A Survey of Normalization Methods in Multiobjective Evolutionary Algorithms

Linjun He, *Student Member, IEEE*, Hisao Ishibuchi, *Fellow, IEEE*, Anupam Trivedi, *Member, IEEE*, Handing Wang, *Member, IEEE*, Yang Nan, and Dipti Srinivasan, *Fellow, IEEE*

***Abstract*—A real-world multiobjective optimization problem (MOP) usually has differently-scaled objectives. Objective space normalization has been widely used in multiobjective optimization evolutionary algorithms (MOEAs). Without objective space normalization, most of the MOEAs may fail to obtain uniformly-distributed and well-converged solutions on MOPs with differently-scaled objectives. Objective space normalization requires information on the Pareto front range, which can be acquired from the ideal and nadir points. Since the ideal and nadir points of a real-world MOP are usually not known a priori, many recently proposed MOEAs tend to estimate and update the two points adaptively during the evolutionary process. Different methods to estimate ideal and nadir points have been proposed in the literature. Due to inaccurate estimation of the two points (i.e., inaccurate estimation of the Pareto front range), objective space normalization may deteriorate the performance of an MOEA. Different methods have also been proposed to alleviate the negative effects of inaccurate estimation. This paper presents a comprehensive survey of objective space normalization methods, including ideal point estimation methods, nadir point estimation methods, and different methods based on the utilization of the estimated Pareto front range.**

***Index Terms*—Evolutionary multiobjective optimization (EMO), objective space normalization, nadir point, ideal point, dominance resistant solution (DRS).**

I. Introduction

In the real world, it is not uncommon to face problems with multiple objectives, which are called multiobjective optimization problems (MOPs). The objectives of a real-world MOP are often of very different scales[1], such as the portfolio optimization problem [4] and the car side impact problem [5]. In a recently presented real-world MOP test suite [6], the approximated Pareto fronts of the 16 real-world MOPs are shown to have differently-scaled objectives. Since differently-scaled objectives are common in the real world, they are widely adopted in test problem design [7], [8]. Many synthetic test problems have this feature, such as WFG [7], MaF4 and MaF5 [9], and MaOP1-10 [10]. Although the popularly used DTLZ test suite [11] does not have differently-scaled objectives, DTLZ problems are often rescaled to emphasize this feature (e.g., SDTLZ and DSDTLZ [12]).

The complexity and variety of MOPs result in the emergence of numerous multiobjective evolutionary algorithms (MOEAs). An important issue in multiobjective optimization is to give equal emphasis to each objective. The importance of normalization has been emphasized in many early works [12]. When each objective has a totally different range of objective values, some MOEAs may fail to give equal emphasis to each objective and may not be able to obtain the desired performance since some objectives are predominant over others [13]. To deal with MOPs whose objectives are badly scaled, objective space normalization is commonly used in the field of multiobjective optimization. Fig. 1 shows a standard framework of MOEAs, in which objective space normalization is applied before environmental selection. A normalization method of an MOEA is an independent algorithmic component [14], which consists of three parts: ideal point estimation, nadir point estimation, and the use of the estimated Pareto front range for normalization.

The Pareto front is bounded by the ideal point $\mathbf{z}^*$ (i.e., lower bound) and the nadir point $\mathbf{z}^{\text{nad}}$ (i.e., upper bound)[2], as illustrated in Fig. 2 (a). These two points can offer the information on the ranges of the objective function values over the Pareto optimal solutions. Along with the ideal point, the nadir point can be used to normalize the objective space (see Fig. 2 (b)), which helps MOEAs to be applied more reliably to problems involving noncommensurable objective functions. It should be noted that the ideal and nadir points are not just for objective normalization. Their estimation has also been studied independent of normalization and can be used for other purposes. For example, some interactive multiobjective approaches (e.g., NAUTILUS [15]) need an estimated nadir point as an input. In this paper, we focus on the use of them for normalization. One can refer to [16]–[18] for their use for other purposes.

A straightforward way to normalize the objective space is

This work was supported by National Natural Science Foundation of China (Grant No. 61876075), Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531), and the National Research Foundation Singapore under its AI Singapore Programme (Award Number: [AISG-RP-2018-004]). (*Corresponding author: Hisao Ishibuchi.*)

L. He, Y. Nan, and H. Ishibuchi are with the Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China. (e-mail: this.helj@gmail.com, nany@mail.sustech.edu.cn, hisao@sustech.edu.cn).

H. Wang is with the School of Artificial Intelligence, Xidian University, China. (e-mail: hdwang@xidian.edu.cn).

A. Trivedi and D. Srinivasan are with the Department of Electrical and Computer Engineering, National University of Singapore, 117575, Singapore. (e-mail: eleatr@nus.edu.sg, dipti@nus.edu.sg) .

L. He is also with the Department of Electrical and Computer Engineering, National University of Singapore, 117575, Singapore.

[1]It is worth noting that there exist real-world MOPs whose ideal and nadir points are known, such as the biobjective feature selection problem [1], [2]. For some MOPs, normalization is not needed, such as the knapsack problem [3].

[2]Throughout this paper, the minimization of all objective is assumed.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation

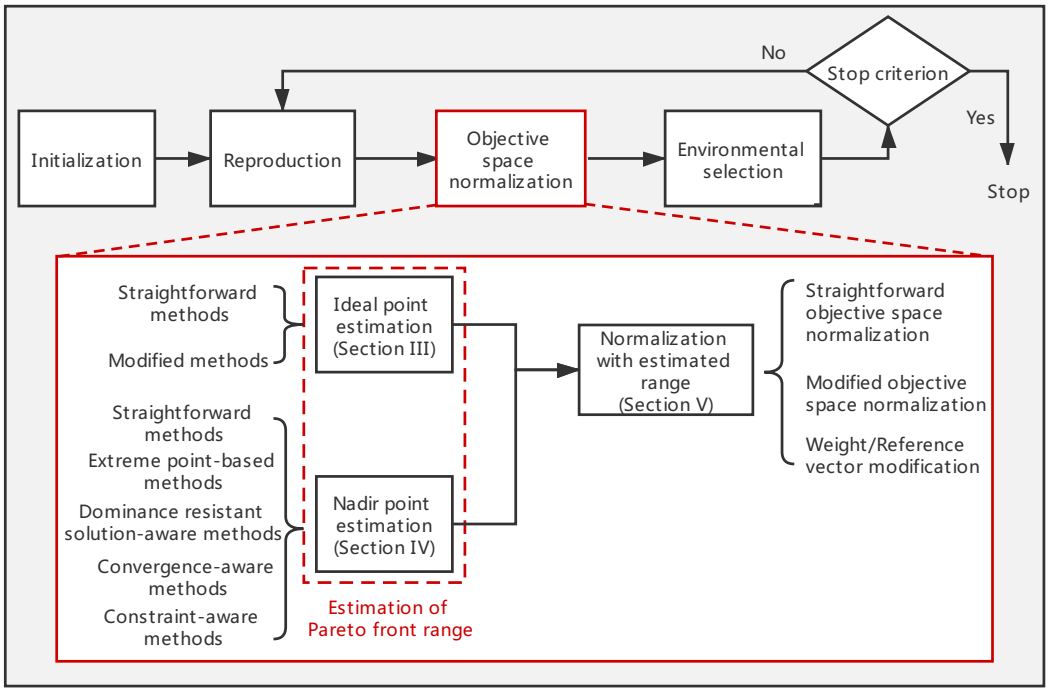


Fig. 1. Standard framework of MOEAs. Objective space normalization is an independent algorithmic block that is applied before environmental selection. A normalization method consists of ideal point estimation, nadir point estimation, and use of the estimated range.

searching for the ideal and nadir points before the evolutionary process. The ideal point can be found by minimizing each objective function individually over the decision space. Unfortunately, the nadir point is much more difficult to be estimated because it requires the knowledge of the Pareto front. There is no constructive way to obtain the exact nadir point for a nonlinear problem without finding the entire Pareto front [16], [17], [19]. Many methods have been proposed to estimate the nadir point in an offline manner (i.e., prior to the evolutionary process). One category (called "surface-to-nadir") of these offline methods uses an MOEA that tries to search for the entire Pareto front and then calculate the nadir point from the approximated Pareto front. It was argued that the surface-to-nadir approach is not recommended since searching for the entire Pareto front can be difficult and computationally expensive, especially when the number of objectives is large. Moreover, an MOEA itself needs the nadir point to facilitate the evolutionary process. One can refer to [16]–[18] for more details on offline nadir point estimation methods.

In the evolutionary multiobjective optimization (EMO) field, the ideal and nadir points are often adaptively estimated during the execution of an MOEA in an online manner, rather than before its execution in an offline manner [20], [21]. At each generation, objective space normalization is performed with the estimated ideal and nadir points, as illustrated in Fig. 1. For MOEAs with weight vectors or reference vectors, instead of normalizing the solutions in the objective space, the weight/reference vector scaling can be an alternative way of normalization, as illustrated in Fig. 2. Weight vector modification has almost the same effect as the objective space normalization in obtaining a uniformly distributed set

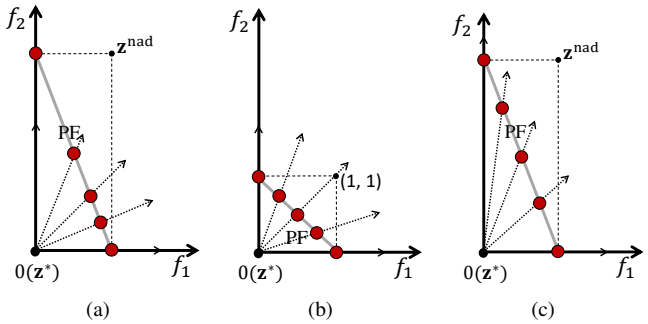


Fig. 2. Illustration of Pareto optimal solutions specified by different weight vectors on a Pareto front (PF) with different scales: (a) Original space and original weight vectors, (b) Normalized space and original weight vectors, and (c) Original space and modified weight vectors. The same solution sets are obtained in (b) and (c).

of solutions as explained in Fig. 2 (b) and (c).

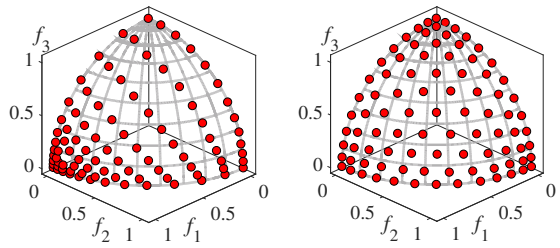
Objective space normalization can be necessary for an MOEA with either the decomposition-based, Pareto dominance-based, or indicator-based framework. For decomposition-based MOEAs, objective space normalization is particularly necessary since both the association-based partition process and the elitism selection strategy are performed in the normalized space [22], [23]. MOEA/D with PBI (MOEA/D-PBI), as one representative decomposition-based MOEA, shows good performance in dealing with problems having similar objective ranges but fails on problems whose objectives are differently scaled. Uniformly distributed solutions are not obtained by MOEA/D-PBI on these badly-scaled problems because the original MOEA/D algorithm [20] has no normalization mechanism [10], [24].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation

Fig. 3 shows the results in the normalized objective space obtained by MOEA/D-PBI with and without normalization on the three-objective WFG4 problem. MOEA/D-PBI without normalization obtains many solutions around the bottom-left corner of the Pareto front, while MOEA/D-PBI with the normalization method in [12] obtains uniformly-distributed solutions over the entire Pareto front. In the literature, we can also observe that normalization methods are almost always used in other decomposition-based MOEAs (e.g., NSGA-III [21], EFR-RR [25], θ-DEA [26], RPD-NSGA-II [27], MOEA/D-LWS [28], SPEA/R [29], ASEA [30], etc.) when handling optimization problems with differently-scaled objectives. To further show the importance of normalization, we have examined the effect of normalization on the performance of MOEAs from the three classes (i.e., decomposition-based, Pareto dominance-based, and indicator-based MOEAs) on a set of real-world MOPs recently presented in [6]. Due to the page limit, all the results and analyses are included in Section I of the supplementary document. The experimental results clearly show that normalization can be necessary for an MOEA with either the decomposition-based, Pareto dominance-based, or indicator-based framework in order to deal with real-world MOPs.

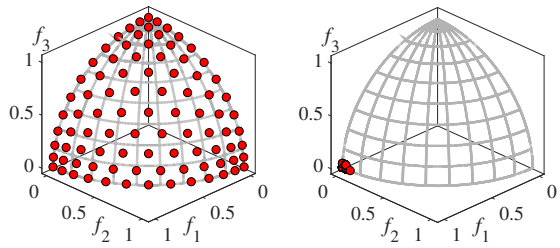
The normalization methods in MOEAs can also have negative effects on the performance of these MOEAs when ideal and nadir points are not correctly estimated. It was reported that some MOEAs (e.g., MOEA/D [20], EFR-RR [25], GrEA [31], KnEA [32], and 1by1EA [33]) with normalization methods can perform well on the badly-scaled problems but might fail on problems with similar objective ranges [33]. Fig. 4 shows the results obtained by MOEA/D-PBI with and without normalization on the three-objective DTLZ3 problem. The original MOEA/D-PBI without normalization obtains well-distributed solutions, while the normalization method in [12] deteriorates the performance of the original MOEA/D-PBI on the three-objective DTLZ3 problem. This is because the estimated nadir point is likely to be inaccurate at the early stage of the evolutionary process [12], especially when the MOP has a vast number of local optima [33]. With the inaccurately estimated nadir point, the original objective space will be transformed into wrong scales, which misleads the search direction and then makes the nadir point estimation even worse [12], [33], [34]. To avoid such negative effects, different methods have been proposed in the literature to improve the quality of the estimated ideal and nadir points (see Sections III and IV). On the other hand, modified normalization equations have also been proposed to reduce the negative effect of inaccurate estimation of the two points (see Section V).

Considering the positive and negative impacts that a normalization method can have on an MOEA's performance, its description and implementation should be carefully treated. The normalization method should be described precisely to make the experimental results reproducible [35]. It was reported in [36] that there exists inconsistency between the description of a normalization method in a paper and the actual implementation in the source code. This inconsistency may lead to an unfair comparison. As reported in [37], a small



(a) Without normalization. (b) With normalization.

Fig. 3. Solutions obtained by MOEA/D-PBI with the median hypervolume value over 21 runs on the three-objective WFG4. Experimental settings are the same as [12].



(a) Without normalization. (b) With normalization.

Fig. 4. Solutions obtained by MOEA/D-PBI with the median hypervolume value over 21 runs on the three-objective DTLZ3. Experimental settings are the same as [12].

implementation difference of the normalization method in NSGA-III [21] between Yuan [26] and jMetal (which follows the early access version of [21]) makes a vast performance difference. Similarly, in [38], a normalization method is used in the actual implementation of MOEA/DD [38] but not described in its paper. MOEA/DD is often outperformed by other algorithms on badly-scaled problems since the implementation of MOEA/DD without normalization is often used [39], [40]. In principle, with a proper normalization method (i.e., a proper ideal point estimation method, a proper nadir point estimation method, and a proper way of using the estimated Pareto front range), an MOEA is able to perform well on MOPs with differently-scaled objectives [21].

The rest of the paper is organized as follows. Starting with the necessary preliminary knowledge in Section II, the paper summarizes and categories a variety of methods for normalization, focusing on the advantages and disadvantages of these methods. Studies on ideal point estimation, including straightforward and modified methods, are surveyed in Section III. Studies on nadir point estimation, including straightforward, extreme point-based, dominance resistant solution-aware, convergence-aware, and constraint-aware methods, are surveyed in Section IV. In Section V, different methods to reduce the negative effect of inaccurate estimation of the two points when using the estimated Pareto front range are surveyed, including straightforward objective space normalization, modified objective space normalization, and weight vector modification. In Section VI, we further discuss the effect of normalization. We conclude the survey and provide several possible future directions in Section VII.

