Many-objective Optimization Using Evolutionary Algorithms: A Survey

Slim Bechikh, Maha Elarbi and Lamjed Ben Said

Abstract Multi-objective Evolutionary Algorithms (MOEAs) have proven their effectiveness and efficiency in solving complex problems with two or three objectives. However, recent studies have shown that the performance of the classical MOEAs is deteriorated when tackling problems involving a larger number of conflicting objectives. Since most individuals become non-dominated with respect to each others, the MOEAs' behavior becomes similar to a random walk in the search space. Motivated by the fact that a wide range of real world applications involves the optimization of more than three objectives, several Many-objective Evolutionary Algorithms (MaOEAs) have been proposed in the literature. In this chapter, we highlight in the introduction the difficulties encountered by MOEAs when handling Many-objective Optimization Problems (MaOPs). Moreover, a classification of the most prominent MaOEAs is provided in an attempt to review and describe the evolution of the field. In addition, a summary of the most commonly used test problems, statistical tests, and performance indicators is presented. Finally, we outline some possible future research directions in this research area.

Keywords Many-objective optimization · Evolutionary algorithms · Scalability · High dimensionality

S. Bechikh (✉) · M. Elarbi · L. Ben Said
SOIE Lab, Computer Science Department, ISG-Tunis, University of Tunis,
Bouchoucha city, 2000 Le Bardo, Tunis, Tunisia
e-mail: slim.bechikh@gmail.com

M. Elarbi
e-mail: arbi.maha@yahoo.com

L. Ben Said
e-mail: lamjed.bensaid@isg.rnu.tn

S. Bechikh et al. (eds.), *Recent Advances in Evolutionary Multi-objective Optimization*, Adaptation, Learning, and Optimization 20,
DOI 10.1007/978-3-319-42978-6_4

1 Introduction

Since the implementation of the first MOEA, different algorithms have been proposed to deal with Multi-objective Optimization Problems (MOPs) [1]. MOEAs have been widely used to solve problems with two or three objectives. In fact, most of the proposed MOEAs use the Pareto-dominance relation to compare solutions of the population. Specially, the population members are ranked using the Pareto-dominance relation and the recombination operator is performed to the best individuals in order to generate solutions that are closer to the Pareto set. However, recent studies on MOEAs have shown that Pareto-based MOEAs struggle to solve problems with more than three objectives. Thus, although the classical MOEAs such as NSGA-II [2] and SPEA2 [3] have been successfully applied to solve many real-world problems with a small number of objectives, they are not well-suited when dealing with problems involving a high number of objectives. This limitation seems to affect only Pareto-based MOEAs but some difficulties are common to most existing multi-objective optimizer. For this reason, motivated by the fact that a wide range of real world applications in industrial [4] and engineering [5] designs involves the optimization of more than three objectives, a wide variety of proposals have been proposed to deal with the difficulties encountered by the current state of the art MOEAs.

In summary, the challenges encountered by the state of the art MOEAs in finding a representative set of Pareto optimal solutions when handling MaOPs can be briefly discussed as follows:

- **Increase of the number of non-dominated solutions**: When the dimensionality of the objective space increases, the proportion of Pareto-non dominated solutions in the population grows which deteriorates the search process ability to converge towards the Pareto front. Thus, the MOEA behavior becomes similar to a random search one. Figure 1 shows how the proportion of non-dominated solutions in the population behaves with respect to the number of objectives. We can see that after a few generations, the population becomes completely non-dominated.

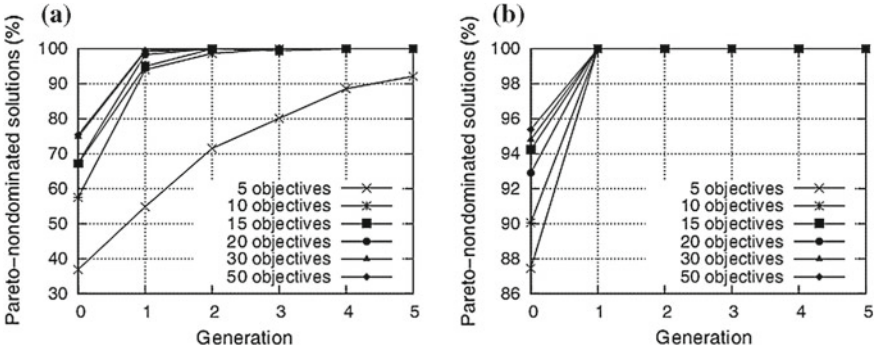

Fig. 1 Proportion of Pareto-non-dominated solutions. From Ref. [6]. **a** DTLZ1. **b** DTLZ6

Table 1 Bounds for the number of points required to represent a Pareto front with resolution $r = 25$

M	Points
2	50
4	62 500
5	1 953 125
7	1 708 984 375

From Ref. [7]

As a consequence, new promising search directions become very hard to find. Another reason is the increment of the number of dominance resistant solutions in the population when we deal with many-objective. In fact, dominance resistant solutions are non-dominated solutions but they are far from the True Pareto Front.

- **Ineffectiveness of crossover and mutation operators**: In a high dimensional space, the population members are likely to be widely distant from each other. Thereby, two distant parent solutions will produce two distant children that are not similar to their parents. In such a case, the effect of the recombination operation becomes inefficient in producing promising offspring individuals.
- **Difficulty to represent the trade-off surface**: Due to the high dimensionality, more points are needed to represent the trade-off surface. In fact, the number of points to represent a Pareto front with M objectives and r resolution is bounded by $O(Mr^{M-1})$. This expression is derived assuming that each solution is contained in a hypercube. Thus, the resolution r represents the number of hypercubes per dimension. Table 1 shows the bound of points required to represent a Pareto front for different number of objectives using a resolution $r = 25$. We note that for 5 objectives the number of points required to represent the Pareto front is about 2 million points.
- **High computational cost of the diversity measure estimation**: In order to determine the extent of crowding of a solution in a population, the identification of neighboring solutions in a population becomes computationally very expensive in high dimensional spaces. For this reason, the use of any approximation in diversity to reduce the computational cost may cause an unacceptable distribution of the solutions.
- **Difficulty of visualization**: It is not a matter that is directly related to optimization. The visualization of a higher dimensional trade-off front becomes difficult. Hence, it is difficult for the Decision Maker (DM) to choose a preferred solution. Several methods were proposed to ease decision making in MaOPs such as Parallel coordinates and self-organizing maps.

2 A Taxonomy of Many-objective Solution Approaches

In this section, a classification of the most relevant approaches to deal with MaOPs is presented.

2.1 Preference Ordering Relation-Based Approaches

- **Expansion Relation**

The Expansion preference relation (ER) was proposed by Sato et al. [8] to control the dominance area of solutions using a user-defined parameter S. This preference ordering relation was proposed in order to induce an appropriate ranking of solutions and to enhance the selection mechanism, so that the performance of MOEAs on combinatorial optimization problems with a variety of objectives is improved. The basic idea consists of expanding and contracting the dominance area by replacing the objective function $f_i(x)$ using the vector S as follows:

$$f_i^{'}(x) = \frac{r\ sin(\omega_i + S_i\ \Pi)}{sin\ (S_i\ \Pi)} \quad \forall i \in \{1, 2, \ldots, m\} \tag{1}$$

where r is the norm of $f(x)$, $f_i(x)$ is the fitness value of the i-th objective, and ω_i is the angle between $f(x)$ and $f_i(x)$. Figure 2 illustrates the fitness modification to change the covered area of dominance when $S_i < 0.5$ and $\phi_i = S_i\ \Pi$. One can notice that the i-th fitness value $f_i(x)$ is increased to $f_i^{'}(x) > f_i(x)$. Thus, if $S_i < 0.5$ a more finer grained ranking of solutions is produced and the dominance area is expanded which strengthen the selection. However, if $S_i > 0.5$ a coarser ranking of solutions is produced and the dominance area is contracted which would weaken the selection. While if $S_i = 0.5$, the usual dominance relation is used. Since in a MaOP we search to produce a finer grained ranking of solutions, the parameter S_i should be less than 0.5. In fact, the main characteristic of this preference relation is that it emphasizes the solutions in the middle region of the Pareto front. The authors used the multi-objective 0/1 Knapsack problem [9] on two up to five objectives and integrated the ER relation into NSGA-II. The experimental results show that contracting or expending the dominance area is better than using conventional dominance in terms of the quality of the obtained solutions. However, the ER was assessed only on problems involving up to five objectives. Hence further experiments with higher dimension problems are

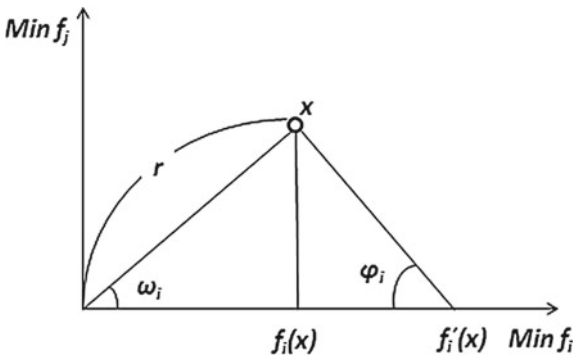

Fig. 2 Fitness modification to change the covered area of dominance for $S_i < 0.5$. From Ref. [8]

required for validation. Moreover, since the expansion relation can improve either convergence or diversity, the authors concluded that it would be better to combine it with other methods.

- ***k*-Optimality Relation**

Farina and Amato [10] proposed the k-optimality relation. This preference relation is based on the number of improved objectives between two solutions. The k-optimality employs three quantities. Assuming that we have two solutions x and y, the first quantity n_b represents the number of objectives where x is better than y. The second one n_e denotes the number of objectives where x is equal to y and the final one n_w where x is worse. Thus, given M objectives the following inequalities holds true:

$$n_b + n_w + n_e = M \tag{2}$$

$$0 < n_b, n_w, n_e < M \tag{3}$$

In fact, by employing these quantities the concepts of (1-k)-dominance and k-optimality can be defined.

Definition 1 *(1-k)-dominance*
A solution x (1-k)-dominates a solution y if and only if:

$$\begin{cases} n_e < M \\ n_b \geq \frac{M-n_e}{1+k} \end{cases} \qquad 0 \leq k \leq 1 \tag{4}$$

From the above definition, one can notice that the 1-dominance (i.e., $k = 0$) represents the Pareto dominance. The parameter k can assume any value in [0,1], but because n_b has to be a natural number, the smallest integer greater than the quantity $\frac{M-n_e}{1+k}$ need to be considered. After defining the (1-k)-dominance, the following definition of the k-optimality can be given:

Definition 2 *k-optimality*
A solution x^* is k-optimum if and only if there is no $x \in \Omega$ such that x k-dominates x^*.

