



# Evolutionary multiobjective optimization: open research areas and some challenges lying ahead

Carlos A. Coello Coello<sup>1</sup> · Silvia González Brambila<sup>1</sup> · Josué Figueroa Gamboa<sup>1</sup> · Ma Guadalupe Castillo Tapia<sup>2</sup> · Raquel Hernández Gómez<sup>3</sup>

Received: 8 May 2019 / Accepted: 28 May 2019 / Published online: 19 June 2019  
© The Author(s) 2019

## Abstract

Evolutionary multiobjective optimization has been a research area since the mid-1980s, and has experienced a very significant activity in the last 20 years. However, and in spite of the maturity of this field, there are still several important challenges lying ahead. This paper provides a short description of some of them, with a particular focus on open research areas, rather than on specific research topics or problems. The main aim of this paper is to motivate researchers and students to develop research in these areas, as this will contribute to maintaining this discipline active during the next few years.

**Keywords** Pythagorean fuzzy value · Pythagorean fuzzy sets · Similarity

## Introduction

In the real world, there exist many problems having two or more (often conflicting) objectives that we aim to optimize at the same time. They are called multiobjective optimization problems (MOPs) and their solution has attracted the

attention of researchers for many years. Because of the conflict among the objectives, solving an MOP produces a set of solutions representing the best possible trade-offs among the objectives (i.e., solutions in which one objective cannot be improved without worsening another one). Such solutions constitute the *Pareto optimal set* and the image of this set (i.e., the corresponding objective function values) form the so-called *Pareto front*.

In spite of the fact that a wide variety of mathematical programming techniques have been developed to tackle MOPs since the 1970s [115], such techniques present a number of limitations, from which the most remarkable are that these algorithms are normally quite susceptible to the shape and/or continuity of the Pareto front and that they usually generate one element of the Pareto optimal set per algorithmic execution. Additionally, some mathematical programming techniques require that the objective functions and the constraints are provided in algebraic form and in many real-world problems we can only obtain such values from a simulator. These limitations have motivated the use of alternative approaches, from which metaheuristics have been a very popular choice, mainly because of their flexibility (i.e., they require little domain specific information) and their ease of use. From the many metaheuristics currently available [158], evolutionary algorithms [57] have certainly been the most popular in the last few years in this area, giving rise to a field now known as evolutionary multiobjective optimization (EMO) [29].

---

On sabbatical leave from the Department of Computer Science, CINVESTAV-IPN, Av. IPN No. 2508, Col. San Pedro Zacatenco, México City, Mexico.

---

✉ Carlos A. Coello Coello  
ccoello@cs.cinvestav.mx

Silvia González Brambila  
sgb@azc.uam.mx

Josué Figueroa Gamboa  
jfgo@azc.uam.mx

Ma Guadalupe Castillo Tapia  
mgct@correo.azc.uam.mx

Raquel Hernández Gómez  
rhernandez@computacion.cs.cinvestav.mx

<sup>1</sup> Departamento de Sistemas, UAM Azcapotzalco, Av San Pablo Xalpa 180 Col. Reynosa Tamaulipas, 02200 Mexico, Mexico

<sup>2</sup> Departamento de Administración, UAM Azcapotzalco, Av San Pablo Xalpa 180 Col. Reynosa Tamaulipas, 02200 Mexico, Mexico

<sup>3</sup> Department of Computer Science (Evolutionary Computation Group), CINVESTAV-IPN, Av. IPN No. 2508 Col. San Pedro Zacatenco, 07360 Mexico, Mexico

The first Multi-Objective Evolutionary Algorithm (MOEA) was called Vector Evaluated Genetic Algorithm (VEGA) and was proposed by Schaffer in the mid-1980s [147–149]. Something interesting is that there was not much interest in EMO research for almost a decade. However, in the mid-1990s, this area started to attract a lot of attention from several research groups around the world, and has maintained a high research activity since then.<sup>1</sup>

The remainder of this paper is organized as follows. In “Basic Concepts” we provide some basic mathematical concepts related to multiobjective optimization, with the aim of making of this a self-contained paper. “Open research areas” provides a list of research areas that present challenges that are particularly relevant from the authors’ perspective. In the subsequent subsections, each of these areas are described, providing in the process a summary of some of the most relevant research that has been conducted in such areas so far. The next section presents the main challenges associated to each of the research areas previously discussed. Finally, our conclusions are presented in the last section.

## Basic concepts

In multiobjective optimization, the aim is to solve problems of the type:<sup>2</sup>

$$\text{minimize } \mathbf{f}(\mathbf{x}) := [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})] \quad (1)$$

subject to:

$$g_i(\mathbf{x}) \leq 0 \quad i = 1, 2, \dots, m, \quad (2)$$

$$h_i(\mathbf{x}) = 0 \quad i = 1, 2, \dots, p, \quad (3)$$

where  $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$  is the vector of decision variables,  $f_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, \dots, k$  are the objective functions and  $g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, \dots, m, j = 1, \dots, p$  are the constraint functions of the problem.

A few additional definitions are required to introduce the notion of optimality used in multiobjective optimization.

**Definition 1** Given two vectors  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^k$ , we say that  $\mathbf{x} \leq \mathbf{y}$  if  $x_i \leq y_i$  for  $i = 1, \dots, k$ , and that  $\mathbf{x}$  *dominates*  $\mathbf{y}$  (denoted by  $\mathbf{x} < \mathbf{y}$ ) if  $\mathbf{x} \leq \mathbf{y}$  and  $\mathbf{x} \neq \mathbf{y}$ .

**Definition 2** We say that a vector of decision variables  $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n$  is *nondominated* with respect to  $\mathcal{X}$ , if there does not exist another  $\mathbf{x}' \in \mathcal{X}$  such that  $\mathbf{f}(\mathbf{x}') < \mathbf{f}(\mathbf{x})$ .

<sup>1</sup> The first author maintains the EMOO repository [28] which currently contains over 12,100 bibliographic references on evolutionary multiobjective optimization. The EMOO repository is located at: <https://emoo.cs.cinvestav.mx>.

<sup>2</sup> Without loss of generality, we will assume only minimization problems.

**Definition 3** We say that a vector of decision variables  $\mathbf{x}^* \in \mathcal{F} \subset \mathbb{R}^n$  ( $\mathcal{F}$  is the feasible region) is *Pareto-optimal* if it is nondominated with respect to  $\mathcal{F}$ .

**Definition 4** The *Pareto optimal set*  $\mathcal{P}^*$  is defined by:

$$\mathcal{P}^* = \{\mathbf{x} \in \mathcal{F} | \mathbf{x} \text{ is Pareto-optimal}\}.$$

**Definition 5** The *Pareto front*  $\mathcal{PF}^*$  is defined by:

$$\mathcal{PF}^* = \{\mathbf{f}(\mathbf{x}) \in \mathbb{R}^k | \mathbf{x} \in \mathcal{P}^*\}.$$

Therefore, our aim is to obtain the Pareto optimal set from the set  $\mathcal{F}$  of all the decision variable vectors that satisfy (2) and (3). Note however, that in practice, not all the Pareto optimal set is normally desirable or achievable, and decision-makers tend to prefer certain types of solutions or regions of the Pareto front [16].

## Open research areas

In spite of the intense research activity of the last 25 years in this area, there are still some open research areas that are worth exploring in the next few years. The following is a non-comprehensive list of them:

1. Algorithmic design
2. Scalability
3. Dealing with expensive objective functions
4. Hyper-heuristics.

Next, we briefly discuss some of the most representative research that has been conducted on these topics.

### Algorithmic design

In their origins, MOEAs were very simple and naive. A good example of this is the Vector Evaluated Genetic Algorithm (VEGA) [148] in which the population of a simple genetic algorithm was subdivided into as many subpopulations as the number of objectives of the MOP to be solved (only problems with two objectives were normally considered at that time). Then, solutions in each subpopulation were selected based on their performance on a single objective (e.g., individuals in the first subpopulation were selected based on the performance on the first objective). Then, the individuals of all the subpopulations were shuffled with the aim of recombining solutions that were the best in the first objective with those that were the best in the second objective. When combined with proportional selection (e.g., the roulette-wheel method), VEGA produced solutions similar to those obtained with the use of a linear aggregating function that combines all the

objective functions into a single scalar value [31]. In spite of the limitations of VEGA, some researchers eventually found applications in which this sort of scheme could be useful (see for example [27]).

Linear aggregating functions were among the most popular approaches adopted in the early days of MOEAs [67], but their incapability of dealing with non-convex Pareto fronts was soon pointed out by some researchers (see for example [36]). Nevertheless, linear aggregating functions and other naive approaches, such as lexicographic ordering have survived in the EMO literature for many years [29].

Goldberg proposed in his seminal book on genetic algorithms [57] a mechanism called *Pareto ranking* for the selection scheme of a MOEA. The core idea of Pareto ranking is to rank the population of an evolutionary algorithm based on Pareto optimality, such that the nondominated solutions obtain the highest (best) possible rank and are sampled at the same rate (i.e., all nondominated solutions have the same probability of survival). Since Goldberg did not provide a specific algorithm for Pareto ranking, several implementations were developed based on his proposal. From them, the two main ones were those provided in the Multi-Objective Genetic Algorithm (MOGA) of Fonseca and Fleming [53] and the Nondominated Sorting Genetic Algorithm (NSGA) of Srinivas and Deb [153]. In the first, the ranking was done in a single pass (by comparing each individual with respect to everybody else, in terms of Pareto optimality), whereas the second required the creation of several layers of solutions, which involved re-ranking the population several times (i.e., NSGA was more computationally expensive than MOGA).

Goldberg [57] realized that in MOEAs, diversity would be a key issue if we aimed to generate as many elements of the Pareto optimal set as possible in a single algorithmic execution. This gave rise to the use of a mechanism that was eventually called *density estimator*, whose task is to maintain different (nondominated) solutions in the population, thus avoiding convergence to a single one (something that eventually happens with any evolutionary algorithm because of stochastic noise [57]). MOGA [53] and NSGA [153] used fitness sharing [58] as their density estimator, but a wide variety of other approaches have been proposed since then: clustering [184], adaptive grids [85], crowding [40], entropy [129] and parallel coordinates [74], among others.

In the late 1990s, another mechanism was incorporated in MOEAs: elitism. The idea of *elitism* is to retain the best solutions obtained by a MOEA so that they are not destroyed by the evolutionary operators (e.g., crossover and mutation). However, since all nondominated solutions are considered equally good (unless we have some preference information), this leads to the generation of a large number of solutions. Zitzler realized this when developing the Strength Pareto Evolutionary Algorithm (SPEA) [184] and also observed that retaining such a large number of solutions diluted the

selection pressure. Thus, he proposed not only to use an external archive to store the nondominated solutions generated by his MOEA, but also proposed to prune such an archive once a certain (user-defined) limit was reached. For this sake, he adopted clustering. Elitism is important not only for practical reasons (it is easier to compare the performance of two MOEAs that produce the same number of nondominated solutions), but also for theoretical reasons, since it has been proved that such a mechanism is required in a MOEA to guarantee convergence [141].

