A Survey of Decomposition-Based Evolutionary Multi-Objective Optimization: Part I—Past and Future

Ke Li[1]

[1]Department of Computer Science, University of Exeter, EX4 4QF, Exeter, UK
✍ k.li@exeter.ac.uk

Abstract: Decomposition has been the mainstream approach in classic mathematical programming for multi-objective optimization and multi-criterion decision-making. However, it was not properly studied in the context of evolutionary multi-objective optimization (EMO) until the development of multi-objective evolutionary algorithm based on decomposition (MOEA/D). In this two-part survey series, we use MOEA/D as the representative of decomposition-based EMO to review the up-to-date development in this area, and systematically and comprehensively analyze its research landscape. In the first part, we present a comprehensive survey of the development of MOEA/D from its origin to the current state-of-the-art approaches. In order to be self-contained, we start with a step-by-step tutorial that aims to help a novice quickly get onto the working mechanism of MOEA/D. Then, selected major developments of MOEA/D are reviewed according to its core design components including weight vector settings, subproblem formulations, selection mechanisms and reproduction operators. Besides, we also overview some selected advanced topics for constraint handling, optimization in dynamic and uncertain environments, computationally expensive objective functions, and preference incorporation. In the final part, we shed some light on emerging directions for future developments.

Keywords: Multi-objective optimization, decomposition, evolutionary computation.

> *"Divide each difficulty into as many parts as is feasible and necessary to resolve it."*
> — *René Descartes*

1 Introduction

Multi-objective optimization problems (MOPs) are ubiquitous in real-world scenarios, spanning a broad spectrum of fields such as machine learning [1], data mining [2], material science [3], cheminformatics [4], aerospace [5], software engineering [6], finance and economics [7]. These problems often involve conflicting objectives, where improving one can compromise others, leading to a set of trade-off solutions known as Pareto-optimal solutions, rather than a single global optimum. Partially due to the implicit parallelism and the population-based nature of evolutionary computation (EC), it has been widely recognized as a major approach for multi-objective optimization (MO). Over the past three decades, significant efforts have been dedicated to developing evolutionary multi-objective optimization (EMO) algorithms. These can be broadly classified into dominance-, indicator-, and decomposition-based frameworks. Representative algorithms include the fast non-dominated sorting genetic algorithm (NSGA-II) [8], the indicator-based evolutionary algorithm (IBEA) [9], and the multi-objective evolutionary algorithm based on decomposition (MOEA/D) [10].

Different from the dominance- and indicator-based frameworks, which are rooted in the EC community, decomposition has been the mainstream in the classic mathematical programming and operations research for MO and multi-criterion decision-making (MCDM) [21]. Its core concept involves aggregating different objectives into a scalarizing function by using a dedicated weight vector, transforming a MOP into a single-objective optimization subproblem. The optimum of this subproblem is a Pareto-optimal solution of the original MOP. Traditional mathematical programming approaches, however, heavily rely on derivative information, which is often notoriously hardly accessible in real-world black-box scenarios. Moreover, in MO, these methods can only obtain one Pareto-optimal solution at a time,
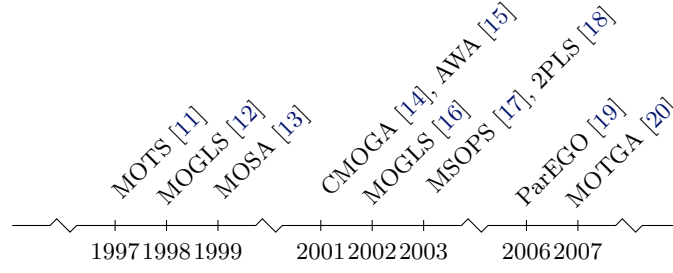
Figure 1. A pragmatic history of decomposition-based EMO algorithms from the past till the development of MOEA/D.

limiting the effectiveness and flexibility in exploring various trade-off alternatives simultaneously. To the best of our knowledge, MOTS [11] is the first that combines the decomposition concept in MCDM with meta-heuristics, tabu search in particular, for MO. As the chronological line shown in Fig. 1, several follow-up works [12–14, 18, 20] use a set of randomly generated weight vectors to navigate the EMO search process. A slightly different idea is in [16, 17, 19] where only a single aggregation function is considered at each iteration and the corresponding weight vector is randomly sampled within a canonical simplex. While [15] also only considers one subproblem at a time, the setting of the under-pinned weight vector is controlled in a predefined manner. In a nutshell, all these earlier algorithms lack a principled way to maintain population diversity due to the random generation of weight vectors, resulting in less effective search process [22]. Further, they fail to consider collaboration among neighboring subproblems, rendering them less capable of tackling problems with complex properties [23]. The decomposition-based EMO is not established, compared to the dominance- and indicator-based counterparts, until the seminal paper of MOEA/D [10].

🔊 **Why is this survey necessary?** There are three compelling reasons for organizing this survey series.

1. As the foremost representative of decomposition-based EMO methods, MOEA/D transcends being merely a dedicated algorithm; it embodies a pragmatic problem-solving paradigm. Its application as a backbone across a wide spectrum of challenging MOPs underscores its significance and versatility. This constitutes our key motivation of using MOEA/D as the central pillar to develop this survey series.

2. While there have been nine survey-style articles published since 2014, only [24] and [25] have offered a comprehensive summary of the development of MOEA/D. Note that these surveys have become somewhat outdated, with some of the topics they highlighted now heavily advanced. In contrast, other articles have focused on more restricted aspects: [26] and [27] concentrated on decomposition methods, particularly subproblem formulation; [28] explored weight vector adjustment methods within MOEA/D; and [29–31] mainly addressed many-objective challenges for MOEA/D.

3. To the best of our knowledge, this survey series is the first to offer a comprehensive analysis of the MOEA/D research landscape from a data science perspective. We have developed a novel data mining pipeline that elucidates the categorization and interconnections of research topics within MOEA/D, alongside a spatial-temporal evolution of these topics. Further, our analysis of citation and collaboration networks offers unprecedented insights into the research community's structure. Our analysis results not only shed light on the structural dynamics of MOEA/D research but also extend insights applicable across the entire EMO community and beyond.

👥 **Who are the audiences?** This survey presents techniques and analysis in a progressive manner, starting with a gentle tutorial before delving into more in-depth analysis. This structured approach makes it accessible to diverse audiences. Newcomers and early-career researchers in the EMO community will find it a valuable resource for acquiring a high-level understanding of the EMO domain and identifying pressing research questions for their endeavors. Further, experienced EMO experts are provided with insights into opportunities for creating new research frontiers. Further, the analytic
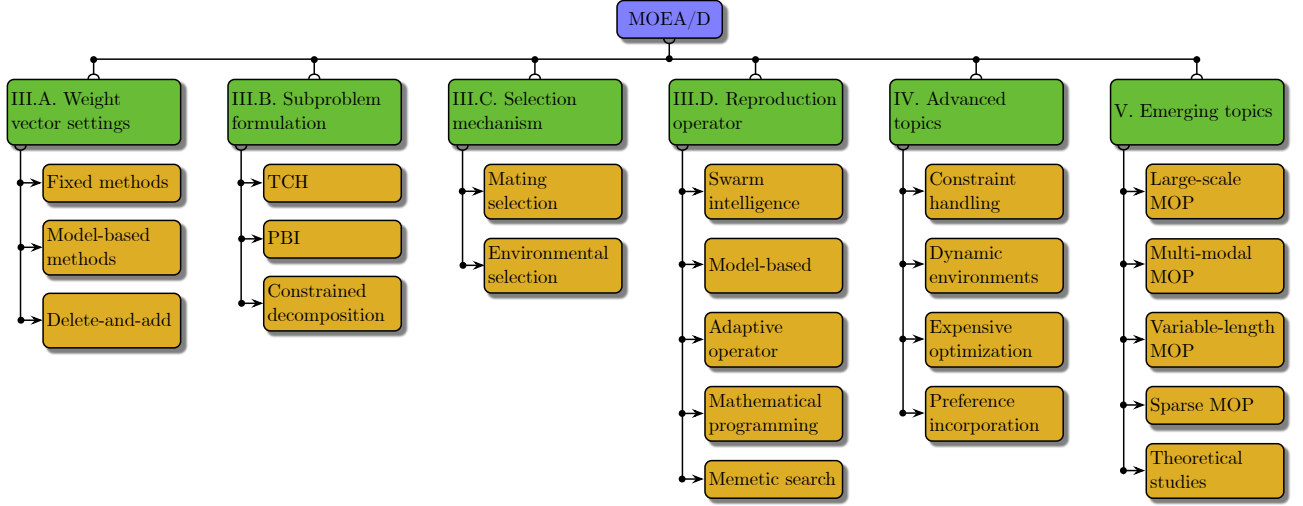
MOEA/D

III.A. Weight vector settings	III.B. Subproblem formulation	III.C. Selection mechanism	III.D. Reproduction operator	IV. Advanced topics	V. Emerging topics
Fixed methods	TCH	Mating selection	Swarm intelligence	Constraint handling	Large-scale MOP
Model-based methods	PBI	Environmental selection	Model-based	Dynamic environments	Multi-modal MOP
Delete-and-add	Constrained decomposition		Adaptive operator	Expensive optimization	Variable-length MOP
			Mathematical programming	Preference incorporation	Sparse MOP
			Memetic search		Theoretical studies

Figure 2. Outline of the topics covered in the PART I of this survey series on decomposition-based EMO.

results from our data mining offer added value across different parties in the academic community, aiding researchers, journal editors, and funding agencies in identifying promising research topics and projects, selecting appropriate journals for submissions, and guiding strategic development focus.
⊹ **Outline:** This survey series comes in two parts.