Therefore, the k-optimality represents a strong version of the Pareto-optimality (0-optimality). The authors extended the (1-k) dominance relation by incorporating fuzzy arithmetic techniques.

- **Average and Maximum Ranking Relations**

Bentley and Wakefield [11] proposed the average ranking (AR) and maximum ranking (MR) preference ordering relations. The AR relation begins by sorting the solutions based on their fitness. Then, a set of different ranking for every solution is obtained for each objective. After that, the average ranking value of each solution is

computed by summing their ranks on each objective. Hence, based on the obtained average ranking values, the solutions can be sorted into order of best average rank. Thus, a solution x dominates a solution y with respect to the AR relation (denoted by $x \prec_{avg} y$) if and only if $R_{avg}(x) < R_{avg}(y)$ where $R_{avg}(x) = \sum_{1 \leq i \leq M} f_i(x)$. The AR distinguishes the non-dominated solutions based on their obtained ranks on different objectives. This preference ordering relation is simple and range-independent [12]. However, Corne and Knowles [13] have reported that the obtained solution set may only concentrate in a subregion of the Pareto front. Thus, it has a lack of diversity maintenance mechanism.

Differently, the MR relation considers the best rank as the global rank of each solution. Therefore, a solution x dominates a solution y with respect to the MR relation (denoted by $x \prec_{max} y$) if and only if $R_{max}(x) < R_{max}(y)$ where $R_{max}(x) = min_{1 \leq i \leq M} \{rank f_i(x)\}$. The main drawback of this method is that it emphasizes solutions with high performance in some objectives, while they have a poor overall performance (i.e., extreme solutions).

- **Favour Relation**

In order to refine the ranking of solutions in MaOPs, Drechsler et al. [14] proposed the favour relation. In their work, the authors used the favour relation and a method called Satisfiability Class Ordering (SCO) where the former is used to compare solutions to each others, while the latter is used to sort solutions. This preference ordering relation can be defined as follows:

Definition 3 *Favour relation*
A solution x dominates a solution y with respect to the favour relation (denoted by $x \prec_f y$) if and only if:

$$|\{i : f_i(x) < f_i(y), 1 \leq i \leq M\}| < |\{j : f_j(y) < f_j(x), 1 \leq j \leq M\}| \quad (5)$$

The main idea behind the favour relation is that the solution x is favoured to y if and only if the number of objectives in which x outperforms y is superior to the number of objectives in which y outperforms x. For example given two solutions x_1 $(4, 2, 1)$, and x_2 $(1, 1, 2)$ then we have that $x_2 \prec_f x_1$ since it has better objective values than x_1 with respect to the first two objectives. This preference ordering relation was used in an algorithm proposed by Drechsler et al. [15]. In fact, it was demonstrated that the favour relation does not create a partial order since it is not transitive but it is able to create a finer grained ranking of solutions than that created by the Pareto dominance when solving MaOPs. However, the main disadvantage of this relation is that it emphasizes extreme solutions.

- **Other Preference Ordering Relation-Based Approaches**

Preference ordering relation-based approaches have been proposed to deal with the first challenge which is the increase of the number of Pareto non-dominated

solutions in the population in a high dimensional space. Therefore, preference ordering relation-based approaches aim mainly to provide a finer ranking of solutions when solving MaOPs. Various methods based on preference ordering relations have been proposed to deal with MaOPs. The preference order ranking was introduced by Di Pierro et al. [16]. The basic idea of this preference relation is to discard the objectives in order to compare the solutions. It is based on the concept of *efficiency of order* proposed by Das [17]. However, the main drawback of this relation is its high computational cost. Sülflow et al. [4] proposed the ε-preferred relation which is based on the favour relation [14]. In the ε-preferred relation, two solutions are compared based on the number of objectives in which one solution exceeds the other using a predefined threshold. Moreover, the favour relation is used to determine which solution is better in case of a tie. The authors replaced the favour relation in the algorithmic framework used in [14] with the ε-preferred relation. The ε-preferred relation has demonstrated good results on the nurse rostering problem [18] with 25 objectives. A summary of some existing preference ordering relations for solving MaOPs is presented in Table 2.

Table 2 Summary of preference ordering relations: MaxObjs means maximum number of objectives

Relations	References	Basic idea	MaxObj	Test problems
ER	Sato et al. [8]	Control the dominance area of solutions using a user defined parameter S	5	MKP
k-Optimality	Farina and Amato [10]	Compare two solutions based on the number of improved objectives between them	12	Test case
AR	Garza-Fabre et al. [6]	Sort the solutions based on their average ranking values	50	DTLZ
MR	Garza-Fabre et al. [6]	Compare the solutions based on their best obtained rank	50	DTLZ
Favour	Drechsler et al. [14]	Favour the solution that outperforms the other one in more objectives	7	5 benchmarks problems
Preference order ranking	Di Pierro et al. [16]	Compare two solutions by discarding objectives	8	DTLZ
ε-Preferred	Sülflow et al. [4]	Compare two solutions based on the number of objectives in which a solution exceeds the other one by using a predefined threshold	25	NRP

2.2 Objective Reduction-Based Approaches

- **PCA-NSGA-II: Principal Component Analysis-NSGA-II**

In this work, Deb and Saxena [19] proposed the Principal Component Analysis-Nsga-II algorithm called PCA-NSGA-II. This latter combines a reduction technique with NSGA-II to deal with MaOPs with redundant objectives. In fact, many real world problems have M objectives, while the true Pareto front is less than M-dimensional. Hence, some of the objectives are redundant. Thus, in order to determine the true Pareto optimal front, the authors suggested to use the PCA procedure. This reduction technique was used to reduce the dimensionality of a data set with a large number of interrelated variables. The PCA-NSGA-II starts with an initial set of objectives $\Pi_0 = \{1, 2, \ldots, M\}$. Then, NSGA-II is executed for a given number of iterations to obtain a population P_t where t represents the current generation. Next, the population P_t is used by the PCA reduction method to get a new set of objectives $\Pi_t \subset \Pi_0$ to be used in the next iterations of NSGA-II. The PCA procedure can be summarized by the following four basic steps:

- **Step 1**: Store the objective values of the population P_t in an initial data matrix D of size $M \times N$, where M is the number of objectives and N is the size of the population;
- **Step 2**: Obtain the standardized matrix X by subtracting the mean from each objective value in matrix D;
- **Step 3**: Compute the covariance matrix V and the correlation matrix R using the standardized matrix X;
- **Step 4**: Compute eigenvalues of the correlation matrix R and eigenvectors that are considered as the PCs;

We note that the most negative and the most positive elements for a given PC are considered as the two most important conflicting objectives. In addition, PCA-NSGA-II approach uses an additional procedure that selects the most negative and most positive elements for the first PC. The experimental results on a modified version of DTLZ5 test problem [19] have demonstrated the ability of PCA-NSGA-II to solve high dimensional problems with redundant objectives. However, the proposed algorithm shows some vulnerability when the task involves finding a large Pareto optimal front due to the difficulties encountered to find the correct combination of objectives in MaOPs with non-redundant objectives.

- **PCSEA: Pareto Corner Search Evolutionary Algorithm**

Singh et al. [20] introduced the Pareto Corner Search Evolutionary Algorithm (PCSEA). The authors proposed a new approach that identifies a reduced set of objectives instead of dealing with the true dimensionality of the true MaOP. Moreover, PCSEA does not approximate the whole Pareto front but it searched for a specific set of non-dominated solutions. More specifically, the authors suggested to use boundaries of the Pareto front called corner solutions in order to predict the

dimensionality of the true Pareto front. In fact, for a two-dimensional optimization problem, a corner solution corresponds to the minimum value of each objective. However, the number of corner solutions increases exponentially with the number of objectives (i.e., $2^M - 1$ possible corners to a M-objective optimization problem), but in reality, test problems such as DTLZ [21] and WFG [22] have M corner solutions for M-objective problems. The proposed approach can be summarized by the following two steps: (1) find the corner solutions, and (2) use the corner solutions to reduce the set of objectives. PCSEA uses the same crossover and mutation operators used in NSGA-II. However, differently to NSGA-II which uses non-dominated sorting and crowding distance-based ranking, PCSEA uses a corner-sort ranking. Details of this method can be found in [20]. After identifying the corner solutions, a heuristic technique is performed to determine the relevant objectives and to eliminate the redundant ones. The reduction process can be described as follows. First, a set F containing the non-dominated solutions produced by PCSEA is formed where only unique solutions are considered in the set F. Second, in order to quantify the change in the number of non-dominated solutions a parameter R was used. The parameter R is defined as follows:

$$R = N_{F_R - f_m} / N_F \tag{6}$$

where N_F is the number of non-dominated solutions in the set F and $N_{F_R - f_m}$ is the number of non-dominated solutions corresponding to the objective set obtained after omitting f_m from the set of relevant objectives F_R. If the value of R is high for a particular objective f_m, it means that this objective can be omitted from the set of relevant objectives. PCSEA does not suffer from the lack of the selection pressure and it has a low computational complexity which makes it suitable for solving MaOPs. However, it should be noted that a large population size is not required when dealing with many objectives for the reason that PCSEA does not approximate the entire Pareto front.

- **Objective Reduction Using a Feature Selection Technique**

In this work, López Jaimes et al. [23] proposed to integrate an unsupervised feature selection technique that was originally introduced by Mitra et al. [24] in NSGA-II. This reduction method is similar to the one used by [19] for the reason that both of them use a correlation matrix to measure the conflict between each pair of objectives and to determine the most conflicting objectives in order to eliminate the redundant ones. Two algorithms have been introduced in this work. The first algorithm finds the minimum subset of non-redundant objectives with the minimum possible error, while the second algorithm finds the minimum set of k-non-redundant objectives that yield to the minimum possible error. The authors described the main steps of their reduction technique by the following three steps:

- **Step 1**: Define the conflict between objectives as distance and divide the objective set into homogeneous neighborhoods of size q around each objective;

- **Step 2**: Select the most compact neighborhood where the most compact neighbors is the neighborhood with the minimum distance to its qth neighbor;
- **Step 3**: Retain the center of the neighborhood and discard q neighbors with least conflict in the current set. The distance to the qth neighbor is considered as the error committed by removing the q objectives;

The reduction techniques used in this work iterated **Step 2** and **Step 3** until the number of desired objectives does not reach the predefined k value or until there are not more considered neighborhoods. The experimental study has mentioned good results in solving: a variation of DTLZ5 [21] with 3, 5, and 10 objectives, a variation of the DTLZ2 [21], and the 0/1 knapsack problem [9] with 10 and 20 objectives. The experiment results show that the proposed methods are competitive compared to the PCA-based reduction method and the reduction method of Brockhoff et al. [25].