Pareto-based MOEAs were very popular in the mid-1990s, but few of the many approaches that were proposed at that time have been actually used by other researchers. With no doubt, the most popular of the Pareto-based MOEAs has been the Nondominated Sorting Genetic Algorithm-II (NSGA-II) [40] which uses a more efficient ranking scheme (called *nondominated sorting*) than its predecessor (NSGA), and adopts a clever mechanism called crowded comparison operator (which does not require any parameters), as its density estimator. NSGA-II is still used today by many researchers, in spite of the well-known limitations of its crowded comparison operator when dealing with MOPs having more than three objectives<sup>3</sup> (the so-called *many-objective optimization problems* [29]).

For over 10 years, Pareto-based MOEAs were, by far, the most popular approaches in the specialized literature. In 2004, a different type of algorithmic design was proposed, although it remained underdeveloped for several years: indicator-based selection. The core idea of this sort of MOEA was introduced in the Indicator-Based Evolutionary Algorithm (IBEA) [183] which consists of an algorithmic framework that allows the incorporation of any performance indicator into the selection mechanism of a MOEA. IBEA was originally tested with the hypervolume [182] and the binary  $\epsilon$  indicator [183].

Indicator-based MOEAs were initially seen as a curiosity in the field, since it was not clear what were their advantages other than providing an alternative mechanism for selecting solutions. However, when the limitations of Pareto-based selection for dealing with many-objective problems became clear, researchers started to get interested in indicator-based MOEAs, which did not seem to have scalability limitations. Much of the interest in this area was produced by the introduction of the S Metric Selection Evolutionary Multiobjective Algorithm (SMS-EMOA) [46]. SMS-EMOA randomly generates an initial population and then produces a single solution per iteration (i.e., it uses steady-state selection) using the crossover and mutation operators from NSGA-II. Then, it applies nondominated sorting

<sup>3</sup> In fact, there is empirical evidence indicating that the crowded comparison operator has difficulties even in MOPs with only 3 objectives (see for example [88]).

(as in NSGA-II). When the last nondominated front has more than one solution, SMS-EMOA uses hypervolume [182] to decide which solution should be removed. Beume et al. [13] proposed a new version of SMS-EMOA in which the hypervolume contribution is not used when, in the nondominated sorting process, we obtain more than one front (i.e., the hypervolume is used as a density estimator). In this case, they use the number of solutions that dominate to a certain individual (i.e., the solution that is dominated by the largest number of solutions is removed). This makes SMS-EMOA a bit more efficient. However, since this MOEA relies on the use of exact hypervolume contributions, it becomes too computationally expensive as we increase the number of objectives [12].

SMS-EMOA started a trend for designing indicator-based MOEAs (several of which rely on the hypervolume indicator) although it is worth indicating that in such approaches, the performance indicator has been mostly used as a density estimator (see for example [77]). The use of “pure” indicator-based selection mechanisms has been very rare in the specialized literature (see for example [114]).

At this point, an obvious question is: why is that the *hypervolume* is such an attractive choice for indicator-based selection? The hypervolume (which is also known as the  $\mathcal{S}$  metric or the Lebesgue measure) of a set of solutions measures the size of the portion of objective space that is dominated by those solutions collectively. One of its main advantages are its mathematical properties, since it has been proved that the maximization of this performance measure is equivalent to finding the Pareto optimal set [52]. Additionally, empirical studies have shown that (for a certain number of points previously determined) maximizing the hypervolume indeed produces subsets of the true Pareto front [46,85]. Also, the hypervolume assesses both convergence and, to a certain extent, also the spread of solutions along the Pareto front (although without necessarily enforcing a uniform distribution of solutions). However, there are several issues regarding the use of the hypervolume. First, the computation of this performance indicator depends of a reference point, which can influence the results in a significant manner. Some people have proposed to use the worst objective function values in the current population, but this requires scaling of the objectives. Nevertheless, the most serious limitation of the hypervolume is its high computational cost. The best algorithms known to compute hypervolume have a polynomial complexity on the number of points used, but such complexity grows exponentially on the number of objectives [12]. This has triggered a significant amount of research regarding algorithms that can reduce the computational cost of computing the hypervolume and the hypervolume contributions,

which is what we need for a hypervolume-based MOEA<sup>4</sup> (see for example [33,62,81,93,142]).

An obvious alternative to deal with this issue is to approximate the actual hypervolume contributions. This is the approach adopted by the Hypervolume Estimation Algorithm for Multi-Objective Optimization (HyPE) [6] in which Monte Carlo simulations were used to approximate exact hypervolume values. In spite of the fact that HyPE can efficiently solve MOPs having a very large number of objectives, the results are not as competitive as when using exact hypervolume contributions.

Another possibility is to use a different performance indicator whose computation is relatively inexpensive. Unfortunately, the hypervolume is the only unary indicator which is known to be Pareto compliant [185], which makes less attractive the use of other performance indicators. Nevertheless, there are some other performance indicators which are weakly Pareto compliant, such as  $R2$  [17] and the Inverted Generational Distance plus (IGD+) [79]. Although several efficient and effective indicator-based MOEAs have been proposed around these performance indicators (see for example [18,72,97,105,106,160]), their use has remained relatively rare in the specialized literature.

In 2007, a different sort of approach was proposed, quickly attracting a lot of interest: the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) [178]. The idea of using decomposition (or scalarization) methods was originally proposed in mathematical programming more than 20 years ago [35] and it consists in transforming an MOP into several single-objective optimization problems which are then solved to generate the nondominated solutions of the original problem. Unlike linear aggregating functions, the use of scalarization (or decomposition) methods allows the generation of non-convex portions of the Pareto front and works even in disconnected Pareto fronts. MOEA/D presents an important advantage with respect to methods proposed in the mathematical programming literature [such as Normal Boundary Intersection (NBI) [35]]: it uses neighborhood search to solve simultaneously all the single-objective optimization problems generated from the transformation. Additionally, MOEA/D is not only effective and efficient, but can also be used for solving problems with more than three objectives although in such cases it will require higher population sizes.

Decomposition-based MOEAs became fashionable at around 2010 and have remained as an active research area since then [144]. In fact, this sort of approach influenced the development of the Nondominated Sorting Genetic

<sup>4</sup> See also:

<http://ls11-www.cs.uni-dortmund.de/rudolph/hypervolume/start>,  
<http://people.mpi-inf.mpg.de/~fried/HYP/>, <http://iridia.ulb.ac.be/~manuel/hypervolume>.

Algorithm-III (NSGA-III) [38] which adopts both decomposition and reference points to deal with many-objective problems. However, it was recently found that decomposition-based MOEAs do not work properly with certain Pareto front geometries [80]. This will certainly trigger a lot of research in the next few years, given the popularity of decomposition-based MOEAs.

## Scalability

As has been pointed out, in the early days of MOEAs, their use was frequently limited to solving problems having only two or three objectives. However, over the years, the need for tackling many-objective problems became more evident. It was soon identified that scalability in objective function space is a serious limitation of Pareto-based MOEAs [75].

However, it is interesting to notice that when Schütze et al. [150] studied the actual source of difficulty in many-objective problems, they concluded that adding more objectives to an MOP does not necessarily makes it harder. According to this study, the difficulty is really associated to the intersection of the descent cones of the objectives (these descent cones are obtained with the combination of the gradients of each objective). This was somehow corroborated by an empirical study conducted by Ishibuchi et al. [78] in which it was shown that NSGA-II could properly solve many-objective knapsack problems in which the objectives were highly correlated. So, the question arises: why is that many-objective problems turn out to be difficult to solve in practice when using Pareto-based MOEAs? A series of experimental [131,165] and analytical studies [32,78,86] have identified the following limitations of Pareto-based MOEAs in many-objective problems:

1. *Deterioration of the search ability* The proportion of non-dominated solutions in a population increases rapidly with the number of objectives [49]. According to Bentley et al. [9], the number of nondominated  $k$ -dimensional vectors on a set of size  $n$  is  $O(\ln^{k-1} n)$ . This implies that in problems with a large number objectives, the selection of solutions is carried out almost at random or guided by the density estimator. In fact, Mostaghim and Schmeck [119] experimentally showed that a random search optimizer could achieve better results than NSGA-II [40] in a problem with ten objectives. This problem is the reason why most indicator-based MOEAs are able to tackle many-objective problems simply by using a more powerful density estimator that guides the search.
2. *Dimensionality of the Pareto front* Due to the ‘curse of dimensionality’ the number of points required to represent accurately a Pareto front increases exponentially with the number of objectives. The number of points necessary to represent a  $k$ -dimensional Pareto front with

resolution  $r$  is given by  $O(kr^{k-1})$  (e.g., see [151]). This poses a challenge both to the data structures to efficiently manage that number of points in the population as well as in the external archive and to the density estimators to achieve an even distribution of the solutions along the Pareto front. In fact, this is a challenge even for performance indicators [78]. In practice, most MOEAs tend to use relatively small population sizes (less than 300 individuals) even when tackling MOPs with more than six objectives in spite of the fact that such population sizes are clearly inappropriate for sampling such high-dimensional Pareto fronts.

3. *Visualization of the Pareto front* Clearly, with more than three objectives is not possible to plot the Pareto front as usual. This is a serious problem since visualization plays a key role for a proper decision-making process. In recent years, a number of visualization techniques have been proposed for many-objective problems (see for example [161]), and this is still an active research area (see for example [14,70,156]).

In order to properly deal with many-objective optimization problems, three main approaches have been normally adopted [8,96,163]:

1. As mentioned before, the use of indicator-based MOEAs has been an important research trend to deal with many-objective optimization problems, in spite of the limitations of some performance indicators such as the hypervolume (see for example [82]).
2. One interesting possibility that was adopted in the early days of many-objective optimization was the use of an optimality relation that yields a solution ordering finer than that produced by Pareto optimality. These are normally called relaxed forms of Pareto dominance. Some examples are:  $k$ -optimality [50], preference order ranking [42], the favour relation [155], and a method that controls the dominance area [145], among others. Besides providing a richer ordering of the solutions, these relations obtain an optimal set that it is usually a subset of the Pareto optimal set.
3. Another interesting approach which is now rarely used is dimensionality reduction in which we reduce the number of objectives of the MOP either during the search process or in an a posteriori manner, during the decision-making process [19,100,146]. The main aim of reduction techniques is to identify redundant objectives (or at least partially redundant) in order to discard them. A redundant objective is one that can be removed without changing the dominance relation induced by the original objective set. Evidently, if each objective is conflicting with respect to every other objective, no reduction is possible. However, this is rarely the case in practice.

