible domain. Figure 3 illustrates the correspondence between the two representations in the decision space, which is constituted by variables and, in the objective space, which is constituted by objective functions (Note that the dimension of the decision space and that of the objective space are not necessary the same.). Given a Pareto front $\mathscr{P}$ of a certain optimisation problem and an archive under evaluation $\mathscr{A}$, that are assumed, without loss of generality, to be finite, the distance between any solution $S_{\mathscr{A}}$ from $\mathscr{A}$ to $\mathscr{P}$, denoted $\text{dist}(S_{\mathscr{A}}, \mathscr{P})$, is measured in the objective space by a Euclidean distance based formula:

$$\text{dist}\left(S_{\mathscr{A}}, \mathscr{P}\right) = \min_{S_{\mathscr{P}} \in \mathscr{P}} d\left(S_{\mathscr{A}}, S_{\mathscr{P}}\right) \qquad (8)$$

where d is the classical Euclidean distance. Yet, the naive formulation of the distance between $\mathscr{A}$ and $\mathscr{P}$

$$\text{dist}\left(\mathscr{A}, \mathscr{P}\right) = \min_{S_{\mathscr{A}} \in \mathscr{A}} \text{dist}\left(S_{\mathscr{A}}, \mathscr{P}\right) = \min_{S_{\mathscr{A}} \in \mathscr{A}, S_{\mathscr{P}} \in \mathscr{P}} d\left(S_{\mathscr{A}}, S_{\mathscr{P}}\right) \quad (9)$$

returns the distance between the nearest pairs from $\mathscr{A}$ and $\mathscr{P}$, which does not reflect the closeness between the two sets as
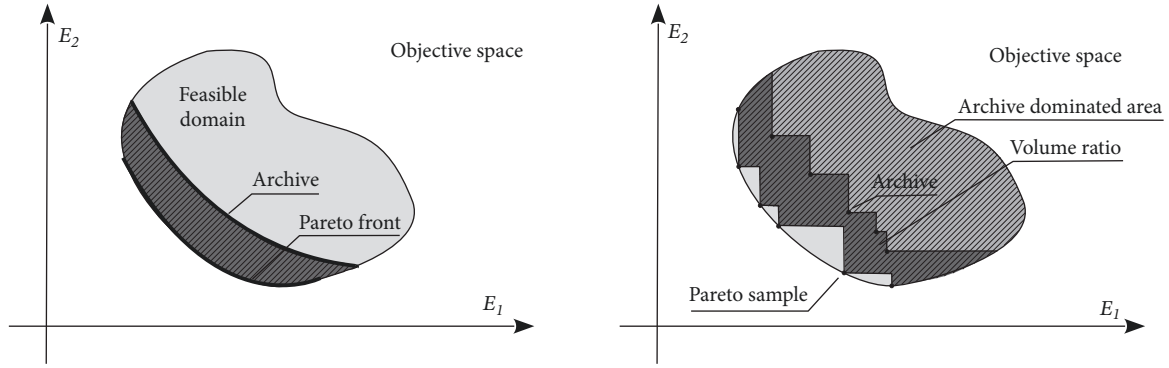
FIGURE 4: Hypothetical volume domination versus practical volume domination.

a matter of course; indeed, if only one solution is very close to the Pareto set, the distance would be lower providing a confusing closeness measurement. Therefore, a sophisticated distance formulation would consist in taking into account the distances between all pairs from $\mathscr{A}$ and $\mathscr{P}$ (Either the considered optimisation problem is discrete or continuous; only a discrete sample of the actual Pareto front would be available, and yet a discrete approximating Pareto front would be solicited.). Accordingly, Smith [51] has proposed to use the median of the Euclidean distance between pairs from the two sets:

$$M(\mathscr{A}, \mathscr{P}) = \underset{S_{\mathscr{A}} \in A}{\text{median}} \{\text{dist}(S_{\mathscr{A}}, \mathscr{P})\} \quad (10)$$

Besides, two metrics that have been widely used in the literature are the generational distance [52], G, and the inverted generational distance [53], IG, that consist of the following:

$$G(\mathscr{A}, \mathscr{P}) = \frac{1}{|\mathscr{A}|} \sqrt{\sum_{S_{\mathscr{A}} \in \mathscr{A}} \text{dist}^2(S_{\mathscr{A}}, \mathscr{P})} \quad (11)$$

$$IG(\mathscr{A}, \mathscr{P}) = \frac{1}{|\mathscr{P}|} \sqrt{\sum_{S_{\mathscr{P}} \in \mathscr{P}} \text{dist}^2(S_{\mathscr{P}}, \mathscr{A})} \quad (12)$$