- **Other Objective Reduction-Based Approaches**

Objective reduction-based approaches aim to tackle MaOPs with redundant objectives. In fact, there are two different timing of incorporating the dimensionality into a MOEA, thus we can identify two classes [28]: (1) offline dimensionality reduction where the dimensionality reduction method is performed after obtaining a set of Pareto optimal solutions or (2) online dimensionality reduction where the number of objectives is introduced gradually by iteratively obtaining solution sets and invoking the dimensionality reduction method during the search process. In fact, the first class can be further divided into three sub-classes: Correlation-based methods, dominance structure-based methods, and feature-based methods. Correlation-based methods consist in examining the correlation among the objectives. In this sub-class, we find the work of Saxena et al. [26] in which they proposed L-PCA and NL-MVU-PCA algorithms based on the PCA method and Maximum Variance Unfolding for linear and nonlinear objective reduction, respectively. The authors investigated the performance of the two algorithms on a wide range of redundant and non-redundant test problems and on two real world problems. Dominance structure-based methods consider the dominance relationships among the solutions obtained by a MOEA. Brockhoff and Zitzler [27] proposed a new notion of conflict and they introduced a quantification δ for measuring the change in the dominance structure based on ε-dominance. In their studies, an exact and a greedy algorithm were proposed to solve the δ-MOSS and the k-EMOSS problems, where the δ-MOSS consists in finding the minimum objective subset corresponding to a given error, while k-EMOSS consists in finding an objective subset of size k with the minimum possible error. The experimental results demonstrated that the exact algorithm yields smaller objective subsets than the greedy algorithm, while the high complexity of the exact method limits its usage. In the third sub-class, we find the work of López Jaimes et al. [23]. Concerning the online dimensionality reduction algorithms, many approaches have been proposed in the literature. PCA-NSGA-II is considered as an online dimensional reduction algorithm since it iteratively obtains solution sets and reduces the objectives using information of correlations among the objectives. Table 3 presents a comparison of the objective reduction-based approaches studied in this subsection.

Table 3 Comparison of objective reduction-based approaches: OnDRA means online dimensionality reduction approach, OfDRA means offline dimensionality reduction approach, C means correlation-based methods, DS means dominance structure-based methods, F means feature-based methods, and MObj means maximum number of objectives

Algorithms	References	Characteristics					MObj	Test problems
		OnDRA	OfDRA					
			C	DS	F	Other		
PCA-NSGA-II	Deb and Saxena [19]	X	–	–	–	–	30	DTLZ5(I,M)
							5	DTLZ2
PCSEA	Singh et al. [20]	–	–	–	–	X	100	DTLZ5(I,M)
							20	DTLZ2
–	López Jaimes et al. [23]	–	–	–	X	–	20	MKP
							20	$DTLZ2_{BZ}$
							10	DTLZ5(I,M)
L-PCA	Saxena et al. [26]	–	X	–	–	–	25	DTLZ
							25	WFG
							50	DTLZ5(I,M)
NL-MVU-PCA	Saxena et al. [26]	–	X	–	–	–	25	DTLZ
							25	WFG
							50	DTLZ5(I,M)
Exact-δ-MOSS/k-EMOSS	Brockhoff and Zitzler [27]	–	–	X	–	–	–	–
Greedy-δ-MOSS/k-EMOSS	Brockhoff and Zitzler [27]	–	–	X	–	–	25	DTLZ
							25	MKP

2.3 Preference Incorporation-Based Approaches

- **R-NSGA-II: Reference Point-Based NSGA-II**

Deb et al. [29] introduced a modified version of NSGA-II that prefers solutions closer to a user-provided reference point set and that de-emphasizes solutions within a ε-neighborhood of a reference point. In fact, the parameter ε controls the extent of the distribution of solutions near the closest Pareto-optimal solution and the reference points are used to guide the search toward the preferred parts of the Pareto front. In this work, the crowding distance used in NSGA-II is modified as follows. For each reference point, the normalized Euclidean distance of each solution of the last considered front is calculated and based on this distance the solutions are sorted in

ascending order. Hence, the closest solution to the reference point is assigned a rank of one. The second closest solution to the reference point is assigned a rank of two and so on. After that, the minimum of the assigned ranks is assigned as the crowding distance to a solution. Thus, the smallest crowding distance of one is assigned to the closest solutions to all reference points. The solutions having next-to-smallest Euclidean distance to all reference points are assigned the next-to-smallest crowding distance of two, and so on. Thereafter, solutions with a smaller crowding distance are preferred. The authors proposed to control the extent of obtained solutions by grouping all solutions having a sum of normalized difference in objective values of ε or less. A randomly picked solution from each group is retained and the rest of all group members are assigned a large crowding distance in order to discourage them to remain in the race. The proposed procedure allows finding multiple ROIs simultaneously in a single simulation run. R-NSGA-II has demonstrated good results on two to five objective test problems but it faces difficulties when using a single reference point since diversity is not well maintained. Moreover, the ε clearing parameter setting is not trivial.

- **PBEA: Preference-Based Evolutionary Algorithm**

In this work, Thiele et al. [30] proposed a new algorithm called PBEA that combines IBEA with the reference point method. In fact, in IBEA, the fitness value of a solution x in a population P can be expressed as follows:

$$F(x) = \sum_{y \in P\{x\}} (-\mathrm{e}^{-I(y,x)}/\kappa) \tag{7}$$

where κ is a scaling factor. In IBEA, the additive epsilon indicator is used which is a Pareto compliant indicator and it is defined as follows:

$$I_{\varepsilon^+}(x, y) = min_{\varepsilon} \{f_m(x) - \varepsilon \leq f_m(y)\ \forall m = 1, 2, \ldots, M\} \tag{8}$$

In order to take the preference information into account, the authors defined a new preference-based quality indicator described as follows:

$$I_p(x, y) = I_{\varepsilon^+}(x, y)/s(g, f(x), \delta) \tag{9}$$

where x and y are two solutions, I_{ε^+} is the additive epsilon indicator, s is a function used to normalize the set of points, and δ is a positive parameter used to specify the minimal value of the normalized function, it allows the DM to control the spread of the obtained Region of Interest (ROI). PBEA was also used in an interactive fashion to offer many possibilities to the DM in directing the search into a preferred part of the Pareto optimal set. The main motivation behind PBEA is that it gives reliable information on the solutions to the DM. Moreover, the used binary quality indicator I_p is Pareto dominance preserving. In addition, the experimental results show that it is suitable to solve MaOPs due to the use of the achievement function. However, the authors noted that adjusting the δ parameter is not an easy task.

- **r-NSGA-II: Reference Solution-Based NSGA-II**

Ben Said et al. [31] proposed a new dominance relation called r-dominance (reference-solution-based dominance) that creates a strict partial order among Pareto equivalent solutions and that has the ability to differentiate between non-dominated solutions in a partial manner based on a user-supplied aspiration level vector. The r-dominance relation represents a hybridization between the Pareto dominance relation and the reference point method (i.e., DM' s preferences). In fact, the key feature of this preference-based dominance relation is to prefer solutions that are closer to the reference point, while preserving the order induced by the Pareto dominance. Thus, in order to determine the closeness of a solution to the reference point, the authors used the weighted Euclidean distance employed by Deb et al. [30] which is expressed as follows:

$$dist(x, g) = \sqrt{\sum_{i=1}^{M} w_i \left(\frac{(f_i(x) - f_i(g))}{(f_i^{max} - f_i^{min})} \right)^2} \; w_i \in]0, 1[\; \sum_{i=1}^{M} w_i = 1 \qquad (10)$$

where x is a solution, g is a reference point, f_i^{max} and f_i^{min} represent the upper and the lower bounds of the i-th objective, respectively, and w_i is the weight associated to each objective. The r-dominance is defined as follows:

Definition 4 *r-dominance*
Assuming a population of individuals P, a reference vector g, and a weight vector w, a solution x is said to r-dominate a solution y (denoted by $x \prec_r y$) if one of the following statements holds true:

1. x dominates y in the Pareto sense,
2. x and y are Pareto-equivalent and $D(x, y, g) < -\delta$, where $\delta \in [0, 1]$ and:

$$d(x, y, g) = \frac{dist(x, g) - dist(y, g)}{dist_{max} - dist_{min}} \qquad (11)$$

$$dist_{max} = Max_{z \in P} \; dist(z, g) \qquad (12)$$

$$dist_{min} = Min_{z \in P} \; dist(z, g) \qquad (13)$$

δ is termed as the non-r-dominance threshold.

After substituting the Pareto dominance with the r-dominance in the NSGA-II algorithm with an adaptive management of the δ parameter, the performance of the resulting preference-based MOEA, named r-NSGA-II, has been assessed on several test problems where the number of objectives is varying between two and ten objectives. The experimental results show that r-NSGA-II outperforms several recent reference point approaches. Moreover, the r-dominance was able to guide the

search using the DM' s preferences and to control the spread of the region of interest. However, r-NSGA-II algorithm has faced difficulties in solving highly multi-modal problems such as ZDT4 [32].