Many other approaches are possible for tackling many-objective problems, including, for example, the use of alternative ranking schemes (different from nondominated sorting) (see for example [54]), the use of machine learning techniques (as in MONEDA [108]), or approaches such as the two-archive MOEA, which uses one archive for convergence and another for diversity [131].

### Dealing with expensive objective functions

In spite of the several advantages that MOEAs can offer for solving complex MOPs (e.g., ease of use and generality), their most important limitation is that they normally require a relatively high number of objective function evaluations to produce a reasonably good approximation of the Pareto front. The reason for this is that MOEAs need to sample the search space in order to identify an appropriate search direction, since they are stochastic search techniques. This is, indeed, a serious limitation and, in some cases, it can make MOEAs inappropriate for solving certain real-world MOPs in which their computational cost becomes prohibitive (e.g., applications in aeronautical and aerospace engineering [122]).

In general, MOEAs can become computationally unaffordable for an application when:

- The evaluation of the fitness functions is computationally expensive (i.e., it takes from minutes to hours, depending on the quality or granularity of the model and the available computational resources).
- The fitness functions cannot be defined in an algebraic form (e.g., when the fitness functions are generated by a computational simulation of the physics of the real system).
- The total number of evaluations of the fitness functions is limited by some financial constraints (i.e., there is a financial cost involved in computing the fitness functions and we cannot exceed a certain pre-defined budget).

As we get access to more computational power each year at more affordable prices, the interest in pursuing research in the development of MOEAs for solving computationally expensive MOPs has significantly increased in the last few years. The main approaches that have been developed in this area can be roughly divided into three main groups [143]:

1. *Parallelism* The use of parallel processing is perhaps the most obvious choice for solving computationally expensive MOPs, particularly with the decrease in the cost of high-speed multi-core processors (see for example [37,83,110,124,139]). Something interesting, however, is that in spite of the existence of an important number of papers on parallel MOEAs [159], basic research

in this area has remained scarce since the origins of MOEAs [29]. Most papers on parallel MOEAs focus on applications [173] or relatively straightforward parallel extensions of well-established MOEAs such as NSGA-II [130] or SMS-EMOA [71]. For many years, the emphasis was placed on developing synchronous parallel MOEAs, but in recent years, the development of asynchronous implementations (which are more appropriate for the heterogeneous computer architectures that are more common nowadays) has become more common [41,68,171,174].

2. *Surrogates* When using this approach, an empirical model that approximates the real problem is built through the use of information gathered from actual objective function evaluations [43,121,157]. Then, the empirical model (on which evaluating the fitness function is computationally inexpensive) is used to predict promising new solutions [157]. Current functional approximation models include Polynomials (response surface methodologies [47,172]), artificial neural networks (e.g., multi-layer perceptrons (MLPs) [5], radial-basis function (RBF) networks [2,177], Gaussian processes [15,179], support vector machines [3,4,176] and Kriging [111,126] models. Although frequently used in engineering applications, surrogate methods can normally be adopted only in problems of low dimensionality, which is an important limitation when dealing with real-world MOPs. Additionally, surrogate models tend to lack robustness which is also an important issue in optimization problems. Nevertheless, there has been recent research oriented towards overcoming the scalability and robustness limitations of surrogate methods (see for example [102,125,134,175]).
3. *Fitness inheritance* This approach was originally proposed by Smith et al. [152] with the aim of reducing the total number of fitness function evaluations performed by a (single-objective) evolutionary algorithm. This mechanism works as follows: when assigning fitness to an individual, in some cases we evaluate the objective function as usual, but the rest of the time, we assign as the fitness value of an individual the average of the fitness of its parents. This allows saving one fitness function evaluation. This idea is based on the assumption of similarity of an offspring to its parents. Evidently, fitness inheritance must not be always applied, since the algorithm needs to use the true fitness function value from time to time, in order to obtain enough information to guide the search. The percentage of time in which fitness inheritance is applied is called *inheritance proportion*. If this inheritance proportion is one (i.e., 100%), the algorithm is most likely to prematurely converge [24]. Extending fitness inheritance to multiobjective optimization involves several issues, mainly related to its apparent limitation for dealing with non-convex Pareto fronts [45]. However,

some researchers have managed to successfully adapt fitness inheritance to MOEAs [55,128,135,169], reporting important savings on the total number of objective function evaluations performed.

## Hyper-heuristics

A hyper-heuristic is a search method or learning mechanism for selecting or generating heuristics to solve computational search problems [20]. Hyper-heuristics are high-level approaches that operate on a search space of heuristics or on a pool of heuristics instead of the search space of the problem at hand. Hyper-heuristics have been promoted with the aim to provide more general search methodologies. Typically, simple heuristics tend to work well on a particular type of problem. However, when facing a new problem or even slightly modified instances of the same problem, such heuristics tend to perform poorly. Additionally, identifying which heuristic works efficiently on a certain problem is a very tedious and time-consuming task whose computational cost may become prohibitive in some applications.

Burke et al. [21] proposed a taxonomy of hyper-heuristics considering two dimensions:

1. The nature of the heuristics' search space, and
2. The different sources of feedback information.

Regarding the nature of the search space, there are two options:

1. Heuristic selection, which are methodologies for choosing existing heuristics,
2. Heuristic generation, which are methodologies for generating new heuristics from the components of existing ones.

Regarding the source of feedback information obtained during the search process, there are three options:

1. *No-learning*, in which there is no learning mechanism and the heuristic selection is based on either a random or an exhaustive process,
2. *Offline learning*, in which knowledge is gathered in the form of rules from a set of training instances, that will hopefully generalize to solve unseen instances, and
3. *Online learning*, in which the learning takes place while the algorithm is solving an instance of a problem.

The use of collaborative approaches which work as hyper-heuristics can be found across Operations Research, Computer Science and Artificial Intelligence. Although the ideas behind hyper-heuristics can be traced back to the early 1960s in single-objective optimization, until relatively

recently, their potential had not been explored in multiobjective optimization. Early attempts in this field date back to 2005, when hyper-heuristics started to be used to solve multiobjective combinatorial optimization problems, such as space allocation and timetabling [22], decision-tree induction algorithms [7], bin packing and cutting stock problems [59], integration and test order problems [63,64,107], spanning trees [89], job shop scheduling [162], knapsack problems [90] and software module clustering [91,92], among others.

Multiobjective hyper-heuristics for continuous search spaces are still rare in the specialized literature. For example Vrugt et al. [164], proposed the Multi-ALgorithm Genetically Adaptive Method (AMALGAM), which is an online selection hyper-heuristic that operates similarly to NSGA-II [40]. However, in this case, the offspring are also created using the variation operators of other stochastic search methods, such as differential evolution [154], particle swarm optimization [84] and adaptive metropolis search [65]. Although all these methods participate during the optimization process, those showing the highest reproductive success are favored. Additionally, the initialization of the population adopts Latin hypercubes sampling.

León et al. [95] proposed the Metaheuristics-based Extensible Tool for Cooperative Optimization (METCO) which is based on the island model and the cooperation of a set of MOEAs, which grants more computational resources to those algorithms that show a more promising behavior. A coordinator node is in charge of maintaining the global solution and selecting the configurations that are executed on the islands. A configuration consists of a MOEA plus the variation operators and the set of parameters which define them (population size, mutation and crossover rates, etc.). These parameters are defined by the user. The global solution set is obtained by merging local results achieved by each of the islands and its size is limited using the crowding distance operator. Besides the global stopping criterion, a local stopping criterion is defined for the execution of the MOEAs on the islands. When the local stopping criterion is reached, the configuration is scored using a performance indicator. Then, the coordinator applies the hyper-heuristic, selecting the configuration that will continue executing on the idle island. If the configuration has changed, the subpopulation is replaced by a random subset of the currently global solution.

Wang and Li [170] proposed the Multi-Strategy Ensemble Dynamic Multi-Objective Evolutionary Algorithm (MS-MOEA), which is an offline selection hyper-heuristic that adopts the fundamental principle of AMALGAM of combining different variation operators. This approach works as  $\epsilon$ -MOEA [39] with an external archive that is pruned to a limit size using the hypervolume indicator. The heuristics for generating new individuals consists of two re-initialization techniques, which are based on random sampling and Gaussian distribution with mean around the previous optimal

solutions; the genetic operators SBX (Simulated Binary Crossover) and polynomial-based mutation; the Differential Evolution strategies DE/rand/1 and DE/current to best/1; as well as Gaussian mutation. This approach was designed to solve dynamic multiobjective optimization problems (i.e., problems in which either the Pareto optimal set or the Pareto optimal front change over time). If there is an environmental change, then one re-initialization technique is applied under certain probability. Genetic operators are used at early stages of the evolutionary process. Once convergence is detected, Differential Evolution is adopted to enhance diversity. Under this scheme, each strategy creates an offspring. After a fixed number of solutions is created, Gaussian mutation is launched for escaping from local optima. It is worth noticing that convergence is detected when the external archive has been full during a certain number of generations.

McClymont et al. [112] proposed the Markov Chain hyper-heuristic (MCHH), which is a selection heuristic with online learning working as a  $(\mu + \lambda)$ -evolution strategy with an unbounded external archive. The pool of heuristics is composed of four variation operators: mutation, replication, transposition, and cloning. All of them operate on the decision variables of a given solution. The heuristic selection mechanism uses a Markov chain, which can be seen as a directed graph where every vertex is connected to each other vertex and to itself. A vertex represents a state (heuristic), and the weight of an edge out represents the probability of moving from the current state to the destination state. All edges out of a state must sum one. The Markov chain stochastically selects the next heuristic biased by these probabilities. The selected heuristic is employed during a certain number of generations. Then, its performance is measured by counting the number of parents that were dominated by each offspring produced by such heuristic. This performance is used to update the corresponding probability in the Markov chain using reinforcement learning.

Maashi [104] proposed an online learning selection choice function based hyper-heuristic framework for multiobjective optimization. Her proposed approach controls and combines the strengths of three well-known MOEAs (NSGA-II, SPEA2, and MOGA), which are adopted as her low-level heuristics.

Gonçalves et al. [60] proposed the MOEA/D Hyper-Heuristic (MOEA/D-HH), which is an online selection hyper-heuristic that is coupled to a MOEA/D variant [178]. In this approach, an adaptive choice function is used to determine the Differential Evolution (DE) strategy that should be applied to generate individuals at each iteration.