where $|\mathscr{A}|$ and $|\mathscr{P}|$ are the cardinality of $\mathscr{A}$ and $\mathscr{P}$ respectively. Another example of metrics used in the literature is the distance of Hausdorff consisting of the following:

$$\begin{aligned} &H(\mathscr{A}, \mathscr{P}) \\ &= \max\left\{\sup_{S_{\mathscr{A}} \in \mathscr{A}} \inf_{S_{\mathscr{P}} \in \mathscr{P}} d(S_{\mathscr{A}}, S_{\mathscr{P}}), \sup_{S_{\mathscr{P}} \in \mathscr{P}} \inf_{S_{\mathscr{A}} \in \mathscr{A}} d(S_{\mathscr{A}}, S_{\mathscr{P}})\right\} \end{aligned} \quad (13)$$

that has been used in an extended version in the works of Schütze et al. [50] and Bogoya et al. [54].

It is straightforward that a multiobjective optimisation should result in a set that is sufficiently covering as well as efficiently close to the actual Pareto set. Deb et al. [55] proposed a spread metric parameter to characterise the covering. Yet, no performance indicator from the aforementioned has got the general approval as an efficient and accurate indicator for both closeness and spread measurement. Volume domination

has been proposed in the literature [56–58] as a promising alternative towards efficient evaluation of the constructed archive.

Hypervolume is a useful metric for characterising the coverage of an archive as well as its closeness to the Pareto front. This metric, which dates back to the works of Zitzler and Thiele [58], consists in calculating in the objective space the size of the area covered by the archive elements. The considered optimisation problem would be reduced to maximising such measurement. Similarly, a certain ratio of the volume dominated by the resultant archive to that dominated by the Pareto front would be an alternative performance indicator. Smith [51] has considered the area dominated by the Pareto front and not dominated by the archive under evaluation. Figure 4 illustrates the conceptual areas delimited by the Pareto front and the archive. Of course, the domination is related to the acceptance rule (weak or strong). Yet, in practice, three areas can be differentiated, namely, the area dominated by both the available Pareto sample and the archive under evaluation (hatched grey zone), the area dominated by the Pareto sample only (hatched dark grey zone), and the unexplored area (grey zone).

Many MOSA methods have used one or more dominance-oriented performance indicators in order to provide an advanced metaheuristic. An early discussion has been conducted by Nam and Park [59] where six acceptance criteria have been suggested and evaluated. Besides, Smith et al. [60] have proposed a composite energy function based on domination as an alternative to the classical weighted combination of objective functions. While the latter approach allows a priori biasing guidance towards a certain region of the Pareto front, thereby making the algorithm sensitive to the weighting given to the competitive objective functions, the proposed approach aims at eschewing any limiting proposal on the feasible domain exploration in favour of dominance concept.

Sankararao and Yoo [61] proposed a robust multiobjective simulated annealing (rMOSA) with the aim to speed up the convergence process and to get a uniform Pareto set approximation. The method is developed with application to chemical engineering problems.

A domination-based method, called Archived Multiobjective Simulated Annealing method (AMOSA), has been

proposed [62] using a parameter called amount of domination. For a given number of objective functions E_i and considering an archived solution S_a and a candidate solution S_c, the amount of domination between the two solutions is given by the following:

$$\Delta dom\,(S_c, S_a) = \prod_{\substack{i \\ E_i(S_c) \neq E_i(S_a)}} \frac{|E_i(S_c) - E_i(S_a)|}{R_i} \quad (14)$$

where R_i is the range of the ith objective function. This domination amount represents a hypervolume for which each side in the Euclidean plane, constituted by certain two objective functions E_i and E_j, is a rectangle delimited by the points S_c, $(E_i(S_a), E_j(S_c))$, S_a, and $(E_i(S_c), E_j(S_a))$. At each iteration, AMOSA constructs a candidate solution from a current solution at hand that will undergo an evaluation in respect of the archive as well as the current solution—an average amount of domination Δdom_{avr} is computed in regard to either the solutions dominated by the candidate solution or those dominating it. Any candidate solution that is nondominated by the current solution or any other solution from the archive is accepted as current and added to the archive. Meanwhile, any dominated solution from the archive is removed. If the candidate solution is dominated by a certain number of solutions from the archive, the considered acceptance rule consists of the following:

$$P\,(\text{Accept } S_c) = \frac{1}{1 + \exp\,(\Delta dom_{avr}\,(S_c) \times T)} \quad (15)$$

where T is the current value of the temperature parameter. If the candidate solution dominates the current solution (which does not necessary belong to the archive) while it is dominated by a certain number of archived solutions, a transition to an archived solution is performed with certain probability. Overall, the possible transitions consist in accepting the candidate solution, keeping the current solution, or moving to a solution from the archive. In order to decide on the resultant archive size, AMOSA defines two limiting parameters called hard limit and soft limit. The archive size is allowed to increase up to the soft limit, after which the number of solutions is reduced to the hard limit by means of a certain clustering process.