- **PICEA-g: Preference-Inspired Co-Evolutionary Algorithm-Goals**

PICEA-g was introduced by Wang et al. [33]. PICEA-g is a posteriori preference-based algorithm where the intervention of the DMs is performed after obtaining a solution set which approximates the real Pareto front. The main idea of this algorithm is to provide DMs with both a proximal and a diverse representation of the entire Pareto front before the elicitation and the application of their preferences. As the search progress, PICEA-g coevolves a family of DM' preferences together with a population of candidate solutions. Thus, the solutions would gain fitness by performing well against the preferences and the preferences would gain fitness by offering comparability between solutions. The general principle of PICEA-g is as follows. The PICEA-g begins by initializing a population of candidate solutions *S* and preference sets *G* of fixed size *N* and *NGoal*, respectively. In each generation *t*, genetic variation operators are applied to the parents *S(t)* in order to produce *N* offspring *Sc(t)*. Simultaneously, *NGoal* new preference sets *Gc(t)*, are randomly regenerated based on the initial bounds. Thereafter, *S(t)* and *Sc(t)* and both *G(t)* and *Gc(t)* are then pooled, respectively. After that, the obtained populations are sorted based on the fitness. Finally, a truncation selection is applied to select *N* solutions to form the new parent population *S(t+1)* and *NGoal* solutions as new preference population *G(t + 1)*. The method to calculate the fitness F_s, of a candidate solution *s* is defined as follows:

$$F_s = 0 + \sum_{g \in G \uplus G_c | s \preccurlyeq g} \frac{1}{n_g} \tag{14}$$

where n_g is the number of solutions that satisfy preference *g*. It should be noted that if *s* does not satisfy any *g*, then F_s is equal to zero. The fitness F_g of a preference *g* can be expressed as follows:

$$F_g = \frac{1}{1+\alpha} \tag{15}$$

where

$$\alpha = \begin{cases} 1 & \text{if } n_g = 0 \\ \frac{n_g - 1}{2N-1} & \text{otherwise} \end{cases} \tag{16}$$

where *N* is the candidate solution population size. After calculating fitness values, the non-dominated solutions in $S \cup S_c$ are identified. Then, based on the fitness, the best *N* non-dominated solutions are selected to constitute the new parent *S(t + 1)*. The authors reported that PICEA-g outperforms several state-of-the-art methods in terms of convergence and spread when compared on WFG test problems with up to 10 objectives.

- **Other Preference Incorporation-Based Approaches**

In the context of incorporating preference information in EMO, many studies have been made [34, 35]. A key point in preference-based approaches is the timing of integrating the preference information into the optimizing process. In fact, the DM can provide his/her preferences before (a priori), after (a posteriori), or during (interactively) the MOEA run [36–38]. Since the search direction is biased towards the area of the Pareto front on which the DM would like to focus (i.e., ROI), the priori and interactive algorithms can reduce the computational load during the search. However, the posteriori preference-based algorithms are inferior to the above mentioned classes since they might obtain a large number of solutions that the DM is not interested in [28]. Deb and Kumar [40] proposed the reference direction-based NSGA-II (RD-NSGA-II). In each iteration, the DM supplies a reference direction in the objective space. Thereafter, the solutions are ranked using an achievement scalarizing function and the crowding distance value. RD-NSGA-II has demonstrated good results when tested on DTLZ functions with up to 10 objectives. However, the population diversity degradation that can be yielded when using a single reference direction remains a significant matter. Preference-inspired co-evolutionary algorithms (PICEAs) represent an example of a posteriori preference-based algorithm that tries to avoid the intervention of the DM before or during the optimization process. In PICEAs preferences are modeled as a set of solutions which co-evolve with the population [41, 42]. In [41, 42], the authors tested a-PICEA-g and PICEA-w on the WFG test problems with up to 7 objectives. Table 4 presents a comparison of the studied preference incorporation-based approaches that are classified into three main classes: (1) Priori preference-based approaches, (2) Interactive preference-based approaches, and (3) Posteriori preference-based approaches.

2.4 Indicator-Based Approaches

- **IBEA: Indicator-Based Evolutionary Algorithm**

IBEA was introduced by Zitzler and Künzli [43]. They proposed a general IBEA where they used a binary performance indicator in the selection process. Initially, IBEA begins by generating an initial population P. Then, the algorithm calculates the fitness value of each solution x in P. In fact, the fitness value is a measure for the loss in quality if a solution x is removed from P. After computing all the fitness values of all individuals in the population, the worst individual is removed from the population and the fitness values of the residual population must be updated. In the following, the selection step is used in creating the mating pool $P^{'}$. When we compare IBEA with the use of two binary performance indicators the additive ε-indicator and the I_{HD}-indicator to Pareto-based MOEAs such as SPEA2 and NSGA-II, we note that IBEA can greatly improve the quality of the generated Pareto set approximation. In addition, IBEA outperforms NSGA-II and SPEA2 in term of convergence. However, the parameter κ which is a scaling factor of the fitness function values should be

Table 4 Comparison of preference incorporation-based approaches for MaOPs: MObjs means maximum number of objectives (inspired by [28, 31])

Classes	Algorithms	References	MObjs	Test problems	Preference information
Priori preference-based approaches	PBEA	Thiele et al. [30]	5	LPMS	Reference points
	SBGA	Gong et al. [39]	20	DTLZ	Preferred regions
Interactive preference-based approaches	R-NSGA-II	Deb et al. [29]	10	DTLZ	Reference points
	r-NSGA-II	Ben Said et al. [31]	10	DTLZ	Reference points
	RD-NSGA-II	Deb and Kumar [40]	10	DTLZ	Reference directions
Posteriori preference-based approaches	PICEA-g	Wang et al. [33]	10	WFG	Weight vectors
	a-PICEA-g	Wang et al. [41]	7	WFG	Goal vectors
	PICEA-w	Wang et al. [42]	7	WFG	Weight vectors

appropriately chosen. The main weakness of IBEA is the computational cost of the quality indicator value. Several variants of IBEAs have been proposed such as the work of Basseur and Bruke [44] in which they extended IBEA and proposed a multi-objective local search algorithm called IBMOLS that uses a local search operator and the work of Wagner et al. [45] that reported good results for MaOPs. Since, IBEAs do not use Pareto dominance, their search ability is not severely deteriorated by the increase of the number of objectives. It should be noted that most of the existing variants use the hypervolume as an indicator but one difficulty arises in using the hypervolume when dealing with a large number of objectives which is the high computational cost of the hypervolume calculation.

- **SMS-EMOA: *S* Metric Selection-Based Evolutionary Multi-objective Algorithm**

One of the most successfully used indicator-based MOEAs, is the S-Metric-Selection-EMOA (SMS-EMOA) proposed by Emmerich et al. [46]. The SMS-EMOA invokes firstly the non-dominated sorting that is used as a ranking criterion. Secondly, it uses the hypervolume indicator as a selection mechanism to discard the individual that contributes the least hypervolume to the worst-ranked front. The SMS-EMOA

algorithm starts with generating a new population P with μ individuals. In each iteration, there is a new individual that is generated by the application of random variation operators. An individual becomes a member of the population if it replaces dominated individuals and contributes to a higher quality of the population. Thus, the selection criterion ensures that the non-dominated individuals could not be replaced by the dominated ones. Then, the algorithm applied the fast-non-dominated-sort-algorithm used in NSGA-II to compute the Pareto fronts. After that, an individual is rejected from the worst ranked front R_I if it contains more than one individual. Thus, the individual $n \in R_I$ that minimizes the following equation is discarded:

$$\Delta_S(n, R_I) = S(R_I) - S(R_I\{n) \tag{17}$$

where the $\Delta_S(n, R_I)$ represents the contribution of n to the S metric value of its appropriate front. The application of this algorithm to several standards benchmark shows that it is suitable for Pareto optimization with two and three objectives. Rather than that, SMS-EMOA outperforms a number of Pareto-based algorithms in term of convergence. It is also shown that it provides solutions that are well distributed on the Pareto Front. The main disadvantage of this indicator based-MOEA is the high computational coast of the S-metric values with problems evolving more than three objectives. Moreover, SMS-EMOA is well-suited for real-world applications with a limited number of function evaluations. Wagner and Neumann [47] have compared SMS-EMOA to a number of Pareto-based algorithms and indicator-based algorithms on MaOPs. The results show that SMS-EMOA is unable to find the front of the high-dimensional DTLZ1 and DTLZ3 test problems.

- **AGE: Approximation-Guided Evolutionary**

AGE was proposed by Bringmann et al. [48]. AGE uses the additive approximation. In fact, the additive approximation of the set B with respect to the set A is expressed as follows:

$$\alpha(A, B) = \max_{a \in A} \min_{b \in B} \max_{1 \le i \le N} (a_i - b_i) \tag{18}$$

It could also use the multiplicative approximation which is similar to the additive approximation by just replacing $a_i - b_i$ with $\frac{a_i}{b_i}$. The goal is to minimize the additive approximation that measures the approximation quality of the population B with respect to the archive A. The archive A contains all non-dominated solutions seen so far. However, the additive approximation is not locally sensitive to the changes of the output population. AGE uses another sensitive indicator that should be minimized which is defined as follows:

$$S_\alpha(A, B) = (\alpha_1, \ldots, \alpha_{|A|}) \tag{19}$$

where $S_\alpha(A, B)$ is the result of sorting decreasingly the set $\alpha(\{a\}, B) | a \in A$. The algorithm begins by generating a population P of μ individuals. In each iteration, we obtain λ new offspring by selecting randomly two individuals from the population and

applying the crossover and the mutation operators. Those λ new offspring individuals are added to P and a new population Q is obtained. After that, only non dominated solutions obtained from Q are added to A. In addition, there are two criteria to add a solution S to A: (1) S is not dominated by any existing individual in A and (2) individuals that dominate S are removed. In each generation, the individual p with lexicographically worst approximation is removed from Q. AGE was compared to several MOEAs and as a result it was proved that AGE outperforms them in term of the quality of the approximation set obtained especially when dealing with many objectives and the covered hypervolume. Wagner and Neumann [47] extended AGE and presented a new version called AGE-II where they control the size of the archive by storing the additive ε-approximation of the non-dominated solutions and they propose a new strategy for the parent selection.

- **MOMBI: Many-objective Meta-Heuristic Based on the R2 Indicator**

MOMBI was introduced by Gomez and Coello Coello [49]. The MOMBI algorithm is based on the R2 indicator which is defined as:

$$R2(A, V, Z^*) = \frac{1}{|V|} \sum_{v \in V} \min_{a \in A} \left\{ \max_{1 \leq j \leq m} v_j |Z_j^* - a_j| \right\} \tag{20}$$

where A is an individual set, V is a set of weight vectors, and Z^* is used as a reference point which is never dominated by any feasible solution. This algorithm produces a non-dominated sorting scheme based on the utility functions. The main idea is to group solutions that optimize the set of utility functions and gives them the first rank. Then, those solutions will be removed and a second rank will be identified in the same manner. The process will continue until all the population members will be ranked. We notice that MOMBI uses the non-dominated sorting scheme without using the usual Pareto dominance. The MOMBI algorithm is described as follows. MOMBI begins by generating a population P randomly. Then, we obtain the objective function values, the ideal and the nadir point, and the R2-ranking of all P members. After that, a binary tournament selection using the rank of the solutions and the mutation and crossover operators are performed to create an offspring population Q. Next, the reference points are updated with the minimum and maximum objective function values and the population R which is the union of both P and Q populations is ranked using the R2 indicator. In order to reduce the population, MOMBI selects the best N individuals according to their ranks. The experimental results show that MOMBI outperforms MOEA/D [50] in most cases. This algorithm performs well when dealing with many-objective. However, its main weakness is its high computational cost.