- PART I focuses on methodological development within the MOEA/D framework where Fig. 2 provides an outline of topics covered therein. Section 2 begins with a pragmatic tutorial, providing background knowledge pertinent to this survey and making it self-contained. Thereafter, major developments regarding weight vector settings, subproblem formulations, selection mechanisms, and reproduction operators are discussed in Section 3, according to the core algorithmic components of MOEA/D. Then, Section 4 and Section 5 reviews selected advanced topics and future directions, respectively, which are not exclusive to MOEA/D but are relevant to the wider EMO community. Finally, Section 6 concludes this survey.

- PART II[1] focuses on the data analytic regarding the current research related to MOEA/D. We will introduce a novel data-driven scientific literature mining platform that acts as the technological foundation to guide data collection, data cleaning, and exploratory data analysis. We believe our data analysis results will offer unprecedented insights into the research community's structure.

2 Background Knowledge

This section starts with some basic concepts pertinent to this survey, followed by a gentle tutorial on the working mechanisms of a vanilla MOEA/D.

2.1 Basic Concepts

The MOP[2] considered in this paper is defined as:

$$
\begin{aligned}
\underset{\mathbf{x}\in\Omega}{\text{minimize}} \quad & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}),\ldots,f_m(\mathbf{x}))^\top \\
\text{subject to} \quad & \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \\
& \mathbf{h}(\mathbf{x}) = \mathbf{0}
\end{aligned}
\tag{1}
$$

where $\Omega = [x_i^{\mathrm{L}}, x_i^{\mathrm{U}}]_{i=1}^n \subseteq \mathbb{R}^n$ defines the search (or decision variable) space, and $\mathbf{x} = (x_1,\ldots,x_n)^\top$ is a candidate solution therein. $\mathbf{F} : \Omega \to \mathbb{R}^m$ constitutes m conflicting objective functions, and $\mathbb{R}^m$

---

[1]The article of PART II can be downloaded from this link, while the APPENDIX file can be found in this link
[2]Note that the definitions herein are interchangeable between minimization and maximization problems.

is the objective space. $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), \ldots, g_p(\mathbf{x}))^\top$ and $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}), \ldots, h_q(\mathbf{x}))^\top$ consists of $p \geq 1$ inequality and $q \geq 1$ equality constraints, respectively. A solution $\mathbf{x} \in \Omega$ is _feasible_ if and only if $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$ and $\mathbf{h}(\mathbf{x}) = \mathbf{0}$.

Definition 1. _Given two feasible solutions $\mathbf{x}^1$ and $\mathbf{x}^2$, $\mathbf{x}^1$ is said to _Pareto dominate_ $\mathbf{x}^2$ (denoted as $\mathbf{x}^1 \preceq \mathbf{x}^2$) if and only if $f_i(\mathbf{x}^1) \leq f_i(\mathbf{x}^2)$, $\forall i \in \{1, \ldots, m\}$ and $\exists j \in \{1, \ldots, m\}$ such that $f_i(\mathbf{x}^1) < f_i(\mathbf{x}^2)$._

Definition 2. _A feasible solution $\mathbf{x}^*$ is _Pareto-optimal_ with respect to (1) if $\nexists \mathbf{x} \in \Omega$ with $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$ and $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ such that $\mathbf{x} \preceq \mathbf{x}^*$._

Definition 3. _The set of all Pareto-optimal solutions is called the _Pareto-optimal set_ (PS). Accordingly, $PF = \{\mathbf{F}(\mathbf{x}) | \mathbf{x} \in PS\}$ is called the _Pareto-optimal front_ (PF)._

Definition 4. _The _ideal objective vector_ $\mathbf{z}^{\mathrm{id}} = (z_1^{\mathrm{id}}, \ldots, z_m^{\mathrm{id}})^\top$ is constructed by the minimum of each objective function, i.e., $z_i^{\mathrm{id}} = \min\limits_{\mathbf{x} \in \Omega} f_i(\mathbf{x})$, $i \in \{1, \ldots, m\}$._

Definition 5. _The _nadir objective vector_ $\mathbf{z}^{\mathrm{nad}} = (z_1^{\mathrm{nad}}, \ldots, z_m^{\mathrm{nad}})^\top$ is constructed by the worst objective functions of the PF, i.e., $z_i^{\mathrm{nad}} = \max\limits_{\mathbf{x} \in PS} f_i(\mathbf{x})$, $i \in \{1, \ldots, m\}$._

2.2 Tutorial of Vanilla MOEA/D

Generally speaking, a MOEA/D consists of two major functional components, i.e., _decomposition_ and _collaboration_. Their working mechanisms are briefly introduced as follows.

2.2.1 Decomposition

Under some mild continuous conditions, a Pareto-optimal solution of a MOP is an optimal solution of a scalar optimization problem whose objective function is an aggregation of all individual objectives [21]. The principle of decomposition is to transform the task of approximating the PF into a number of scalarized subproblems. There have been various approaches for constructing such subproblems [21], which will be further discussed in Section 3.2. Here we introduce the three most widely used ones within the MOEA/D framework.

- _Weighted sum (WS)_: it is a convex combination of all individual objectives. Let $\mathbf{w} = (w_1, \ldots, w_m)^\top$ be a weight vector where $w_i \geq 0$, $\forall i \in \{1, \ldots, m\}$ and $\sum_{i=1}^{m} w_i = 1$, the WS is formulated as:

$$\min_{\mathbf{x} \in \Omega} g^{\mathrm{ws}}(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i f_i(\mathbf{x}). \qquad (2)$$

- _Weighted Tchebycheff (TCH)_: in this approach, the scalar optimization problem is formulated as:

$$\min_{\mathbf{x} \in \Omega} g^{\mathrm{tch}}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = \max_{1 \leq i \leq m} \left\{ w_i |f_i(\mathbf{x}) - z_i^*| \right\}, \qquad (3)$$

where $\mathbf{z}^*$ is the approximated ideal objective vector estimated by using the current population.

- _Penalty-baed boundary intersection (PBI)_: this is a variant of the normal-boundary intersection method [32], whose equality constraint is handled by a penalty function. Formally, it is defined as:

$$\min_{\mathbf{x} \in \Omega} g^{\mathrm{pbi}}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = d_1 + \theta d_2, \qquad (4)$$

where $\theta > 0$ is a user-defined penalty parameter, $d_1 = \|(\mathbf{F}(\mathbf{x}) - \mathbf{z}^*)^\top \mathbf{w}\| / \|\mathbf{w}\|$ and $d_2 = \|\mathbf{F}(\mathbf{x}) - (\mathbf{z}^* + d_1 \mathbf{w})\|$.

Note that MOEA/D applies the Das and Dennis's method [32] to specify a set of weight vectors evenly distributed along a unit simplex. Its basic idea is to divide each coordinate into $H > 0$ equally spaced segments. The weight vectors are constituted by picking up a sliced coordinate along each axis in an iterative manner. In total, this method can generate $\binom{H+m-1}{m-1}$ weight vectors, each of which corresponds to a unique subproblem in MOEA/D. Fig. A1 of the APPENDIX gives an example that illustrates 21 weight vectors generated in a three-dimensional space when $H = 5$.
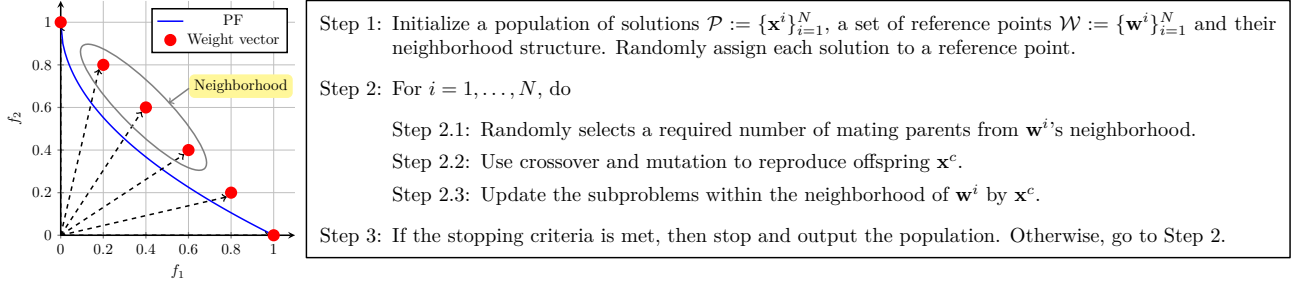
Figure 3. The left panel provides an illustrative example of the weight vector distribution and the neighborhood structure in MOEA/D. The right panel gives the working mechanism of MOEA/D as a three-step process.