Walker and Keedwell [166] proposed the indicator-based multiobjective sequence-based hyper-heuristic (MOSSH) algorithm. This seems to be the first attempt to use a hyper-heuristic in many-objective problems. This online selection hyper-heuristic is based on a hidden Markov model to deter-

mine the mutation strategy to be applied for generating a single child from the current parent. Thus, this approach works as a  $(1 + 1)$ -evolution strategy complemented with an external archive, which keeps all the nondominated solutions discovered so far. The pool of seven mutation heuristics consists primarily of (1) adding noise to the current solution using three different continuous probability distributions, and (2) replacing the parent (or only a variable) with another one, whether randomly created or taken from the archive. At each iteration, the child replaces the parent if the former dominates the second. However, in a further paper [167], this comparison rule was changed by strategies based on the hypervolume indicator [182], the *favour* relation [44] and the average rank [10]. Moreover, the hidden Markov model is updated if the child is added to the archive and if it was better than the parent.

More recently, Hernández Gómez and Coello [73] proposed a hyper-heuristic which combines the strengths and compensates for the weaknesses of different scalarizing functions. The selection is conducted through an indicator called *s-energy* [69], which measures the even distribution of a set of points in  $k$ -dimensional manifolds.

## Challenges

After reviewing some of the most relevant research done on the topics selected in “Open research areas”, we provide here some of the challenges that lie ahead in each of them.

- *Algorithmic design* Although there has been some debate regarding the paradigm that will become more common in the next few years, it seems quite obvious at this point that decomposition-based MOEAs are the clear winner so far. For indicator-based MOEAs to become more popular, other performance indicators need to be proposed. Although there has been some research activity in this regard (see for example [113,140]) none of these other performance indicators has become as popular as the hypervolume. Another interesting idea is the combination of performance indicators in order to take advantage of their strengths and compensate for their limitations (see for example [48]).

Nevertheless, a more relevant question in this area is the following: can we design MOEAs in a different way? This is a very important question, since algorithmic development has been at the heart of research on EMO and it is quite important that new algorithmic proposals are made in the next few years in order to keep this research area alive. Evidently, it is not trivial to produce a selection mechanism that is not Pareto-based, decomposition-based or indicator-based, but this is indeed possible. For example, Molinet Berenguer and



Coello [11,118], proposed an approach that transforms an MOP into a linear assignment problem using a set of weight vectors uniformly scattered. Uniform design is adopted to obtain the set of weights, and the Kuhn–Munkres (Hungarian) algorithm [87] is used to solve the resulting assignment problem. This approach was found to perform quite well (and at a low computational cost) even in many-objective optimization problems. As such, this approach does not belong to any of the three algorithmic families previously discussed and it constitutes an intriguing new family of MOEAs.

So, it should be clear that a challenge is to develop selection mechanisms for MOEAs that are different from those that have been developed so far. Additionally, popularizing the use of such MOEAs is certainly another (perhaps more difficult) challenge.

Another challenge in this area is to gain a deeper understanding of the limitations of current MOEAs. For example, knowing that some scalarizing functions offer advantages over others is very useful to design good decomposition-based and even indicator-based MOEAs (see for example [127]).

Another interesting idea is to combine components of MOEAs under a single framework that allows to exploit their advantages. This is the basic idea of Borg [66], which adopts  $\epsilon$ -dominance, a measure of convergence speed called  $\epsilon$  progress, an adaptive population size, multiple recombination operators and a steady-state selection mechanism.

A relevant question in this regard is if this sort of scheme could lead us to the automated design of MOEAs as has been suggested by researchers from automated parameter tuning for single-objective evolutionary algorithms [76].

- *Scalability* This area presents several challenges. For example, what sort of performance indicator should we use to assess diversity in many-objective problems? In low-dimensional Pareto fronts, the aim is normally to achieve a uniform distribution of solutions along the Pareto front. However, what would be a desirable distribution in an MOP having, for example, ten objectives, if we are using only 300 solutions to sample it? In recent years, the use of some performance indicators such as the Riesz  $s$ -energy [69] have shown promise in this regard, but more research in this area is required.

In contrast with the significant interest that researchers have had on many-objective optimization in recent years, scalability in decision variable space (i.e., the solution of the so-called large-scale problems) has been only recently studied in the context of multiobjective optimization (see for example [103,116,181]). This is remarkable if we consider that large-scale multiobjective optimization problems (i.e., problems having more than 100 decision variables) are not rare in real-world applications (see for

example [101]). In this area, the use of cooperative coevolutionary approaches (which have been very successful in single-objective large-scale optimization) is the most common research trend. However, new test suites are required for large-scale multiobjective optimization and some work has already been done in this direction (see for example [25]).

Another challenge in this area is the solution of large-scale many-objective problems which is a very recent research topic in which some work has been recently published (see for example [23,180]).

- *Dealing with expensive objective functions* Other approaches for dealing with expensive objective functions are also possible. For example, some researchers have adopted cultural algorithms [30,34,133,136], which gather knowledge during the evolutionary process and use it to perform a more efficient search at the expense of a significantly larger memory usage. Cultural algorithms were proposed by Reynolds [137,138], as an approach that tries to add domain knowledge to an evolutionary algorithm during the search process, avoiding the need to add it a priori. This approach uses, in addition to the population space commonly adopted in evolutionary algorithms, a belief space, which encodes the knowledge obtained from the search points and their evaluation, in order to influence the evolutionary operators that guide the search. However, the belief space is commonly designed based on the group of problems that is to be solved. At each generation, the cultural algorithm selects some exemplar individuals from the population, in order to extract information from them that can be useful during the search. Such an information is used to update the belief space. The belief space will then influence the operators of the evolutionary algorithm, to transform them into informed operators that can enhance the search process. Cultural algorithms can be an effective means of saving objective function evaluations, but since a map of decision variable space must be kept at all times, their cost will soon become prohibitive even for problems of moderate dimensionality (in decision variable space).
- *Hyper-heuristics* We certainly need theoretical studies on the use of hyper-heuristics in multiobjective optimization and some work in that direction has been already done. Qian et al. [132] provided a theoretical study on the effectiveness of selection hyper-heuristics for multiobjective optimization. This paper concluded that applying selection hyper-heuristics to any of the three major components of a MOEA (selection, mutation and acceptance), can exponentially speed up the optimization process.

Another interesting idea is to combine different performance indicators within an indicator-based MOEA as proposed by Falcón-Cardona and Coello [48]. In this

case,  $IGD^+$ ,  $\epsilon+$ ,  $\Delta_p$  and  $R2$  are adopted as possible density estimators (i.e., the low-level heuristics).

One more interesting area of research would be the use of genetic programming to generate components of MOEAs (e.g., evolutionary operators or even scalarizing functions) that can improve their performance when adopted within a multiobjective hyper-heuristic.

## Conclusions

As we have seen in this paper, EMO still has plenty of topics to be explored. However, it is important to emphasize that some of them require us to move outside the main stream of research currently being conducted in this area. Besides the topics previously indicated, there are several more that have been already explored, but are worth re-visiting. For example, we need new performance indicators, particularly for many-objective optimization. For instance, we have very few performance indicators for assessing diversity in many-objective optimization (see for example [99,168]), but there are other interesting choices that are also worth exploring (see for example, the  $s$ -energy indicator [69]).

It is also important to design new mechanisms (e.g., operators, encodings, etc.) for MOEAs based on specific features of real-world problems (e.g., variable length encodings, expensive objective functions, uncertainty, etc.). See for example [98]. The way in which coevolutionary approaches can help us to solve complex multiobjective optimization problems is another interesting venue for future research. Besides large-scale problems, coevolution can help us in other domains (e.g., dynamic multiobjective optimization problems [56]), but its potential has been scarcely studied in this area (see [117]).

However, it is important to keep in mind that a great source of diversity regarding research ideas is the knowledge coming from other disciplines. For example, EMO has adopted advanced data structures (e.g., red–black trees [51]), concepts from computational geometry (e.g., convex hulls [26,109], quadtrees [120] and Voronoi maps [123]), and from economics (e.g., game theory [61]) to design novel MOEAs and operators.

We also need to explore more ways of bridging the gap between Operations Research and EMO. An example is the development of hybrid approaches that combine a MOEA with a mathematical programming technique (see for example [94]). Another one is the use of the Karush–Kuhn–Tucker optimality conditions to estimate proximity of a solution to the Pareto optimal set (see [1]).

Summarizing, we claim that EMO is still a very promising research area which should remain active for several more years. However, we need to increase diversity in our research

topics and to be more disruptive. If we only do work by analogy, we will suffer stagnation!

**Acknowledgements** The first author acknowledges support from CONA-CyT project no. 2016-01-1920 (*Investigación en Fronteras de la Ciencia 2016*) and from a project from the 2018 SEP-Cinvestav Fund (application no. 4).

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

1. Abouhawwash M, Deb K (2016) Karush–Kuhn–Tucker proximity measure for multi-objective optimization based on numerical gradients. In: 2016 genetic and evolutionary computation conference (GECCO'2016), ACM Press, Denver, Colorado, USA, 20–24, pp 525–532 (ISBN: 978-1-4503-4206-3)
2. Akhtar T, Shoemaker CA (2016) Multi objective optimization of computationally expensive multi-modal functions with RBF surrogates and multi-rule selection. *J Glob Optim* 64(1):17–32
3. Alves RVH, Reynoso-Meza G (2018) Multi-objective Support Vector Machines Ensemble Generation for Water Quality Monitoring. In: 2018 IEEE congress on evolutionary computation (CEC'2018), IEEE Press, Rio de Janeiro, Brazil, July 8–13, pp 608–613 (ISBN: 978-1-5090-6017-7)
4. Aytug H, Sayin S (2009) Using support vector machines to learn the efficient set in multiple objective discrete optimization. *Eur J Oper Res* 193(2):510–519
5. Azzouz N, Bechikh S, Said LB (2014) Steady state IBEA assisted by MLP neural networks for expensive multi-objective optimization problems. In: 2014 genetic and evolutionary computation conference (GECCO 2014), ACM Press, Vancouver, Canada, July 12–16, pp 581–588 (ISBN: 978-1-4503-2662-9)
6. Bader J, Zitzler E (2011) HypE: an algorithm for fast hypervolume-based many-objective optimization. *Evol Comput* 19(1):45–76
7. Basgalupp MP, Barros RC, Pogdorelec V (2015) Evolving decision-tree induction algorithms with a multi-objective hyper-heuristic. In: Proceedings of the 30th annual ACM symposium on applied computing (SAC'15), ACM Press, New York, USA, pp 110–117 (ISBN: 978-1-4503-3196-8)
8. Bechikh S, Elarbi M, Said LB (2017) Many-objective optimization using evolutionary algorithms: a survey. In: Bechikh S, Datta R, Gupta A (eds) Recent advances in evolutionary multi-objective optimization. Springer, Switzerland, pp 105–137 (ISBN: 978-3-319-42977-9)
9. Bentley JL, Kung HT, Schkolnick M, Thompson CD (1978) On the average number of maxima in a set of vectors and applications. *J Assoc Comput Mach* 25(4):536–543
10. Bentley PJ, Wakefield JP (1997) Finding acceptable solutions in the pareto-optimal range using multiobjective genetic algorithms. In: Chawdhry PK, Roy R, Pant RK (eds) Soft computing in engineering design and manufacturing, Part 5. Springer, London, pp 231–240 (Presented at the 2nd On-line World Conference on Soft Computing in Design and Manufacturing (WSC2))
11. Berenguer JAM, Coello CAC (2015) Evolutionary many-objective optimization based on Kuhn–Munkres' algorithm. In: Gaspar-Cunha A, Antunes CH, Coello CC (eds) Evolutionary