Suman et al. [63] have presented an extension of AMOSA method on the basis of the orthogonal experiment design (OED), called Orthogonal Simulated Annealing (OSA). The main contribution of this method lies in the use of the OED for guiding the selection of good neighbouring solutions, by means of an orthogonal table and a fractional factorial analysis. OED aims at allowing effective search while reducing computational cost. Figure 5 presents the OSA method flowchart. The equilibrium condition is set to impose an upper bound on the number of iterations that should be performed for the same current solution at each temperature level.

On the basis of AMOSA method, Sengupta and Saha [64] proposed a reference point based many-objective simulated annealing algorithm, referred to as RSA. Rather than

moving from one solution to a neighbouring one, the work introduced the archive-to-archive concept that consists in moving from an archive of solutions to another one. The idea is analogous to the use of population of solutions in genetic algorithms. The method involves in addition a switching strategy, called mutation switching, which consists in periodically switching between different mutation techniques. Besides, the method uses the reference point based clustering algorithm in order to get a uniform spread of the returned archive.

Due to its computational efficiency and simplicity, simulated annealing has been widely used to solve multiobjective optimisation problems, as well as mono-objective ones, in various fields including clustering [65–67], jobshop problems [68], and scheduling [69]. Besides, genetic algorithms and particle swarm optimisation methods have received much attention in the recent years, and many commercial solvers permit simple implementation of such methods. Accordingly, few hybridisations of multiobjective simulated annealing with one of the aforementioned metaheuristics have been introduced in the literature [70, 71] in addition to many applications in supply chain [72, 73], distribution networks [74, 75], facility layout design [76, 77], design optimisation[66], and scheduling [69, 78–80].

4. Summary and Concluding Remarks

Simulated annealing is a metaheuristic belonging to the local search methods family. It guarantees gradual convergence to a near-optimal solution and provides the ability of escaping local optima and being memoryless. Due to its adaptability to many combinatorial and continuous optimisation problems, a wide range of adaptations have hitherto emerged. Such a success has motivated an extension of the simulated annealing meta-heuristic to the multiobjective optimisation context that consists in approximating the optimal solutions set by means of an archive that holds the nondominated solutions found while exploring the feasible domain. Accordingly, many variants of multiobjective simulated annealing have been introduced in the literature. Nevertheless, the following remarks are observed:

(i) Compared to other metaheuristics, simulated annealing is a powerful tool for many applications due to its asymptotic convergence. However, the algorithm process is rather slow especially for some cooling schemes and using another heuristic seems to give better result for problems with few optima.

(ii) The use of an archive within a sequential storage process has emerged in almost all Pareto optimisation including swarm optimisation and evolutionary algorithms (e.g., [81, 82]). The process involves an updating mechanism that becomes restrictive when the nondominated solutions number gets larger. A discussion has been conducted on the use of archive regarding the performance of metaheuristics in Pareto optimisation [83].