II. Preliminaries

A. Multiobjective Optimization Problems

A multiobjective optimization problem (MOP) can be mathematically formulated as follows:

$$\begin{array}{ll} \text{minimize} & \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}))^T, \\ \text{subject to} & \mathbf{x} \in \Omega \subseteq \mathbb{R}^n, \end{array} \tag{1}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ is a solution in the feasible region Ω. Here, $\mathbf{f} : \Omega \to \mathbb{R}^m$ is an objective vector involving m conflicting objectives f_i to be minimized ($i = 1, 2, ..., m$), and $\mathbb{R}^m$ is the objective space. Due to the conflicting nature of the objectives, there is no single optimal solution that can optimize all the objectives at the same time. Instead, a set of optimal trade-off solutions is usually obtained where no single objective can be further improved without deteriorating at least one of the other objectives. This trade-off relationship is defined by the following definitions:

Definition 1. *Given two solutions* $\mathbf{x}, \ \mathbf{y} \in \Omega$*, we say that* $\mathbf{x}$ *(Pareto) dominates* $\mathbf{y}$*, denoted by* $\mathbf{x} \prec \mathbf{y}$*, if and only if* $f_i(\mathbf{x}) \leq f_i(\mathbf{y})$ *for all* $i = 1, 2, \ldots, m$*, and* $\mathbf{f}(\mathbf{x}) \neq \mathbf{f}(\mathbf{y})$[3].

Definition 2. *Given a set* $P \subseteq \Omega$*, we say that a solution* $\mathbf{x}^*$ *is a nondominated solution in* P *if no other solution in* P *dominates it. Here,* $\mathbf{x}^*$ *is a Pareto optimal solution if* $P = \Omega$*, and* $\mathbf{f}(\mathbf{x}^*)$ *is then called a Pareto optimal objective vector.*

Definition 3. *The Pareto set, denoted by* PS*, is a set of all Pareto optimal solutions:* $PS = \{\mathbf{x} \in \Omega \mid \mathbf{x}$ *is Pareto optimal*$\}$*, and the Pareto front, denoted by* PF*, is a set of all Pareto optimal objective vectors:* $PF = \{\mathbf{f}(\mathbf{x}) \in \mathbb{R}^m \mid \mathbf{x} \in PS\}$.

B. Ideal and Nadir Points

The ideal point $\mathbf{z}^* = (z_1^*, z_2^*, \ldots, z_m^*)^T$ is an objective vector consists of the minimum value of each objective over the decision space Ω, where $z_i^* = \min_{\mathbf{x} \in \Omega} f_i(\mathbf{x}), \ i \in \{1, 2, \ldots, m\}$. Hence, the ideal point dominates all the Pareto optimal solutions $\mathbf{x}^*$ (and all solutions $\mathbf{x}$).

The nadir point $\mathbf{z}^{\text{nad}} = \left(z_1^{\text{nad}}, z_2^{\text{nad}}, \ldots, z_m^{\text{nad}}\right)^T$ is an objective vector consists of the maximal objective values over the Pareto set, where $z_i^{\text{nad}} = \max_{\mathbf{x} \in PS} f_i(\mathbf{x}), \ i \in \{1, 2, \ldots, m\}$.

The ideal and nadir points are usually unknown in a real-world MOP and they need to be estimated due to the necessity of normalization. Different methods have been proposed to estimate them adaptively during the evolutionary process. These methods will be surveyed in Section III for the ideal point and Section IV for the nadir point. In this paper, the estimated ideal and nadir points are denoted by $\mathbf{z}^{\text{lb}} = \left(z_1^{\text{lb}}, z_2^{\text{lb}}, \ldots, z_m^{\text{lb}}\right)^T$ and $\mathbf{z}^{\text{ub}} = \left(z_1^{\text{ub}}, z_2^{\text{ub}}, \ldots, z_m^{\text{ub}}\right)^T$, respectively.

C. Objective Space Normalization

Each individual in the current population can be translated according to the following transformation:

$$f_i'(\mathbf{x}) = f_i(\mathbf{x}) - z_i^*, \tag{2}$$

where $i = 1, 2, \ldots, m$, and $f_i'(\mathbf{x})$ is the translated objective. The true ideal point $\mathbf{z}^*$ becomes the origin of the coordinate system in the translated objective space.

After the objective value translation, each translated objective vector $f_i'(\mathbf{x})$ can be scaled by the following transformation:

$$\widetilde{f}_i(\mathbf{x}) = \frac{f_i'(\mathbf{x})}{z_i^{\text{nad}} - z_i^*} = \frac{f_i(\mathbf{x}) - z_i^*}{z_i^{\text{nad}} - z_i^*}, \tag{3}$$

where $i = 1, 2, \ldots, m$, and $\widetilde{f}_i(\mathbf{x})$ denotes the i-th normalized objective function. The denominator is based on the range of the original Pareto front. After the objective space scaling, the Pareto front has the same range $[0, 1]$ on each objective. That is, the Pareto front is included in the unit hypercube $[0, 1]^m$.

In practice, the true ideal and nadir points are usually unknown for a real-world MOP. They are usually replaced by the estimated ideal point $\mathbf{z}^{\text{lb}}$ and the estimated nadir point $\mathbf{z}^{\text{ub}}$ in MOEAs. Different variants of Eq. (3) have also been proposed to avoid the negative effects of using inaccurately estimated ideal and nadir points. These equations will be surveyed in Section V.

D. Dominance Resistant Solutions

Dominance resistance phenomena [41], [42] caused by the increasing number of objectives is known as the main obstacle of Pareto dominance-based MOEAs in handling MOPs with more than three objectives. As the number of objectives increases, the proportion of nondominated solutions in a population grows enormously [42], [43], i.e., most solutions become incomparable in terms of the Pareto dominance relation. Although dominance resistant solutions (DRSs)[4] [41] are far from the Pareto front, they are regarded as nondominated solutions and cannot be removed by the Pareto dominance relation. The performance of Pareto dominance-based algorithms can be severely degraded by DRSs even in three-objective problems [44].

Many test problems are endowed with such DRSs, such as block-separable problems [41] and mDTLZ [45]. DRSs are also observed in the frequently-used DTLZ problems [11], [46], [47]. Wang et al. [45] analyzed the existence conditions of hardly dominated boundaries. The hardly dominated boundary is a common problem feature existing in many problems, especially when the number of objectives is large [45].

Most of the existing studies only focus on the detrimental effect of DRSs in the offspring generation, i.e., an offspring generated from DRSs by crossover and mutation is likely to be far away from the Pareto front. Although these solutions may present good diversity over the objective space, they can bias the search toward the solutions with the poor proximity to the Pareto front. It has been reported in many studies [42] that these DRSs lead to a decline in selection pressure, which slows down the search process to a certain extent. Consequently, the obtained solutions may have good diversity over the objective space but fail to converge toward the Pareto front.

Several attempts are made to relax the strict definition of Pareto dominance (e.g., α-dominance [41], ϵ-dominance [46],

[3]That is, $f_i(\mathbf{x}) < f_i(\mathbf{y})$ for at least one index $i \in \{1, 2, \ldots, m\}$.

[4]DRSs are solutions with extremely good values (i.e., nearly the best values) in some objectives, but with very poor values in other objectives.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation

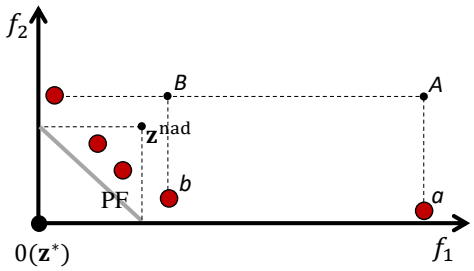


Fig. 5. Illustration of relationship between DRSs and nadir point estimation.

[48]). It is also reported in [36] that the use of the PBI function might help reduce the effect of DRSs. In [47], Kondoh et al. reported that the number of DRSs might be reduced by discretizing the decision space with a lower precision.

However, the effect of DRSs in nadir point estimation has rarely been discussed in the literature [45], [49]. One popular way of estimating the nadir point is to take the maximum values of nondominated solutions in the current population. As illustrated in Fig. 5, the estimated nadir point A is far away from the true nadir point $\mathbf{z}^{\text{nad}}$ due to the existence of a DRS (i.e., point a). After removing point a, the estimated nadir point B is much closer to $\mathbf{z}^{\text{nad}}$. The presence of DRSs will heavily influence the resulting nadir point, create an undesirable bias in scaled objective values, and thus imbalance the diversity. Detection and elimination of DRSs are far from trivial and further research is required to address this issue.

III. Ideal Point Estimation

The normalization methods in MOEAs can have negative effects on the performance of these MOEAs when ideal and nadir points are not correctly estimated. In the EMO field, the ideal point is often estimated adaptively during the evolutionary process [20], [21]. Compared with nadir point estimation, the ideal point estimation is much easier [16], [17]. The estimated ideal point approaches the true ideal point and settles down very quickly [50]. There are fewer studies on ideal point estimation than nadir point estimation. In this section, we review ideal point estimation methods.

A. Straightforward Ideal Point Estimation Methods

Given an initial population P_1, the ideal point $\mathbf{z}^{\text{lb}}$ is estimated by $z_i^{\text{lb}} = \min_{\mathbf{x}\in P_1} f_i(\mathbf{x}),\ i \in \{1, 2, \ldots, m\}$. During the evolutionary process, the ideal point will be updated at each generation. Considering the t-th generation, a set of offspring Q_t is generated from the population P_t. Note that Q_t can contain only one offspring if the algorithm has a generation update mechanism of $(\mu + 1)$ type. The ideal point can be updated in two different manners.

- The ideal point is updated by taking the minimum value of the ideal point from the previous generation and the newly generated offspring Q_t as shown in Eq. (4). This method for updating the ideal point has been used in several state-of-the-art MOEAs, such as MOEA/D [20] and NSGA-III [21].

$$z_i^{\text{lb}} = \min\left\{z_i^{\text{lb}}, \min_{\mathbf{x}'\in Q} f_i(\mathbf{x}')\right\},\ i \in \{1, 2, \ldots, m\}. \quad (4)$$

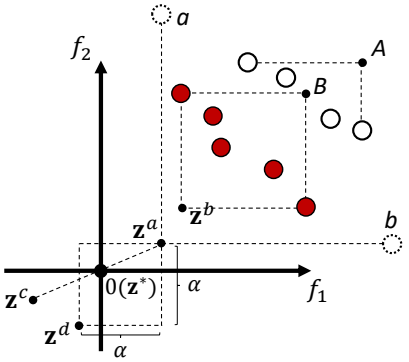


Fig. 6. Illustration of ideal point estimation methods. Red points are nondominated solutions, and black circles are dominated solutions. Dashed circles a and b are solutions that are generated in previous generations but not included in the current population. Note that the true ideal point $\mathbf{z}^*$ can be different from the origin of the objective space. We assume $\mathbf{z}^*$ is the same as the origin here for illustration.

- The ideal point is updated by taking the minimum value of the merged population (i.e., the union of the current population P_t and the newly generated offspring Q_t). This method for updating the ideal point has also been used in several state-of-the-art MOEAs, such as CDG-MOEA [51], NSGA-II/SDR [52], and DDEA [53].

$$z_i^{\text{lb}} = \min_{\mathbf{x}\in P_t \bigcup Q_t} f_i(\mathbf{x}),\ i \in \{1, 2, \ldots, m\}. \quad (5)$$

The estimation of the ideal point requires $O(mN)$ computations. The estimated ideal points by Eq. (4) and Eq. (5) can be the same in some algorithms. For example, in NSGA-II [54], extreme solutions are in the first nondominated front and have the largest crowding distance. Thus, they are never removed. However, in other algorithms (e.g., MOEA/D [20] and NSGA-III [21]), it is not guaranteed that solutions contributing to the ideal point always survive during the evolutionary process [36], [55]. For this reason, the ideal point updated by Eq. (4) is a more accurate estimation [55]. As illustrated in Fig. 6, the ideal point $\mathbf{z}^a$ estimated by Eq. (4) is often closer to the true ideal point than the ideal point $\mathbf{z}^b$ estimated by Eq. (5).

B. Other Ideal Point Estimation Methods

Estimated ideal points by the methods in Eq. (4) and Eq. (5) are usually far away from the true ideal point at the beginning of the search. Several other ideal point estimation methods have been proposed to reduce the negative effects caused by such inaccurate estimation.

Fan et al. [56] proposed an improved ideal point setting method for MOEA/D. A temporary ideal point $\mathbf{z}^a$ is first obtained by Eq. (5). Then, the ideal point $\mathbf{z}^{\text{lb}}$ (i.e., $\mathbf{z}^c$ in Fig. 6) is determined by taking the symmetric point of $\mathbf{z}^a$ (i.e., $\mathbf{z}^c = -\mathbf{z}^a$). In the early stage of the evolutionary process, the estimated ideal point $\mathbf{z}^c$ is much better than the temporary ideal point $\mathbf{z}^a$. As $\mathbf{z}^a$ approaches the origin of the coordinate, the estimated ideal point $\mathbf{z}^c$ also approaches the origin. Note that this method implicitly assumes that the true ideal point $\mathbf{z}^*$ is the same as the origin of the objective space. When the true ideal point $\mathbf{z}^*$ is different from the origin, the estimated

ideal point $\mathbf{z}^c$ cannot converge to $\mathbf{z}^*$ even when the true Pareto front is found.

Ishibuchi et al. [57] proposed a new idea of ideal point estimation for MOEA/D. A temporary ideal point $\mathbf{z}^a$ is first obtained by Eq. (5). Then, the ideal point $\mathbf{z}^{\text{lb}}$ (i.e., $\mathbf{z}^d$ in Fig. 6) is determined by the temporary ideal point $\mathbf{z}^a$ subtracted by a parameter α (i.e., $\mathbf{z}^d = \mathbf{z}^a - \alpha$). In [57], the value of α linearly decreases with the evolutionary process. That is, the estimated ideal point $\mathbf{z}^d$ is much better than the temporary ideal point $\mathbf{z}^a$ at the early evolutionary stage and it gradually approaches to $\mathbf{z}^a$ during the evolutionary process. Experimental results in [57] show that MOEA/D-PBI with the new estimated ideal point $\mathbf{z}^d$ performs much better than the original MOEA/D-PBI on multiobjective knapsack problems.

Wang et al. [58] systematically investigated the effect of the value α on the behavior of MOEA/D. In [58], three ideal point estimation strategies are examined: pessimistic specification (i.e., a small α), optimistic specification (i.e., a large α), and dynamic specification (i.e., the value of α varies from a large value to a small value as in [57]). Experimental results in [58] show that the pessimistic specification strategy emphasizes the exploitation of existing regions (i.e., convergence), while the optimistic specification strategy emphasizes the exploration of new regions (i.e., diversity). MOEA/D with either strategy only performs well on certain problems. The dynamic specification strategy can balance exploitation and exploration and helps MOEA/D perform well on various types of problems.

C. Discussions and Summary

In summary, the survey of studies on ideal point estimation reveals the following.

- In general, the ideal point estimated by Eq. (4) is closer to the true ideal point than that estimated by Eq. (5).
- It is worth investigating whether the estimated ideal point is very close to the true ideal point in some objectives but not close in other objectives (i.e., whether there exists a large difference in optimizing each objective). The reason is that such an ideal point leads to a heavily biased estimation of the range of the Pareto front even when the nadir point is accurately estimated.
- The ideal point estimated by either Eq. (4) or Eq. (5) is usually far away from the true ideal point at early generations. The use of a better point as in [56]–[58] than the estimated point may help to improve the estimation accuracy and the performance of MOEAs.