- **Other Indicator-Based Approaches**

Indicator-based approaches are yet another direct way to solve MaOPs [51]. In fact, in an indicator-based algorithm, an indicator is not only used to evaluate the obtained approximation set according to the indicator but also indicator values are used to guide the search process. Although, an emerging trend is the use of a quality indicator to solve a MaOP. We identify two indicators that have been applied

Table 5 Comparison of indicator-based approaches for MaOPs: MaxObjs means maximum number of objectives

Classes	Algorithms	References	MaxObjs	Test problems
Hypervolume-based approaches	IBEA	Zitzler and Künzli [43]	4	EXPO
	SMS-EMOA	Wagner and Neumann [47]	20	DTLZ
				WFG
				LZ
	HypE	Bader and Zitzler [52]	50	DTLZ
				WFG
				MKP
R2 indicator-based approaches	AGE	Bringmann et al. [48]	20	DTLZ
	AGE-II	Wagner and Neumann [47]	20	DTLZ
	MOMBI	Gomez and Coello Coello [49]	8	DTLZ
				WFG
	R2-MOGA	Manriquez et al. [54]	10	DTLZ
	R2MODE	Manriquez et al. [54]	10	DTLZ

by most indicator-based approaches for MaOPs: Hypervolume indicator and R2 indicator (cf. Table 5). In fact, tow main issues arise when using the hypervolume indicator to solve MaOPs. First, the computational cost of the hypervolume value is high. Second, the hypervolume might not be appropriate when the DM aims to find a uniform spread optimal set.

In order to deal with the high computational cost of computing the exact hypervolume values, Bader and Zitzler [52] introduced the Hypervolume Estimation Algorithm (HypE) where they used a Monte Carlo algorithm [53] in order to approximate the exact hypervolume values. In this algorithm, the non-dominated solutions are compared according to their hypervolume-based fitness values. Specifically, HypE uses an environmental selection to create a new population from the best solutions in the union set of the parent and offspring populations and estimate the hypervolume value by sampling solutions in different fronts. The experimental results showed that HypE achieved competitive performance in terms of the average hypervolume on a number of test problems with up to 50 objectives. Manriquez et al. [54] proposed two R2-indicator-based approaches which are: R2-MOGA and R2MODE. Those latter present a modified version of Goldberg's non-dominated sorting method. The obtained results on DTLZ with up to 10 objectives indicate that these algorithms can outperform SMS-EMOA in term of computational time.

2.5 Decomposition-Based Approaches

- **MOGLS: Multi-objective Genetic Local Search**

MOGLS was first proposed by Ishibuchi and Murata [55] and improved by Jaszkiewicz [56]. In fact, the Genetic Local Search (GLS) is a metaheuristic that hybridizes recombination operators with local search or with other local improvement heuristics. The basic idea of MOGLS is to transform the original MaOP into a simultaneous optimization of a collection of weighted Tchebycheff functions or weighted sum functions. At each iteration, the algorithm generates a random weight vector to evaluate the current population and uses an external population to store the non-dominated solutions. The Jaszkiewicz's MOGLS can be described as follows:

- **Step 1**: An initialization step is performed to initialize a set of current solutions *CS* with *S* solutions, a vector $z = (z_1, z_2, \ldots, z_m)^T$ where z_i is the largest value found so far for the objective f_i, and an external population *EP* to store the non-dominated solutions of *CS*;
- **Step 2**: Then, the external population *EP* is updated as follows:
  1. A randomly weight vector w is generated, k (i.e., the size of temporary elite population) best solutions with regard to the used scalarizing function are selected to form a temporary elite population T, and a new solution y is generated by applying the genetic operators to two randomly chosen solutions from T;
  2. A solution y' is generated by applying a local improvement heuristic to y;
  3. The vector z is updated: For each $j = 1, \ldots, m$, if $z_j < f_j(y')$, then set $z_j = f_j(y')$. This step is performed only in the case where the Tchebycheff approach is used, this step is removed otherwise;
  4. The solution y' is added to *CS*, if y' is better than the worst solution in T with regard to the used scalarizing function and different from the solutions in T with regard to the m real-valued objective functions. In the case where the size of *CS* is larger than $K \times S$ the oldest solution is deleted from *CS*.
  5. All the solutions in *EP* that are dominated by y' are removed and y' is added to *EP* if there is no solutions that dominate it.

The above described steps are repeated until a stopping criterion is satisfied. The experimental results have shown that MOGLS may work well on MaOPs. However, the use of the recombination operator and the appropriate selection of the solutions for recombination influence the performance of MOGLS. Moreover, as reported in [50], the upper bound of the size of *CS* which is equal to $K \times S$ influences the space complexity of MOGLS.

- **MOEA/D: Multi-objective Evolutionary Algorithm Based on Decomposition**

MOEA/D is one of the most popular decomposition-based algorithm proposed by Zhang and Li [50]. MOEA/D decomposes the MaOP into N sub-problems (N is the population size) that are optimized simultaneously. It uses a set of well-distributed

weight vectors λ_j to cover the whole Pareto front. The algorithm begins by determining a neighborhood of T weight vectors for each λ_j. After that, the population members are assigned to the weight vectors. Thereafter, two solutions from neighboring weight vectors are mated and an offspring solution is created. The offspring solution is then evaluated using a scalarizing function. This generated new solution can also replace several current solutions of its neighboring sub-problems when it outperforms them. Three versions of scalarizing functions are adopted for MOEA/D: (1) weighted sum approach [57], (2) weighted Tchebycheff approach [57], and (3) boundary intersection approach [58, 59]. Ishibuchi et al. [60] studied the relation between the neighborhood size and the performance of MOEA/D in solving many-objective problems. In this work, it was proved that a large replacement neighborhood improves the search ability of MOEA/D in the objective space. However, a small replacement and mating neighborhood are beneficial to maintain the diversity. MOEA/D has demonstrated very interesting results on several MaOPs. However, its main shortcoming is the degradation of diversity and solution distribution when tackling scaled problems.

- **NSGA-III: Non-dominated Sorting Genetic Algorithm III**

Deb and Jain [61] proposed NSGA-III which remains similar to the NSGA-II algorithm with some changes in its selection mechanism. The general principle of this MaOEA can be described as follows. Differently to MOEA/D, NSGA-III makes the decomposition based on a set of well-distributed reference points. Afterwords, a randomly parent population P_t with N individuals is generated. The following steps are iterated until the termination criterion is satisfied. The algorithm begins by creating an offspring population Q_t with N individuals obtained by applying genetic operators to P_t. Thereafter, the two populations P_t and Q_t are merged with each other to form a new population R_t of size $2N$. After that, the combined population R_t is sorted into several fronts using the non-dominated sorting as done in NSGA-II. Then, a new population S_t is constructed starting from the first front F_1 until the size of the population S_t becomes equal to N or for the first time greater than N. Let us suppose that the last accepted level is the lth level. Therefore, all solutions from level $(l + 1)$ onwards are rejected. In most cases, the last front F_1 is accepted partially. NSGA-II uses a niching strategy to choose individuals from the last front which are situated in the least crowding regions in F_1. However, the crowding distance is not well-suited for MaOPs. For this reason, the selection mechanism was modified in NSGA-III. Figure 3 illustrates the two mechanisms used in (a) NSGA-II and (b) NSGA-III to maintain diversity among solutions. The principle of the selection mechanism is as follows. It begins by normalizing the population members and the supplied reference points. Then, it calculates the perpendicular distance between a solution in S_t and each of the reference lines that join the ideal point with the reference points. So that, each individual in S_t is associated with the reference point having the minimum perpendicular distance. Thereafter, a niche preservation operation is performed and it can be summarized by this two following steps:

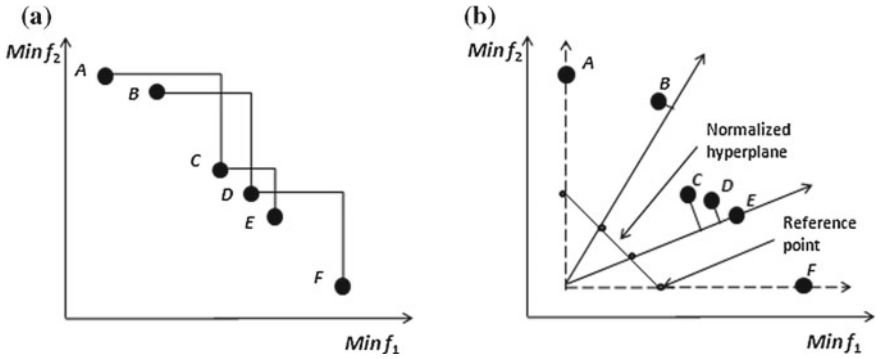

Fig. 3 Illustration of working principles of **a** NSGA-II versus **b** NSGA-III (inspired by [62])

- **Step 1**: Count the number of individuals from $P_{t+1} = S_t / F_l$ that are associated with each reference point;
- **Step 2**: Define a reference point set that contains the reference points having the minimum niche count ρ. If this set contains more than one point, we choose one of them at random.

Hence, four scenarios are identified which are detailed in [61]. After that, we update the different niche counts. It should be noted that this procedure is repeated until the population size of P_{t+1} becomes equal to N. NSGA-III has demonstrated very good results on problems involving up to 15 objectives. The major advantage of NSGA-III is its ability to find well-converged and well-diversified solutions. Another advantage is that it does not require any additional parameters to be set such in MOEA/D.