(iii) Controlling the number of returned solutions keeps challenging in multiobjective simulated annealing.
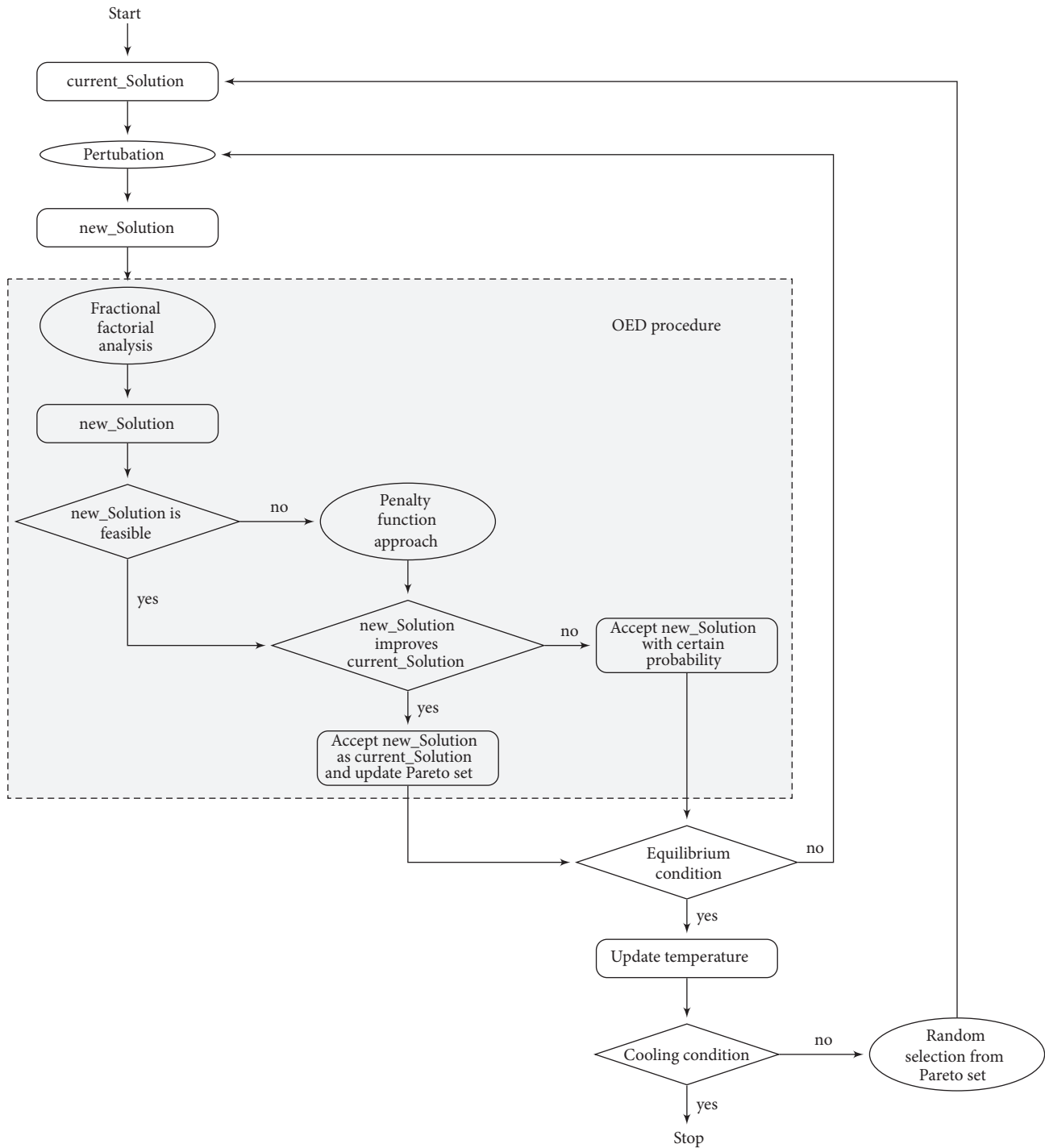
FIGURE 5: Orthogonal simulated annealing flowchart.

Generally, all metaheuristics involving an archive to represent an approximation of the actual Pareto solutions set should strive for this archive to be bounded and of limited size [84]. Yet, it may be difficult to decide on the archive cardinality in practice, and the same metaheuristic may lead to a very concise archive for a certain problem occurrence and to a very large one for others. Knowles and Corne [84]

have addressed a theoretical study of metaheuristics archiving, as a sequential storage process, leading to the no significant performance of any proposed approach, so far, to control the archive size. Nevertheless, Gaspar-Cunha et al. [85] have proposed a solution clustering mechanism with the aim to reduce the returned near-Pareto front when it is of large size.

(iv) Tuning simulated annealing parameters is challenging in both single and multiple criteria optimisation. As a wise choice of parameters would result on fairly representative approximation of the Pareto-optimal solutions set, many works [86–88] have required preliminary experiments. Yet, experiments would require some knowledge on the actual Pareto front in terms of range or cardinality, which is not often the case. The situation becomes more problematic if the experiments would involve comparison with other metaheuristics.
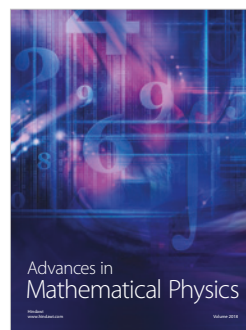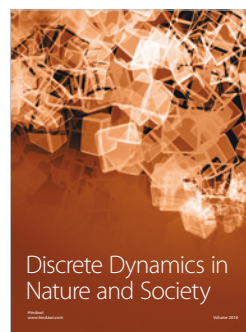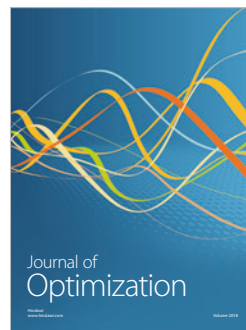
Conflicts of Interest

The author declares that they have no conflicts of interest.

References

[1] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[2] V. Černý, "Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm," *Journal of Optimization Theory and Applications*, vol. 45, no. 1, pp. 41–51, 1985.

[3] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[4] S. Alizamir, S. Rebennack, and P. M. Pardalos, "Improving the neighborhood selection strategy in simulated annealing using the optimal stopping problem," in *Simulated Annealing*, C. M. Tan, Ed., pp. 363–382, InTech, Rijeka, Croatia, 2008.