IV. Nadir Point Estimation

Nadir point estimation is one of the three main parts of a normalization method. Inaccurate estimation of the nadir point can cause performance deterioration of MOEAs. Since the nadir point estimation needs the information about the entire Pareto front (i.e., the entire Pareto optimal solution set of the given multi-objective optimization problem), the nadir point estimation task is usually much more difficult than estimating the ideal point [16], [59].

Given a solution set S, the nadir point can be estimated as follows,

$$z_i^{\text{ub}} = \max_{\mathbf{x} \in S} f_i(\mathbf{x}),\ i \in \{1, 2, \ldots, m\}. \tag{6}$$

When the solution set S is the same as the Pareto optimal solution set, the true nadir point is obtained by Eq. (6). Since the Pareto optimal solution set is unknown, different methods have been used to specify the solution set S in the literature [12], [36], [60]. Straightforward estimation methods use Eq. (6) to estimate the nadir point.

In addition, several efforts have been made to improve the quality of the nadir point estimation. As illustrated in Fig. 7, the solution set S in Eq. (6) can be preprocessed, and the processed solution set S' (instead of the original solution set S) is used for nadir point estimation. After nadir point estimation, the estimated nadir point can also be updated when additional information is available.

According to the additional information used to preprocess the solution set or update the estimated nadir point, we categorize these sophisticated nadir point estimation methods into four classes: extreme point-based methods, DRS-aware methods, convergence-aware methods, and constraint-aware methods. In this section, both straightforward and sophisticated nadir point estimation methods are reviewed in detail.

A. Straightforward Nadir Point Estimation Methods

The straightforward nadir point estimation methods directly estimate the nadir point from the solution set S using Eq. (6) [29], [55]. These methods have been used in several state-of-the-art MOEAs, such as Two_Arch2 [61] and VaEA [62]. Due to its simplicity, the normalization using these nadir point estimation methods is also called simple normalization or naïve normalization [25], [63]. In this subsection, the studies on the specification of the solution set S in Eq. (6) are surveyed.

The use of the merged population (i.e., the union of the current and offspring populations) is a frequently-used method to specify the solution set S in Eq. (6) [27], [29], [58], [62], [64]–[75], which emphasizes the population's distribution [22]. The nadir point estimation using this set as the solution set requires $O(mN)$ computations, where m and N are the number of objectives and the population size, respectively.

Only nondominated solutions in the merged population [29], [34], [51], [76], [77] or the best fronts (based on the nondominated sorting) containing at least N solutions from the merged population [21], [22], [25], [26], [78], [79] are also popularly used as the solution set S to give priority to individuals with good convergence [22]. Note that to obtain these solution sets, nondominated sorting is performed on the merged population, which requires $O(mN^2)$ computations.

To reduce the computational cost when the set of nondominated solutions in the merged population is used as the solution set, Cai et al. proposed [51] to preserve only a subset S' of solutions in the merged population that are close to the corner solutions. Then, the nadir point is estimated from the nondominated solutions in the subset S'. The computational complexity

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation

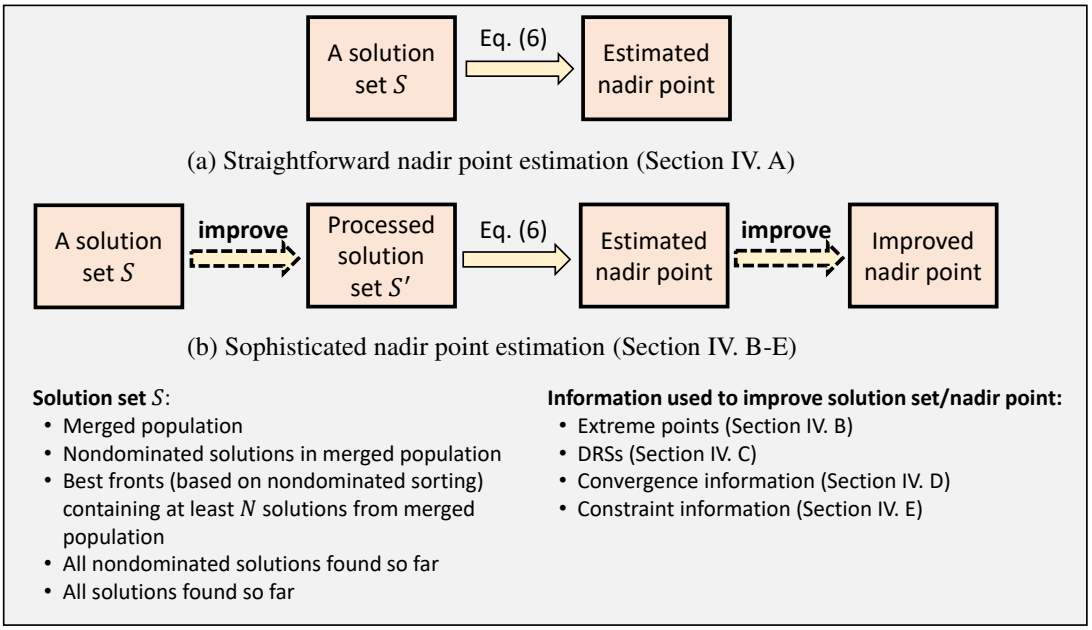


Fig. 7. Flowcharts of nadir point estimation. The straightforward nadir point estimation methods directly estimate the nadir point from a solution set S. The sophisticated nadir point estimation methods either attempt to improve the quality of the solution set, or update the estimated nadir point, making use of extreme points, DRSs, convergence information, or constraint information.

for estimating the nadir point is reduced to $O\left(mL^2\right)$, where L is the size of S'.

There exist some other specification methods of the solution set S in Eq. (6). For example, all nondominated solutions found so far (which are stored in an archive) [28], [80] and all solutions found so far [38] are used as the solution set S.

Considering different straightforward ways of estimating the ideal and nadir points, there can be different combinations of ideal and nadir point estimation, as illustrated in Fig. 6. In [36], He et al. examined the effects of normalization using four different combinations of ideal and nadir points on the performance of MOEA/D. Two frequently-used methods (i.e., the merged population and the nondominated solutions in the merged population) are considered for nadir point estimation. Experimental results show that different combinations have different effects on the performance of MOEA/D. Each combination shows significant performance deterioration for some problems in DTLZ and WFG. This study suggests that a robust normalization method is needed.

B. Extreme Point-based Nadir Point Estimation Methods

Extreme point-based nadir point estimation methods include extreme point identification and are often followed by hyperplane construction.

To be more specific, as shown in Fig. 8, given a solution set S, m extreme points $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_m\}$ are identified first. Then, the nadir point $\mathbf{z}^{\text{ub}}$ can be directly estimated from the m extreme points [81]–[83]. Alternatively, a hyperplane can be constructed using the m extreme points and $\mathbf{z}^{\text{ub}}$ can be derived from the intercept of the hyperplane with each axis of the translated objective space [21], [26], [84]. When each of the m extreme points is the global optimal solution for each objective, the hyperplane is also called the convex hull of the individual minima (CHIM) [85], [86].

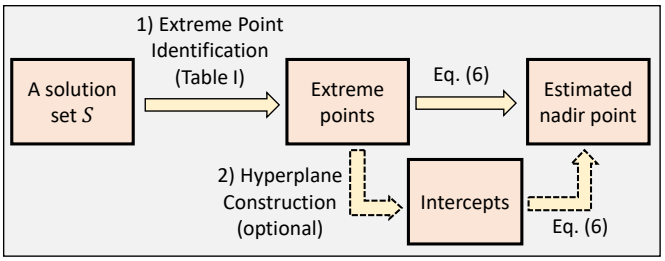


Fig. 8. Overview of extreme point-based nadir point estimation. The extreme points identified by methods in Table I can be directly used as the processed solution set S' to estimate the nadir point. Alternatively, the extreme points can be used for hyperplane construction and the nadir point is estimated from the intercepts of the hyperplane with the axes of the translated objective space.

Usually, when the extreme point-based nadir point estimation methods are used, the ideal point is estimated from the best values found so far (i.e., Eq. (4) in Section III) [24], [30], [83], [87], [88]. However, there are different ways of identifying the extreme points. In this subsection, studies related to extreme point identification and hyperplane construction are surveyed. MOEAs designed with the extreme-point-first principle are also surveyed.

1) Extreme Point Identification Methods: Various extreme point identification methods have been proposed in the literature. We summarize the extreme point identification methods in Table I. These methods can be classified into three categories based on extreme point definitions: solutions with the minimum objective values, solutions closest to the axis vectors, and corner solutions.

TABLE I

SUMMARY OF EXTREME POINT IDENTIFICATION METHODS. EXTREME POINT IDENTIFICATION FUNCTIONS FOR IDENTIFYING THE EXTREME POINT $\mathbf{e}_i$ CORRESPONDING TO THE i-TH OBJECTIVE f_i ARE SUMMARIZED. FOR A GIVEN VECTOR $\mathbf{f} \in \mathbb{R}^m$ IN THE OBJECTIVE SPACE, WE DEFINE $\mathbf{f}^{\bar{i}} := (f_1, \ldots, f_{i-1}, f_{i+1}, \ldots, f_m)^T \in \mathbb{R}^{m-1}$, WHICH HAS THE i-TH ELEMENT REMOVED (DEFINITION 1 IN THE SUPPLEMENTARY DOCUMENT).

Category	Identification function		Description	Reference
Solutions with the minimum objective values	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} f_i(\mathbf{x})$	(7)	The solution that has the minimum objective function value for the i-th objective f_i is identified as the extreme point for f_i.	[14]
	–		One extreme point is first identified by Eq. (7). Then, the remaining $m-1$ extreme points are identified one by one by selecting the solution that is farthest from the identified solutions.	DSS [89]
Solutions closest to the axis vectors	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} \mathrm{ASF1}(\mathbf{x}, \mathbf{s}_i)$	(8)	The solution closest to each axis vector is identified as an extreme point. The closeness is measured by the achievement scalarizing function (approximated Chebyshev distance).	NSGA-III [21]
	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} \left\| (\mathbf{f}(\mathbf{x}) - \mathbf{z}^{\mathrm{lb}})^{\bar{i}} \right\|_1$	(9)	The only difference from NSGA-III [21] is that the closeness is measured by the Manhattan distance (L^1 norm).	1by1EA [33]
	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} \left\| (\mathbf{f}(\mathbf{x}) - \mathbf{z}^{\mathrm{lb}})^{\bar{i}} \right\|_2$	(10)	The only difference from NSGA-III [21] is that the closeness is measured by the Euclidean distance (L^2 norm).	AGE-MOEA [90]
	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} \left\| (\mathbf{f}(\mathbf{x}) - \mathbf{z}^{\mathrm{lb}})^{\bar{i}} \right\|_\infty$	(11)	The only difference from NSGA-III [21] is that the closeness is measured by the exact Chebyshev distance (L^∞ norm).	PaRP/EA [91]
	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S\cup\{\mathbf{e}_1,\mathbf{e}_2,\ldots,\mathbf{e}_m\}} \mathrm{ASF1}(\mathbf{x}, \mathbf{s}_i)$	(12)	The only difference from NSGA-III [21] is that the extreme points identified at the previous generation are involved in the solution set such that the extreme points are updated instead of being straightforwardly replaced.	[55]
	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} \mathrm{ASF2}(\mathbf{x}, \mathbf{s}_i)$	(13)	The only difference from NSGA-III [21] is that the achievement scalarizing function is calculated in the normalized objective space. The objective space is normalized using the ideal and nadir points estimated at the previous generation.	θ-DEA [26]
	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} \left\| (\mathbf{f}(\mathbf{x}) - \mathbf{z}^{\mathrm{lb}})^{\bar{i}} \right\|_2 + \lambda \lvert f_i(\mathbf{x}) - z_i^{\mathrm{lb}} \rvert$, where $\lambda = 0.01$	(14)	A penalty term is introduced to prevent $f_i(\mathbf{x})$ from becoming too large.	MaOEA/IGD [92]
	$\mathbf{e}_i = \arg\min_{\mathbf{x}\in S} \arccos\left((\mathbf{f}(\mathbf{x}) - \mathbf{z}^{\mathrm{lb}}) \mathbf{s}_i^T \right)$	(15)	The solution closest to each axis vector is identified as an extreme point. The closeness is measured by the angle of each axis vector and the vector starting from the estimated ideal point to the solution.	MaOEA-IT [93]
	$\mathbf{e}_i = \arg\max_{\mathbf{x}\in S} \left\| (\mathbf{z}^{\mathrm{ub}} - \mathbf{f}(\mathbf{x}))^{\bar{i}} \right\|_2$	(16)	The solution with the largest perpendicular distance from the line specified by $z_i = z_i^{\mathrm{ub}}$ for $i = 2, 3, \ldots, m$ (i.e., d_{ub} in Fig. 10), where $\mathbf{z}^{\mathrm{ub}} = (z_1^{\mathrm{ub}}, z_2^{\mathrm{ub}}, \ldots, z_m^{\mathrm{ub}})$ is the estimated nadir point by a straightforward method, is identified as an extreme point. The closeness is measured by the Euclidean distance (L^2 norm).	1by1EA [33]
Corner solutions	–		A hybrid method of the solutions with the minimum objectives and the solutions closest to the axis vectors.	I-DBEA [87]

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation

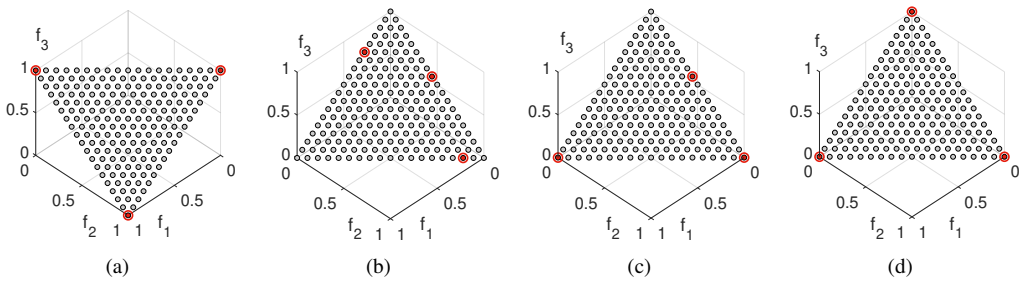


Fig. 9. Illustration of the identified extreme points by approaches in [14] and [89] on an inverted triangular Pareto front (a), by the approach in [14] on a regular triangular Pareto front (b), by the approach in [89] on a regular triangular Pareto front (c), and by the approach modified from [89] on a regular triangular Pareto front (d).

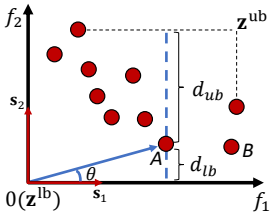


Fig. 10. Illustration of extreme point identification methods based on axis vectors.

a) Minimum objective value-based methods: Tanabe et al. [14] defined the extreme points as solutions that have the minimum objective value for each objective (i.e., Eq. (7) in Table I). Extreme points can be correctly identified when the shape of the Pareto front is inverted triangular, as illustrated in Fig. 9 (a). However, it was reported in [89] that this approach may capture none of the extreme points when the optimization problem has a regular triangular Pareto front since all solutions on a Pareto front boundary minimize the same objective. As illustrated in Fig. 9 (b), the identified three extreme points are different from the true extreme points.

Singh et al. [89] proposed to start with only a single extreme point for an arbitrary objective instead of trying to find all the m extreme points at the same time. To be more specific, one extreme point $\mathbf{e}_1$ for an arbitrary objective is found in the same manner as in [14] and added to the solution set S. Then, starting from the extreme point $\mathbf{e}_1$, the solution with the farthest distance from the solution set S is selected one by one. This approach ensures the identification of the $m-1$ extreme solutions (i.e., $\mathbf{e}_2, \mathbf{e}_3, \ldots, \mathbf{e}_m$), as illustrated in Fig. 9 (c).

b) Axis vector-based methods: The axis vector-based extreme point identification methods identify the extreme point based on axis vectors. The definition of axis vectors is shown in Definition 4. For example, for a three-objective optimization problem, the three axis vectors are $\mathbf{s}_1 = (1, 0, 0)$, $\mathbf{s}_2 = (0, 1, 0)$, and $\mathbf{s}_3 = (0, 0, 1)$.