- **DBEA-Eps: Decomposition Based Evolutionary Algorithm for Many-objective Optimization with Systematic Sampling and Adaptive Epsilon Control**

Asafuddoula et al. [63] proposed a decomposition-based algorithm that generates a structured set of reference points, that uses an adaptive epsilon comparison to manage the balance between the convergence and the diversity, and that adopts an adaptive epsilon formulation to deal with constraints. DBEA-Eps begins with a generation of a set of reference points using the normal boundary intersection method (NBI). Thereafter, the neighborhood of each reference point (i.e., T closest reference points computed based on a Euclidean distance amongst them) is determined. Similarly to NSGA-III, DBEA-Eps normalizes the population based on intercepts calculated using M extreme points of the non-dominated set and computes the same two distance measures d_1 and d_2 used in NSGA-III to control diversity and convergence of the algorithm. Figure 4 illustrates the two distance measures d_1 and d_2 in a two objectives minimization problem. It also uses a mating partner selection to select a parent from the neighborhood of the current solution P_i with a given mating probability δ and a method of recombination using information from neighboring sub-problems. In order

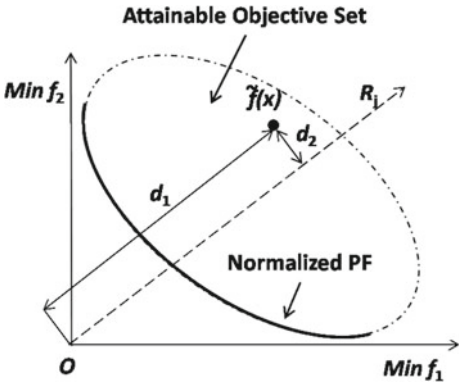

Fig. 4 Illustration of the distance measures d_1 and d_2 with respect to a reference direction. From Ref. [63]

to manage the balance between convergence and diversity, the authors proposed to use an adaptive epsilon comparison, where a child solution replaces a single parent based on the following equation:

$$(d_1, d_2) < \varepsilon_{CD}(d_1, d_2) \Rightarrow \begin{cases} d_1 < d_2, & \text{if } d_2,\ d_2 < \varepsilon_{CD} \\ d_1 < d_2, & \text{if } d_2 = d_2 \\ d_1 < d_2, & \text{otherwise} \end{cases} \tag{21}$$

where d_{2i} is the d_2 measure of the i-th individual, W is the number of reference points, and the average deviation ε_{CD} is defined as follows:

$$\varepsilon_{CD} == \frac{\sum_{i=1}^{W} d_{2i}}{W} \tag{22}$$

In this work, an epsilon level comparison is used to compare the solutions. The DBEA-Eps has demonstrated its outperformance on the DTLZ1 and DTLZ2 problems and on the three constrained engineering design optimization problems with three to seven constraints (car side impact [64], water resource management [65], and a general aviation aircraft design problem [66]). Thus, it is able to deal with unconstrained and constrained MaOPs. However, the performance is dependent on the choice of a number of parameters and several adaptive rules.

The same authors [67] have proposed the improved decomposition based evolutionary algorithm (I-DBEA) which is a modified version of DBEA-Eps. I-DBEA eliminates the use of the neighborhood size T and the mating probability δ such that the entire population is considered as a neighborhood and a first encounter replacement strategy has been adopted. Comparisons between solutions were based on an adaptive epsilon level of d_2. However, in I-DBEA, a simple precedence rule is used, where d_2 has a precedence over d_1. In the proposed algorithm, a corner-sort is used to identify M extreme points that are used to create the hyperplane and to compute the intercepts. The experimental results indicate that I-DBEA is able to deal with unconstrained and constrained MaOPs. However, as noted by the authors, this approach

is not suitable to solve problems evolving a large number of reference directions (i.e., a large population is not practical).

- **Other Decomposition-Based Approaches**

Inspired by Preference based approaches, the researchers have proposed to direct the search towards multiple well-distributed ROIs in order to cover the whole Pareto front for MaOPs. In fact, decomposition-based approaches decompose the original MOP into a collection of sub-problems that will be simultaneously optimized. Several scalarizing functions have been used to convert the problem of approximation of the PF into a number of scalar optimization sub-problems such as the weighted sum method, the Tchebycheff method, and the boundary intersection method. Those scalarizing functions have been used to decompose the problem into single objective sub-problems which are defined with the help of weight vectors (Miettinen and Mäkelä [68]). Li et al. [69] proposed MOEA/DD which presents a unified paradigm that combines dominance and decomposition-based approaches for many-objective optimization to balance between convergence and diversity. MOEA/DD uses an update procedure that depends on Pareto dominance, local density estimation, and scalarizing functions, sequentially. The authors have also proposed a modified version of MOEA/DD called C-MOEA/DD to solve constrained problems. The performance of the two algorithms was investigated on a set of unconstrained benchmark problems with up to fifteen objectives and on a number of constrained optimization problems. The obtained results have demonstrated the outperformance of both algorithms in solving problems with a high number of objectives. However, MOEA/DD is sensitive to the two parameters T and δ which represent the neighborhood size and the probability of selecting mating parents from neighboring sub-regions, respectively. Different MOEA/D variants have been proposed in the literature to tackle MaOPs such as MOEA/D-DRA (Zhang et al. [70]) and UMOEA/D (Tan et al. [71]). Yuan et al. [72] proposed the θ-NSGA-III witch is an improved version of NSGA-III, but the main difference between the two algorithms is that the θ-NSGA-III replaces the Pareto dominance used in NSGA-III with a new dominance relation which is called the θ-dominance. θ-NSGA-III outperforms MOEA/D and NSGA-III in terms of convergence. However, it was proved that this algorithm is insensitive to the parameter θ. Elarbi et al. [73] proposed a new dominance relation called TSD-dominance to deal with MaOPs. The TSD-NSGA-II represents a new many-objective version of NSGA-II where the Pareto dominance is replaced by the TSD-dominance. TSD-NSGA-II was found to be highly competitive in dealing with constrained and unconstrained problems. However, MaOPs involving the characteristics of DTLZ6-7 represent the limits of TSD-NSGA-II. Table 6 provides a comparison of some of the most prominent decomposition-based approaches for MaOPs. From the different discussed works in this chapter, we remark that the choice of a specific scalarizing function to use influences the performance of the decomposition-based algorithm

Table 6 Comparison of decomposition-based approaches for MaOPs: MaxObjs means maximum number of objectives, CP means constrained problems, UP means unconstrained problems, WV means weight vectors, RP means reference points

Algorithms	References	MaxObjs	CP	UP	WV	RP
MOGLS	Jaszkiewicz [56]	4	X	–	X	–
MOEA/D	Zhang and Li [50]	4	X	X	X	–
NSGA-III	Deb and Jain [61]	15	–	X	–	X
DBEA-Eps	Asafuddoula et al. [63]	15	X	X	–	X
I-DBEA	Asafuddoula et al. [67]	15	X	X	–	X
MOEA/DD	Li et al. [69]	15	X	X	X	–
MOEA/D-DRA	Zhang et al. [70]	5	–	X	X	–
UMOEA/D	Tan et al. [71]	5	X	X	X	–
θ-NSGA-III	Yuan et al. [72]	20	–	X	–	X
TSD-NSGA-II	Elarbi et al. [73]	20	–	X	–	X

[74]. Moreover, different methods have been used to generate a set of weight vectors (i.e., reference points) such as the systematic approach [58] and the on-the-fly weighting vector generating method [75]. However, how to configure the weight vectors is still a big challenge for decomposition-based algorithms, since those latter dramatically affect the diversity performance.

3 Performance Assessment of MaOEAs

3.1 Test Problems and Statistical Analysis

Several test problems have been used to investigate MaOEAs capabilities in approximating the Pareto front. In the literature, among the most used test function suites we find: (1) the scalable DTLZ (Deb-Thiele-Laumans- Zitzler) suite and the scalable WFG (Walking Fish Group) Toolkit. MaOEAs have also been used in some combinatorial problems such as knapsack problems. Recently, a number of scalable constrained test problems having three up to 15 objectives have been introduced [64]. Those latter are characterized with various types of difficulties to an algorithm. Table 7 summarizes the above mentioned test problems in this chapter.

Many existing works use the median and the interquartile range values obtained by a specific performance metric in order to compare the performance of different algorithms. However, each algorithm can behave differently from one run to another. Hence, the use of a statistical testing approach is necessary. Firstly, we should check whether the obtained results are normally distributed or not using the Kolmogorov–Smirnov test. If data follow a normal distribution, we can use the t-test when comparing between two algorithms and the ANOVA one if the comparison

Table 7 Summary of the mentioned test problems (inspired by [28])

Test problems	References	Remarks
DTLZ	Deb et al. [21]	Scalable problem
WFG	Huband et al. [22]	Scalable problem
MKP	Zitzler et al. [9]	Multi-objective 0/1 Knapsack Problem
NRP	Burke et al. [18]	Nurse Restoring Problem
DTLZ5(I,M)	Deb et al. [19]	Scalable problem
$DTLZ2_{BZ}$	Brockhoff et al. [77]	Modified version of DTLZ2
LPMS	Miettinen et al. [78]	Locating a pollution monitoring station
EXPO	Thiele et al. [79]	A network processor application comprising problem
LZ	Li et al. [80]	Continuous test problems
Car side impact	Jain and Deb [64]	Engineering constrained problem
Water resource management	Ray et al. [65]	Engineering constrained problem
General aviation aircraft design	Hadka et al. [66]	Engineering constrained problem

involves more than two algorithms. To avoid verifying data normality, we can directly use the Wilcoxon test and the Kruskal-Wallis as non-parametric alternatives of the t-test and the ANOVA one respectively. For more details about statistical testing in evolutionary computation, the reader could refer to [76].

3.2 Performance Metrics

In the literature, fewer are the performance metrics that have been used to evaluate the performance of MaOEAs. The most common used performance metrics are described in this subsection.

- **The Inverted Generational Distance (IGD)**

The IGD is a performance metric that measures the distance between the true Pareto front and the closest individual in an approximation set. It is expressed as follows [81]:

$$I_{IGD} = \frac{(\sum_{i=1}^{PF^*} d_i^q)^{\frac{1}{q}}}{PF^*} \tag{23}$$

where d_i^q is the Euclidean distance between an individual from the Pareto front PF^* to its nearest individual in the approximation set and $q = 2$. In fact, the lower are the I_{IGD} values, the better are the obtained sets. Moreover, the IGD can measure both convergence and diversity. The IGD metric requires a reference true Pareto front in the calculation. However, it is difficult to determine the reference true Pareto front of MaOPs. Thus, an exact method to generate a set of uniformly well-spread points over the true Pareto front has been proposed for the DTLZ1-DTLZ4 test problems [69]. This method locates exactly the intersecting points of the reference points generated by the algorithm and the Pareto-optimal surface since the exact Pareto-optimal surfaces of DTLZ1 to DTLZ4 are known a priori. For DTLZ1, given a reference point $r = (r_1, \ldots, r_M)^T$, the i-th objective function of a Pareto-optimal solution x^* is computed as follows:

$$f_i(x^*) = 0.5 \times \frac{r_i}{\sum_{j=1}^{M} r_j} \tag{24}$$

As for DTLZ2 to DTLZ4, given a reference point $r = (r_1, \ldots, r_M)^T$, the i-th objective function of a Pareto-optimal solution x^* is computed as follows:

$$f_i(x^*) = \frac{r_i}{\sqrt{\sum_{j=1}^{M} r_j^2}} \tag{25}$$

- **The Generational Distance (GD)**

The GD metric evaluates an average distance of an approximation set P from the true Pareto front PF^* [82]. It is defined as follows:

$$I_{GD} = \frac{\sqrt{\sum_{i=1}^{P} d_i^2}}{|P|} \tag{26}$$

where d_i is the Euclidean distance between the solution $i \in P$ and the nearest member of PF^*. A value of $I_{GD} = 0$ indicates that all the individuals of the approximation set P are in the true Pareto front PF^*. This metric evaluate only the convergence of an algorithm.