[5] L. Goldstein and M. Waterman, "Neighborhood size in the simulated annealing algorithm," *American Journal of Mathematical and Management Sciences*, vol. 8, no. 3-4, pp. 409–423, 1988.

[6] E. Aarts, J. Korst, and W. Michiels, "Simulated annealing," in *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, K Burke and G. Kendall, Eds., pp. 91–120, Springer US, Boston, Mass, USA, 2nd edition, 2014.

[7] E.-G. Talbi, *Metaheuristics: from Design to Implementation*, John Wiley & Sons, Hoboken, New Jersey, NJ, USA, 2009.

[8] E. Triki, Y. Collette, and P. Siarry, "A theoretical study on the behavior of simulated annealing leading to a new cooling schedule," *European Journal of Operational Research*, vol. 166, no. 1, pp. 77–92, 2005.

[9] P. N. Strenski and S. Kirkpatrick, "Analysis of finite length annealing schedules," *Algorithmica*, vol. 6, no. 1, pp. 346–366, 1991.

[10] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.

[11] K. Sekihara, H. Haneishi, and N. Ohyama, "Details of simulated annealing algorithm to estimate parameters of multiple current dipoles using biomagnetic data," *IEEE Transactions on Medical Imaging*, vol. 11, no. 2, pp. 293–299, 1992.

[12] M. Lundy and A. Mees, "Convergence of an annealing algorithm," *Mathematical Programming*, vol. 34, no. 1, pp. 111–124, 1986.

[13] L. Ingber, "Very fast simulated re-annealing," *Mathematical and Computer Modelling*, vol. 12, no. 8, pp. 967–973, 1989.

[14] K. Amine, "Insights into simulated annealing," in *Handbook of Research on Modeling, Analysis, and Application of Nature-Inspired Metaheuristic Algorithms*, S. Dash, B. K. Tripathy, and A. Rahman, Eds., Advances in Computational Intelligence and Robotics, pp. 121–139, IGI Global, Hershey, Pa, USA, 2018.

[15] K. A. Dowsland and J. M. Thompson, "Simulated annealing," in *Handbook of Natural Computing*, G. Rozenberg, T. Bäck, and J. N. Kok, Eds., pp. 1623–1655, Springer-Verlag, Berlin, Germany, 2012.

[16] D. Fouskakis and D. Draper, "Stochastic optimization: a review," *International Statistical Review*, vol. 70, no. 3, pp. 315–349, 2002.

[17] A. Franzin and T. Stützle, "Revisiting simulated annealing: a component-based analysis," *Computers & Operations Research*, vol. 104, pp. 191–206, 2019.

[18] N. Siddique and H. Adeli, "Simulated annealing, its variants and engineering applications," *International Journal on Artificial Intelligence Tools*, vol. 25, no. 6, Article ID 1630001, 2016.

[19] B. Suman, N. Hoda, and S. Jha, "Orthogonal simulated annealing for multiobjective optimization," *Computers & Chemical Engineering*, vol. 34, no. 10, pp. 1618–1631, 2010.

[20] J. Korhonen and Y. Wang, "Effect of packet size on loss rate and delay in wireless links," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'05)*, vol. 3, pp. 1608–1613, New Orleans, Louisiana, La, USA, 2005.

[21] K. Sörensen and J. Springael, "Progressive multi-objective optimization," *International Journal of Information Technology & Decision Making*, vol. 13, no. 5, pp. 917–936, 2014.

[22] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proceedings of the 1st International Conference on Genetic Algorithms*, J. J. Grefenstette, Ed., pp. 93–100, Lawrence Erlbaum Associates, Inc., Hillsdale, NJ, USA, 1985.

[23] P. M. Hansen, "Tabu search for multiobjective optimization: MOTS," in *Proceedings of the 13th International Conference on Multiple Criteria Decision Making (MCDM 97)*, Cape Town, South Africa, 1996.

[24] L. M. Gambardella, É. Taillard, and G. Agazzi, "MACS-VRPTW: a multiple ant colony system for vehicle routing problems with time windows," in *New Ideas in Optimization*, D. Corne, M. Dorigo, F. Glover et al., Eds., pp. 63–76, McGraw-Hill Ltd, Maidenhead, UK, 1999.

[25] E. L. Ulungu, J. Teghem, P. H. Fortemps, and D. Tuyttens, "MOSA method: a tool for solving multiobjective combinatorial optimization problems," *Journal of Multi-Criteria Decision Analysis*, vol. 8, no. 4, pp. 221–236, 1999.