Definition 4. *For an m-objective optimization problem, there are m axis vectors $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_m\}$. An axis vector $\mathbf{s}_i$ is a weight vector with the axis direction corresponding to the objective f_i. All elements of the axis vector $\mathbf{s}_i$ are zeros except for the i-th element with one.*

We use Fig. 10 to summarize the general idea of different axis vector-based methods. The corresponding extreme point identification functions of these methods can be found in Eqs. (8)-(16) in Table I. As illustrated in Fig. 10, all solutions are first translated to the translated objective space so that the estimated ideal point of the translated population becomes a zero vector. For the axis vector $\mathbf{s}_1$, the solution closest to $\mathbf{s}_1$ is identified as the extreme point $\mathbf{e}_1$. The closeness of a solution to $\mathbf{s}_1$ can be measured in various manners such as

- the perpendicular distance of the solution from $\mathbf{s}_1$ (i.e., d_{lb} in Fig. 10 and Eqs. (8)-(14) in Table I),
- the angle between the axis vector $\mathbf{s}_1$ and the vector starting from the estimated ideal point $\mathbf{z}^{\text{lb}}$ to the solution (i.e., θ in Fig. 10 and Eq. (15) in Table I), and
- the perpendicular distance of the solution from the line specified by $z_i = z_i^{\text{ub}}$ for $i = 2, 3, \ldots, m$ (i.e., d_{ub} in Fig. 10 and Eq. (16) in Table I) where $\mathbf{z}^{\text{ub}} = (z_1^{\text{ub}}, z_2^{\text{ub}}, \ldots, z_m^{\text{ub}})$ is the estimated nadir point by a straightforward method.

While the first and second distances are to be minimized, the third distance is maximized. Various approaches based on these three distances have been proposed in the literature, which are explained later.

In NSGA-III [21], Deb and Jain proposed to minimize the following achievement scalarizing function (ASF) for finding an extreme point for each objective,

$$\text{ASF1}(\mathbf{x}, \mathbf{s}_i) = \max_{j=1}^{m} f'_j(\mathbf{x})/s_{i,j},\ i \in \{1, 2, \ldots, m\}, \quad (17)$$

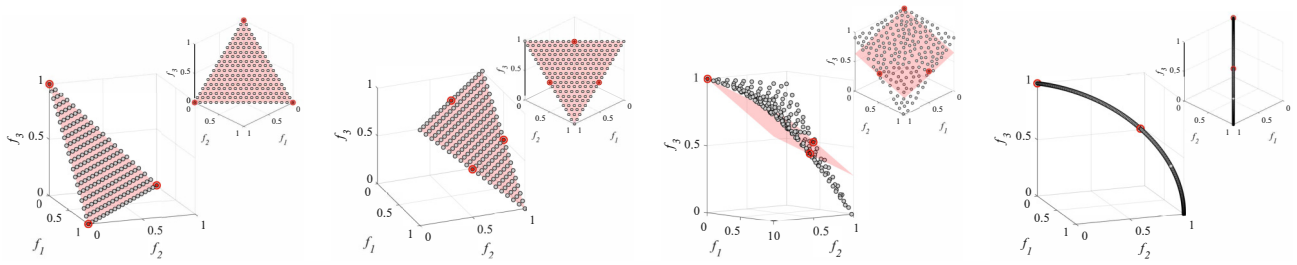
where f'_i is the i-th objective translated by Eq. (2) such that the estimated ideal point becomes the origin of the coordinate system. Notice that zeros in the axis vector $\mathbf{s}_i$ are replaced with a small number 10^{-6}. For the axis vector $\mathbf{s}_i$, the solution $\mathbf{x} \in S$ minimizing the ASF function in Eq. (17) is identified as the extreme point $\mathbf{e}_i$ for the objective f_i (i.e., Eq. (8) in Table I). This method has also been used in DBEA-Eps [94], DBEA [95], DoD [73], LEAF [78], and DECAL [24].

The geometric property of Eq. (17) is as follows. For proof of Theorem 1, refer to the supplementary document.

Theorem 1. *Let $\mathbf{s}_i = (s_{i,1}, s_{i,2}, \ldots, s_{i,i}, \ldots, s_{i,m}) = (10^{-6}, 10^{-6}, \ldots, 1, \ldots, 10^{-6})$ be the axis vector corresponding to the i-th objective f_i. The ASF function in Eq. (8) is the approximation of the infinity norm (also called Chebyshev distance) between the solution $\mathbf{x}$ and the axis vector $\mathbf{s}_i$.*

That is, the distance d_{lb} in Fig. 10 is calculated by the approximated L^∞ norm in NSGA-III [21]. The distance d_{lb}

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation



(a) (b) (c) (d)

Fig. 11. Illustration of the identified extreme points by approaches in NSGA-III [21] on (a) a regular triangular Pareto front, (b) an inverted triangular Pareto front, (c) the Pareto front of the kite test problem [8], and (d) the degenerated Pareto front of DTLZ5 [11]. The gray points denote the Pareto optimal solutions, where the identified extreme points are highlighted in red. The red plane denotes the hyperplane constructed using the identified extreme points. No unique hyperplane can be constructed in (d) due to duplicate extreme points.

in Fig. 10 has also been calculated by other distance metrics in the literature. For example, these metrics include the L^1 norm in 1by1EA [33], L^2 norm in AGE-MOEA [90], and exact L^∞ norm in PaRP/EA [91] (i.e., Eq. (9)-(11) in Table I). As pointed out by Xiang et al. [91], when the exact L^∞ norm is used, DRSs are likely to be identified as the extreme points (which will be explained in Section IV-C).

Blank et al. [55] improved the method in NSGA-III [21] by including the extreme points identified at the previous generation in the solution set S to avoid an abrupt change in the nadir point (see Eq. (12) in Table I).

Yuan et al. [26] proposed to calculate the distance d_{lb} in Fig. 10 in the normalized objective space. A slightly different ASF function was proposed in θ-DEA [26] as follows,

$$\mathrm{ASF2}\left(\mathbf{x}, \mathbf{s}_i\right) = \max_{j=1}^{m} \left\{ \left| \frac{f_j(\mathbf{x}) - z_j^{\mathrm{lb}}}{z_j^{\mathrm{ub}} - z_j^{\mathrm{lb}}} \right| / s_{i,j} \right\}. \qquad (18)$$

The extreme points are identified by minimizing the ASF2 function (see Eq. (13) in Table I). Similar to NSGA-III [21], this method is actually identifying the closest solution toward each objective axis by considering the approximated L^∞ norm. The only difference from the ASF function (17) in NSGA-III [21] is that the approximated L^∞ norm is calculated in the normalized objective space. The objective space is normalized using the estimated ideal and nadir points at the previous generation. This might help to more accurately identify the extreme points because the L^p norm is a scaling-dependent distance metric. However, if the estimated ideal and nadir points from the previous generation are not accurate, this method may deteriorate the identification process. This method has also been used by MOEA/D-DU [25], EFR-RR [25], MOEA/D-AU [96], ASEA [30], and MaOEA/SRV [22].

To identify the extreme point $\mathbf{e}_i$ for the i-th objective, Eqs. (8)-(13) calculate the perpendicular distances (i.e., d_{lb} in Fig. 10) of solutions from the axis vector $\mathbf{s}_i$ in different manners. Sun et al. [92] pointed out that the i-th objective value was neglected in such calculations when identifying the extreme point $\mathbf{e}_i$. As illustrated in Fig. 10, if point B is a DRS that is much worse than point A in terms of the first objective, it will be identified as an extreme point because it is slightly closer to $\mathbf{s}_1$ than point A. Sun et al. [92] highlighted this limitation and proposed to introduce a penalty term on the i-th objective value to prevent it from becoming too large (see Eq. (14)). This method has also been used in B-NSGA-III [97].

Sun et al. [93] proposed to minimize the angle θ between the axis vector and the vector starting from the estimated ideal point $\mathbf{z}^{\mathrm{lb}}$ (the origin of the translated objective space) to the solution in MaOEA-IT [93], as illustrated in Fig. 10. The solution with the minimum angle toward each axis vector is identified as an extreme point (i.e., Eq. (15) in Table I). This method has also been used in MaOEA/RVs [22].

Liu et al. [33] proposed to maximize the perpendicular distance of the solution from the line specified by $z_i = z_i^{\mathrm{ub}}$ for $i = 2, 3, \ldots, m$ (i.e., d_{ub} in Fig. 10) where $\mathbf{z}^{\mathrm{ub}} = (z_1^{\mathrm{ub}}, z_2^{\mathrm{ub}}, \ldots, z_m^{\mathrm{ub}})$ is the estimated nadir point by a straightforward method. That is, the distance d_{ub} as illustrated in Fig. 10. The solution with the maximum distance d_{ub} is identified as an extreme point (i.e., Eq. (16) in Table I).

These methods have a common issue. The extreme point identification methods searching for solutions closest to the axis vectors may fail to identify the corners that give the correct nadir point [8]. Only when the Pareto front is regular triangular can the extreme points be correctly identified by these methods. As shown in Fig. 11, with the method in NSGA-III [21], the three corners of the triangular Pareto front are correctly found, whereas the extreme points of irregular Pareto fronts (e.g., the inverted triangular Pareto front, the Pareto front of the kite test problem [8], and the degenerated Pareto front of DTLZ5 [11]) cannot be correctly identified.

c) Corner solution methods: The Pareto corner (or the corner solution) was first defined in [98]. In I-DBEA [87], a corner sort ranking method was proposed by hybridizing the abovementioned two approaches: (a) solutions with the minimum objective values and (b) solutions close to the axis vectors. First, $2m$ solutions are identified, and then, m solutions are selected as the corner solutions. The first m solutions are m solutions with the minimum value of each objective (i.e., solutions that minimize one of the m objectives). The other m solutions are m solutions with the minimum distance from each objective axis considering the L^2 norm (e.g., the solution that minimizes $f_2^2 + f_3^2 + \ldots + f_m^2$ for the f_1 axis $\mathbf{s}_1$). After those $2m$ solutions are identified, m corner solutions are selected. In [87] and [99], the point with the maximum objective value is selected for each objective. In

the source code of [87], nondominated solutions are selected first, and then, the point with the maximum objective value is selected for each objective from the nondominated solutions. The motivation of the hybrid approach is to utilize the two features:

- the minimum value-based approaches (in (a)) are useful for inverted triangular Pareto fronts as shown in Fig. 9 (a), and
- the axis vector-based approaches (in (b)) are useful for triangular Pareto fronts (as shown in Fig. 11 (a)).

2) Hyperplane Construction: We have surveyed various extreme point identification methods. When extreme points are identified, the nadir point $\mathbf{z}^{\text{ub}}$ can be directly estimated from the identified m extreme points $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_m\}$. In this case, however, the estimated nadir point is not correct when the extreme points are not correctly identified. For example, in Fig. 9 (b), the nadir point directly estimated from the three identified extreme points is completely different from the true nadir point. Alternatively, an m-dimensional linear hyperplane passing through these m points can be constructed and the nadir point $\mathbf{z}^{\text{ub}}$ can be derived from the intercept of the hyperplane with each axis, as illustrated in Fig. 8.

The equation of the plane is obtained by calculating its m intercepts with the axes of the objective space from the following linear system [26], [88]:

$$(\mathbf{a})^{\circ -1} = \mathbf{E}^{-1}\mathbf{u}, \tag{19}$$

where the matrix $\mathbf{E} = (\mathbf{f}(\mathbf{e}_1) - \mathbf{z}^{\text{lb}}, \mathbf{f}(\mathbf{e}_2) - \mathbf{z}^{\text{lb}}, \ldots, \mathbf{f}(\mathbf{e}_m) - \mathbf{z}^{\text{lb}})^{\text{T}}$, $\mathbf{a} = (a_1, a_2, \ldots, a_m)^T$ is a vector that consists of the intercepts of the hyperplane with all translated objective axes (i.e., a_i is the intercept of the hyperplane with the i-th axis of the translated objective space), $\mathbf{u} = (1, 1, \ldots, 1)^{\text{T}}$, and $(\mathbf{a})^{\circ -1} = ((a_1)^{-1}, (a_2)^{-1}, \ldots, (a_m)^{-1})^T$. The intercepts can be obtained by solving the linear system (19). Thus, the intercepts can be viewed as showing the estimated range of the Pareto front. The estimated nadir point in the original objective space can be computed by $\mathbf{z}^{\text{ub}} = \mathbf{a} + \mathbf{z}^{\text{lb}}$.

The overall process requires $O(m^3)$ computations, which results from the matrix inversion of size $m \times m$. Given its computational cost, hyperplane construction is expected to improve the quality of the estimated nadir point. For example, as shown in Fig. 9 (b), the hyperplane passing through the three identified extreme points is the same as the hyperplane with the true extreme points (i.e., the true linear Pareto front). In this case, the estimated nadir point is the same as the true nadir point, although none of the extreme points are correctly identified.

The hyperplane construction was analyzed on an implementation level in [55]. Blank et al. [55] pointed out two practical issues about the hyperplane construction: construction failure and abnormal intercepts. These issues were also reported in [25], [26]. The m extreme points may fail to construct an m-dimensional hyperplane. When linear-dependent extreme points are identified (e.g., in the degenerate case where duplicate extreme points are identified), the rank of the matrix $\mathbf{E}$ is less than m, resulting in a linear system with infinitely many solutions [25], [26]. On some problems, especially those with degenerate Pareto fronts, searching along different axis directions is very likely to produce duplicate extreme solutions [29], [55], as shown in Fig. 11 (d). Even when the hyperplane is constructed, the computed intercepts can be infinite [26] or negative [25], [26], [55] in some axis directions, leading to abnormal normalization results.

In the original NSGA-III paper [21], the handling of these issues was not mentioned. To address these issues, Yuan et al. [25] adopted the following strategy: z_i^{nad} is specified by the largest value of f_i in the nondominated solutions of S_t for each $i \in \{1, 2, \ldots, m\}$.

Blank et al. [55] revised the algorithm by specifying the upper and lower bounds of the estimated nadir point $\mathbf{z}^{\text{ub}}$. The upper bound of $\mathbf{z}^{\text{ub}}$ is defined by the worst point $\mathbf{z}^{\text{w}}$ having the worst value observed for each objective so far, while the lower bound of $\mathbf{z}^{\text{ub}}$ is defined by a slightly worse point than the estimated ideal point $\mathbf{z}^{\text{lb}}$. For each objective, when the intercept is very small or the estimated nadir point is larger than the worst point $\mathbf{z}^{\text{w}}$, the hyperplane is not useful [55]. Once the hyperplane construction fails, the nadir point is estimated by a straightforward method. Experimental results in [55] show that the estimated nadir point by the hyperplane construction is much better than that by the extreme points on the DTLZ [11] and scaled DTLZ [12] problems.

To investigate the effectiveness of hyperplane construction, we show the constructed hyperplanes for different Pareto fronts using the extreme points identified by approaches from NSGA-III [21] in Fig. 11. Our observations are as follows.