2.2.2 Collaboration

Different from the classic mathematical programming method for MO, in which only one subproblem is considered at a time, MOEA/D applies a population-based meta-heuristic approach to concurrently solve all subproblems in a collaborative manner. Each subproblem is associated with its best-so-far solution during the evolutionary problem-solving process. The basic assumption behind this collaboration mechanism is that neighboring subproblems are more likely to share similar properties, e.g., similar objective functions and/or optimal solutions. In particular, the neighborhood of a subproblem is determined by the Euclidean distance of its corresponding weight vector with respect to the others. By this means, solutions associated with the suproblems lying in the same neighborhood can collaborate with each other to push the evolution forward by sharing and exchanging elite information. Specifically, the mating parents are picked up from the underlying neighborhood (as depicted in the left panel of Fig. 3) for offspring reproduction while the newly generated offspring is used to update the subproblem(s) within the same neighborhood. Note that the solution associated with a subproblem can be updated/replaced by the newly generated offspring if and only if it has a better fitness value.

As in the right panel of Fig. 3, the skeleton of MOEA/D consists of four algorithmic components, including the setting of weight vectors, the subproblem formulation, the mating or environmental selection, and the offspring reproduction. In the next section, our discussion is mainly guided by the primary studies dedicated to one or multiple pieces of these above algorithmic components within the MOEA/D framework.

3 Selected Developments on Major Components of MOEA/D

In this section, we review some selected developments in the four major components of MOEA/D.

3.1 Weight Vector Setting

In MOEA/D, weight vectors act as carriers for subproblems, with their configuration being crucial for achieving a well-balanced convergence and diversity [33]. However, employing evenly distributed weight vectors may prove detrimental for PFs that are not regular simplexes, exhibiting disconnected, badly scaled, or strongly convex shapes, as illustrated in Fig. 4. To address this issue, a significant body of work has focused on proactively adapting weight vectors. This subsection reviews selected developments in this area, categorized by the type of heuristics used for adaptation, as summarized in Table A1 of the APPENDIX.

3.1.1 Fixed methods

Rather than progressively tuning weight vectors, this approach employs a fixed strategy to set the weight vectors based on certain regularities *a priori*. The simplest way is random sampling along the canonical simplex. For instance, MOGLS [12, 16, 34] guides a local search on an elite solution with a
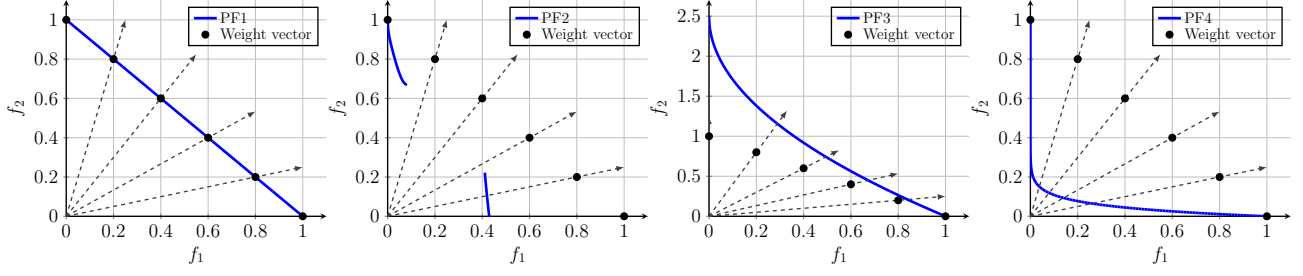
Figure 4. MOEA/D obtains evenly distributed solutions by using evenly distributed weight vectors merely when the PF is a simplex. Otherwise, some weight vectors do not have Pareto-optimal solutions like PF2; or the distribution of the corresponding Pareto-optimal solutions is highly biased like PF3 and PF4.

randomly generated weight vector for each generation. In [35], evenly distributed weight vectors are switched to a random sampling when the search stagnates.

Beyond pure randomness, principled methods from optimal experimental design, such as uniform design [36–38], have been utilized for weight vector generation. Deb and Li et al. [39, 40] introduced a two-layer method to enhance weight vector distribution within the canonical simplex for many-objective optimization. Jin et al. [15] proposed a periodical variation method for two-objective problems, where a weight vector $\mathbf{w}(t) = (w_1(t), w_2(t))^\top$ changes gradually with the evolution process as $w_1(t) = |\sin(2\pi t)/F|, w_2(t) = 1.0 - w_1(t)$, with F controlling the change frequency. In [41], the Riesz s-energy metric was used to iteratively generate well-spaced weight vectors in high-dimensional spaces.

3.1.2 Model-based adaptation methods

The basic concept behind model-based adaptation methods in MOEA/D is to proactively estimate the PF shape by learning from the evolutionary population or an archive, and then resample weight vectors from this estimated PF. These methods fall into two main categories based on their modeling approach.

The first research direction changes the PF assumption from a simplex format, $\sum_{i=1}^{m} f_i(\mathbf{x}) = 1$, to a parameterized polynomial format, $\sum_{i=1}^{m} f_i^p(\mathbf{x}) = 1$. Practical implementations estimate the exponential term p either by solving a nonlinear least square problem [42] or by maximizing Hypervolume [43]. However, this approach struggles with complex PF shapes, thus [44] proposed an asymmetric Pareto-adaptive scheme using a curve model for two-objective PFs. DMOEA/D [45] addresses disconnected PF segments with piecewise hyperplanes estimated from non-dominated solutions.

In contrast to fixed assumptions, another research strand employs parametric models learned from data, such as the evolutionary population. For example, in [46] and [47], Gaussian process regression was employed to model the PF under a general weighted polynomial assumption. This approach involves initially sampling a large number of weight vectors from the model, then refining the sample set by removing dominated or high-variance predictions. This method effectively filters out samples in disconnected regions or beyond the PF. More recently, Wang et al. [48] proposed a cooperative strategy to alternate between global and local weight vectors, thus improving the distribution and uniformity of the population on complicated bi-objective optimization problems. MOEA/D-SOM uses the self-organizing map (SOM) [49] to learn a latent model of the PF and the topological properties are preserved through the use of neighborhood functions [50]. A set of uniformly distributed weight vectors is thereafter sampled from the SOM model. Later, the growing neural gas model [51], a variant of SOM, is applied in a similar manner to facilitate the weight vector generation in many-objective optimization [52, 53]. Wang et al. [54] proposed using a supervised self-organizing network LVQ to model the distribution of weight vectors in the feasible region(s).

3.1.3 Delete-and-add methods

They focus on local tuning by removing invalid weight vectors—those in regions without feasible solutions or overly crowded areas—and adding new ones in promising regions. A representative approach is MOEA/D-AWA [55], which periodically deletes weight vectors from crowded regions, guided by an external archive, and introduces new ones in sparser areas. This strategy aims to improve exploration in many-objective optimization problems by initially generating weight vectors randomly [56, 57] and focusing on under-exploited areas [58, 59].

Contrary to targeting sparse regions, some methods generate new weight vectors away from current promising areas. RVEA*, for example, partitions the population and replaces associated weight vectors of empty subpopulations with unit vectors generated within specific ranges [60]. This approach aims to diversify exploration by introducing variability in the weight vector distribution. Another strategy involves generating new weight vectors within a regulated neighborhood, such as creating a simplex with neighboring weight vectors for each crowded one [61–63]. This method seeks to maintain a balance between exploration and exploitation by focusing on less crowded but potentially promising areas. Some methods reset weight vectors based on the distribution of non-dominated solutions within local niches, directly using archive solutions to adjust weight vectors [64–66]. It enhances the algorithm's ability to adapt to the evolving distribution of solutions, aiming for a more targeted exploration in the objective space.

> 🎯 Key bottlenecks of the current weight vector adaptation methods include: ▶ Fixed methods are notorious for being too rigid to accommodate versatile PF shapes in the real world. ▶ While both model-based adaptation and delete-and-add methods are designed to adapt to the PF shape on the fly, their success heavily depends on the effectiveness of the evolutionary process itself. There is a risk that the weight vector adaptation may be compromised by the populations trapped in local optima or deceptive regions of the search space. ▶ Due to curse-of-dimensionality, representing a high-dimensional space with a limited set of weight vectors is highly challenging, while evolving with a huge number of weight vectors significantly increases computational demands.