[26] R. P. Beausoleil, "'MOSS' multiobjective scatter search applied to non-linear multiple criteria optimization," *European Journal of Operational Research*, vol. 169, no. 2, pp. 426–449, 2006.

[27] I. Das and J. E. Dennis, "A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems," *Journal of Structural Optimization*, vol. 14, no. 1, pp. 63–69, 1997.

[28] P. Serafini, "Simulated annealing for multi objective optimization problems," in *Multiple Criteria Decision Making: Proceedings of the Tenth International Conference: Expand and Enrich the Domains of Thinking and Application*, G. H. Tzeng, H. F. Wang, U. P. Wen, and P. L. Yu, Eds., pp. 283–292, Springer-Verlag, New York, NY, USA, 1994.

[29] M. H. Alrefaei and A. H. Diabat, "A simulated annealing technique for multi-objective simulation optimization," *Applied Mathematics and Computation*, vol. 215, no. 8, pp. 3029–3035, 2009.

[30] P. Czyżak and A. Jaszkiewicz, "Pareto simulated annealinga metaheuristic technique for multiple-objective combinatorial optimization," *Journal of Multi-Criteria Decision Analysis*, vol. 7, no. 1, pp. 34–47, 1998.

[31] B. Suman, "Simulated annealing-based multiobjective algorithms and their application for system reliability," *Engineering Optimization*, vol. 35, no. 4, pp. 391–416, 2003.

[32] A. Suppapitnarm, K. A. Seffen, G. T. Parks, and P. J. Clarkson, "A Simulated annealing algorithm for multiobjective optimization," *Engineering Optimization*, vol. 33, no. 1, pp. 59–85, 2000.

[33] J. Teghem, D. Tuyttens, and E. L. Ulungu, "An interactive heuristic method for multi-objective combinatorial optimization," *Computers & Operations Research*, vol. 27, no. 7-8, pp. 621–634, 2000.

[34] O. Tekinalp and G. Karsli, "A new multiobjective simulated annealing algorithm," *Journal of Global Optimization*, vol. 39, no. 1, pp. 49–77, 2007.

[35] M. Villalobos-Arias, C. A. Coello Coello, and O. Hernández-Lerma, "Asymptotic convergence of a simulated annealing algorithm for multiobjective optimization problems," *Mathematical Methods of Operations Research*, vol. 64, no. 2, pp. 353–362, 2006.

[36] E. L. Ulungu, J. Teghem, and P. Fortemps, "Heuristic for multi-objective combinatorial optimization problems by simulated annealing," in *MCDM: Theory and Applications*, J. Gu, G. Chen, Q. Wei, and S. Wang, Eds., pp. 229–238, Sci-Tech, 1995.

[37] R. Baños, C. Gil, B. Paechter, and J. Ortega, "Parallelization of population-based multi-objective meta-heuristics: an empirical study," *Applied Mathematical Modelling*, vol. 30, no. 7, pp. 578–592, 2006.

[38] M. Hapke, A. Jaszkiewicz, and R. Słowiński, "Pareto simulated annealing for fuzzy multi-objective combinatorial optimization," *Journal of Heuristics*, vol. 6, no. 3, pp. 329–345, 2000.

[39] W. J. Gutjahr, "Two metaheuristics for multiobjective stochastic combinatorial optimization," in *Stochastic Algorithms: Foundations and Applications*, O. B. Lupanov, O. M. Kasim-Zade, A. V. Chaskin, and K. Steinhöfel, Eds., pp. 116–125, Springer, Berlin, Germany, 2005.

[40] H. Li and D. Landa-Silva, "An adaptive evolutionary multi-objective approach based on simulated annealing," *Evolutionary Computation*, vol. 19, no. 4, pp. 561–595, 2011.

[41] H. Li and D. Landa-Silva, "Evolutionary multi-objective simulated annealing with adaptive and competitive search direction," in *Proceedings of the 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, IEEE, Hong Kong, June 2008.

[42] Q. Zhang and H. Li, "MOEA/D: a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[43] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 263–282, 2002.

[44] P. Engrand, "A multi-objective approach based on simulated annealing and its application to nuclear fuel management," in *Proceedings of the 5th International Conference on Nuclear Engineering: ICONE-5*, pp. 416–423, American Society of Mechanical Engineers, New York, NY, USA, 1997.

[45] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[46] K. Deb, "Multi-objective genetic algorithms: problem difficulties and construction of test problems," *Evolutionary Computation*, vol. 7, no. 3, pp. 205–230, 1999.

[47] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.

[48] Y. Jin and B. Sendhoff, "Constructing dynamic optimization test problems using the multi-objective optimization concept," in *Applications of Evolutionary Computing*, G. R. Raidl, S. Cagnoni, and J. Branke, Eds., pp. 525–536, Springer, Berlin, Germany, 2004.