- Triangular linear Pareto front. If the three extreme points are included in the solution set, they are found as the extreme points. Then, the hyperplane constructed by those three points is the same as the linear Pareto front, as shown in Fig. 11 (a). The estimated ideal and nadir points are the same as the true ideal and nadir points, respectively.
- Inverted triangular Pareto front. Even when the three extreme points are included in the solution set, they are not selected by the ASF function (17). Three solutions (each of which is around the middle of each side of the Pareto front) are found, as shown in Fig. 11 (b). If those three points are on the Pareto front, the identified plane is the same as the Pareto front (exactly speaking, the Pareto front is on the identified plane and the Pareto front is a subset of the identified plane). However, the nadir point estimated by the intercepts is much larger (worse) than the true nadir point.
- Pareto front of the kite test problem [8]. Only one corner can be correctly identified as shown in Fig. 11 (c). Although the hyperplane can be constructed, the nadir point estimated by the intercepts is different from the true nadir point because the Pareto front of the kite problem is not linear.
- Degenerated Pareto front of the DTLZ5 [11]. Only one corner can be correctly identified as shown in Fig. 11 (d). The plane cannot be uniquely identified due to duplicate extreme points.

The identified plane is the same as the Pareto front when the following conditions are satisfied: (a) the Pareto front is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation

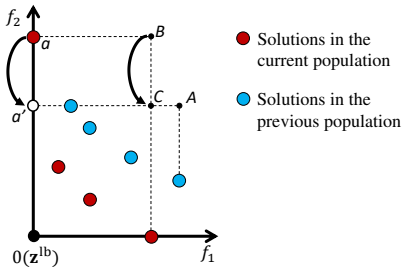


Fig. 12. Illustration of nadir point translation in [101].

linear, (b) all the obtained m points are on the Pareto front, and (c) all the obtained m points are linearly independent. When the condition (c) is not satisfied, inappropriate results can be obtained (i.e., a hyperplane cannot be uniquely determined).

3) Extreme-Point-First Principle: In the EMO field, the extreme points are usually selected first in order to obtain information about the boundary of the Pareto front [22], [100]. Some MOEAs are designed based on the extreme-point-first principle. For example, in MaOEA/D-2ADV [69], at early generations, search is only conducted along the objective axes in order to fast search for the extreme solutions. In B-NSGA-III [97], after the search for the extreme points converges, the search for other solutions starts. In VaEA [62], accurate boundary values are found for each objective, which may be attributed to the survival of extreme solutions in environmental selection.

C. Dominance Resistant Solution-Aware Nadir Point Estimation Methods

The presence of DRSs in the solution set can seriously inflate the estimated nadir point in certain objective directions. Some careful treatments are needed to avoid or weaken this phenomenon during the nadir point estimation. The methods considering the existence of DRSs are called DRS-aware nadir point estimation methods in this paper.

Asafuddoula et al. proposed a nadir point translation method in g-DBEA [101]. At each generation, the nadir point is estimated by a straightforward estimation method discussed in IV-A with nondominated solutions in the current population as the solution set. The estimated nadir point can be translated by taking the minimum value of the nadir points from the previous and current generations. As illustrated in Fig. 12, points A and B are the nadir points estimated from nondominated solutions in the previous (blue points) and current (red points) generation, respectively. If the solution a is a DRS solution, it is translated parallel to the f_2-axis such that its f_2 value is equal to the closest point in the previous generation. Correspondingly, the nadir point is translated from B to C. Nadir point translation is performed regularly (e.g., every 10 generations when the total number of generations is 100) [101].

Six sigma strategies are commonly used in statistical quality control [102] and robustness analysis [103]. Bhattacharjee et al. [49] proposed a six sigma-based DRS removal scheme to minimize the influence of DRSs. Nondominated solutions along each axial direction are categorized as DRSs if they have an objective value more than the mean plus six times its standard deviation. These DRSs will be removed from the solution set. The remaining solution set is then used to compute the amended nadir point which in turn is used for normalization.

Xiang et al. [91] proposed a hypercube-based method in PaRP/EA [91] to deal with DRSs. The hypercube is defined by the estimated ideal and nadir points. The nadir point is estimated by the nadir point estimation method in NSGA-III [21]. That is, the ASF function (17) is used to identify the extreme points. Solutions outside the hypercube are regarded as DRSs and are removed in environmental selection.

D. Convergence-aware Nadir Point Estimation Methods

When some solutions in the current population are far away from the Pareto front, the estimated nadir point by a straightforward method from the current population is not accurate. The location of the estimated nadir point can be strongly biased. This is why some studies have proposed methods to use convergence information in nadir point estimation. Such methods are called convergence-aware nadir point estimation methods in this paper.

Hernández Gómez and Coello Coello [104] proposed a nadir point update mechanism by monitoring convergence of the population in MOMBI-II [104]. At each generation, a temporary nadir point $\mathbf{z}^{\text{temp}}$ is estimated by taking the maximum value of each objective in the current population. A pivot nadir point $\mathbf{z}^{\text{ub}}$ is initialized by $\mathbf{z}^{\text{temp}}$ at the first generation and updated by $\mathbf{z}^{\text{temp}}$ at each generation. The temporary nadir points of the previous five generations are recorded, and the variance of the five points for each objective is used as a metric to evaluate the proximity of the current population to the true Pareto front. Based on the proximity, the nadir point is updated in different ways. The update of the nadir point requires $O(mN)$ computations. This method is also used in iMOACO$_{\mathbb{R}}$ [105].

Qi et al. [106] proposed an adaptive nadir point setting strategy for MOEA/D-IPBI [107]. The nadir point is first estimated by the worst value of each objective in the current population. When the estimated ideal point remains unchanged for G generations, the nadir point is set to the upper bound of the extreme points. Note that an extreme point can be defined in different manners, as shown in Table I. In [106], an extreme point is defined as a solution with the best function value for one objective function in the current population. The parameter G is set as 90 in [106]. This strategy can reduce the negative effects of dominated solutions in nadir point estimation [107].

Blank et al. [55] proposed to consider multiple nondominated fronts adaptively as the solution set. Degenerate cases may occur when the set of nondominated solutions in the current population are selected for nadir point estimation. The estimated nadir point by this method might be the same as the estimated ideal point if the population has only one nondominated solution. To avoid the division by zero problem, better fronts by the nondominated sorting are included in the solution set S one by one until the range $z_i^{\text{ub}} - z_i^{\text{lb}}$ for each objective is larger than a prespecified threshold.

Saxena and Kapoor [108] proposed to intermittently update the nadir point. A stability tracking procedure based upon weight vectors was proposed for decomposition-based MOEAs to check if the population is stabilized. This procedure has been incorporated into NSGA-III [21]. At every generation, this procedure is performed and the nadir point is only updated when the stability tracking procedure indicates that the population is stabilized. This method has also been used in HFiDEA [109].

E. Constraint-aware Nadir Point Estimation Methods

When dealing with constrained MOPs, the feasibility of solutions has a large impact on nadir point estimation. However, constraints are usually ignored in estimating the nadir point. The methods taking into consideration the constraint information are called constraint-aware nadir point estimation methods in this paper.

Fukumoto and Oyama [60] proposed a constraint-aware estimation method for the ideal and nadir points. All solutions are sorted into different fronts as follows. Feasible solutions are ranked by the nondominated sorting and infeasible solutions are ranked by constraint violation values. The fronts of feasible solutions are ranked before (i.e., better than) the fronts of infeasible solutions. Then these fronts are added to the solution set S one by one to estimate the ideal and nadir points until the number of solutions inside the hyperrectangle bounded by the two points is larger than 5% of the population size. This stopping criterion guarantees that the number of solutions in the solution set S is not too small and the ideal-nadir hyperrectangle is near the most feasible region even when there are only infeasible solutions. Experimental results show that the proposed estimation method improves the performance of MOEA/D-M2M [110] in solving a real-world constrained MOP (i.e., the Mazda CdMOBP problem [111]).

F. Discussions and Summary

In summary, the survey of studies on nadir point estimation reveals the following.

- The extreme point identification in [14] is based on the assumption that the Pareto front is inverted triangular, which may fail to identify any corner solutions when the Pareto front is regular triangular, as explained in Fig. 9 (b). Extreme point identification methods using the axis vectors are based on the assumption that the Pareto front is regular triangular. However, as discussed in [45], [112], this shape is not realistic. It is advisable to consider both shapes as in I-DBEA [87].
- In DSS [89], starting from the extreme point $\mathbf{e}_1$ identified by Eq. (7), the most distant solutions from the identified solutions are selected one by one. This method guarantees that at least $m-1$ extreme points can be correctly identified, as explained in Fig. 9 (c). This approach can be modified by discarding the first identified extreme point and finding another solution farthest from the other identified solutions. As shown in Fig. 9 (a) and (d), all of the three extreme points can be correctly identified by the modified approach on the regular and inverted triangular Pareto fronts.
- There exist different definitions of the closeness of a solution to each axis vector in axis vector-based extreme point identification methods. It is worth comparing them and examining the effects of different definitions.
- MOEAs based on the extreme-point-first principle may help nadir point estimation.
- Hyperplane construction in the extreme point-based nadir point estimation methods is computationally expensive, particularly for many-objective problems [21], [29].
- Hyperplane construction has two issues (i.e., construction failure and abnormal intercepts) that should be carefully handled as in [25], [26], [55].
- The effectiveness of hyperplane construction is dependent on Pareto front shapes. While the estimated nadir point is the same as the true nadir point on triangular linear Pareto fronts, it is often much worse than the true nadir point on inverted triangular Pareto fronts and different from the true nadir point on nonlinear Pareto fronts (e.g., the kite test problem [8]). The hyperplane cannot be uniquely constructed for degenerated Pareto fronts (e.g., DTLZ5 [11]).
- DRSs are often identified as extreme points, which may severely deteriorate the performance of the extreme point-based nadir point estimation methods. It is important to take the existence of DRSs into consideration when estimating the nadir point as used in DBEA-DS [49], g-DBEA [101], and PaRP/EA [91].
- It is interesting to use convergence information, as in [55], [106] and MOMBI-II [104], to improve the quality of the estimated nadir point since the nadir point estimated by a straightforward method is not accurate when the current population is far away from the Pareto front.
- Constraint information can be considered in nadir point estimation when handling constrained MOPs as in [60].

V. Normalization with Estimated Pareto Front Range

Given a population P with N individuals, the objective space normalization generates a set of normalized points $\{\widetilde{\mathbf{f}}(\mathbf{x}^1), \widetilde{\mathbf{f}}(\mathbf{x}^2), \ldots, \widetilde{\mathbf{f}}(\mathbf{x}^N)\}$ in the m-dimensional objective space. Note that, in this section, we assume that the estimated ideal point $\mathbf{z}^{\text{lb}}$ and the estimated nadir point $\mathbf{z}^{\text{ub}}$ have already been obtained. Since the estimated ideal and nadir points can be inaccurate, several modified normalization equations have also been proposed to reduce the negative effect of inaccurate estimation of the two points. In this section, a simple straightforward normalization method and its modifications are surveyed. Alternative attempts of normalization in MOEAs using weight vectors or reference vectors are also surveyed in this section. These methods are summarized in Table II of the supplementary document due to the page limit.

A. Straightforward Objective Space Normalization

The straightforward equation for objective space normalization directly replaces the true ideal and nadir points in Eq. (3) with the estimated ideal and nadir points as follows [21], [67], [113]:

$$\widetilde{f}_i(\mathbf{x}) = \frac{f'_i(\mathbf{x})}{z_i^{\text{ub}} - z_i^{lb}} = \frac{f_i(\mathbf{x}) - z_i^{\text{lb}}}{z_i^{\text{ub}} - z_i^{\text{lb}}},\ i \in \{1, 2, \ldots, m\}. \quad (20)$$

Each individual in the population P is first translated such that the estimated ideal point of the translated population becomes a zero vector. After objective value translation, each translated objective vector $f'_i(\mathbf{x})$ is scaled so that solutions inside the hyperrectangle bounded by the estimated ideal and nadir points are in the range of $[0, 1]$ [113].

The objective space translation is important, especially for decomposition-based MOEAs. This is because the objective space translation makes sure that the weight vectors start from the origin, which maximizes the coverage of the objective space by the weight vectors [67]. In addition, the objective space translation guarantees that all individuals are in the first quadrant. That is, all the objective values are strictly nonnegative. This is a necessary condition for some operations (e.g., binary ϵ-indicator I_ϵ [114] and density estimator [83]).

The significance of the objective value translation has been empirically studied in [67]. It was reported in [67] that the number of intersection points between the weight vectors and the Pareto front in decomposition-based MOEAs was increased by the objective value translation. Thus, the density of the approximated Pareto optimal solutions was increased.

As pointed out in [12], the use of Eq. (20) has a zero division issue, which can occur when the diversity of the population is small. Several variants of the Eq. (20) are recommended to avoid this issue, which are surveyed in the following subsections.

B. Modified Objective Space Normalization

Different modifications of the normalization equations have been proposed to reduce the possible negative effects of inaccurate estimation of the ideal and nadir points. They can be classified into four categories: use of small values in numerator and denominator, use of conditions on the range of objective values, use of conditions on constraints, and consideration of the reliance on the estimated Pareto front range.

1) Small Values in The Numerator and The Denominator: There exist different kinds of modifications for Eq. (20). The most commonly-used modification is as follows:

$$\widetilde{f}_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - z_i^{\text{lb}} + \alpha}{z_i^{\text{ub}} - z_i^{\text{lb}} + \beta}, \quad (21)$$

where α and β are user-defined parameters. The main advantage of this modification is to prevent the equation from zero division in the processes of computation [115], [116].

Usually, the setting of $\alpha = 0$ and $\beta = \epsilon$ is adopted. Ishibuchi et al. [12] investigated the effect of the value of ϵ in MOEA/D. Different values of ϵ (i.e., 10^{-12}, 10^{-6}, 1) are considered. Experimental results in [12] show that an appropriate specification of ϵ is problem dependent. Recently, Pang et al. [117] proposed a hyper-heuristic method to tune the parameter ϵ (i.e., from 10^{-6} to 25) in an offline manner. The auto-tuned MOEA/D has different values of ϵ for different problems. Experimental results in [117] show that an appropriate specification of ϵ for each problem is able to remedy the diversity deterioration caused by inaccurately estimated ideal and nadir points.

In some papers [115], [116], the setting that $\alpha = \beta = \epsilon$ is used. In this case, it is interesting to note that Eq. (21) can be reformatted as follows:

$$\widetilde{f}_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - (z_i^{\text{lb}} - \epsilon)}{z_i^{\text{ub}} - (z_i^{\text{lb}} - \epsilon)} = \frac{f_i(\mathbf{x}) - z_i^{**}}{z_i^{\text{ub}} - z_i^{**}}, \quad (22)$$

where z_i^{**} is the i-th objective of the estimated utopian point[5]. That is, the range of the Pareto front is bounded by the estimated nadir point and the estimated utopian point [19]. It is worth noting that the setting of α and β is highly related to the ideal point estimation methods, which we have discussed in Section III.

2) Conditions on The Range of Objective Values: Ishibuchi et al. [12] reported that a very small range of objective values is the main reason for the severe negative effects of normalization on the performance of MOEA/D. Different attempts were made in the literature to constrain the range of objective values.

He et al. [53] proposed to fix the normalized objective value based on the range of objective values. The value of the normalized objective $\widetilde{f}_i$ is fixed (i.e., changed) to a small value when the estimated range of that objective is very small. That is, $\widetilde{f}_i(\mathbf{x}) = 10^{-10}$ when $z_i^{\text{ub}} - z_i^{\text{lb}} < 10^{-10}$.

Liu et al. [33] fixed the estimated range of an objective to 1 in 1by1EA [33] when the estimated range of that objective is very small. That is, $z_i^{\text{ub}} - z_i^{\text{lb}} = 1$ when $z_i^{\text{ub}} - z_i^{\text{lb}} < 10^{-21}$.

Tian et al. [52] used the following strategy in NSGA-II-SDR [52]: the normalization is performed only when the maximum range is smaller than 20 times the minimum range. That is, $\max\limits_{j=1,\ldots,m}(z_j^{\text{ub}} - z_j^{\text{lb}}) < 20 \times \min\limits_{j=1,\ldots,m}(z_j^{\text{ub}} - z_j^{\text{lb}})$. Note that the effect of this multiplicity parameter was not studied in [52].