3.2 Subproblem Formulations

Subproblem formulation determines the search behavior of MOEA/D. Most subproblem formulations used in MOEA/D are derived from the three traditional ones introduced in Section 2.2.1. In this paper, our focus, as the selected works summarized in Table A2 of the Appendix, is on the shape of the contour line and the size of the improvement region, which defines areas with superior solutions. Specifically, a larger improvement region indicates stronger selection pressure towards convergence, whereas a smaller one emphasizes diversity.
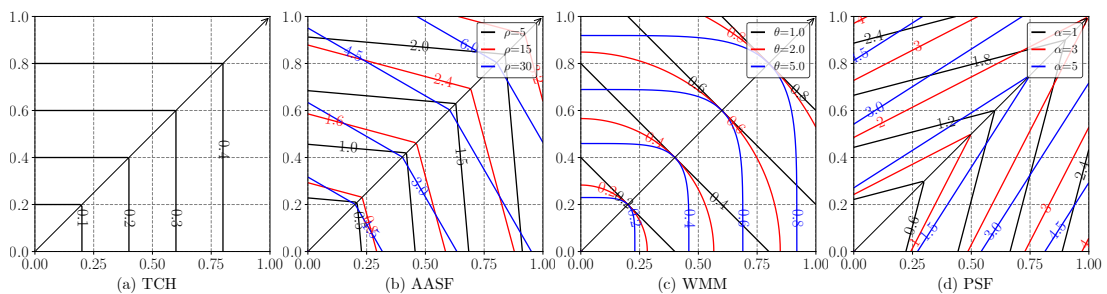
3.2.1 TCH variants

Figure 5. Contour lines of the Tchebycheff approach and its three variants.

The shape of the contour line in the TCH approach and its variants primarily depends on the distance measure relative to the ideal point. As illustrated in Fig. 5(a), the Tchebycheff distance,

utilized in the original TCH approach, generates a contour line that mirrors the Pareto dominance relation. This resemblance underpins the TCH approach's asymptotic equivalence to the Pareto dominance relation in identifying superior solutions [67]. Despite the TCH approach not being limited by the convexity of the PF, unlike the WS approach, it struggles to distinguish between weakly dominated solutions. A direct alternative to overcome this limitation is the implementation of an augmented achievement scalarizing function (AASF) [21], defined as:

$$\min_{\mathbf{x}\in\Omega} \quad g^{\mathrm{aasf}}(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m}\left\{\frac{f_i(\mathbf{x})-z_i^*}{w_i}\right\} + \rho\sum_{i=1}^{m}\left\{\frac{f_i(\mathbf{x})-z_i^*}{w_i}\right\}, \tag{5}$$

where $\rho > 0$ is a parameter that adjusts the opening angle of the contour line, as demonstrated in SectionFig. 5(b). However, setting the ρ value in practice to control the improvement region lacks a thumb rule. Another notable application of the AASF is found in [68], where an adversarial decomposition approach is proposed to capitalize on the complementary characteristics of the AASF and conventional PBI approaches.

Another critique of the TCH approach concerns its crisp improvement region, which can lead to a loss of population diversity. To address this problem, a natural idea is to adapt the shape of the contour line to each local niche of the corresponding subproblem. For instance, Wang et al. [69, 70] explored the characteristics of a family of weighted metrics methods within the MCDM community, defined as follows:

$$\min_{\mathbf{x}\in\Omega} \quad g^{\mathrm{wmm}}(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \left(\sum_{i=1}^{m} w_i|f_i(\mathbf{x})-z_i^*|^{\theta}\right)^{\frac{1}{\theta}}, \tag{6}$$

where $\theta \ge 1$ controls the geometric characteristics of the contour lines, as depicted in Fig. 5(c). By varying θ within a predefined set, the improvement region can be fine-tuned for each subproblem. Expanding on this concept, Jiang et al. [71] introduced a multiplicative scalarizing function (MSF), further refining the adjustment of contour shapes to balance convergence and diversity:

$$\min_{\mathbf{x}\in\Omega} \quad g^{\mathrm{msf}}(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \frac{[\max_{1\le i\le m}(\frac{1}{w_i}|f_i(\mathbf{x})-z_i^*|)]^{1+\alpha}}{[\min_{1\le i\le m}(\frac{1}{w_i}|f_i(\mathbf{x})-z_i^*|)]^{\alpha}}, \tag{7}$$

where α is a parameter that controls the shape of the contours of the MSF, which becomes the TCH when $\alpha = 0$ while it overlaps with the corresponding weight vector when $\alpha = +\infty$. To have an adaptive control of α, [71] suggested that:

$$\alpha^i = \beta(1 - g/G_{\max})\{m \times \min_{1<j<m}(w_j^i)\}, \tag{8}$$

where α^i is the α value associated with the i-th subproblem. Meanwhile, the authors developed another TCH variant called penalty-based scalarizing function (PSF), the contour lines of which are given in Fig. 5(d), and it is defined as:

$$\min_{\mathbf{x}\in\Omega} \quad g^{\mathrm{psf}}(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m}(\frac{1}{w_i}|f_i(\mathbf{x})-z_i^*|) + \alpha d, \tag{9}$$

where d is the same as d_2 in the PBI approach and α is a parameter that controls the balance between convergence and diversity. In particular, a larger α value prioritizes the diversity. In practice, the authors proposed to linearly decrease α with the evolution progress and it becomes zero at the end. This makes both MSF and PSF gradually degenerate to TCH. Thus, the corresponding improvement region for each subproblem is accordingly enlarged leading to a more emphasis on the convergence and relaxation of the diversity.
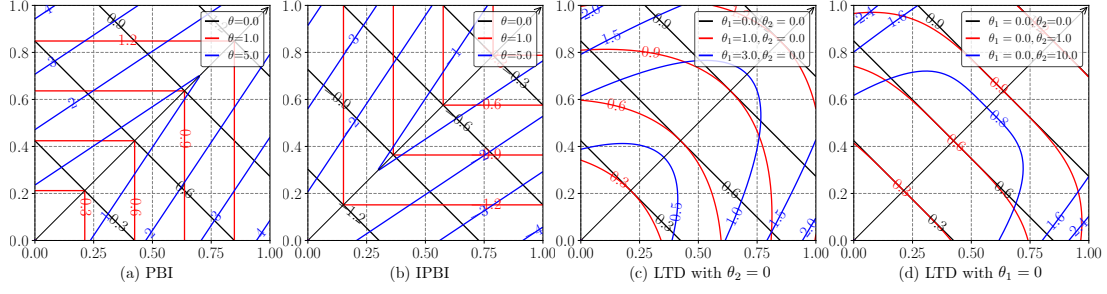
Figure 6. Contour lines of the PBI approach and its three variants.

3.2.2 PBI variants

As a parameterized combination of two distance measures, the improvement region of the PBI approach is controlled by θ in equation (4). As shown in Fig. 6(a), it becomes the WS approach when $\theta = 0.0$ and the TCH approach when $\theta = 1.0$. Its improvement region becomes smaller with the increase of θ. One typical idea of the PBI variants is largely about the adaptation of θ with respect to the shape of the PF [72, 73]. For example, Yang et al. [72] proposed two θ adaptation methods. One is called the adaptive penalty scheme (APS) that gradually increases the θ value with the progression of evolution:

$$\theta = \theta_{\min} + (\theta_{\max} - \theta_{\min})\frac{t}{t_{\max}}, \tag{10}$$

where t is the generation count and $t_{\max}$ is the maximum number of generations. $\theta_{\min} = 1.0$ and $\theta_{\max} = 10.0$ are the lower and upper boundary of θ, respectively. The basic idea of APS is to promote convergence during the early stage of evolution while gradually shifting to diversity. The alternative way, called a subproblem-based penalty scheme (SPS), is to assign an independent θ value for each subproblem:

$$\theta_i = e^{ab_i}, b_i = \max_{1 \leq j \leq m} \left\{ w_i^j \right\} - \min_{1 \leq j \leq m} \left\{ w_i^j \right\}, \tag{11}$$

where θ_i means a θ value for the i-th subproblem in PBI and w_i^j is the j-th element for the i-th weight vector. $a > 0$ is a scaling factor that controls the magnitude of the penalty.

Instead of tweaking the penalty term in the PBI approach, Sato proposed an inverted PBI (IPBI) [74, 75]:

$$\min_{\mathbf{x} \in \Omega} \quad g^{\mathrm{ipbi}}(\mathbf{x}|\mathbf{w}, \mathbf{z}^{\mathrm{nad}}) = d_1^{\mathrm{nad}} - \theta d_2^{\mathrm{nad}}, \tag{12}$$

where $d_1^{\mathrm{nad}} = \|\mathbf{z}^{\mathrm{nad}} - \mathbf{F}(\mathbf{x})\mathbf{w}\|/\|\mathbf{w}\|$ and $d_2^{\mathrm{nad}} = \|\mathbf{z}^{\mathrm{nad}} - (\mathbf{F}(\mathbf{x} + d_1^{\mathrm{nad}}\frac{\mathbf{w}}{\|\mathbf{w}\|}))\|$. Its basic idea is to push the solution away from $\mathbf{z}^{\mathrm{nad}}$ as much as possible. From the examples shown in Fig. 6(b), we find that the IPBI approach essentially shares similar characteristics as the conventional PBI approach but it can search a wider range of the objective space. In [47], Wu et al. proposed an augmented format of the PBI approach as:

$$\min_{\mathbf{x} \in \Omega} y(\mathbf{x}|\mathbf{n}^*, \mathbf{z}^*) = h(\overline{\mathbf{F}}(\mathbf{x})|\mathbf{n}^*, \mathbf{z}^*) = d_1 + \theta_1 d_2^2 + \theta_2 d_2^4, \tag{13}$$

where $\theta_1 > 0$ and $\theta_2 > 0$ are parameters that control the shape and distribution of the opening angle and curvature contours and thus search behaviors of the underlying subproblem as shown in Fig. 6. In practice, θ_1 and θ_2 are set according to the manifold structure of the underlying PF estimated by a Gaussian process regression model.