[49] G. Rudolph, O. Schütze, C. Grimme, C. Domínguez-Medina, and H. Trautmann, "Optimal averaged Hausdorff archives for bi-objective problems: theoretical and numerical results," *Computational Optimization and Applications*, vol. 64, no. 2, pp. 589–618, 2016.

[50] O. Schütze, X. Esquivel, A. Lara, and C. A. C. Coello, "Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 4, pp. 504–522, 2012.

[51] K. I. Smith, *A Study of Simulated Annealing Techniques for Multi-Objective Optimisation [Ph.D. thesis]*, University of Exeter, England, UK, 2006.

[52] D. A. van Veldhuizen and G. B. Lamont, "Evolutionary computation and convergence to a pareto front," in *Proceedings of the Late Breaking Papers at the 1998 Genetic Programming Conference*, J. R. Koza, Ed., Omni Press, University of Wisconsin, Madison, Wis, USA, July 1998.

[53] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima, "Modified distance calculation in generational distance and inverted generational distance," in *Evolutionary Multi-Criterion Optimization*, A. Gaspar-Cunha, C. H. Antunes, and C. C. Coello, Eds., vol. 9019 of *Lecture Notes in Computer Science*, pp. 110–125, Springer International Publishing, Cham, Switzerland, 2015.

[54] J. M. Bogoya, A. Vargas, O. Cuate, and O. Schütze, "A (p, q)-averaged Hausdorff distance for arbitrary measurable sets," *Mathematical & Computational Applications*, vol. 23, no. 3, Article ID 51, 2018.

[55] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, Wiley, Chichester, UK, 2001.

[56] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler, "Hypervolume-based multiobjective optimization: theoretical foundations and practical implications," *Theoretical Computer Science*, vol. 425, pp. 75–103, 2012.

[57] J. Wu and S. Azarm, "Metrics for quality assessment of a multiobjective design optimization solution set," *Journal of Mechanical Design*, vol. 123, no. 1, pp. 18–25, 2001.

[58] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[59] D. Nam and C. Park, "Multiobjective simulated annealing: a comparative study to evolutionary algorithms," *International Journal Fuzzy Systems*, vol. 2, no. 2, pp. 87–97, 2000.

[60] K. I. Smith, R. M. Everson, J. E. Fieldsend, C. Murphy, and R. Misra, "Dominance-based multiobjective simulated annealing,"

IEEE Transactions on Evolutionary Computation, vol. 12, no. 3, pp. 323–342, 2008.

[61] B. Sankararao and C. K. Yoo, "Development of a robust multiobjective simulated annealing algorithm for solving multiobjective optimization problems," *Industrial & Engineering Chemistry Research*, vol. 50, no. 11, pp. 6728–6742, 2011.

[62] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, "A simulated annealing-based multiobjective optimization algorithm: AMOSA," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 3, pp. 269–283, 2008.

[63] B. Suman and P. Kumar, "A survey of simulated annealing as a tool for single and multiobjective optimization," *Journal of the Operational Research Society*, vol. 57, no. 10, pp. 1143–1160, 2006.

[64] R. Sengupta and S. Saha, "Reference point based archived many objective simulated annealing," *Information Sciences*, vol. 467, pp. 725–749, 2018.

[65] S. G. Devi and M. Sabrigiriraj, "A hybrid multi-objective firefly and simulated annealing based algorithm for big data classification," *Concurrency and Computation: Practice and Experience*, Article ID e4985, 2018.

[66] S. L. Ho, S. Yang, H. C. Wong, and G. Ni, "A simulated annealing algorithm for multiobjective optimizations of electromagnetic devices," *IEEE Transactions on Magnetics*, vol. 39, no. 3, pp. 1285–1288, 2003.

[67] S. Saha and S. Bandyopadhyay, "A new multiobjective clustering technique based on the concepts of stability and symmetry," *Knowledge and Information Systems*, vol. 23, no. 1, pp. 1–27, 2010.

[68] R. K. Suresh and K. M. Mohanasundaram, "Pareto archived simulated annealing for job shop scheduling with multiple objectives," *The International Journal of Advanced Manufacturing Technology*, vol. 29, no. 1-2, pp. 184–196, 2006.

[69] V. Yannibelli and A. Amandi, "Hybridizing a multi-objective simulated annealing algorithm with a multi-objective evolutionary algorithm to solve a multi-objective project scheduling problem," *Expert Systems with Applications*, vol. 40, no. 7, pp. 2421–2434, 2013.

[70] A. Abubaker, A. Baharum, and M. Alrefaei, "Multi-objective particle swarm optimization and simulated annealing in practice," *Applied Mathematical Sciences*, vol. 10, no. 42, pp. 2087–2103, 2016.