3) Conditions on Constraints: Liu and Wang [118] experimentally showed that the normalization for constrained MOPs may mislead the search process since some infeasible solutions far away from the feasible region can have small (i.e., good) objective function values. It was argued that it is impractical to employ only the feasible solutions to normalize the objective functions since all solutions in the population may be infeasible in the early stage of evolution [60], [118].

Li et al. [119] proposed to control the timing of normalization by considering the constraint information in C-TAEA [119]. The objective space is normalized before density estimation only when the number of feasible solutions is larger than the population size N. That is, normalization is only performed when there are enough feasible solutions.

[5]A utopian point $\mathbf{z}^{\star\star} = \left(z_1^{\star\star}, z_2^{\star\star}, \ldots, z_m^{\star\star}\right)^T$ is a vector that is strictly better than the ideal point. In practice, a utopian point is specified by subtracting a small positive value ϵ from the ideal point. Thus, we have $z_i^{\text{nad}} - z_i^{\star\star} > 0$ for all $i = 1, \ldots, m$.

4) Consideration of Reliance on The Estimated Pareto Front Range: At early generations, the population is far away from the Pareto front and the estimated ideal and nadir points (i.e., the estimated Pareto front range) are not accurate [12]. Although this has been well recognized in the literature, objective space normalization is still often performed completely relying on the estimated range even at the beginning of the evolutionary process. Several methods have been proposed to reduce the negative effects of the inaccurately estimated Pareto front range.

He et al. [120] proposed a dynamic normalization strategy to control the reliance of normalization on the estimated Pareto front range. The straightforward normalization equation Eq. (20) is reformulated as follows:

$$\widetilde{f}_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - z_i^{\text{lb}}}{L_i + \epsilon}, i = 1, 2, \ldots, m, \tag{23}$$

where $L_i = (z_i^{\text{ub}} - z_i^{\text{lb}})/((1-\alpha)(z_i^{\text{ub}} - z_i^{\text{lb}} - 1) + 1)$, and α is a parameter with a value in the range of $[0, 1]$ that controls the reliance of normalization on the estimated Pareto front range. When $\alpha = 0$, we have $L_i = 1$, which makes Eq. (23) independent of the estimated nadir point. That is, only the objective function translation occurs. When $\alpha = 1$, we have $L_i = z_i^{\text{ub}} - z_i^{\text{lb}}$ (i.e., exactly the same as the straightforward normalization equation, Eq. (20)). By controlling the value of α from 0 to 1, the reliance of Eq. (23) on the estimated Pareto front range increases. In [120], the value of α increases from 0 to 1 linearly or based on a sigmoid function. Experimental results show that the dynamic change in α makes the normalization more robust than the case of $\alpha = 1$ (i.e., the straightforward normalization equation, Eq. (20)).

Unlike [120], Saxena and Kapoor [108] adopted a step function to control the reliance on the estimated Pareto front range. The objective function translation (i.e., the value of α in Eq. (23) is 0 and the objective space does not rely on the estimated nadir point) is performed at early generations when the population is not stable. A stability tracking method was proposed for NSGA-III to check if the population is stable. Normalization (i.e., the value of α in Eq. (23) is 1) is performed only when the population stabilizes. This method has also been used in HFiDEA [109].

In [121], two populations in different objective spaces are evolved collaboratively, one in the translated objective space (i.e., the objective space does not rely on the estimated nadir point) and another in the normalized objective space. The nondominated solutions among all solutions in the merged populations are used to estimate the ideal and nadir points. The experimental results show that the use of the two objective spaces in normalization improves the performance of MOEA/D in solving various MOPs.

C. Weight/Reference Vector Modification

For MOEAs with weight vectors or reference vectors, instead of normalizing the solutions in the objective space, the weight/reference vector scaling can be an alternative way of normalization. Weight vector modification seems to have almost the same effect as objective space normalization in obtaining a uniformly distributed set of solutions, as explained in Fig. 2 (b) and (c).

Cheng et al. [67] proposed a reference vector scaling strategy in RVEA [67] based on the minimum and maximum values of each objective. That is, the reference vectors are scaled according to the range of solutions, instead of normalizing the solutions in the objective space, as illustrated in Fig. 2. This strategy is periodically performed (e.g., every 10 generations). The sensitivity analysis of the frequency of this strategy shows that frequent use (e.g., every generation) of this strategy leads to a significant performance deterioration of RVEA on DTLZ3. This is also pointed out by Giagkiozis et al. [122].

Tian et al. [72] adopted this strategy in AR-MOEA [72]. Unlike RVEA [67], the scaled weight vectors are not normalized, but they are generated in the region of $\prod_{i=1}^{m} \left[0, z_i^{\text{ub}} - z_i^{\text{lb}}\right]$. Habib et al. [74] adopted this strategy in HSMEA [74]. Unlike RVEA [67], the weight vector modification in AR-MOEA [72] and HSMEA [74] is performed at every generation.

Li et al. [84] investigated the weight vector modification in MOEA/D-PBI [20] on the three-objective SDTLZ problems. The original MOEA/D-PBI, MOEA/D-PBI with objective space normalization, and MOEA/D-PBI with weight vector modification are compared. Experimental results show that MOEA/D-PBI with weight vector modification is more robust.

Sun et al. [92] proposed a reference point generation method for IGD calculation in MaOEA/IGD [92]. The reference points for IGD calculation are generated by rescaling and transferring the points generated by the Das and Dennis's method [86] using the ideal and nadir points. This method has also been used in MaOEA-IT [93].

D. Discussions and Summary

In summary, the survey of studies on normalization with the estimated Pareto range reveals the following.

- In [117], the value of the parameter β in Eq. (21) is automatically tuned for each problem to reduce the negative effects of normalization. It is interesting to design a robust normalization method by automatically selecting appropriate methods for ideal point estimation, nadir point estimation and the use of the estimated range.
- The effects of reference point specification in decomposition-based MOEAs have been well studied [12]. While reference point specification is closely related to objective space normalization, the effects of these two factors have not been discussed in a unified manner. Addition of the same small value to the numerator and the denominator of the basic normalization formulation may have a positive effect for decomposition-based MOEAs since it is equivalent to the use of a utopian point as the reference point.
- The estimated range of the Pareto front can be very small in some objectives, especially when population diversity is poor. Different measures have been taken to prevent the normalization from being too much or too small [33], [52], [53]. It is interesting to compare these measures and examine their effectiveness.

- For constrained MOPs, constraint information can be used to control the timing of normalization as in C-TAEA [119] to avoid improper normalization.
- The inaccuracy of the estimated Pareto front range at early generations should be considered as in [108], [109], [120], [121]. The reliance of normalization on the estimated range can be controlled according to the accuracy of the estimated range.
- For MOEAs with weight vectors or reference vectors, the weight/reference vector scaling can be an alternative way of normalization as in RVEA [67], AR-MOEA [72], and MaOEA/IGD [92]. It is interesting to compare the weight/reference vector scaling with the objective space normalization methods. The frequency of performing the weight/reference vector scaling is also an interesting topic that is worth investigating.

VI. Further Discussions on The Effect of Normalization

In the previous sections, we have surveyed various objective space normalization methods, including ideal point estimation methods, nadir point estimation methods, and different methods based on the utilization of the estimated Pareto front range. In this section, we attempt to further discuss the effect of normalization.

In an MOEA, solutions are ranked based on a certain comparison method for environmental selection (and mating selection). When the ordering of solutions is not affected by the scaling of the objectives, the comparison method is scaling invariant or scaling independent [123], [124]. The scaling invariance in the ranking of solutions is an important property since the ranking result with this property does not depend on the absolute objective function values [124]. For example, if a comparison method is not scaling invariant, the ranking result of solutions can be changed by the choice of a unit for length such as "mm", "cm" and "m". Since the scaling-invariant property is closely related to the effect of objective space normalization, we discuss this property by explaining some related theorems.

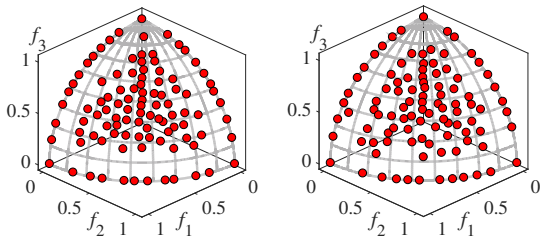
To the best of the authors' knowledge, only two types of comparison methods that are not dependent on scaling are as follows [123], [125], [126],

1) Objective-wise comparison (e.g., $favour$ relation [127], Pareto dominance relation [54]);
2) Volume-based comparison (e.g., hypervolume indicator [126], I_{HD} [64]).

These two types of comparison methods are scaling invariant as shown by the following theorems (For proofs of Theorem 2-4, refer to the supplementary document.):

Theorem 2. *Let $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^m$ be two solutions in the objective space. The objective-wise comparison (i.e., comparing the values of each objective independently from the others) between $\mathbf{x}$ and $\mathbf{y}$ is scaling invariant.*

Theorem 3. *Let $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^m$ be two solutions in the objective space. The comparison between the volume covered by $\mathbf{x}$ and the volume covered by $\mathbf{y}$ is scaling invariant.*



(a) Without normalization. (b) With normalization.

Fig. 13. Solutions obtained by IBEA_{HD} with the median hypervolume value over 21 runs on the three-objective WFG4. Experimental settings are the same as [12].

In the ranking of solutions for environmental selection (and mating selection), the given m objectives are combined to calculate a single scalar fitness value for each solution. Except for the abovementioned two types of comparison methods, the comparison result of solutions depends on the scaling of each objective [123], [126]. Objective space normalization is necessary in order to treat different objectives equally.

Theorem 4. *Let $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^m$ be two solutions in the objective space. When the fitness values of solutions are obtained by adding different objectives together, the comparison between $\mathbf{x}$ and $\mathbf{y}$ is not scaling invariant.*

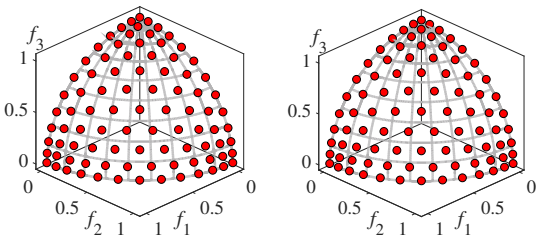
Based on Theorem 2-4, MOEAs can be classified into three classes: scaling-independent MOEAs, scaling-insensitive MOEAs, and scaling-sensitive MOEAs. In the following, these three classes are briefly explained.

When only scaling-independent comparisons are used in the environmental selection, the MOEA is scaling independent. SMS-EMOA [128], HypE [129], and FV-EMOA [130] are representative scaling-independent MOEAs since only Pareto dominance relation and hypervolume are used in the environmental selection. In these algorithms, the Pareto dominance relation is used as the first selection criteria to emphasize convergence, and the diversity is maintained using the hypervolume indicator. IBEA_{HD} [64] is also a scaling-independent MOEA since only the scaling-independent indicator I_{HD} [64] is used in environmental selection. These MOEAs are able to deal with MOPs with differently-scaled objectives without objective space normalization. As shown in Fig. 13, similar solution sets are obtained by IBEA_{HD} with and without normalization on the three-objective WFG4.

In MOEAs with only distance-based fitness evaluation mechanisms in environmental selection (e.g., MOEA/D [20] and 1by1EA [33]), normalization is necessary in order to deal with MOPs with differently-scaled objectives [123]. These MOEAs are categorized as scaling-sensitive MOEAs. That is, these MOEAs are sensitive to normalization. The use of objective space normalization in these MOEAs often leads to poor results on MOPs with similar objective ranges (i.e., MOPs for which the normalization is not needed) when the estimated ideal and nadir points are inaccurate during the evolutionary process [12], [33].

In NSGA-II [54], SPEA2 [131], NSGA-III [21], and θ-DEA [26], solutions are forced to converge to the Pareto front

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2021.3076514, IEEE Transactions on Evolutionary Computation



(a) Without normalization. (b) With normalization.

Fig. 14. Solutions obtained by NSGA-III with the median hypervolume value over 21 runs on the three-objective DTLZ3. Experimental settings are the same as [12].

by Pareto dominance-based fitness evaluation mechanisms. However, normalization is still needed because of the use of distance-based density measures (e.g., crowding distance, k nearest neighbor, weight vectors) to emphasize diversity [132]. These MOEAs, which are categorized as scaling-insensitive MOEAs, are less sensitive to normalization compared to MOEAs with only distance-based fitness evaluation mechanisms (e.g., MOEA/D [20]). As shown in Fig. 14, well-distributed solutions are obtained by NSGA-III with or without normalization on the three-objective DTLZ3. That is, a normalization method does not affect the performance of NSGA-III on the three-objective DTLZ3 (see Fig. 14), while it affects the performance of MOEA/D on the three-objective DTLZ3 (see Fig. 4).

It is worth noting that although the hypervolume calculation has the scaling-independent property, it is very computationally expensive, which makes it inappropriate when the number of objectives is large. To reduce the computational cost of hypervolume calculation, several works have been proposed to approximate the hypervolume instead of calculating it exactly. For example, the hypervolume contribution is approximated by the $R2$ indicator R_2^{HVC} [133] in R2HCA-EMOA [75]. However, although this approximation can significantly reduce the computational cost of hypervolume calculation, it makes normalization a necessary part. In addition, the reference point specification for many-objective problems can be recognized as the main difficulty in the use of the hypervolume indicator [134], [135]. As reported in [134], [135], while the hypervolume indicator is scaling invariant, it is sensitive to the location of the reference point. The ordering of solutions in environmental selection strongly depends on the location of the reference point. Moreover, the reference point specification usually depends on the estimated nadir point, which is closely related to objective space normalization. In this sense, only the first type of comparison methods can be viewed as purely independent of objective space normalization.

VII. Conclusion and Future Directions

As an independent algorithmic component, objective space normalization plays an important role in the design of MOEAs since the objectives of a real-world MOP by nature are often of very different scales. This paper comprehensively surveyed studies on normalization methods in three parts: ideal point estimation, nadir point estimation, and objective space normalization methods (including weight/reference vector modification methods). From the analysis of each normalization method, we found that although a number of attempts have been made in objective space normalization, there is still lack of an effective and robust normalization method for MOEAs, and consequently, it is still an open research area. Some directions worth pursing in the future are summarized below.

1) The ideal points estimated by the straightforward methods are usually inaccurate at early generations since the population at early generations is far away from the true Pareto front. Dynamic ideal point estimation methods [57], [58] have been used in MOEA/D and promising results are obtained on MOPs with similar objective ranges. It is worth investigating whether these methods can help to estimate a more accurate ideal point for objective space normalization.
2) The straightforward nadir point estimation methods cannot accurately estimate the nadir point when the available solution set is not appropriate for the estimation (e.g., due to the inclusion of DRSs). DRS removal strategies (as in [49], g-DBEA [101], and PaRP/EA [91]) and convergence information (as in [55], [106] and MOMBI-II [104]) can be used to improve the quality of the solution set or remedy the inaccurate nadir point.
3) Extreme points are important for nadir point estimation. Emphasizing the importance of extreme points by adopting an extreme-point-first principle in algorithm design as in MaOEA/D-2ADV [69], B-NSGA-III [97], and VaEA [62] may improve the accuracy of the estimated nadir point and make an MOEA more robust in dealing with real-world MOPs.
4) Most of the extreme point identification methods are based on the assumption that the Pareto front is either regular triangular or inverted triangular. It is interesting to consider corner solutions [87], which are selected from the union of solutions identified by the methods of the previous two categories.
5) In [89], starting from the extreme point $\mathbf{e}_1$ identified by Eq. (7), the solutions with the farthest distance from the identified solutions are selected one by one. This method guarantees that at least $m-1$ corner points can be correctly identified. It is interesting to examine if m corner points can be correctly identified by discarding the first identified extreme point and finding another solution farthest from the identified solutions.
6) In axis vector-based extreme point identification methods, the closeness of a solution to each axis vector is defined in many different ways, as shown in Fig. 10. It is interesting to investigate their difference.
7) The estimated range of the Pareto front can be very small in some objectives, especially when population diversity is poor. Different measures have been taken to prevent the normalization from being too much or too small [33], [52], [53]. It is interesting to compare these measures and examine their effectiveness.
8) Investigation into the impact of objective space normalization should also be carried out for constrained MOPs.