3.2.3 Constrained Decomposition

Instead of working on the entire search space, constrained decomposition (CD) is to restrict the improvement region of each subproblem to be a local niche [76]. This idea is similar to some TCH and
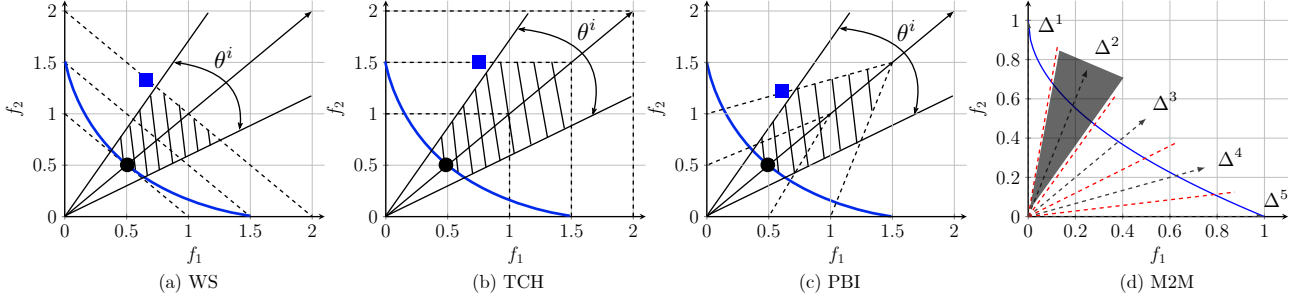
Figure 7. Contour lines of different variants CD (black circle is not comparable with the blue square due to the constrained angle θ^i).

PBI variants (e.g., [47, 69, 70, 72, 73]) that applied a dedicated parameterization on each subproblem. Specifically, a constrained optimization subproblem in CD is defined as:

$$\begin{aligned}
\min \quad & g(\mathbf{x}|\mathbf{w}^i, \mathbf{z}^*) \\
\text{subject to} \quad & \mathbf{x} \in \Omega, \langle \mathbf{w}^i, \mathbf{F}(\mathbf{x}) - \mathbf{z}^* \rangle \leq 0.5\theta^i
\end{aligned} \tag{14}$$

where $g(\cdot)$ can be any subproblem formulation and θ^i controls the improvement region of the i-th subproblem. Based on this definition, the one that meets the constraint is superior to the violated one(s). If both solutions meet the constraint or not, they are compared based on the $g(\cdot)$ function. As depicted in Fig. 7(a) to (c), the improvement region of the constrained subproblem is much smaller than its conventional version, enabling a better population diversity. In [77], a local constraint is imposed for WS to come up with a localized WS:

$$\begin{aligned}
\min \quad & g(\mathbf{x}|\mathbf{w}^i, \mathbf{z}^*) \\
\text{subject to} \quad & \mathbf{x} \in \Omega, \langle \mathbf{w}^i, \mathbf{F}(\mathbf{x}) \rangle \leq \omega^i
\end{aligned} \tag{15}$$

where $\omega^i = \sum_{j=1}^m \frac{\theta^{ij}}{m}$ is the constraint for the i-th subproblem and θ^{ij} is an acute angle between the i-th weight vector and its j-th closest weight vector. In particular, $g(\mathbf{x}^*|\mathbf{w}^i, \mathbf{z}^*)$ is set to infinity when $\langle \mathbf{w}^i, \mathbf{F}(\mathbf{x}^*) \rangle > \omega^i$, i.e., $\mathbf{x}^*$ is out of comparison when consider the ω^i constraint. Each constraint imposes a hypercone region regarding each subproblem, similar to the effect of the example in Fig. 7(a). By doing so, we can expect to make solution(s) lying in the non-convex region survive in the update procedure when using the localized WS approach.

Instead of decomposing the original MOP into a scalar optimization problem, Liu et al. [78] proposed MOEA/D-M2M that decomposes a MOP into $K > 1$ simplified MOPs where the i-th one is defined as:

$$\begin{aligned}
\min \quad & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))^\top \\
\text{subject to} \quad & \mathbf{x} \in \Omega, \mathbf{F}(\mathbf{x}) \in \Delta^i
\end{aligned} \tag{16}$$

where $\Delta^i = \{\mathbf{F}(\mathbf{x}) \in \mathbb{R}^m | \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^i \rangle \leq \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^j \rangle\}$, $j \in \{1, \ldots, N\}$ and $\langle \mathbf{F}(\mathbf{x}), \mathbf{w} \rangle$ is the acute angle between $\mathbf{F}(\mathbf{x})$ and $\mathbf{w}$. Fig. 7(d) provides an illustrative example of five simplified MOPs in a two-objective space. In practice, each simplified MOP is solved by NSGA-II collaboratively. Since each subregion is enforced with a population of solutions, MOEA/D-M2M naturally strikes a balance between convergence and diversity. The effectiveness of this M2M decomposition approach is validated on problems with some challenging bias towards certain regions of the PF as well as those with disconnected segments [79, 80].

💣 Key bottlenecks of the existing subproblem formulations include: ▶ All existing subproblem formulations within MOEA/D are parametric. Thereby, their search behaviors are influenced by specific parameter settings. However, there is no rule of thumb for selecting the optimal parameters for black-box problems. ▶ Attempts to adaptively set parameters based on the current population lead to the risk of biasing the search towards premature convergence. ▶ Further, the varying difficulties of different subproblems present another challenge—optimally allocating computational

3.3   Selection Mechanisms in MOEA/D

Selection in MOEA/D plays a crucial role in two key processes—*mating selection*, which determines the selection of high-quality parents for offspring reproduction, and *environmental selection*, which ensures the survival of the fittest and guides the evolution. Table A3 of APPENDIX summarizes related works on both selection mechanisms.
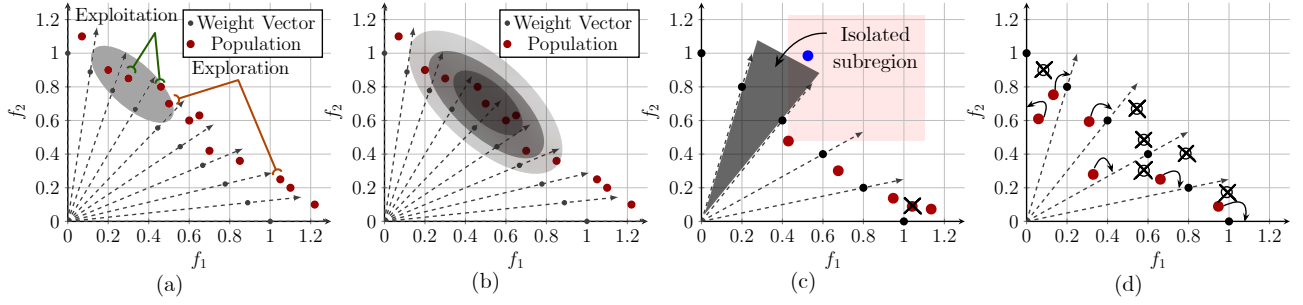

Figure 8. Illustrative examples of selected mating selection mechanisms and environmental selection mechanisms in MOEA/D. (a) green and red lines represent mating selection within and outside the neighborhood, respectively. (b) three different neighborhood sizes are shaded in different colors. (c) the blue point is dominated but lies in an isolated subregion. (d) the eliminated solutions are marked with crosses.

3.3.1   Mating selection

In MOEA/D [10], the mating selection is restricted to the neighborhood of each subproblem. This is based on the assumption that neighboring subproblems share certain structural similarities thus their corresponding solutions are expected to help each other. However, this mechanism is too exploitive, leading to a risk of being trapped by local optima. A simple strategy, as depicted in Fig. 8(a), is to allow mating parents to be picked up from the entire population with a low probability [23]. Differently, [81] proposed taking the local crowdedness into consideration when selecting mating parents. If a neighborhood is associated with too many solutions, the mating parents will be selected outside of this neighborhood. Likewise, [82] proposed a self-adaptive mating restriction strategy that dynamically sets the mating selection probability according to the estimated quality of solutions associated with each subproblem. In particular, if a subproblem is frequently updated, it is assumed to be associated with a poor solution. Accordingly, it is assigned with a small mating selection probability and vice versa.