- **The Hypervolume (HV)**

The HV indicator is a unary indicator that calculates the volume of the hypercube dominated by an approximation set. This indicator can be expressed as follows:

$$I_{HV} = \bigcup_i S(i) | i \in PF^* \tag{27}$$

where $S(i)$ is the hypercube bounded by a solution i and a reference point. In fact, the choice of the reference point is important because it influences the outcome of this metric. The reference point can be constructed by the worst objective function

values. This measure captures both convergence and diversity. A large value of the HV metric is desirable.The main drawback of this metric is the high computational cost needed to compute the exact HV [53].

- **The Spread** (Δ)

The Δ metric measures the deviation among neighboring solutions in the non-dominated solution set P furnished by the MOEA [83]. Analytically, it is stated as follows:

$$I_{\Delta} = \sum_{i=1}^{|P|} \frac{|d_i - \overline{d}|}{|P|} \tag{28}$$

where d_i is the Euclidean distance between two neighbor solutions in P and $\overline{d}$ is the average of these distances. In fact, a smaller value of Δ indicates that the algorithm is able to find a diverse set of non-dominated solutions.

4 Conclusion and Future Research Paths

In this chapter, we have first described the related issues that MOEAs encounter when the dimensionality of the objective space increases. Then, we have surveyed the most prominent MaOEAs. We have proposed to classify the existing MaOEAs into five classes: Preference ordering relation-based approaches, objective reduction-based approaches, preference incorporation-based approaches, indicator-based approaches, and decomposition-based approaches.

The preference ordering relation-based approach aims to propose a preference relation that induce a finer order than that induced by the Pareto dominance relation. Hence, the non-dominated solutions are further ranked using this relation. This method has the ability to increase the selection pressure towards the Pareto front. However, it decreases the diversity of the solutions. Thus, it will be interesting to propose new flexible selection methods and new diversity mechanisms for preference ordering relation-based approaches.

The objective reduction-based approach finds the relevant objectives and eliminates the redundant objectives that are not essential to describe the Pareto optimal front. In other words, it identifies the non-conflicting objectives and discards them to reduce the number of objectives of the MaOPs during the search process. Two reduction methods can be identified: (1) the offline dimensionality reduction method and (2) the online dimensionality reduction method. The main advantage of this approach is that it reduces the computational cost of the MaOEAs. However, for MaOps with non-redundant objectives this approach may fail to reduce the number of objectives.

The preference incorporation-based approach exploits the DM preferences in order to differentiate between Pareto equivalent solutions. It focuses the search process on a specific region of the Pareto front. Preference incorporation-based

methods can be classified into the three following subclasses: (1) priori preference-based approaches, (2) interactive preference-based approaches, and (3) posteriori preference-based approaches. In fact, one of the issues that arises when using the a posteriori preference-based approach is that it may obtain a high number of solutions that the DM is not interested in [28].

The indicator-based approach transforms the MaOP into the problem of optimizing an indicator by evaluating the solutions using a performance metric. The high computational cost of the hypervolume calculation represents a difficulty for this approach when dealing with high dimensional objective space. Hence, it will be interesting to propose new indicators that are well-adapted for MaOPs.

The decomposition-based approach decomposes the problem into several subproblems that will be simultaneously optimized using scalarizing functions. Actually, decomposition is the most successful approach to solve MaOPs. Both the scalarizing function and the method used to generate a structured set of reference points (or weight vectors) influence the performance of a decomposition-based algorithm. For this reason, more future research are needed on proposing new methods for generating weight vectors.

References

1. Bechikh, S., Chaabani, A., Said, L.B.: An efficient chemical reaction optimization algorithm for multiobjective optimization. IEEE Trans. Cybern. **45**(10), 2051–2064 (2015)
2. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Trans. Evolut. Comput. **6**(2), 182–197 (2002)
3. Zitzler, E., Laumanns, M., Thiele, L., Zitzler, E., Zitzler, E., Thiele, L., Thiele, L.: Spea 2: Improving the strength pareto evolutionary algorithm (2001)
4. Sülflow, A., Drechsler, N., Drechsler, R.: Robust multi-objective optimization in high dimensional spaces. In: Evolutionary Multi-criterion Optimization, pp. 715–726. Springer, Heidelberg (2007)
5. Kasprzyk, J.R., Reed, P.M., Kirsch, B.R., Characklis, G.W.: Managing population and drought risks using many-objective water portfolio planning under uncertainty. Water Resour. Res. **45**(12) (2009)
6. Garza-Fabre, M., Pulido, G. Coello, C.A.C.: Ranking methods for many-objective optimization. In: MICAI 2009: Advances in Artificial Intelligence, pp. 633–645. Springer, Heidelberg (2009)
7. Jaimes, A.L.: Técnicas para resolver problemas de optimización con muchas funciones objetivo usando algoritmos evolutivos. Ph.D. thesis (2011)
8. Sato, H., Aguirre, H.E., Tanaka, K.: Controlling dominance area of solutions and its impact on the performance of moeas. In: Evolutionary Multi-criterion Optimization, pp. 5–20. Springer, Heidelberg (2007)
9. Zitzler, E., Thiele, L.: Multiobjective optimization using evolutionary algorithms-a comparative case study. In: Parallel Problem Solving from Nature-PPSN V, pp. 292–301. Springer, Heidelberg (1998)
10. Farina, M., Amato, P.: On the optimal solution definition for many-criteria optimization problems. In: Proceedings of the NAFIPS-FLINT International Conference, pp. 233–238 (2002)
11. Bentley, P.J., Wakefield, J.P.: Finding acceptable solutions in the pareto-optimal range using multiobjective genetic algorithms. In: Soft Computing in Engineering Design and Manufacturing, pp. 231–240. Springer, Heidelberg (1998)

12. Li, M., Zheng, J., Li, K., Yuan, Q., Shen, R.: Enhancing diversity for average ranking method in evolutionary many-objective optimization. In: Parallel Problem Solving from Nature, PPSN XI, pp. 647–656. Springer, Heidelberg (2010)
13. Corne, D.W., Knowles, J.D.: Techniques for highly multiobjective optimisation: some nondominated points are better than others. In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, pp. 773–780. ACM, New York (2007)
14. Drechsler, N., Drechsler, R., Becker, B.: Multi-objective optimisation based on relation favour. In: Evolutionary Multi-criterion Optimization, pp. 154–166. Springer, Heidelberg (2001)
15. Drechsler, N., Drechsler, R., Becker, B.: Multi-objective optimization in evolutionary algorithms using satisfiability classes. In: Computational Intelligence, pp. 108–117. Springer, Heidelberg (1999)
16. Di Pierro, F., Khu, S.-T., Savic, D.A.: An investigation on preference order ranking scheme for multiobjective evolutionary optimization. IEEE Trans. Evolut. Comput. **11**(1), 17–45 (2007)
17. Das, I.: A preference ordering among various pareto optimal alternatives. Struct. Optim. **18**(1), 30–35 (1999)
18. Burke, E.K., De Causmaecker, P., Berghe, G.V., Van Landeghem, H.: The state of the art of nurse rostering. J. Sched. **7**(6), 441–499 (2004)
19. Deb, K., Saxena, D.: Searching for pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems In: Proceedings of the World Congress on Computational Intelligence (WCCI-2006), pp. 3352–3360 (2006)
20. Singh, H.K., Isaacs, A., Ray, T.: A pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems. IEEE Trans. Evolut. Comput. **15**(4), 539–556 (2011)
21. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable multi-objective optimization test problems. In: Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC'02, vol. 1, pp. 825–830. IEEE, New York (2002)
22. Huband, S., Hingston, P., Barone, L., While, L.: A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans. Evolut. Comput. **10**(5), 477–506 (2006)
23. Jaimes, A.L, Coello, C.A.C., Chakraborty, D.: Objective reduction using a feature selection technique. In: Proceedings of the 10th annual conference on Genetic and evolutionary computation, pp. 673–680. ACM, New York (2008)
24. Mitra, P., Murthy, C., Pal, S.K.: Unsupervised feature selection using feature similarity. IEEE Trans. Pattern Anal. Mach. Intell. **24**(3), 301–312 (2002)
25. Brockhoff, D., Saxena, D.K., Deb, K., Zitzler, E.: On handling a large number of objectives a posteriori and during optimization. In: Multiobjective Problem Solving from Nature, pp. 377–403. Springer, Heidelberg (2008)
26. Saxena, D.K., Duro, J.A., Tiwari, A., Deb, K., Zhang, Q.: Objective reduction in many-objective optimization: linear and nonlinear algorithms. IEEE Trans. Evolut. Comput. **17**(1), 77–99 (2013)
27. Brockhoff, D., Zitzler, E.: Are all objectives necessary? on dimensionality reduction in evolutionary multiobjective optimization. In: Parallel Problem Solving from Nature-PPSN IX, pp. 533–542. Springer, Heidelberg (2006)
28. Li, B., Li, J., Tang, K., Yao, X.: Many-objective evolutionary algorithms: a survey. ACM Comput. Surv. (CSUR) **48**(1), 13 (2015)
29. Deb, K., Sundar, J.: Reference point based multi-objective optimization using evolutionary algorithms. Int. J. Comput. Intell. Res. **2**, 273–286 (2006)
30. Thiele, L., Miettinen, K., Korhonen, P.J., Molina, J.: A preference-based evolutionary algorithm for multi-objective optimization. Evolut. Comput. **17**(3), 411–436 (2009)
31. Said, L.B, Bechikh, S., Ghédira, K.: The r-dominance: a new dominance relation for interactive evolutionary multicriteria decision making. IEEE Trans. Evolut. Comput., **14**(5), 801–818 (2010)
32. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: empirical results. Evolut. Comput. **8**(2), 173–195 (2000)