For constrained MOPs, the constraint information can also be considered for better nadir point estimation [60]. It can be used to control the timing of normalization to avoid improper normalization [119].

9) At early generations of MOEAs, the current population cannot be used to accurately approximate the shape of the Pareto front. As a result, the estimated ideal and nadir points are likely to be inaccurate. It is interesting to dynamically adjust the extent of normalization based on the accuracy (reliability) of the estimated ideal and nadir points as in [108], [109], [120].
10) For MOEAs with weight/reference vectors, the weight/reference vector scaling can be an alternative way of normalization as in RVEA [67], AR-MOEA [72], and MaOEA/IGD [92]. It is interesting to compare the weight/reference vector scaling with the objective space normalization methods. The effect of the frequency of performing the weight/reference vector scaling and the nadir point update is also an interesting research topic.
11) Although a substantial number of methods have been proposed for objective space normalization, there is still lack of a benchmarking study of these methods. No metric has been proposed to measure the effect of normalization.
12) In [117], the value of the parameter β in Eq. (21) is automatically tuned for each problem to reduce the negative effects of normalization. It is interesting to configure a robust normalization method by automatically selecting an appropriate method for each of the three components of normalization (i.e., ideal point estimation, nadir point estimation, and the use of the estimated range).
13) Scaling-independent comparison methods are encouraged in MOEA design to reduce its sensitivity to the inaccurately estimated ideal and nadir points. Design of scaling-independent MOEAs can be a promising future direction.
14) For all of the abovementioned topics, it is important to use realistic MOPs (e.g., [6]). Design of realistic test suites is also an important future research topic.

REFERENCES

[1] B. H. Nguyen, B. Xue, P. Andreae, H. Ishibuchi, and M. Zhang, "Multiple reference points-based decomposition for multiobjective feature selection in classification: Static and dynamic mechanisms," *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 170–184, Feb. 2020.

[2] Y. Tian, S. Yang, X. Zhang, and Y. Jin, "Using PlatEMO to solve multi-objective optimization problems in applications: A case study on feature selection," in *Proc. IEEE Congr. Evol. Comput.*, Wellington, New Zealand, Jun. 2019, pp. 1710–1717.

[3] J. Knowles, L. Thiele, and E. Zitzler, "A tutorial on the performance assessment of stochastic multiobjective optimizers," Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland, Technical Report, 2006.

[4] Q. Zhang, H. Li, D. Maringer, and E. Tsang, "MOEA/D with NBI-style Tchebycheff approach for portfolio management," in *Proc. IEEE Congr. Evol. Comput.*, Barcelona, Spain, Jul. 2010, pp. 1–8.

[5] H. Jain and K. Deb, "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. Evol. Comput.*, vol. 18, no. 4, pp. 602–622, Aug. 2014.

[6] R. Tanabe and H. Ishibuchi, "An easy-to-use real-world multi-objective optimization problem suite," *Appl. Soft Comput.*, vol. 89, Apr. 2020.

[7] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.

[8] S. Zapotecas-Martinez, C. A. Coello Coello, H. E. Aguirre, and K. Tanaka, "A review of features and limitations of existing scalable multiobjective test suites," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 130–142, Feb. 2019.

[9] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, and X. Yao, "A benchmark test suite for evolutionary many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 1, pp. 67–81, Mar. 2017.

[10] H. Li, K. Deb, Q. Zhang, P. Suganthan, and L. Chen, "Comparison between MOEA/D and NSGA-III on a set of novel many and multi-objective benchmark problems with challenging difficulties," *Swarm Evol. Comput.*, vol. 46, pp. 104–117, May 2019.

[11] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proc. IEEE Congr. Evol. Comput.*, Honolulu, USA, May 2002, pp. 825–830.

[12] H. Ishibuchi, K. Doi, and Y. Nojima, "On the effect of normalization in MOEA/D for multi-objective and many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 4, pp. 279–294, Dec. 2017.

[13] L. Paquete and T. Stützle, "A two-phase local search for the biobjective traveling salesman problem," in *Proc. Evol. Multi-Criter. Optim.*, Faro, Portugal, Apr. 2003, pp. 479–493.

[14] R. Tanabe, H. Ishibuchi, and A. Oyama, "Benchmarking multi- and many-objective evolutionary algorithms under two optimization scenarios," *IEEE Access*, vol. 5, pp. 19 597–19 619, Sep. 2017.

[15] K. Miettinen, P. Eskelinen, F. Ruiz, and M. Luque, "NAUTILUS method: An interactive technique in multiobjective optimization based on the nadir point," *European Journal of Operational Research*, vol. 206, no. 2, pp. 426–434, Oct. 2010.

[16] K. Deb and K. Miettinen, "A review of nadir point estimation procedures using evolutionary approaches: A tale of dimensionality reduction," in *Proc. Multiple Criter. Decis. Making Conf.*, Auckland, New Zealand, Jan. 2009, pp. 1–14.

[17] K. Deb, K. Miettinen, and S. Chaudhuri, "Toward an estimation of nadir objective vector using a hybrid of evolutionary and local search approaches," *IEEE Trans. Evol. Comput.*, vol. 14, no. 6, pp. 821–841, Dec. 2010.

[18] H. Wang, S. He, and X. Yao, "Nadir point estimation for many-objective optimization problems based on emphasized critical regions," *Soft Comput.*, vol. 21, no. 9, pp. 2283–2295, May 2017.

[19] K. Miettinen, M. M. Mäkelä, and K. Kaario, "Experiments with classification-based scalarizing functions in interactive multiobjective optimization," *European Journal of Operational Research*, vol. 175, no. 2, pp. 931–947, Dec. 2006.

[20] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[21] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints." *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

[22] S. Liu, Q. Lin, K.-C. Wong, C. A. C. Coello, J. Li, Z. Ming, and J. Zhang, "A self-guided reference vector strategy for many-objective optimization," *IEEE Trans. Cybern.*, 2020 (Early Access).

[23] J. Blank and K. Deb, "Pymoo: multi-objective optimization in Python," *IEEE Access*, vol. 8, pp. 89 497–89 509, Apr. 2020.

[24] Y.-H. Zhang, Y.-J. Gong, T.-L. Gu, H.-Q. Yuan, W. Zhang, S. Kwong, and J. Zhang, "DECAL: Decomposition-based coevolutionary algorithm for many-objective optimization," *IEEE Trans. Cybern.*, vol. 49, no. 1, pp. 27–41, Jan. 2019.

[25] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 180–198, Apr. 2015.

[26] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 1, pp. 16–37, Feb. 2016.

[27] M. Elarbi, S. Bechikh, A. Gupta, L. Ben Said, and Y.-S. Ong, "A new decomposition-based NSGA-II for many-objective optimization," *IEEE Trans. Syst. Man Cybern, Syst.*, vol. 48, no. 7, pp. 1191–1210, Jul. 2018.

[28] R. Wang, Z. Zhou, H. Ishibuchi, T. Liao, and T. Zhang, "Localized weighted sum method for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 3–18, Feb. 2016.

[29] S. Jiang and S. Yang, "A strength Pareto evolutionary algorithm based on reference direction for multiobjective and many-objective

optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 329–346, Jun. 2017.

[30] C. Liu, Q. Zhao, B. Yan, S. Elsayed, T. Ray, and R. Sarker, "Adaptive sorting-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 247–257, Apr. 2018.

[31] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 5, pp. 721–736, Oct. 2013.

[32] X. Zhang, Y. Tian, and Y. Jin, "A knee point-driven evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 761–776, Dec. 2015.

[33] Y. Liu, D. Gong, J. Sun, and Y. Jin, "A many-objective evolutionary algorithm using a one-by-one selection strategy," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2689–2702, Sep. 2017.

[34] Y. Liu, H. Ishibuchi, Y. Nojima, N. Masuyama, and K. Shang, "Improving 1by1EA to handle various shapes of Pareto fronts," in *Proc. Int. Conf. Parallel Prob. Solv. Nat.*, Coimbra, Portugal, Sep. 2018, pp. 311–322.

[35] A. Jaszkiewicz, "Evaluation of multiple objective metaheuristics," in *Metaheuristics for Multiobjective Optimisation*. Berlin, Heidelberg: Springer, 2004, vol. 535, pp. 65–89.

[36] L. He, Y. Nan, K. Shang, and H. Ishibuchi, "A study of the naïve objective space normalization method in MOEA/D," in *IEEE Symp. Ser. Comput. Intell.*, Xiamen, China, Dec. 2019, pp. 1834–1840.

[37] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "Performance comparison of NSGA-II and NSGA-III on various many-objective test problems," in *Proc. IEEE Congr. Evol. Comput.*, Vancouver, BC, Canada, Jul. 2016, pp. 3045–3052.

[38] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 694–716, Oct. 2014.

[39] H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes," *IEEE Trans. Evol. Comput.*, vol. 21, no. 2, pp. 169–190, Apr. 2017.

[40] Y. Zhou, Y. Xiang, Z. Chen, J. He, and J. Wang, "A scalar projection and angle-based evolutionary algorithm for many-objective optimization problems," *IEEE Trans. Cybern.*, vol. 49, no. 6, pp. 2073–2084, Jun. 2019.

[41] K. Ikeda, H. Kita, and S. Kobayashi, "Failure of Pareto-based MOEAs: Does non-dominated really mean near to optimal?" in *Proc. IEEE Congr. Evol. Comput.*, Seoul, South Korea, May 2001, pp. 957–962.

[42] R. C. Purshouse and P. J. Fleming, "On the evolutionary optimization of many conflicting objectives," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 770–784, Dec. 2007.

[43] M. Li, S. Yang, and X. Liu, "Bi-goal evolution for many-objective optimization problems," *Artificial Intelligence*, vol. 228, pp. 45–65, Nov. 2015.

[44] H. Ishibuchi, T. Fukase, N. Masuyama, and Y. Nojima, "Effects of dominance resistant solutions on the performance of evolutionary multi-objective and many-objective algorithms," in *Proc. Conf. Genet. Evol. Comput.*, Cancún, Mexico, Jun. 2020, pp. 507–515.

[45] Z. Wang, Y.-S. Ong, and H. Ishibuchi, "On scalable multiobjective test problems with hardly dominated boundaries," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 217–231, Apr. 2019.

[46] M. Li, L. Liu, and D. Lin, "A fast steady-state ϵ-dominance multi-objective evolutionary algorithm," *Comput. Optim. Appl.*, vol. 48, no. 1, pp. 109–138, Jan. 2011.

[47] T. Kondoh, T. Tatsukawa, A. Oyama, T. Watanabe, and K. Fujii, "Effects of discrete design-variable precision on real-coded genetic algorithm," in *IEEE Symp. Ser. Comput. Intell.*, Athens, Greece, Dec. 2016, pp. 1–8.

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

[49] K. S. Bhattacharjee, H. K. Singh, T. Ray, and Q. Zhang, "Decomposition based evolutionary algorithm with a dual set of reference vectors," in *Proc. IEEE Congr. Evol. Comput.*, San Sebastian, Spain, Jun. 2017, pp. 105–112.

[50] L. Chen, K. Deb, H.-L. Liu, and Q. Zhang, "Effect of objective normalization and penalty parameter on penalty boundary intersection decomposition based evolutionary many-objective optimization algorithms," *Evol. Comput.*, pp. 1–25, Jun. 2020.

[51] X. Cai, Z. Mei, Z. Fan, and Q. Zhang, "A constrained decomposition approach with grids for evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 4, pp. 564–577, Aug. 2018.

[52] Y. Tian, R. Cheng, X. Zhang, Y. Su, and Y. Jin, "A strengthened dominance relation considering convergence and diversity for evolutionary many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 331–345, Apr. 2019.

[53] X. He, Y. Zhou, Z. Chen, and Q. Zhang, "Evolutionary many-objective optimization based on dynamical decomposition," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 361–375, Jun. 2018.

[54] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[55] J. Blank, K. Deb, and P. C. Roy, "Investigating the normalization procedure of NSGA-III," in *Proc. Evol. Multi-Criter. Optim.*, vol. 11411, East Lansing, MI, USA, Mar. 2019, pp. 229–240.

[56] Z. Fan, W. Li, X. Cai, H. Li, K. Hu, and H. Yin, "An improved ideal point setting in multiobjective evolutionary algorithm based on decomposition," in *International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration*, Wuhan, China, Dec. 2015, pp. 63–70.

[57] H. Ishibuchi, K. Doi, and Y. Nojima, "Reference point specification in MOEA/D for multi-objective and many-objective problems," in *Proc. Int. Conf. Syst., Man, Cybern.*, Budapest, Hungary, Oct. 2016, pp. 4015–4020.

[58] R. Wang, J. Xiong, H. Ishibuchi, G. Wu, and T. Zhang, "On the effect of reference point in MOEA/D for multi-objective optimization," *Appl. Soft Comput.*, vol. 58, pp. 25–34, Sep. 2017.

[59] K. Deb, S. Chaudhuri, and K. Miettinen, "Towards estimating nadir objective vector using evolutionary approaches," in *Proc. IEEE Congr. Evol. Comput.*, Seattle, Washington, USA, Jul. 2006, p. 643.

[60] H. Fukumoto and A. Oyama, "Impact of estimation method of ideal/nadir points on practically-constrained multi-objective optimization problems for decomposition-based multi-objective evolutionary algorithm," in *IEEE Symp. Ser. Comput. Intell.*, Xiamen, China, Dec. 2019, pp. 2138–2145.

[61] H. Wang, L. Jiao, and X. Yao, "Two_Arch2: An improved two-archive algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 524–541, Aug. 2015.

[62] Y. Xiang, Y. Zhou, M. Li, and Z. Chen, "A vector angle-based evolutionary algorithm for unconstrained many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 1, pp. 131–152, Feb. 2017.

[63] L. Chen, K. Deb, and H.-L. Liu, "Explicit control of implicit parallelism in decomposition-based evolutionary many-objective optimization algorithms," *IEEE Comput. Intell. Mag.*, vol. 14, no. 4, pp. 52–64, Nov. 2019.

[64] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. Int. Conf. Parallel Prob. Solv. Nat.*, vol. 3242, Birmingham, United Kingdom, Sep. 2004, pp. 832–842.

[65] N. Chen, W.-N. Chen, Y.-J. Gong, Z.-H. Zhan, J. Zhang, Y. Li, and Y.-S. Tan, "An evolutionary algorithm with double-level archives for multiobjective optimization," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 1851–1863, Sep. 2015.

[66] K. Li, S. Kwong, Q. Zhang, and K. Deb, "Interrelationship-based selection for decomposition multiobjective optimization," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2076–2088, Oct. 2015.

[67] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 5, pp. 773–791, Oct. 2016.

[68] Q. Lin, G. Jin, Y. Ma, K.-C. Wong, C. A. C. Coello, J. Li, J. Chen, and J. Zhang, "A diversity-enhanced resource allocation strategy for decomposition-based multiobjective evolutionary algorithm," *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2388–2401, Aug. 2018.

[69] X. Cai, Z. Mei, and Z. Fan, "A decomposition-based many-objective evolutionary algorithm with two types of adjustments for direction vectors," *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2335–2348, Aug. 2018.