Since mating selection is mainly implemented within the neighborhood of each subproblem, it is anticipated that the size of each neighborhood can influence the balance of exploration versus exploitation (EvE). Instead of using a constant neighborhood size for each subproblem, [83] and [84] proposed to maintain an ensemble of neighborhood size settings that compete with each other to implement an adaptive EvE balance, as depicted in Fig. 8(b). Instead of using the neighborhood of each subproblem, an alternative is to use the neighborhood of solutions in the mating selection, such as clusters of solutions by using k-means algorithm [85, 86] or solutions close to each other in the objective space [87].

3.3.2   Environmental selection

MOEA/D uses a steady-state environmental selection mechanism that the offspring solution is directly used to update the parent population right after its generation. The offspring can replace as many parent solutions as it can within the mating neighborhood. Such a greedy strategy is detrimental to the population diversity [23]. To mitigate this, it uses the same strategy as in the mating selection

to give a small chance to update the subproblems outside of the mating neighborhood. To give a more robust fitness regarding a neighboring area, [88] proposed an ensemble fitness ranking method to assign fitness value to each solution. It aggregates the ranks of scalarizing function values for a group of subproblems with regard to a solution.

Different from most environmental selection mechanisms that always prioritize the survival of non-dominated solutions against dominated ones, MOEA/DD [40] is the pioneer that suggests emphasizing the survival of solutions lying in isolated regions with few neighbors even if they are dominated or infeasible ones, as depicted in Fig. 8(c). The unpinned assumption is that these regions are likely to be under-exploited. Therefore, they are beneficial to the population diversity and help mitigate the risk of being trapped by local optima. Later, the importance of maintaining isolated solutions regardless of their convergence and feasibility is empirically validated in [89]. Inspired by MOEA/DD, there have been many further attempts that try to allocate certain survival chances to solutions that are relatively inferior in terms of convergence but contribute more in terms of population diversity, e.g., [90–93]. In addition, MOEA/DD is an early work that aims to leverage the complementary effect of dominance- and decomposition-based selection strategies. Along this line, there have been some variants developed to exploit the benefits of combining various selection strategies under the same paradigm to achieve a better trade-off between convergence and diversity [94].

Instead of a steady-state scheme, [95, 96] proposed the first generational environmental selection mechanism for MOEA/D. Its basic idea is to transform the environmental selection process as a matching problem among subproblems and solutions, where subproblems aim to find their appropriate solutions and vice versa, as depicted in Fig. 8(d). In particular, the preference of subproblems is convergence while that of solutions is diversity. A stable matching among subproblems and solutions naturally achieves a balance between convergence and diversity. This stable matching-based selection mechanism inspires a series of follow-up MOEA/D variants that leverage the matching relationship between subproblems and solutions. For instance, [97] proposed another generational environmental selection mechanism that first builds the interrelationship among subproblems and solutions. Each solution is first attached to its most matching subproblem. Thereafter, the fittest solutions from those nonempty subproblems are chosen to survive to the next round while the remaining slot is filled with some less favorable solutions. This idea is extended as a global replacement mechanism that first finds the most matching subproblem for each offspring solution before updating the parent population [76, 98–101], a local niche competition [102, 103] and a correlative selection [104]. Meanwhile, relevant MOEA/D variants have been applied for adversarial robustness in neural networks [105–110].

> ☄* Selection, i.e., survival of the fittest, is one of the most critical steps in both natural evolution and EC. In a nutshell, the major target of both mating and environmental selections is to achieve an EvE dilemma and also a balance between convergence and diversity. Key bottlenecks of the selection mechanisms in MOEA/D include: ▶ Given the black-box nature of the underlying problem and also the dynamics of evolution, there is no rule of thumb to evaluate neither exploration nor exploitation (the same for the convergence and diversity). ▶ Environmental selection is the springboard for analyzing the search dynamics of MOEA/D. However, related theoretical analysis has been significantly overlooked in the literature.

3.4 Reproduction Operators in MOEA/D

Reproduction operators are used to generate offspring thus determining the way to explore the search space. MOEA/D [10] applied the simulated binary crossover and polynomial mutation for offspring reproduction whereas the differential evolution considered in [23] has a much wider uptake thereafter. In principle, any reproduction operator proposed in the EC community can be used within MOEA/D. Here we review some representative operators being used in MOEA/D according to their types, as summarized in Table A4 of APPENDIX.

3.4.1 Swarm intelligence (SI)

SI represents the collective intelligent behavior of decentralized, self-organized systems, such as groups of simple agents. It includes behaviors observed in nature, like social insects' foraging and cooperative tasks.

- *Particle swarm optimization (PSO)*: Among the most popular SI methods, PSO has been widely adopted within MOEA/D [111–113]. They maintain a swarm of particles where each particle possesses its own position, velocity, and personal best vectors, alongside a population of global best vectors corresponding to different subproblems. The fitness of each particle is often assessed by a scalarizing function, e.g., [114–118].

- *Ant colony optimization (ACO)*: MOEA/D-ACO marked the first ACO adaptation within MOEA/D [119], assigning each ant to a subproblem. Ants in groups maintain a pheromone matrix to guide the search, constructing new solutions from both group pheromone and personal heuristic information. Further developments include using opposition-based learning [120], negative pheromone [121] and local optima avoidance [122].

Beyond PSO and ACO, other SI methods, such as artificial bee colony [123], cuckoo search [124], grey wolf optimization [125] and whale optimization [126], have been adapted to the MOEA/D framework.

3.4.2 Model-based methods

Traditional reproduction operators in EC, including crossover and mutation as well as SI methods, rely on fixed heuristic rules or strategies. They primarily interact with individual solutions and seldom adapt to the evolving environment, which can rapidly change as evolution progresses. Additionally, the complex interdependencies between variables can twist or rotate the problem landscape. These potentially diminish the effectiveness of traditional operators due to their static nature and inability to learn from environmental changes. In response, a growing number of efforts have aimed to enhance reproduction operators with learning capabilities. The fundamental approach involves substituting heuristic operators with machine learning models to generate new candidate solutions. These models, trained on candidate solutions sampled from the current environmental context, offer a dynamic means to navigate the search space.

- *Estimation of distribution algorithm (EDA)*: Its core idea involves estimating the distribution of promising candidate solutions by training machine learning models within the decision space. MEDA/D pioneered the use of multi-objective EDA based on decomposition for tackling the multi-objective traveling salesman problem (MOTSP) [127, 128]. By decomposing a MOTSP into subproblems, MEDA/D assigns each subproblem with a probabilistic model incorporating both heuristic information and insights learned from the evolving population. New candidate solutions, or tours, are generated by sampling from these models. The effective approximation of the PF for MOTSP relies on the collaborative efforts of neighboring subproblems. This approach has been extended by using more expressive probabilistic models, such as probabilistic graphical models [129], population-based incremental learning [130], and cross-entropy [131].

- *Covariance matrix adaptation evolution strategy (CMA-ES)*: Initial attempts to incorporate CMA-ES into MOEA/D, such as those in [132] and [133], involved associating each subproblem with a unique Gaussian model, leading to exponentially increased computational complexity with population size. MOEA/D-CMA [134] addresses this by adopting a hybrid strategy that combines differential evolution with CMA-ES, optimizing one subproblem at a time within a group.

- *Probability collectives (PC)*: Originating from game theory and statistical physics [135], PC emphasizes manipulating probability distributions rather than constructing point-wise models. MOPC/D [136] uses this approach, along with local search, to build distributions centered around the PS's neighborhood.

- *Pareto set model*: Focused on capturing regularities through learning techniques, this strategy guides evolutionary search towards promising offspring solutions. [137] initially approximated the PS using a linear model, while [138] later employed a generative topographic mapping to navigate offspring reproduction by understanding the population's manifold distribution.

3.4.3 Adaptive operator selection and hyper-heuristics

Different reproduction operators have their unique characteristics and search dynamics, underscoring the challenge that no single operator can universally excel across a diverse array of problems. Addressing the EvE dilemma requires a dynamic adjustment strategy that favors exploration in the early stages of problem-solving and shifts to exploitation as the search converges on the optimum. Adaptive operator selection (AOS) emerges as a solution, dynamically selecting the optimal reproduction operator based on real-time feedback. MOEA/D-FRRMAB [139] is the first instance that applies AOS for multi-objective optimization. Based on FRRMAB, many other AOS variants have been developed afterward (e.g., [140–143]). In addition to AOS, [144–148] also consider adaptively controlling the parameters associated with the reproduction operator in order to achieve the best algorithm setup. Rather than adaptively selecting the 'appropriate' reproduction operator on the fly, another line of research (e.g., [149–151]) is to build an ensemble of reproduction operators and use them simultaneously.

Different from AOS, hyper-heuristics (HH) offer a high-level strategy for heuristic selection and generation, capable of creating new operators that evolve with the optimization process. Early HH works within the MOEA/D framework, such as [152–155], mainly use a variant of FRRMAB for heuristic generation or selection. HH's integration with additional heuristic strategies, such as irace [156] and bandit models [157, 158], a two-level approach for complex problems.