- multi-criterion optimization, 8th international conference, EMO 2015, Springer, Guimarães, Portugal. Lecture notes in computer science, vol 9019, March 29–April 1, pp 3–17
12. Beume N, Fonseca CM, Lopez-Ibanez M, Paquete L, Vahrenhold J (2009) On the complexity of computing the hypervolume indicator. *IEEE Trans Evol Comput* 13(5):1075–1082
  13. Beume N, Naujoks B, Emmerich M (2007) SMS-EMOA: multi-objective selection based on dominated hypervolume. *Eur J Oper Res* 181(3):1653–1669
  14. Blasco X, Herrero JM, Sanchis J, Martinez M (2008) A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization. *Inf Sci* 178(20):3908–3924
  15. Bradford E, Schweidtmann AM, Lapkin A (2018) Efficient multi-objective optimization employing Gaussian processes, spectral sampling and a genetic algorithm. *J Glob Optim* 71(2):407–438
  16. Branke J, Deb K (2005) Integrating user preferences into evolutionary multi-objective optimization. In: Jin Y (ed) *Knowledge incorporation in evolutionary computation*. Springer, Berlin, pp 461–477 (ISBN: 3-540-22902-7)
  17. Brockhoff D, Wagner T, Trautmann H (2012) On the properties of the R2 indicator. In: 2012 genetic and evolutionary computation conference (GECCO'2012), ACM Press, Philadelphia, USA, pp 465–472 (ISBN: 978-1-4503-1177-9)
  18. Brockhoff D, Wagner T, Trautmann H (2015) R2 indicator-based multiobjective search. *Evol Comput* 23(3):369–395
  19. Brockhoff D, Zitzler E (2009) Objective reduction in evolutionary multiobjective optimization: theory and applications. *Evol Comput* 17(2):135–166
  20. Burke EK, Gendreau M, Hyde M, Kendall G, Ochoa G, Özcan E, Qu R (2013) Hyper-heuristics: a survey of the state of the art. *J Oper Res Soc* 64:1695–1724
  21. Burke EK, Hyde M, Kendall G, Ochoa G, Özcan E, Woodward JR (2010) A classification of hyper-heuristic approaches. In: Gendreau M, Potvin JY (eds) *Handbook of metaheuristics*. Springer, Boston, pp 449–468 (ISBN: 978-1-4419-1663-1)
  22. Burke EK, Silva JDL, Soubiega E (2005) Multi-objective hyper-heuristic approaches for space allocation and timetabling. In: *Meta-heuristics: progress as real problem solvers, selected papers from the 5th metaheuristics international conference (MIC 2003)*, Springer, pp 129–158
  23. Cao B, Zhao J, Lv Z, Liu X, Yang S, Kang X, Kang K (2017) Distributed parallel particle swarm optimization for multi-objective and many-objective large-scale optimization. *IEEE Access* 5:8214–8221
  24. Chen JH, Goldberg DE, Ho SY, Sastry K (2002) Fitness inheritance in multi-objective optimization. In: Langdon WB, Cantú-Paz E, Mathias K, Roy R, Davis D, Poli R, Balakrishnan K, Honavar V, Rudolph G, Wegener J, Bull L, Potter MA, Schultz AC, Miller JF, Burke E, Jonoska N (eds) *Proceedings of the genetic and evolutionary computation conference (GECCO'2002)*, San Francisco, California, Morgan Kaufmann Publishers, pp 319–326
  25. Cheng R, Jin Y, Olhofer M, Sendhoff B (2017) Test problems for large-scale multiobjective and many-objective optimization. *IEEE Trans Cybern* 47(12):4108–4121
  26. Cococcioni M, Ducange P, Lazzerini B, Marcelloni F (2007) A new multi-objective evolutionary algorithm based on convex hull for binary classifier optimization. In: 2007 IEEE congress on evolutionary computation (CEC'2007), IEEE Press, Singapore, pp 3150–3156
  27. Coello CAC (2000) Treating constraints as objectives for single-objective evolutionary optimization. *Eng Optim* 32(3):275–308
  28. Coello CAC (2006) The EMOO repository: a resource for doing research in evolutionary multiobjective optimization. *IEEE Comput Intell Mag* 1(1):37–45
  29. Coello CAC, Lamont GB, Van Veldhuizen DA (2007) *Evolutionary algorithms for solving multi-objective problems*, 2nd edn. Springer, New York (ISBN: 978-0-387-33254-3)
  30. Coello CAC, Baccara RL (2003) Evolutionary multiobjective optimization using a cultural algorithm. In: 2003 IEEE swarm intelligence symposium proceedings, Indianapolis, Indiana, USA, pp 6–13. IEEE Service Center
  31. Coello CAC (1996) An empirical study of evolutionary techniques for multiobjective optimization in engineering design. PhD Thesis, Department of Computer Science, Tulane University, New Orleans, Louisiana, USA
  32. Corne D, Knowles J (2007) Techniques for highly multiobjective optimisation: some nondominated points are better than others. In: Thierens D (ed) 2007 genetic and evolutionary computation conference (GECCO'2007), vol 1, ACM Press, London, UK, pp 773–780
  33. Cox W, While L (2016) Improving the IWFG algorithm for calculating incremental hypervolume. In: 2016 IEEE congress on evolutionary computation (CEC'2016). IEEE Press, Vancouver, Canada, 24–29, pp 3969–3976 (ISBN: 978-1-5090-0623-6)
  34. Daneshyari M, Yen GG (2011) Cultural-based multiobjective particle swarm optimization. *IEEE Trans Syst Man Cybern Part B Cybern* 41(2):553–567
  35. Das I, Dennis JE (1998) Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM J Optim* 8(3):631–657
  36. Das I, Dennis J (1997) A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. *Struct Optim* 14(1):63–69
  37. de Oliveira LB, Marcelino CG, Milanés A, Almeida PEM, Carvalho LM (2016) A successful parallel implementation of NSGA-II on GPU for the energy dispatch problem on hydroelectric power plants. In: 2016 IEEE congress on evolutionary computation (CEC'2016), 24–29, IEEE Press, Vancouver, Canada, pp 4305–4312 (ISBN: 978-1-5090-0623-6)
  38. Deb K, Jain H (2014) An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: solving problems with Box constraints. *IEEE Trans Evol Comput* 18(4):577–601
  39. Deb K, Mohan M, Mishra S (2005) Evaluating the  $\epsilon$ -domination based multi-objective evolutionary algorithm for a quick computation of pareto-optimal solutions. *Evol Comput* 13(4):501–525
  40. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197
  41. Depolli M, Trobec R, Filipic B (2013) Asynchronous master-slave parallelization of differential evolution for multi-objective optimization. *Evol Comput* 21(2):261–291
  42. di Pierro F, Khu S-T, Savić DA (2007) An investigation on preference order ranking scheme for multiobjective evolutionary optimization. *IEEE Trans Evol Comput* 11(1):17–45
  43. Diaz-Manriquez A, Toscano G, Barron-Zambrano JH, Tello-Leal E (2016) A review of surrogate assisted multiobjective evolutionary algorithms. *Comput Intell Neurosci* (Article Number: 9420460)
  44. Drechsler N, Drechsler R, Becker B (2001) Multi-objective optimisation based on relation favour. In: Zitzler E, Deb K, Thiele L, Coello CAC, Corne D (eds) *First international conference on evolutionary multi-criterion optimization*. Lecture Notes in Computer Science No. 1993, Springer, pp 154–166
  45. Ducheyne EI, De Baets B, De Wulf R (2003) Is fitness inheritance useful for real-world applications? In: Fonseca CM, Fleming PJ, Zitzler E, Deb K, Thiele L (eds) *Evolutionary multi-criterion optimization*. second international conference. Lecture notes in computer science, vol 2632, EMO 2003, Springer, Faro, Portugal, pp 31–42