33. Wang, R., Purshouse, R.C., Fleming, P.J.: Preference-inspired coevolutionary algorithms for many-objective optimization. IEEE Trans. Evolut. Comput. **17**(4), 474–494 (2013)
34. Bechikh, S., Kessentini, M., Said, L.B., Ghédira, K.: Chapter four-preference incorporation in evolutionary multiobjective optimization: a survey of the state-of-the-art. Adv. Comput. **98**, 141–207 (2015)
35. Bechikh, S.: Incorporating decision maker's preference information in evolutionary multi-objective optimization. Ph.D. thesis, University of Tunis, ISG-Tunis, Tunisia (2013)
36. Bechikh, S., Said, L.B., Ghédira, K.: Negotiating decision makers' reference points for group preference-based evolutionary multi-objective optimization. In: 2011 11th International Conference on Hybrid Intelligent Systems (HIS), pp. 377–382. IEEE, New York (2011)
37. Bechikh, S., Said, L.B., Ghédira, K.: Group preference based evolutionary multi-objective optimization with nonequally important decision makers: Application to the portfolio selection problem. Int. J. Comput. Inf. Syst. Ind. Manag. Appl. **5**(278–288), 71 (2013)
38. Kalboussi, S., Bechikh, S., Kessentini, M., Said, L.B.: Preference-based many-objective evolutionary testing generates harder test cases for autonomous agents. In: Search Based Software Engineering, pp. 245–250. Springer, Heidelberg (2013)
39. Dunwei, G., Gengxing, W., Xiaoyan, S.: Set-based genetic algorithms for solving many-objective optimization problems. In: 2013 13th UK Workshop on Computational Intelligence (UKCI), pp. 96–103 (2013)
40. Deb, K., Kumar, A.: Interactive evolutionary multi-objective optimization and decision-making using reference direction method. In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, pp. 781–788. ACM, New York (2007)
41. Wang, R., Purshouse, R.C., Fleming, P.J.: On finding well-spread pareto optimal solutions by preference-inspired co-evolutionary algorithm. In: Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation, pp. 695–702. ACM, New York (2013)
42. Wang, R., Purshouse, R.C., Fleming, P.J.: Preference-inspired co-evolutionary algorithm using weights for many-objective optimization. In: Proceedings of the 15th Annual Conference Companion on Genetic and Evolutionary Computation, pp. 101–102. ACM, New York (2013)
43. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Parallel Problem Solving from Nature-PPSN VIII, pp. 832–842. Springer, Heidelberg (2004)
44. Basseur, M., Burke, E.K.: Indicator-based multi-objective local search. In: IEEE Congress on Evolutionary Computation, 2007. CEC 2007, pp. 3100–3107. IEEE, New York (2007)
45. Wagner, T., Beume, N., Naujoks, B.: Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In: Evolutionary Multi-criterion Optimization, pp. 742–756. Springer, Heidelberg (2007)
46. Emmerich, M., Beume, N., Naujoks, B.: An emo algorithm using the hypervolume measure as selection criterion. In: Evolutionary Multi-Criterion Optimization, pp. 62–76. Springer, Heidelberg (2005)
47. Wagner, M., Neumann, F.: A fast approximation-guided evolutionary multi-objective algorithm. In: Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation, pp. 687–694. ACM, New York (2013)
48. Bringmann, K., Friedrich, T., Neumann, F., Wagner, M.: Approximation-guided evolutionary multi-objective optimization. IJCAI Proc. Int. Joint Conf. Artif. Intell. **22**, 1198–1203 (2011)
49. Gomez, R.H., Coello, C.C.: Mombi: A new metaheuristic for many-objective optimization based on the r2 indicator. In: 2013 IEEE Congress on Evolutionary Computation (CEC), pp. 2488–2495. IEEE, New York (2013)
50. Zhang, Q., Li, H.: Moea/d: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evolut. Comput. **11**(6), 712–731 (2007)
51. Azzouz, N., Bechikh, S., Said, L.B.: Steady state ibea assisted by mlp neural networks for expensive multi-objective optimization problems. In: Proceedings of the 2014 Conference on Genetic and Evolutionary Computation, pp. 581–588. ACM, New York (2014)
52. Bader, J., Zitzler, E.: Hype: an algorithm for fast hypervolume-based many-objective optimization. Evolut. Comput. **19**(1), 45–76 (2011)

53. Bader, J., Deb, K., Zitzler, E.: Faster hypervolume-based search using monte carlo sampling. In: Multiple Criteria Decision Making for Sustainable Energy and Transportation Systems, pp. 313–326. Springer, Heidelberg (2010)
54. Diaz-Manriquez, A., Toscano-Pulido, G., Coello, C.A.C., Landa-Becerra, R.: A ranking method based on the r2 indicator for many-objective optimization. In: 2013 IEEE Congress on Evolutionary Computation (CEC), pp. 1523–1530. IEEE, New York (2013)
55. Ishibuchi, H., Murata, T.: A multi-objective genetic local search algorithm and its application to flowshop scheduling. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **28**(3), 392–403 (1998)
56. Jaszkiewicz, A.: On the performance of multiple-objective genetic local search on the 0/1 knapsack problem-a comparative experiment. IEEE Trans. Evolut. Comput. **6**(4), 402–412 (2002)
57. Miettinen, K.: Nonlinear Multiobjective Optimization, International Series in Operations Research and Management Science, vol. 12 (1999)
58. Dennis, J., Das, I.: Normal-boundary intersection: a new method for generating pareto optimal points in nonlinear multicriteria optimization problems. SIAM J. Optim. **8**(3), 631–657 (1998)
59. Messac, A., Ismail-Yahaya, A., Mattson, C.A.: The normalized normal constraint method for generating the pareto frontier. Struct. Multidiscip. Optim. **25**(2), 86–98 (2003)
60. Ishibuchi, H., Akedo, N., Nojima, Y.: Relation between neighborhood size and moea/d performance on many-objective problems. In: Evolutionary Multi-Criterion Optimization, pp. 459–474. Springer, Heidelberg (2013)
61. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. IEEE Trans. Evolut. Comput. **18**(4), 577–601 (2014)
62. Seada, H., Deb, K.: U-nsga-iii: A unified evolutionary algorithm for single, multiple, and many-objective optimization, COIN report, no. 2014022
63. Asafuddoula, M., Ray, T., Sarker, R.: A decomposition based evolutionary algorithm for many objective optimization with systematic sampling and adaptive epsilon control. In: Evolutionary Multi-Criterion Optimization, pp. 413–427. Springer, Heidelberg (2013)
64. Jain, H., Deb, K.: An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: handling constraints and extending to an adaptive approach. IEEE Trans. Evolut. Comput. **18**(4), 602–622 (2014)
65. Ray, T., Tai, K., Seow, C.: An evolutionary algorithm for multiobjective optimization. Eng. Optim. **33**(3), 399–424 (2001)
66. Hadka, D., Reed, P.M., Simpson, T.W.: Diagnostic assessment of the borg moea for many-objective product family design problems. In: 2012 IEEE congress on Evolutionary computation (CEC), pp. 1–10. IEEE, New York (2012)
67. Asafuddoula, M., Ray, T., Sarker, R.: A decomposition-based evolutionary algorithm for many objective optimization. IEEE Trans. Evolut. Comput. **19**(3), 445–460 (2015)
68. Miettinen, K., Mäkelä, M.M.: On scalarizing functions in multiobjective optimization. OR spectrum **24**(2), 193–213 (2002)
69. Li, K., Deb, K., Zhang, Q., Kwong, S.: An evolutionary many-objective optimization algorithm based on dominance and decomposition. IEEE Trans. Evolut. Comput. **19**(5), 694–716 (2015)
70. Zhang, Q., Liu, W., Li, H.: The performance of a new version of moea/d on cec09 unconstrained mop test instances. IEEE Congr. Evolut. Comput. **1**, 203–208 (2009)
71. Tan, Y.-Y., Jiao, Y.-C., Li, H., Wang, X.-K.: Moea/d+ uniform design: a new version of moea/d for optimization problems with many objectives. Comput. Oper. Res. **40**(6), 1648–1660 (2013)
72. Yuan, Y., Xu, H., Wang, B.: An improved nsga-iii procedure for evolutionary many-objective optimization. In: Proceedings of the 2014 Conference on Genetic and Evolutionary Computation, pp. 661–668. ACM, New York (2014)
73. Elarbi, M., Bechikh, S., Said, L.B.: Solving many-objective problems using targeted search directions. In: Proceedings of the 31th Annual ACM Symposium on Applied Computing, pp.89–96. ACM, Pisa, Italy (2016)

74. Derbel, B., Brockhoff, D., Liefooghe, A., Verel, S..: On the impact of multiobjective scalarizing functions. In: Parallel Problem Solving from Nature–PPSN XIII, pp. 548–558. Springer, Heidelberg (2014)
75. Hughes, E.J.: Msops-ii: A general-purpose many-objective optimiser. In: IEEE Congress on Evolutionary Computation, 2007. CEC 2007, pp. 3944–3951. IEEE, New York (2007)
76. Derrac, J., García, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evolut. Comput. **1**(1), 3–18 (2011)
77. Brockhoff, D., Zitzler, E.: Offline and online objective reduction in evolutionary multiobjective optimization based on objective conflicts. TIK Report **269** (2007)
78. Miettinen, K.: Graphical illustration of pareto optimal solutions. In: Multi-objective Programming and Goal Programming, pp. 197–202. Springer, Heidelberg (2003)
79. Thiele, L., Chakraborty, S., Gries, M., Kunzli, S.: Design space exploration of network processor. Netw. Process. Des. Issues Pract. **1**, 55–89 (2002)
80. Li, H., Zhang, Q.: Multiobjective optimization problems with complicated pareto sets, moea/d and nsga-ii. IEEE Trans. Evolut. Comput. **13**(2), 284–302 (2009)
81. Villalobos, C.A.R., Coello, C.A.C.: A new multi-objective evolutionary algorithm based on a performance assessment indicator. In: Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation, pp. 505–512. ACM, New York (2012)
82. Van Veldhuizen, D.A., Lamont, G.B.: On measuring multiobjective evolutionary algorithm performance. In: Proceedings of the 2000 Congress on Evolutionary Computation, 2000, vol. 1, pp. 204–211. IEEE, New York (2000)
83. Deb, K.: Multi-objective Optimization Using Evolutionary Algorithms, vol. 16. Wiley, New Jersey (2001)