3.4.4 Mathematical programming methods

Reproduction operators, as previously discussed, are designed to adaptively manage the EvE dilemma throughout the evolutionary process. However, they often excel more in global search scenarios. To enhance exploitation in specific promising regions, hybrid methods that integrate mathematical programming techniques—both gradient-based and derivative-free—into MOEA/D have been explored. Among these, the Nelder and Mead's method [159], a popular derivative-free optimization approach that conducts a nonlinear simplex search through reflection, expansion, and contraction, has been widely applied within MOEA/D [160–165]. Additionally, the Karush-Kuhn-Tucker optimality theory-based proximity measure, KKTPM [166], originally developed to assess the convergence of non-dominated solutions to the PS, has been utilized as a key component or termination criterion in local search strategies within NSGA-III [167–169].

3.4.5 Memetic search methods

Memetic algorithm [170], an extension of the traditional genetic algorithm, incorporates individual learning or local search techniques to reduce the risk of premature convergence. Within MOEA/D, memetic algorithms have been tailored for combinatorial MOPs, offering a distinct approach from the mathematical programming-based local search. We briefly discuss several memetic MOEA/D variants based on their local search strategies.

- *Pareto local search (PLS)*: It focuses on exploring solution neighborhoods to identify new non-dominated solutions or better local optima [171]. MOMAD was the pioneering integration of PLS with MOEA/D, utilizing an auxiliary population for the PLS process [17]. Shi et al. [172] enhanced PLS efficiency through parallel computation and problem decomposition, significantly accelerating the PF approximation process.

- *Neighborhood search (NS)*: NS searches optimal or near-optimal solutions by exploring a current solution's neighborhood, using diverse operators for varied neighborhood sizes or behaviors [173–

176]. NS has been integrated into MOEA/D to enhance solution quality through different neighborhood dynamics [177–180].

- *Simulated annealing (SA)*: SA is a technique inspired by statistical physics [181], effective for combinatorial optimization and incorporated into MOEA/D for improved solution exploration and optimization [182, 183]. Later, the same idea was applied for solving software next release [184] and hybrid flow shop problems [185].

- *Greedy randomized adaptive search procedure (GRASP)*: It is merged with MOEA/D to form a hybrid evolutionary meta-heuristic that leverages GRASP and path-relinking for enhanced search efficiency [186, 187].

- *Other local serach*: In addition to the classic local search, there are some other local search operators designed by considering bespoke problem knowledge. For instance, [188] proposed two local search operators for exam tabling problems. One is called period-supplement local search that improves the distribution of non-dominated solutions. The other is called bi-directed local search operator that improves the quality of the solutions. [189] proposed a path solution representation along with several neighborhoods and local search operators for multi-factory supply chain problems.

💣 Key bottlenecks of the existing subproblem formulations include: ▶ Although many reproduction operators have been applied in MOEA/D, there does not exist one winner takes all in view of the no-free-lunch theorem. In other words, a reproduction operator might only excel in one type of problems or the heuristics even need to be tailored according to the problem characteristics. ▶ Although both AOS and HH are expected to autonomously identify the bespoke operator, they are essentially an online optimization problem, which is challenging to solve on top of the underlying MOP.

4 Advanced Topics

This section plans to review selected important developments of MOEA/D variants for four specific scenarios.

4.1 Constraint Handling

Real-life optimization scenarios inherently come with various constraints. However, constraint handling has surprisingly been ignored not only in the seminal paper of MOEA/D [10], but also been lukewarm until [190]. Constraints not only create infeasible regions in the search space, posing obstacles to effective evolutionary search but can also distort the PF in terms of its shape and integrity. The current constraint handling approaches fall into three categories, as summarized in Table A5 of APPENDIX. Note that this categorization is not limited within MOEA/D, but is general to any EMO.

4.1.1 Feasibility-driven methods

As the first endeavor in this domain, [191] incorporated penalty terms within the scalarizing function to favor feasible solutions. Later, g-DBEA [192,193] developed an adaptive method to calculate constraint violations, tailoring the violation threshold to the constraints, size of the feasible region, and search outcomes. The successful constraint-domination principle from NSGA-II [8] was also adapted for NSGA-III variants [61, 194]. In [40], a novel approach was proposed considering the prioritization of an infeasible solution over feasible ones if located in an isolated subregion. Inspired by early results in [195], the ϵ-constraint method was modified for MOEA/D, with variants such as [196] dynamically adjusting the ϵ level based on the proportion of feasible solutions in the population. In [197, 198], a push and pull search framework was proposed. Initially, the push stage explores the feasible region; followed by the pull stage, where ϵ-constraint is used to navigate the search within the infeasible region.

4.1.2 Weight vector adaptation

The second category focuses on adapting the distribution of weight vectors to the PF distortion caused by constraints, as those introduced in Section 3.1. The core idea involves progressively identifying and removing weight vectors targeting infeasible parts of the objective space, while introducing new ones in feasible areas, e.g., [60] and [91]. [199] proposed maintaining two types of weight vectors to explore both feasible and infeasible regions. It uses infeasible weights to preserve a set of well-distributed infeasible solutions, thereby utilizing valuable information from these regions. These weights are dynamically adjusted throughout the evolution process to favor individuals with better objective values and smaller constraint violations.

4.1.3 Trade-off among convergence, diversity and feasibility

Traditional constraint handling methods prioritize driving the population toward feasible regions before addressing the balance between convergence and diversity. This approach risks in local optima or feasible regions, especially when these regions are narrow or widely separated in the search space. C-TAEA [190] is the pioneering work incorporating this three-way trade-off into its algorithm design. It maintains two complementary archives: a convergence archive to guide the population towards the PF, and a diversity archive to explore underutilized areas, including infeasible regions. This concept has sparked many follow-up studies using multiple populations for constraint handling, (e.g., [200–205]).

4.2 Optimization in Dynamic and Uncertain Environments

In addition to the challenge of highly constrained search spaces, another complexity of real-world black-box optimization problems comes from dynamic and uncertain environments. These conditions lead to time-varying objective functions, resulting in temporally evolving PSs and PFs. In principle, prevalent approaches in the realm of dynamic multi-objective optimization can be readily adapted to MOEA/D. Among them, the most popular one is the use of predictive models to estimate the seeding population after environmental changes. For instance, [206] proposed leveraging the Kalman filter to predict the evolving trajectories of PSs and PFs with time. This concept has been further developed to address unknown dynamic environments through inverse modeling [207] and autoregressive models [208–210]. Besides, MOEA/D has the unique capability to tailor its search directions to evolving environments by managing the distribution of weight vectors. A representative work along this line is [211] that pioneered the idea of dynamic multi-objective optimization with a changing number of objectives. This idea has been explored in [212–214]. Additionally, transfer optimization has been another popular method that selectively leverages the historical data to boost the evolutionary search in the new environment either by seeding a new initialized population [215] or informing the adjustment of weight vectors [216,217]. Last but not least, given the complexities of uncertain environments, employing an ensemble of dynamic response strategies offers resilience and versatility [218–220].

4.3 Surrogate-Assisted Methods for Expensive Optimization

Real-world problems are characterized by time- or resource-intensive physical experiments or numerical simulations. For instance, evaluating a single function based on computational fluid dynamics simulations can take anywhere from minutes to hours. Traditional evolutionary algorithms, which require a large number of function evaluations, become impractical. To address this, surrogate models have emerged as a widely adopted solution, serving as efficient proxies for expensive objective functions, as summarized in Table A6 of `Appendix`.

ParEGO [19] is the first one that introduces surrogate modeling into the decomposition-based EMO algorithms, even earlier than the canonical MOEA/D [10]. It aggregates a MOP into a scalar function using an augmented Tchebycheff function and a randomly sampled normalized weight vector for each iteration. This method extends the efficient global optimization (EGO) strategy [221], where a Gaussian process (GP) model represents the expensive objective function, and the search is driven by maximizing expected improvement based on the GP model, to address expensive MOPs.

Four years later, Zhang et al. [222] proposed MOEA/D-EGO, adapting EGO for MOEA/D. Given the co-evolution of a population of subproblems in MOEA/D, building a separate surrogate model for each would be computationally prohibitive. Instead, MOEA/D-EGO builds a GP model for each objective, adapting EGO for subproblem optimization using specific mean and variance formats related to weighted sum and Tchebycheff functions. K-RVEA [223], addresses computationally expensive many-objective optimization by utilizing the uncertainty estimates from Kriging models. It adapts the distribution of weight vectors and the positioning of solutions in the objective space to balance convergence and diversity. Very recently, [224] proposed a new framework to interpolate new approximate Pareto-optimal solutions by an integration of gradient-based search with EMO. This method has been further extended to software and hardware co-design problems [225, 226].