46. Emmerich M, Beume N, Naujoks B (2005) An EMO algorithm using the hypervolume measure as selection criterion. In: Coello CAC, Aguirre AH, Zitzler E (eds) Evolutionary multi-criterion optimization. Lecture notes in computer science, vol 3410. Third international conference, EMO 2005, Springer, Guanajuato, México, pp 62–76
47. Esfe MH, Hajmohammad MH, Wongwises S (2018) Pareto optimal design of thermal conductivity and viscosity of ND-Co3O4 nanofluids by MOPSO and NSGA II using response surface methodology. *Curr Nanosci* 14(1):62–70
48. Falcón-Cardona JG, Coello CAC (2018) A multi-objective evolutionary hyper-heuristic based on multiple indicator-based density estimators. In: 2018 genetic and evolutionary computation conference (GECCO'2018), ACM Press, Kyoto, Japan, July 15–19, pp 633–640 (ISBN: 978-1-4503-5618-3)
49. Farina M, Amato P (2002) On the optimal solution definition for many-criteria optimization problems. In: Proceedings of the NAFIPS-FLINT international conference'2002, Piscataway, New Jersey. IEEE Service Center, pp 233–238
50. Farina M, Amato P (2004) A fuzzy definition of “optimality” for many-criteria optimization problems. *IEEE Trans Syst Man Cyber Part A Syst Hum* 34(3):315–326
51. Fieldsend JE, Everson RM, Singh S (2003) Using unconstrained elite archives for multiobjective optimization. *IEEE Trans Evol Comput* 7(3):305–323
52. Fleischer M (2003) The measure of pareto optima. Applications to multi-objective metaheuristics. In: Fonseca CM, Fleming PJ, Zitzler E, Deb K, Thiele L (eds) Evolutionary multi-criterion optimization. Second international conference. Lecture notes in computer science, vol 2632, EMO 2003, Springer, Faro, Portugal, pp 519–533
53. Fonseca CM, Fleming PJ (1993) Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. In: Forrest S (ed) Proceedings of the fifth international conference on genetic algorithms, Morgan Kaufman Publishers, San Mateo, California, USA, pp 416–423
54. Fabre MG, Pulido GT, Coello CAC (2009) Ranking methods for many-objective problems. In: MICAI 2009: advances in artificial intelligence. Lecture notes in artificial intelligence, vol 5845. 8th Mexican international conference on artificial intelligence, Springer, Guanajuato, México, pp 633–645
55. Giannakoglou KC, Kampolis IC (2010) Multilevel optimization algorithms based on metamodel- and fitness inheritance-assisted evolutionary algorithms. *Computational intelligence in expensive optimization problems*. Springer, Berlin, pp 61–84 (ISBN: 978-3-642-10700-9)
56. Goh C-K, Tan KC (2009) A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. *IEEE Trans Evol Comput* 13(1):103–127
57. Goldberg DE (1989) Genetic algorithms in search, optimization and machine learning. Addison-Wesley Publishing Company, Reading
58. Goldberg DE, Richardson J (1987) Genetic algorithm with sharing for multimodal function optimization. In: Grefenstette JJ (ed) Genetic algorithms and their applications: proceedings of the second international conference on genetic algorithms. Lawrence Erlbaum, Hillsdale, New Jersey, pp 41–49
59. Gomez JC, Terashima-Marín H (2010) Approximating multi-objective hyper-heuristics for solving 2D irregular cutting stock problems. In: Sidorov G, Aguirre AH, García CAR (eds) Advances in soft computing, 9th Mexican international conference on artificial intelligence, MICAI 2010. Lecture notes in artificial intelligence, vol 6438, Springer, Berlin, Nov 8–13, pp 349–360
60. Gonçalves RA, Kuk JN, Almeida CP, Venske SM (2015) MOEA/D-HH: a hyper-heuristic for multi-objective problems. In: Evolutionary multi-criterion optimization, 8th international conference, EMO 2015. Lecture notes in computer science, vol 9018, Springer, Guimarães, Portugal, March 29–April 1, pp 94–108
61. Greiner D, Periaux J, Emperador JM, Galvan B, Winter G (2017) Game theory based evolutionary algorithms: a review with nash applications in structural engineering optimization problems. *Arch Comput Methods Eng* 24(4):703–750
62. Guerreiro AP, Fonseca CM (2018) Computing and updating hypervolume contributions in up to four dimensions. *IEEE Trans Evol Comput* 22(3):449–463
63. Guizzo G, Fritsche GM, Vergilio SR, Pozo ATR (2015) A hyper-heuristic for the multi-objective integration and test order problem. In: 2015 genetic and evolutionary computation conference (GECCO 2015), ACM Press, Madrid, Spain, July 11–15, pp 1343–1350 (ISBN: 978-1-4503-3472-3)
64. Guizzo G, Vergilio SR, Pozo ATR, Fritsche GM (2017) A multi-objective and evolutionary hyper-heuristic applied to the integration and test order problem. *Appl Soft Comput* 56:331–344
65. Haario H, Saksman E, Tamminen J (2001) An adaptive metropolis algorithm. *Bernoulli* 7(2):223–242
66. Hadka D, Reed P (2013) Borg: an auto-adaptive many-objective evolutionary computing framework. *Evol Comput* 21(2):231–259
67. Hajela P, Lin CY (1992) Genetic search strategies in multicriterion optimal design. *Struct Optim* 4:99–107
68. Harada T, Takadama K (2017) Performance comparison of parallel asynchronous multi-objective evolutionary algorithm with different asynchrony. In: 2017 IEEE congress on evolutionary computation (CEC'2017), San Sebastián, Spain, June 5–8, IEEE Press, pp 1215–1222 (ISBN: 978-1-5090-4601-0)
69. Hardin DP, Saff EB (2004) Discretizing manifolds via minimum energy points. *Not AMS* 51(10):1186–1194
70. He Z, Yen GG (2016) Visualization and performance metric in many-objective optimization. *IEEE Trans Evol Comput* 20(3):386–402
71. Hernández-Gómez R, Coello CAC, Alba E (2016) A parallel version of SMS-EMOA for many-objective optimization problems. In: Parallel problem solving from nature—PPSN XIV, 14th international conference. Lecture notes in computer science, vol 9921, Springer, Edinburgh, UK, September 17–21, pp 568–577 (ISBN: 978-3-319-45822-9)
72. Gómez RH, Coello CAC (2015) Improved metaheuristic based on the  $R_2$  indicator for many-objective optimization. In: 2015 genetic and evolutionary computation conference (GECCO 2015), Madrid, Spain, ACM Press, July 11–15 2015, pp 679–686 (ISBN: 978-1-4503-3472-3)
73. Gómez RH, Coello CAC (2017) A hyper-heuristic of scalarizing functions. In: 2017 genetic and evolutionary computation conference (GECCO'2017), ACM Press, Berlin, Germany, July 15–19, pp 577–584 (ISBN: 978-1-4503-4920-8)
74. Gómez RH, Coello CAC, Torres EA (2016) A multi-objective evolutionary algorithm based on parallel coordinates. In: 2016 genetic and evolutionary computation conference (GECCO'2016), ACM Press, Denver, Colorado, USA, 20–24, pp 565–572 (ISBN: 978-1-4503-4206-3)
75. Hughes EJ (2005) Evolutionary many-objective optimisation: many once or one many? In: 2005 IEEE congress on evolutionary computation (CEC'2005), vol 1, Edinburgh, Scotland, IEEE Service Center, pp 222–227
76. Hutter F, Hoos HH, Leyton-Brown K, Stützle T (2009) ParamILS: an automatic algorithm configuration framework. *J Artif Intell Res* 36:267–306
77. Igel C, Hansen N, Roth S (2007) Covariance matrix adaptation for multi-objective optimization. *Evolu Comput* 15(1):1–28

78. Ishibuchi H, Akedo N, Nojima Y (2015) Behavior of multi-objective evolutionary algorithms on many-objective knapsack problems. *IEEE Trans Evol Comput* 19(2):264–283
79. Ishibuchi H, Masuda H, Tanigaki Y, Nojima Y (2015) Modified distance calculation in generational distance and inverted generational distance. In: Gaspar-Cunha A, Antunes CH, Coello CC (eds) *Evolutionary multi-criterion optimization*, 8th international conference, EMO 2015, lecture notes in computer science, vol 9019, Springer, Guimarães, Portugal, March 29–April 1, pp 110–125
80. Hisao Ishibuchi Y, Setoguchi HM, Nojima Y (2017) Performance of decomposition-based many-objective algorithms strongly depends on pareto front shapes. *IEEE Trans Evol Comput* 21(2):169–190
81. Jaszakiewicz A (2018) Improved quick hypervolume algorithm. *Comput Oper Res* 90:72–83
82. Jiang S, Zhang J, Ong Y-S, Zhang AN, Tan PS (2015) A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm. *IEEE Trans Cybern* 45(10):2202–2213
83. Kato T, Shimoyama K (2016) Evolutionary algorithm with parallel evaluation strategy using constrained penalty-based boundary intersection. In: 2016 IEEE congress on evolutionary computation (CEC'2016), IEEE Press, Vancouver, Canada, 24–29, pp 3702–3709 (ISBN: 978-1-5090-0623-9)
84. Kennedy J, Eberhart RC (2001) *Swarm intelligence*. Morgan Kaufmann Publishers, San Francisco
85. Knowles J, Corne D (2003) Properties of an adaptive archiving algorithm for storing nondominated vectors. *IEEE Trans Evol Comput* 7(2):100–116
86. Knowles J, Corne D (2007) Quantifying the effects of objective space dimension in evolutionary multiobjective optimization. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) *Evolutionary multi-criterion optimization*, 4th international conference, EMO 2007. Lecture notes in computer science, vol 4403, Springer, Matshushima, Japan, pp 757–771
87. Kuhn HW (1955) The Hungarian method for the assignment problem. *Naval Res Logist Q* 2(1–2):83–97
88. Kukkonen S, Deb K (2006) Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems. In: 2006 IEEE congress on evolutionary computation (CEC'2006), Vancouver, BC, Canada, IEEE, pp 1164–1171
89. Kumar R, Bal BK, Rockett P (2009) Multiobjective genetic programming approach to evolving heuristics for the bounded diameter minimum spanning tree problem. In: 2009 genetic and evolutionary computation conference (GECCO'2009), ACM Press, Montreal, Canada, July 8–12, pp 309–316 (ISBN: 978-1-60558-325-9)
90. Kumar R, Joshi AH, Banka KK, Rockett PI (2008) Evolution of hyperheuristics for the biobjective 0/1 Knapsack problem by multiobjective genetic programming. In: 2008 genetic and evolutionary computation conference (GECCO'2008), ACM Press, Atlanta, USA, pp 1227–1234 (ISBN: 978-1-60558-131-6)
91. Charan Kumari A, Srinivas K (2016) Hyper-heuristic approach for multi-objective software module clustering. *J Syst Softw* 117:384–401
92. Kumari AC, Srinivas K, Gupta MP (2013) Software module clustering using a Hyper-heuristic based Multi-objective genetic algorithm. In: *Proceedings of the 2013 3rd IEEE international advance computing conference*, IEEE Press, Ghaziabad, India, Feb 22–23, pp 813–818 (ISBN: 978-1-4673-4528-6)
93. Lacour R, Klamroth K, Fonseca CM (2017) A box decomposition algorithm to compute the hypervolume indicator. *Comput Oper Res* 79:347–360
94. Lara A, Sanchez G, Coello CAC, Schütze O (2010) HCS: a new local search strategy for memetic multi-objective evolutionary algorithms. *IEEE Trans Evol Comput* 14(1):112–132
95. León C, Miranda G, Segura C (2008) Parallel hyperheuristic: a self-adaptive island-based model for multi-objective optimization. In: 2008 genetic and evolutionary computation conference (GECCO'2008), ACM Press, Atlanta, USA, pp 757–758 (ISBN: 978-1-60558-131-6)
96. Li B, Li J, Tang K, Yao X (2015) Many-objective evolutionary algorithms: a survey. *ACM Comput Surv* 48:1
97. Li F, Cheng R, Liu J, Jin Y (2018) A two-stage R2 indicator based evolutionary algorithm for many-objective optimization. *Appl Soft Comput* 67:245–260
98. Li H, Deb K (2017) Challenges for evolutionary multiobjective optimization algorithms in solving variable-length problems. In: 2017 IEEE congress on evolutionary computation (CEC'2017), IEEE Press, San Sebastián, Spain, June 5–8, pp 2217–2224 (ISBN: 978-1-5090-4601-0)
99. Li M, Yang S, Liu X (2014) Diversity comparison of pareto front approximations in many-objective optimization. *IEEE Trans Cybern* 44(12):2568–2584
100. Jaimes AL, Coello CAC, Chakraborty D (2008) Objective reduction using a feature selection technique. in: 2008 genetic and evolutionary computation conference (GECCO'2008), ACM Press, Atlanta, USA, pp 674–680 (ISBN: 978-1-60558-131-6)
101. Lu R, Guan X, Li X, Hwang I (2016) A large-scale flight multi-objective assignment approach based on multi-island parallel evolution algorithm with cooperative coevolutionary. *Sci China Inf Sci* 59:7 (Article number: 823876)
102. Luo C, Shimoyama K, Obayashi S (2015) Effects of the number of design variables on performances in kriging-model-based many-objective optimization. In: 2015 IEEE congress on evolutionary computation (CEC'2015), IEEE Press, Sendai, Japan, 25–28, pp 1901–1908 (ISBN: 978-1-4799-7492-4)
103. Ma X, Liu F, Qi Y, Wang X, Li L, Jiao L, Yin M, Gong M (2016) A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables. *IEEE Trans Evol Comput* 20(2):275–298
104. Maashi M (2014) An investigation of multi-objective hyperheuristics for multi-objective optimisation. PhD thesis, The University of Nottingham, UK
105. Lopez EM, Coello CAC (2016) IGD<sup>+</sup>-EMOA: a multi-objective evolutionary algorithm based on IGD<sup>+</sup>. In: 2016 IEEE congress on evolutionary computation (CEC'2016), IEEE Press, Vancouver, Canada, 24–29, pp 999–1006 (ISBN: 978-1-5090-0623-9)
106. Lopez EM, Coello CAC (2018) An improved version of a reference-based multi-objective evolutionary algorithm based on IGD<sup>+</sup>. In: 2018 genetic and evolutionary computation conference (GECCO'2018), ACM Press, Kyoto, Japan, July 15–19, pp 713–720 (ISBN: 978-1-4503-5618-3)
107. Mariani T, Guizzo G, Vergilio SR, Pozo ATR (2016) Grammatical evolution for the multi-objective integration and test order problem. In: 2016 genetic and evolutionary computation conference (GECCO'2016), ACM Press, Denver, Colorado, USA, 20–24, pp 1069–1076 (ISBN: 978-1-4503-4206-3)
108. Martí L, García J, Berlanga A, Molina JM (2008) Introducing MONEDA: scalable multiobjective optimization with a neural estimation of distribution algorithm. In: 2008 genetic and evolutionary computation conference (GECCO'2008), ACM Press, Atlanta, USA, pp 689–696 (ISBN: 978-1-60558-131-6)
109. Martínez SZ, Coello CAC (2010) An archiving strategy based on the convex hull of individual minima for MOEAs. In: 2010 IEEE congress on evolutionary computation (CEC'2010), IEEE Press, Barcelona, Spain, July 18–23, pp 912–919
110. Hussain MM, Fujimoto N (2018) Parallel multi-objective particle swarm optimization for large swarm and high dimensional problems. In: 2018 IEEE congress on evolutionary computation