[70] Y. Qi, Q. Zhang, X. Ma, Y. Quan, and Q. Miao, "Utopian point based decomposition for multi-objective optimization problems with complicated Pareto fronts," *Appl. Soft Comput.*, vol. 61, pp. 844–859, Dec. 2017.

[71] R. Tanabe and H. Ishibuchi, "A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization," in *Proc. Int. Conf. Parallel Prob. Solv. Nat.*, Coimbra, Portugal, Sep. 2018, pp. 249–261.

[72] Y. Tian, R. Cheng, X. Zhang, F. Cheng, and Y. Jin, "An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility," *IEEE Trans. Evol. Comput.*, vol. 22, no. 4, pp. 609–622, Aug. 2018.

[73] D. Sharma, S. Z. Basha, and S. A. Kumar, "Diversity over dominance approach for many-objective optimization on reference-points-based framework," in *Proc. Evol. Multi-Criter. Optim.*, East Lansing, MI, USA, Mar. 2019, pp. 278–290.

[74] A. Habib, H. K. Singh, T. Chugh, T. Ray, and K. Miettinen, "A multiple surrogate assisted decomposition-based evolutionary algorithm for expensive multi/many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 6, pp. 1000–1014, Dec. 2019.

[75] K. Shang and H. Ishibuchi, "A new hypervolume-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, 2020 (Early Access).

[76] A. B. Ruiz, R. Saborido, and M. Luque, "A preference-based evolutionary algorithm for multiobjective optimization: the weighting achievement scalarizing function genetic algorithm," *J. Glob. Optim.*, vol. 62, no. 1, pp. 101–129, May 2015.

[77] J. Luo, X. Sun, Y. Qi, and J. Xie, "Approximating the irregularly shaped Pareto front of multi-objective reservoir flood control operation problem," *Applied Mathematical Modelling*, vol. 54, pp. 502–516, Feb. 2018.

[78] D. Sharma and P. K. Shukla, "Line-prioritized environmental selection and normalization scheme for many-objective optimization using reference-lines-based framework," *Swarm Evol. Comput.*, vol. 51, Dec. 2019.

[79] H. Seada and K. Deb, "A unified evolutionary optimization procedure for single, multiple, and many objectives," *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 358–369, Jun. 2016.

[80] R. Wang, H. Ishibuchi, Y. Zhang, X. Zheng, and T. Zhang, "On the effect of localized PBI method in MOEA/D for multi-objective optimization," in *IEEE Symp. Ser. Comput. Intell.*, Athens, Greece, Dec. 2016, pp. 1–8.

[81] S. Zapotecas Martínez, V. A. Sosa Hernández, H. Aguirre, K. Tanaka, and C. A. Coello Coello, "Using a family of curves to approximate the Pareto front of a multi-objective optimization problem," in *Proc. Int. Conf. Parallel Prob. Solv. Nat.*, vol. 8672, Ljubljana, Slovenia, Sep. 2014, pp. 682–691.

[82] R. Hernández Gómez, C. A. Coello Coello, and E. Alba, "A parallel version of SMS-EMOA for many-objective optimization problems," in *Proc. Int. Conf. Parallel Prob. Solv. Nat.*, Coimbra, Portugal, Sep. 2016, pp. 568–577.

[83] R. Hernández Gómez, C. A. Coello Coello, and E. Alba Torres, "A multi-objective evolutionary algorithm based on parallel coordinates," in *Proc. Conf. Genet. Evol. Comput.*, Denver, Colorado, USA, Jul. 2016, pp. 565–572.

[84] H. Li, J. Sun, Q. Zhang, and Y. Shui, "Adjustment of weight vectors of penalty-based boundary intersection method in MOEA/D," in *Proc. Evol. Multi-Criter. Optim.*, East Lansing, MI, USA, Mar. 2019, pp. 91–100.

[85] I. Das, "Nonlinear multicriteria optimization and robust optimality," PhD Dissertation, Rice University, Houston Texas, Apr. 1997.

[86] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems," *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 631–657, Aug. 1998.

[87] M. Asafuddoula, T. Ray, and R. Sarker, "A decomposition-based evolutionary algorithm for many objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 3, pp. 445–460, Jun. 2015.

[88] M. Elarbi, S. Bechikh, C. A. C. Coello, M. Makhlouf, and L. B. Said, "Approximating complex Pareto fronts with pre-defined normal-boundary intersection directions," *IEEE Trans. Evol. Comput.*, 2019 (Early Access).

[89] H. K. Singh, K. S. Bhattacharjee, and T. Ray, "Distance-based subset selection for benchmarking in evolutionary multi/many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 904–912, Oct. 2019.

[90] A. Panichella, "An adaptive evolutionary algorithm based on non-euclidean geometry for many-objective optimization," in *Proc. Conf. Genet. Evol. Comput.*, Prague, Czech Republic, Jul. 2019, pp. 595–603.

[91] Y. Xiang, Y. Zhou, X. Yang, and H. Huang, "A many-objective evolutionary algorithm with Pareto-adaptive reference points," *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 99–113, Feb. 2020.

[92] Y. Sun, G. G. Yen, and Z. Yi, "IGD indicator-based evolutionary algorithm for many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 173–187, Apr. 2019.

[93] Y. Sun, B. Xue, M. Zhang, and G. G. Yen, "A new two-stage evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 748–761, Oct. 2019.

[94] M. Asafuddoula, T. Ray, and R. Sarker, "A decomposition based evolutionary algorithm for many objective optimization with systematic sampling and adaptive epsilon control," in *Proc. Evol. Multi-Criter. Optim.*, Sheffield, United Kingdom, Mar. 2013, pp. 413–427.

[95] S. Jiang and S. Yang, "Convergence versus diversity in multiobjective optimization," in *Proc. Int. Conf. Parallel Prob. Solv. Nat.*, Coimbra, Portugal, Sep. 2016, pp. 984–993.

[96] S. Ying, L. Li, Z. Wang, W. Li, and W. Wang, "An improved decomposition-based multiobjective evolutionary algorithm with a better balance of convergence and diversity," *Appl. Soft Comput.*, vol. 57, pp. 627–641, Aug. 2017.

[97] H. Seada, M. Abouhawwash, and K. Deb, "Multiphase balance of diversity and convergence in multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 503–513, Jun. 2019.

[98] H. K. Singh, A. Isaacs, and T. Ray, "A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 15, no. 4, pp. 539–556, Aug. 2011.

[99] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, Nov. 2017.

[100] H. Wang and X. Yao, "Corner sort for Pareto-based many-objective optimization," *IEEE Trans. Cybern.*, vol. 44, no. 1, pp. 92–102, Jan. 2014.

[101] M. Asafuddoula, H. K. Singh, and T. Ray, "An enhanced decomposition-based evolutionary algorithm with adaptive reference vectors," *IEEE Trans. Cybern.*, vol. 48, no. 8, pp. 2321–2334, Aug. 2018.

[102] G. Lei, J. G. Zhu, Y. G. Guo, J. F. Hu, W. Xu, and K. R. Shao, "Robust design optimization of PM-SMC motors for six sigma quality manufacturing," *IEEE Trans. Magn.*, vol. 49, no. 7, pp. 3953–3956, Jul. 2013.

[103] M. Asafuddoula, H. K. Singh, and T. Ray, "Six-sigma robust design optimization using a many-objective decomposition-based evolutionary algorithm," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 490–507, Aug. 2015.

[104] R. Hernández Gómez and C. A. Coello Coello, "Improved metaheuristic based on the R2 indicator for many-objective optimization," in *Proc. Conf. Genet. Evol. Comput.*, Madrid, Spain, Jul. 2015, pp. 679–686.

[105] J. G. Falcón-Cardona and C. A. Coello Coello, "A new indicator-based many-objective ant colony optimizer for continuous search spaces," *Swarm. Intell.*, vol. 11, no. 1, pp. 71–100, Mar. 2017.

[106] Y. Qi, J. Yu, X. Li, Y. Quan, and Q. Miao, "Enhancing robustness of the inverted PBI scalarizing method in MOEA/D," *Appl. Soft Comput.*, vol. 71, pp. 1117–1132, Oct. 2018.

[107] H. Sato, "Analysis of inverted PBI and comparison with other scalarizing functions in decomposition based MOEAs," *Journal of Heuristics*, vol. 21, no. 6, pp. 819–849, 2015.

[108] D. K. Saxena and S. Kapoor, "On timing the nadir-point estimation and/or termination of reference-based multi- and many-objective evolutionary algorithms," in *Proc. Evol. Multi-Criter. Optim.*, vol. 11411, East Lansing, MI, USA, Mar. 2019, pp. 191–202.

[109] D. K. Saxena, S. Mittal, S. Kapoor, and K. Deb, "A localized high fidelity dominance based many-objective evolutionary algorithm," COIN Lab, Michigan State University, COIN Report No. 2021002, Jan. 2021.

[110] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 450–455, Jun. 2014.

[111] T. Kohira, H. Kemmotsu, O. Akira, and T. Tatsukawa, "Proposal of benchmark problem based on real-world car structure design optimization," in *Proc. Conf. Genet. Evol. Comput.*, Kyoto, Japan, Jul. 2018, pp. 183–184.

[112] H. Ishibuchi, L. He, and K. Shang, "Regular Pareto front shape is not realistic," in *Proc. IEEE Congr. Evol. Comput.*, Wellington, New Zealand, Jun. 2019, pp. 2034–2041.

[113] R. E. Steuer and R. Steuer, *Multiple Criteria Optimization: Theory, Computation, and Application*. Wiley New York, 1986.

[114] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, and V. da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, Apr. 2003.

[115] R. Cheng and M. Gen, "Evolution program for resource constrained project scheduling problem," in *Proc. IEEE Congr. Evol. Comput.*, vol. 2, Orlando, FL, USA, Jun. 1994, pp. 736–741.

[116] R. Cheng and M. Gen, *Genetic Algorithms and Engineering Design*. John Wiley, 1997.

[117] L. M. Pang, H. Ishibuchi, and K. Shang, "Decomposition-based multi-objective evolutionary algorithm design under two algorithm frameworks," *IEEE Access*, vol. 8, pp. 163 197–163 208, 2020.

[118] Z.-Z. Liu and Y. Wang, "Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 870–884, Oct. 2019.

[119] K. Li, R. Chen, G. Fu, and X. Yao, "Two-archive evolutionary algorithm for constrained multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 303–315, Apr. 2019.

[120] L. He, H. Ishibuchi, A. Trivedi, and D. Srinivasan, "Dynamic normalization in MOEA/D for multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput.*, Glasgow, Scotland, United Kingdom, Jul. 2020.

[121] L. He, K. Shang, and H. Ishibuchi, "Simultaneous use of two normalization methods in decomposition-based multi-objective evolutionary algorithms," *Appl. Soft Comput.*, vol. 92, Jul. 2020.

[122] I. Giagkiozis, R. C. Purshouse, and P. J. Fleming, "Towards understanding the cost of adaptation in decomposition-based optimization algorithms," in *IEEE Int. Conf. Syst., Man, Cybern.*, Manchester, UK, Oct. 2013, pp. 615–620.

[123] J. D. Knowles, "Local-search and hybrid evolutionary algorithms for Pareto optimization," phdthesis, Department of Computer Science, University of Reading, Reading, U.K., Jan. 2002.

[124] E. Zitzler, J. Knowles, and L. Thiele, "Quality assessment of Pareto set approximations," in *Multiobjective Optimization*. Berlin, Heidelberg: Springer, 2008, vol. 5252, pp. 373–404.

[125] E. Zitzler, "Evolutionary algorithms for multiobjective optimization: Methods and applications," PhD Dissertation, Swiss Federal Institute of Technology, Zurich, Switzerland, Nov. 1999.

[126] J. Knowles and D. Corne, "On metrics for comparing nondominated sets," in *Proc. IEEE Congr. Evol. Comput.*, Honolulu, HI, USA, 2002, pp. 711–716.

[127] N. Drechsler, R. Drechsler, and B. Becker, "Multi-objective optimisation based on relation *favour*," in *Proc. Evol. Multi-Criter. Optim.*, Zurich, Switzerland, Mar. 2001, pp. 154–166.

[128] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, Sep. 2007.

[129] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, Mar. 2011.

[130] S. Jiang, J. Zhang, Y.-S. Ong, A. N. Zhang, and P. S. Tan, "A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2202–2213, Oct. 2015.

[131] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," *TIK-report*, vol. 103, 2001.

[132] A. J. Keane, "Statistical improvement criteria for use in multiobjective design optimization," *AIAA Journal*, vol. 44, no. 4, pp. 879–891, Apr. 2006.

[133] K. Shang, H. Ishibuchi, and X. Ni, "R2-based hypervolume contribution approximation," *IEEE Trans. Evol. Comput.*, Apr. 2019 (Early Access).

[134] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "Reference point specification in hypervolume calculation for fair comparison and efficient search," in *Proc. Conf. Genet. Evol. Comput.*, Jul. 2017, pp. 585–592.

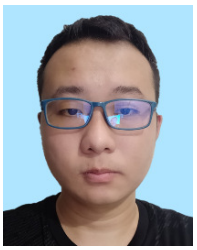
[135] ——, "How to specify a reference point in hypervolume calculation for fair performance comparison," *Evol. Comput.*, vol. 26, no. 3, pp. 411–440, Sep. 2018.

Linjun He received the B.Eng. degree in electrical and electronic engineering from the Southern University of Science and Technology (SUSTech), Shenzhen, China, in July 2018. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. His research interests include evolutionary multi-objective optimization, large-scale optimization, and machine learning.

Hisao Ishibuchi (M'93–SM'10–F'14) received the B.S. and M.S. degrees from Kyoto University in 1985 and 1987, respectively, and the Ph.D. degree from Osaka Prefecture University in 1992. He was with Osaka Prefecture University in 1987-2017. Since 2017, he is a Chair Professor at Southern University of Science and Technology, China. His research interests include fuzzy rule-based classifiers, evolutionary multi-objective and many-objective optimization, memetic algorithms, and evolutionary games. Dr. Ishibuchi was the IEEE CIS Vice-President for Technical Activities in 2010-2013 and the Editor-in-Chief of the IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE in 2014-2019. Currently he is an AdCom member of the IEEE CIS (2014-2019, 2021-2023).

Anupam Trivedi (M'15) received the dual degree (integrated bachelor's and master's) in civil engineering from the Indian Institute of Technology Bombay, Mumbai, India, in 2009, and the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore, in 2015. He is currently a Senior Research Fellow at the National University of Singapore. His research interests include evolutionary algorithms, multi-objective optimization, constrained optimization, and smart grid optimization.

Handing Wang (S'10-M'16) received the B.Eng. and Ph.D. degrees from Xidian University, Xi'an, China, in 2010 and 2015, respectively. She is currently a professor with School of Artificial Intelligence, Xidian University, Xi'an, China. Dr. Wang is an Associate Editor of IEEE Computational Intelligence Magazine, Memetic Computing, and Complex & Intelligent Systems. Her research interests include nature-inspired computation, multiobjective optimization, multiple criteria decision making, surrogate-assisted evolutionary optimization, and real-world problems.



Yang Nan received his B.S. degree in Electrical and Electronic Engineering with the Southern University of Science and Technology (SUSTech), Shenzhen, China, in July 2018. He is currently a research assistant at the Department of Electrical and Computer Engineering in SUSTech. His current research interests include evolutionary multi-objective optimization and its applications.

Dipti Srinivasan (M'89–SM'02) received her M.S. and Ph.D. degrees from National University of Singapore, Singapore, in 1991 and 1994, respectively. She is currently a professor with the Department of Electrical & Computer Engineering, National University of Singapore. Her recent research projects are in the broad areas of optimization and control, wind and solar power prediction, electricity price prediction, deep learning, and development of multi-agent systems for system operation and control.