[71] P. Vasant, "Hybrid simulated annealing and genetic algorithms for industrial production management problems," *International Journal of Computational Methods*, vol. 7, no. 2, pp. 279–297, 2010.

[72] V. Babaveisi, M. M. Paydar, and A. S. Safaei, "Optimizing a multi-product closed-loop supply chain using NSGA-II, MOSA, and MOPSO meta-heuristic algorithms," *Journal of Industrial Engineering International*, vol. 14, no. 2, pp. 305–326, 2017.

[73] T. Dereli and G. Sena Das, "A hybrid simulated annealing algorithm for solving multi-objective container-loading problems," *Applied Artificial Intelligence*, vol. 24, no. 5, pp. 463–486, 2010.

[74] C. H. Antunes, P. Lima, E. Oliveira, and D. F. Pires, "A multi-objective simulated annealing approach to reactive power compensation," *Engineering Optimization*, vol. 43, no. 10, pp. 1063–1077, 2011.

[75] J. Marques, M. Cunha, and D. Savić, "Many-objective optimization model for the flexible design of water distribution networks," *Journal of Environmental Management*, vol. 226, pp. 308–319, 2018.

[76] S. Turgay, "Multi objective simulated annealing approach for facility layout design," *International Journal of Mathematical, Engineering and Management Sciences*, vol. 3, no. 4, pp. 365–380, 2018.

[77] U. R. Tuzkaya, T. Ertay, and D. Ruan, "Simulated annealing approach for the multi-objective facility layout problem," in *Intelligent Data Mining*, D. Ruan, G. Chen, E. E. Kerre, and G. Wets, Eds., Studies in Computational Intelligence, pp. 401–418, Springer, Berlin, Germany, 2005.

[78] C. Akkan and A. Gülcü, "A bi-criteria hybrid genetic algorithm with robustness objective for the course timetabling problem," *Computers & Operations Research*, vol. 90, pp. 22–32, 2018.

[79] H. J. Fraire Huacuja, J. Frausto-Solís, J. D. Terán-Villanueva, J. C. Soto-Monterrubio, J. J. González Barbosa, and G. Castilla-Valdez, "AMOSA with analytical tuning parameters for heterogeneous computing scheduling problem," in *Nature-Inspired Design of Hybrid Intelligent Systems*, P. Melin, O. Castillo, and J. Kacprzyk, Eds., vol. 667 of *Studies in Computational Intelligence*, pp. 701–711, Springer International Publishing, Cham, Switzerland, 2017.

[80] B. Shahul Hamid Khan and K. Govindan, "A multi-objective simulated annealing algorithm for permutation flow shop scheduling problem," *International Journal of Advanced Operations Management*, vol. 3, no. 1, pp. 88–100, 2011.

[81] S. Agrawal, B. K. Panigrahi, and M. K. Tiwari, "Multiobjective particle swarm algorithm with fuzzy clustering for electrical power dispatch," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 5, pp. 529–541, 2008.

[82] X. Cai, Y. Li, Z. Fan, and Q. Zhang, "An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 4, pp. 508–523, 2015.

[83] A. Jaszkiewicz and T. Lust, "ND-tree-based update: a fast algorithm for the dynamic nondominance problem," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 5, pp. 778–791, 2018.

[84] J. Knowles and D. Corne, "Bounded pareto archiving: theory and practice," in *Metaheuristics for Multiobjective Optimisation*, X. Gandibleux, M. Sevaux, K. Sörensen, and V. T'kindt, Eds., pp. 39–64, Springer, Berlin, Germany, 2004.

[85] A. Gaspar-Cunha, P. Oliveira, and J. A. Covas, "Use of genetic algorithms in multicriteria optimization to solve industrial problems," in *Proceedings of the Seventh International Conference on Genetic Algorithms: Michigan State University, East Lansing, MI, July 19-23, 1997*, T. Back, Ed., pp. 682–688, Morgan Kaufmann Publishers Inc., San Francisco, Calif, USA, 1997.

[86] P. Cabrera-Guerrero, G. Guerrero, J. Vega, and F. Johnson, "Improving simulated annealing performance by means of automatic parameter tuning," *Studies in Informatics and Control*, vol. 24, no. 4, pp. 419–426, 2015.

[87] W. G. Jackson, E. Özcan, and R. I. John, "Tuning a simulated annealing metaheuristic for cross-domain search," in *Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1055–1062, San Sebastián, Spain, June 2017.

[88] S. M. Sait and H. Youssef, *Iterative Computer Algorithms with Applications in Engineering: Solving Combinatorial Optimization Problems*, IEEE Computer Society Press, Los Alamitos, Calif, USA, 1999.