- (CEC'2018), IEEE Press, Rio de Janeiro, Brazil, July 8–13, pp 1546–1555 (ISBN: 978-1-5090-6017-7)
111. Mazumdar A, Chugh T, Miettinen K, López-Ibáñez M (2019) On dealing with uncertainties from Kriging models in offline data-driven evolutionary multiobjective optimization. In: Evolutionary multi-criterion optimization, 10th international conference, EMO 2019. Lecture notes in computer science, vol 11411, Springer, East Lansing, MI, USA, March 10–13, pp 463–474 (ISBN: 978-3-030-12597-4)
  112. McClymont K, Keedwell EC (2011) Markov chain hyper-heuristic (MCHH): an online selective hyper-heuristic for multi-objective continuous problems. In: 2011 genetic and evolutionary computation conference (GECCO'2011), ACM Press, Dublin, Ireland, July 12–16, pp 2003–2010
  113. Menchaca-Mendez A, Coello CAC (2016) Selection mechanisms based on the maximin fitness function to solve multi-objective optimization problems. *Inf Sci* 332:131–152
  114. Menchaca-Mendez A, Coello CAC (2017) An alternative hypervolume-based selection mechanism for multi-objective evolutionary algorithms. *Soft Comput* 21(4):861–884
  115. Kaisa MM (1999) Nonlinear multiobjective optimization. Kluwer Academic Publishers, Boston
  116. Luis MA, Coello CAC (2016) Decomposition-based approach for solving large scale multi-objective problems. In: Handl J, Hart E, Lewis PR, López-Ibáñez M, Ochoa G, Paechter B (eds) Parallel problem solving from nature—PPSN XIV, 14th international conference. Lecture notes in computer science, vol 9921. Springer, Edinburgh, UK, September 17–21, pp 525–534 (ISBN 978-3-319-45822-9)
  117. Antonio LM, Coello CAC (2018) Coevolutionary multiobjective evolutionary algorithms: survey of the state-of-the-art. *IEEE Trans Evol Comput* 22(6):851–865
  118. Antonio LM, Berenguer JAM, Coello CAC (2018) Evolutionary many-objective optimization based on linear assignment problem transformations. *Soft Comput* 22(16):5491–5512
  119. Mostaghim S, Schmeck H (2008) Distance based ranking in many-objective particle swarm optimization. In: Rudolph G, Jansen T, Lucas S, Poloni C, Beume N (eds) Parallel problem solving from nature—PPSN X. Lecture notes in computer science, vol 5199, Springer, Dortmund, Germany, pp 753–762
  120. Mostaghim S, Teich J (2005) Quad-trees: a data structure for storing pareto sets in multiobjective evolutionary algorithms with Elitism. In: Evolutionary multiobjective optimization: theoretical advances and applications, Springer, London, pp 81–104 (ISBN: 1-85233-787-7)
  121. Muller J (2017) SOCEMO: surrogate optimization of computationally expensive multiobjective problems. *Inf J Comput* 29(4):581–596
  122. Arias-Monta no A, Coello CAC, Mezura-Montes E (2012) Multi-objective evolutionary algorithms in aeronautical and aerospace engineering. *IEEE Trans Evol Comput* 16(5):662–694
  123. Okabe T, Jin Y, Sendhoff B, Olhofer M (2004) Voronoi-based estimation of distribution algorithm for multi-objective optimization. In: 2004 congress on evolutionary computation (CEC'2004), vol 2, IEEE Service Center, Portland, Oregon, USA, pp 1594–1601
  124. Ortega G, Filatovas E, Garzon EM, Casado LG (2017) Non-dominated sorting procedure for pareto dominance ranking on multicore CPU and/or GPU. *J Glob Optim* 69(3):607–627
  125. Palar PS, Shimoyama K (2017) Multiple metamodells for robustness estimation in multi-objective robust optimization. In: Evolutionary multi-criterion optimization, 9th international conference, EMO 2017. Lecture notes in computer science, vol 10173, Springer, Münster, Germany, March 19–22, pp 469–483 (ISBN: 978-3-319-54156-3)
  126. Palar PS, Shimoyama K (2017) On multi-objective efficient global optimization via universal Kriging surrogate model. in: 2017 IEEE congress on evolutionary computation (CEC'2017), IEEE Press, San Sebastián, Spain, June 5–8, pp 621–628 (ISBN: 978-1-5090-4601-0)
  127. Pescador-Rojas M, Gómez RH, Montero E, Rojas-Morales N, Riff MC, Coello CAC (2017) An overview of weighted and unconstrained scalarizing functions. In: Evolutionary multi-criterion optimization, 9th international conference, EMO 2017. Lecture notes in computer science, vol 10173, Springer, Münster, Germany, March 19–22, pp 499–513 (ISBN: 978-3-319-54156-3)
  128. Pilato C, Palermo G, Tumeo A, Ferrandi F, Sciuto D, Lanzi PL (2007) Fitness inheritance in evolutionary and multi-objective high-level synthesis. In: 2007 IEEE congress on evolutionary computation (CEC'2007), IEEE Press, Singapore, pp 3459–3466
  129. Pires EJS, Machado JAT, de Moura Oliveira PB (2013) Entropy diversity in multi-objective particle swarm optimization. *Entropy* 15(12):5475–5491
  130. Powell D, Hollingsworth J (2007) A NSGA-II, web-enabled, parallel optimization framework for NLP and MINLP. In: 2007 genetic and evolutionary computation conference (GECCO'2007), vol 2, ACM Press, London, UK, pp 2145–2150
  131. Praditwong K, Yao X (2007) How well do multi-objective evolutionary algorithms scale to large problems. In: 2007 IEEE congress on evolutionary computation (CEC'2007), IEEE Press, Singapore, pp 3959–3966
  132. Qian C, Tang K, Zhou ZH (2016) Selection hyper-heuristics can provably be helpful in evolutionary multi-objective optimization. In: Parallel problem solving from nature—PPSN XIV, 14th international conference. Lecture notes in computer science, vol 9921, Springer, Edinburgh, UK, September 17–21, pp 835–846 (ISBN: 978-3-319-45822-9)
  133. Qin H, Zhou J, Youlin L, Li Y, Zhang Y (2010) Multi-objective cultured differential evolution for generating optimal trade-offs in reservoir flood control operation. *Water Resour Manag* 24(11):2611–2632
  134. Ray T, Smith W (2006) A surrogate assisted parallel multiobjective evolutionary algorithm for robust engineering design. *Eng Optim* 38(8):997–1011
  135. Sierra M, Coello CAC (2005) Fitness inheritance in multi-objective particle swarm optimization. In: 2005 IEEE swarm intelligence symposium (SIS'05), IEEE Press, Pasadena, California, USA, pp 116–123
  136. Reynolds R, Liu D (2011) Multi-objective cultural algorithms. In: 2011 IEEE congress on evolutionary computation (CEC'2011), IEEE Service Center, New Orleans, Louisiana, USA, 5–8, pp 1233–1241
  137. Reynolds RG (1994) An introduction to cultural algorithms. In: Sebald AV, Fogel LJ (eds) Proceedings of the third annual conference on evolutionary programming, World Scientific, River Edge, New Jersey, pp 131–139
  138. Reynolds RG, Michalewicz Z, Cavaretta M (1995) Using cultural algorithms for constraint handling in GENOCOP. In: McDonnell JR, Reynolds RG, Fogel DB (eds) Proceedings of the fourth annual conference on evolutionary programming, MIT Press, Cambridge, Massachusetts, pp 298–305
  139. Rocha H, Peretta IS, Lima GFM, Marques LG, Yamanaka K (2016) Exterior lighting computer-automated design based on multi-criteria parallel evolutionary algorithm: optimized designs for illumination quality and energy efficiency. *Expert Syst Appl* 45:208–222
  140. Villalobos CAR, Coello CAC (2012) A new multi-objective evolutionary algorithm based on a performance assessment indicator. In: 2012 genetic and evolutionary computation conference (GECCO'2012), ACM Press, Philadelphia, USA, pp 505–512 (ISBN: 978-1-4503-1177-9)
  141. Rudolph G, Agapie A (2000) Convergence properties of some multi-objective evolutionary algorithms. In: Proceedings of the

- 2000 conference on evolutionary computation, vol 2, IEEE Press, Piscataway, New Jersey, pp 1010–1016
142. Russo LMS, Francisco AP (2016) Extending quick hypervolume. *J Heuristics* 22(3):245–271
  143. Santana-Quintero LV, Montaña AA, Coello CAC (2010) A review of techniques for handling expensive functions in evolutionary multi-objective optimization. In: Tenne Y, Goh CK (eds) *Computational intelligence in expensive optimization problems*, Springer, Berlin, Germany, pp 29–59 (ISBN: 978-3-642-10700-9)
  144. Santiago A, Huacuja HJF, Dorronsoro B, Pecero JE, Santillan CG, Barbosa JJJ, Monterrubio JCS (2014) A survey of decomposition methods for multi-objective optimization. In: Castillo O, Melin P, Pedrycz W, Kacprzyk J (eds) *Recent advances on hybrid approaches for designing intelligent systems*. Springer, Berlin, pp 453–465 (ISBN: 978-3-319-05170-3)
  145. Sato H, Aguirre HE, Tanaka K (2007) Controlling dominance area of solutions and its impact on the performance of MOEAs. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) *Evolutionary multi-criterion optimization, 4th international conference, EMO 2007*. Lecture notes in computer science, vol 4403, Springer, Matshushima, Japan, pp 5–20
  146. Saxena DK, Duro JA, Tiwari A, Deb K, Zhang Q (2013) Objective reduction in many-objective optimization: linear and nonlinear algorithms. *IEEE Trans Evol Comput* 17(1):77–99
  147. Schaffer JD (1984) Multiple objective optimization with vector evaluated genetic algorithms. PhD thesis, Vanderbilt University
  148. Schaffer JD (1985) Multiple objective optimization with vector evaluated genetic algorithms. In: *Genetic algorithms and their applications: proceedings of the first international conference on genetic algorithms*, Lawrence Erlbaum, pp 93–100
  149. Schaffer JD, Grefenstette JJ (1985) Multiobjective learning via genetic algorithms. In: *Proceedings of the 9th international joint conference on artificial intelligence (IJCAI-85)*, Los Angeles, California, pp 593–595. AAAI
  150. Schütze O, Lara A, Coello CAC (2011) On the influence of the number of objectives on the hardness of a multiobjective optimization problem. *IEEE Trans Evol Comput* 15(4):444–455
  151. Sen P, Yang JB (1998) Multiple criteria decision support in engineering design. Springer, London
  152. Smith RE, Dike BA, Stegmann SA (1995) Fitness inheritance in genetic algorithms. In: *SAC '95: proceedings of the 1995 ACM symposium on applied computing*, ACM Press, New York, NY, USA, pp 345–350
  153. Srinivas N, Deb K (1994) Multiobjective optimization using non-dominated sorting in genetic algorithms. *Evol Comput* 2(3):221–248
  154. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim* 11(4):341–359
  155. Süllflow A, Drechsler N, Drechsler R (2007) Robust multi-objective optimization in high dimensional spaces. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) *Evolutionary multi-criterion optimization, 4th international conference, EMO 2007*. Lecture notes in computer science, vol 4403, Springer, Matshushima, Japan, pp 715–726
  156. Suzuki N, Okamoto T, Koakutsu S (2015) Visualization of pareto optimal solutions using MIGSOM. In: *2015 IEEE congress on evolutionary computation (CEC'2015)*, IEEE Press, Sendai, Japan, 25–28, pp 2556–2564 (ISBN: 978-1-4799-7492-4)
  157. Tabatabaei M, Hakanen J, Hartikainen M, Miettinen K, Sindhya K (2015) A survey on handling computationally expensive multi-objective optimization problems using surrogates: non-nature inspired methods. *Struct Multidiscip Optim* 52(1):1–25
  158. Talbi E-G (2009) *Metaheuristics. From design to implementation*. Wiley, Hoboken
  159. Talbi EG, Mostaghim S, Okabe T, Ishibuchi H, Rudolph G, Coello CAC (2008) Parallel approaches for multi-objective optimization. In: Branke J, Deb K, Miettinen K, Slowinski R (eds) *Multi-objective optimization. interactive and evolutionary approaches*, Springer, Lecture notes in computer science, vol 5252, Berlin, Germany, pp 349–372
  160. Tian Y, Zhang X, Cheng R, Jin Y (2016) A multi-objective evolutionary algorithm based on an enhanced inverted generational distance metric. In: *2016 IEEE congress on evolutionary computation (CEC'2016)*, IEEE Press, Vancouver, Canada, 24–29, pp 5222–5229 (ISBN: 978-1-5090-0623-9)
  161. Tušar T, Filipič B (2015) Visualization of pareto front approximations in evolutionary multiobjective optimization: a critical review and the projection method. *IEEE Trans Evol Comput* 19(2):225–245
  162. Vazquez-Rodriguez JA, Petrovic S (2010) A new dispatching rule based genetic algorithm for the multi-objective job shop problem. *J Heuristics* 16(6):771–793
  163. von Lüken C, Baran B, Brizuela C (2014) A survey on multi-objective evolutionary algorithms for many-objective problems. *Comput Optim Appl* 58(3):707–756
  164. Vrugt JA, Robinson BA (2007) Improved evolutionary optimization from genetically adaptive multimethod search. *Proc Natl Acad Sci USA* 104(3):708–711
  165. Wagner T, Beume N, Naujoks B (2007) Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) *Evolutionary multi-criterion optimization, 4th international conference, EMO 2007*. Lecture notes in computer science, vol 4403, Springer, Matshushima, Japan, pp 742–756
  166. Walker DJ, Keedwell E (2016) Multi-objective optimisation with a sequence-based selection hyper-heuristic. In: *Proceedings of the 2016 on genetic and evolutionary computation conference companion*, ACM Press, New York, USA, July 20–24, pp 81–82
  167. Walker DJ, Keedwell E (2016) Towards many-objective optimisation with hyper-heuristics: identifying good heuristics with indicators. In: *Parallel problem solving from nature—PPSN XIV, 14th international conference*. Lecture notes in computer science, vol 9921, Springer, Edinburgh, UK, September 17–21, pp 493–502 (ISBN: 978-3-319-45822-9)
  168. Wang H, Jin Y, Yao X (2017) Diversity assessment in many-objective optimization. *IEEE Trans Cybern* 47(6):1510–1522
  169. Wang TC, Ting CK (2018) Fitness inheritance assisted MOEA/D-CMAES for complex multi-objective optimization problems. In: *2018 IEEE congress on evolutionary computation (CEC'2018)*, IEEE Press, Rio de Janeiro, Brazil, July 8–13, pp 1013–1020 (ISBN: 978-1-5090-6017-7)
  170. Wang Y, Li B (2010) Multi-strategy ensemble evolutionary algorithm for dynamic multi-objective optimization. *Memet Comput* 2(1):3–24
  171. Wessing S, Rudolph G, Menges DA (2016) Comparing asynchronous and synchronous parallelization of the SMS-EMOA. In: *Parallel problem solving from nature—PPSN XIV, 14th international conference*. Lecture notes in computer science, vol 9921, Springer, Edinburgh, UK, September 17–21, pp 558–567 (ISBN: 978-3-319-45822-9)
  172. Wismans L, Van Berkum E, Bliemer M (2014) Acceleration of solving the dynamic multi-objective network design problem using response surface methods. *J Intell Transport Syst* 18(1):17–29
  173. Wong ML, Cui G (2013) Data mining using parallel multi-objective evolutionary algorithms on graphics processing units. In: *Massively parallel evolutionary computation on GPGPUs*, Springer, pp 287–307 (ISBN: 978-3-642-37958-1)
  174. Yagoubi M, Schoenauer M (2012) Asynchronous master/slave MOEAs and heterogeneous evaluation costs. In: *2012 genetic*

- and evolutionary computation conference (GECCO'2012), ACM Press, Philadelphia, USA, pp 1007–1014 (ISBN: 978-1-4503-1177-9)
175. Yang D, Sun Y, di Stefano D, Turrin M, Sariyildiz S (2016) Impacts of problem scale and sampling strategy on surrogate model accuracy. An application of surrogate-based optimization in building design. In: 2016 IEEE congress on evolutionary computation (CEC'2016), IEEE Press, Vancouver, Canada, 24–29, pp 4199–4207 (ISBN: 978-1-5090-0623-6)
  176. Yun Y, Yoon M, Nakayama H (2009) Multi-objective optimization based on meta-modeling by using support vector regression. *Optim Eng* 10(2):167–181
  177. Martínez SZ, Coello CAC (2013) MOEA/D assisted by RBF networks for expensive multi-objective optimization problems. In: 2013 genetic and evolutionary computation conference (GECCO'2013), ACM Press, New York, USA, July 6–10, pp 1405–1412 (ISBN: 978-1-4503-1963-8)
  178. Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Comput* 11(6):712–731
  179. Zhang Q, Liu W, Tsang E, Virginas B (2010) Expensive multiobjective optimization by MOEA/D with Gaussian process model. *IEEE Trans Evol Comput* 14(3):456–474
  180. Zhang X, Tian Y, Cheng R, Jin Y (2018) A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization. *IEEE Trans Evol Comput* 22(1):97–112
  181. Zille H, Ishibuchi H, Mostaghim S, Nojima Y (2018) A framework for large-scale multiobjective optimization based on problem transformation. *IEEE Trans Evol Comput* 22(2):260–275
  182. Zitzler E (1999) Evolutionary algorithms for multiobjective optimization: methods and applications. PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland
  183. Zitzler E, Künzli S (2004) Indicator-based selection in multi-objective search. In: Yao X (ed) *Parallel problem solving from nature—PPSN VIII. Lecture notes in computer science*, vol 3242, Springer, Birmingham, UK, pp 832–842
  184. Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Trans Evol Comput* 3(4):257–271
  185. Zitzler E, Thiele L, Laumanns M, Fonseca CM, da Fonseca VG (2003) Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans Evol Comput* 7(2):117–132

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.