There have been various variants by introducing other machine learning models as surrogates, such as radial basis function [162, 227], extreme learning machines [228, 229], multi-level surrogate models [230], classifiers [231], support vector machines [232], and Walsh functions [233]. In addition to maximizing expected improvement, new infill criteria have been developed, such as bi-objective formulations [234], dynamic weighted aggregations [235], and closed-form improvements for PBI/IPBI functions [236]. Efforts also include leveraging latent information of surrogate models [224, 226], complementary infill criteria [237] and the use of multiple surrogate models [238]. To tackle the curse of dimensionality, dimensionality reduction like principal component analysis [239] and feature selection [240] have been employed. Moreover, the advent of transfer learning has facilitated knowledge sharing and reuse within MOEA/D, such as [241] and [242]. Leveraging MOEA/D's piecewise linear assumption and the similarity among neighboring subproblems, [243–246] proposed multi-task GP models for collaborative surrogate modeling of representative subproblems.

4.4 Preference Incorporation in MOEA/D

The methodologies reviewed thus far aim to approximate the entire PF. However, as [247] discussed, presenting decision-makers (DMs) with a wide array of trade-off alternatives not only burdens the DM but can also introduce irrelevant or noisy information into the decision-making process. Over the past two decades, there has been an increasing trend towards integrating EMO and MCDM to incorporate DM preferences directly into the evolutionary search [247–249]. This approach focuses on approximating the region of interest (ROI), which is often just a partial region of the PF. In MOEA/D, the subproblems' configuration, determined by the distribution of weight vectors, influences the location of approximated Pareto-optimal solutions. Therefore, integrating DM preferences into MOEA/D involves adjusting weight vector distribution based on the DM's provided preference information. As outlined in Table A7 of `Appendix`, we will review current research in MOEA/D's preference incorporation, categorized by the preference elicitation method, whether *a priori* or *interactively*.

4.4.1 A priori preference elicitation

The basic strategy for incorporating *a priori* preference information into MOEA/D involves biasing the distribution of weight vectors based on the DM's preferences. Weight vectors, initially distributed evenly, are adjusted to focus the search towards the ROI. For instance, [250] introduced a method named NUMS that maps evenly distributed weight vectors to positions closer to the DM's reference point on a simplex. Unlike this static approach, other studies [251–253] dynamically shift weight vectors towards the ROI as the population evolves, favoring solutions near the DM's reference point. Further, approaches like a-PICEA-g [254] and cwMOEA/D [255] co-evolve the population and weight vectors, approximating the ROI simultaneously. a-PICEA-g generates a new population of weight vectors randomly in each iteration, while cwMOEA/D employs a mutation operator to create new candidate weight vectors. Another method involves defining a preference region, usually around a DM-provided reference point, within a specified boundary (e.g., a tolerance angle [256] or a radius [257]), from which new weight vectors are sampled. In [258], a scaled simplex was used to specify the preference region, starting with biased weight vectors at the simplex's corners and iteratively adding weight vector pairs inside of the simplex.

4.4.2 Interactive preference elicitation

Differently, *interactive* preference elicitation allows the DM to progressively learn about the problem characteristics and adjust their preferences accordingly. One of the earliest works in this area, PMA [259], periodically engages the DM to rank pairs of solutions, constructing a set of linear value functions for environmental selection and ROI approximation. iMOEA/D [260] proposed a virtual interaction mechanism, where the DM periodically selects promising solutions to refine the utility function and adjust weight vectors. [261–263] developed an interactive framework within MOEA/D. It consists of three modules, i.e., consultation, preference elicitation, and optimization. Specifically, after every several generations, the DM is asked to score a few candidate solutions in a consultation session. Thereafter, an approximated value function, which models the DM's preference information, is progressively learned from the DM's behavior. In the preference elicitation session, the preference information learned in the consultation module is translated into the form that can be used in a decomposition-based EMO algorithm, i.e., a set of reference points that are biased toward the ROI. Most recently, this framework was generalized to Pareto- and indicator-based EMO, and the preference is learned from implicit information [264]. In [265], IEMO/D proposed employing Monte Carlo simulations and rejection sampling to model the DM's preferences.

5 Emerging Topics and Future Directions

In this section, we explore some emerging topics that transcend the realm of decomposition-based EMO, reflecting future research directions for our EMO community.

5.0.1 Large-scale MOPs (LSMOPs)

Real-world optimization problems often involve a large number of decision variables, known as LSMOPs. One of the most recognized approaches in the EC community for tackling LSMOPs is cooperative co-evolution. Its basic idea is to decompose the original problem into smaller, manageable subproblems based on decision variable analysis (DVA) that identifies variable interactions. This allows for a collaborative solution process. As the first attempt, [266] introduced a DVA technique that categorizes decision variables by their control properties—either convergence- or spread-related. This approach analyzes the interaction structures of these variables separately, employing random permutations. Further developments include extending this DVA technique into a distributed solution approach [267], a graph representation [268], and an adaptive localized version [269]. Another strategy [141] involves a self-adaptive operator selection mechanism, allowing MOEA/D to address LSMOPs without variable interaction analysis. However, compared to real-world scenarios like the GPT-4, which contains 1.76 trillion parameters, current EMO algorithms for LSMOPs—handling usually less than $10,000$ decision variables—fall short of industrial-level requirements. To elevate the scalability of EMO, we need to investigate novel problem transformation and dimension reduction techniques, which can compress the high-dimensional search space, making optimal solutions achievable with an acceptable number of function evaluations.

5.0.2 Multimodal MOPs (MMOPs)

In some real-world optimization scenarios, such as space mission design problems [270], diesel engine design problems [271] and rocket engine design problems [272], there exist some dissimilar solutions in the decision space but lead to similar or equivalent quality in the objective space, as known as MMOPs. Solving an MMOP aims to locate (almost) equivalent Pareto-optimal solutions as many as possible. To maintain the diversity of the population in both the objective and decision spaces, [273] and [274] proposed a general framework that uses the environmental selection mechanism of the baseline EMO algorithm to maintain diversity in the objective space while the diversity in the decision space is maintained by a simple niching criterion. Recently, a graph Laplacian based optimization assisted by decomposition in both decision and objective spaces is proposed in [275] for solving MMOPs.

Very recently, [276] proposed transforming a MMOP into a bi-objective optimization problem, thus promoting convergence in the objective space while maintaining diversity in the decision space. Future developments are expected to focus on more effective distance metrics and diversity preservation strategies to better identify multiple equivalent Pareto-optimal solutions simultaneously.

5.0.3   Variable-length MOPs

In conventional optimality theory, the length of decision variables is always assumed to be fixed. However, it is not uncommon that the optimal length of a decision variable is unknown *a priori*, let alone the optimum, in many real-world applications, such as composite laminate stacking problem [277], sensor coverage problem [278] and wind farm layout problem [279]. This can lead to a variable-length Pareto structure where the number of variables in two Pareto-optimal solutions might be different. One of its key challenges is the varying length of variables that disrupt the genetic reproduction operators in EA. Furthermore, due to the existence of variable-length Pareto structure, approximating the PF can be biased as the local region with a larger variable size is more difficult than those with smaller variable sizes [280]. To address these issues, MOEA/D-VLP [281] proposed a two-level decomposition strategy where the global level is the same as the conventional MOEA/D while the local level decomposes an MOP into multiple MOPs with different ranges of dimensionality to approximate various local PF segments. In addition, an adaptive strategy is proposed to allocate computational efforts towards different parts of the PF to mitigate the bias towards any relatively easier PF regions. Recently, [282] proposed using a simple dimensionality incremental learning strategy to choose representative solutions. As a next step, it is desirable to develop bespoke subproblem formulations for different regions of the PF and a balanced way to strategically allocate computational budget.

5.0.4   Sparse optimization

Sparse optimization problems widely appear in many real-world applications such as data mining, variable selection, visual coding, signal and image processing. Its ultimate goal is to find a sparse representation of a system, which usually aims to minimize a data-fitting term and a sparsity term simultaneously. As a dominating strategy in sparse optimization, the classic iterative thresholding often aggregates these two terms into a single function, where a relaxed parameter is used to balance the error and the sparsity. In particular, the choice of this relaxed parameter is sensitive to the performance of the iterative thresholding methods. There have been some initial attempts [283–285] that transform the sparse optimization problem as a MOP and they apply tailored MOEA/D to solve it. Due to the unknown sparsity level of the underlying problem, it is challenging to maintain the diversity of the population. In addition, it is imperative to design bespoke performance indicators to measure the convergence, diversity, and sparsity of the population simultaneously.

5.0.5   Parallel computing

In principle, the decentralized nature of decomposition is ideal for distributed computing resources. However, integrating parallelism into decomposition approaches presents challenges, given the heterogeneous and complex nature of modern large-scale computing platforms. It requires that parallelism be tightly integrated with decomposition components, rather than treated as a separate entity. There have been limited studies on distributed MOEA/D implementations, such as the classic master-slave model [286] and GPU-based CUDA technology for real-time MOPs [287]. As for a future direction along this line, developing a mapping function to align decomposed subproblems with the parallelism offered by the computing platforms is crucial. Additionally, designing adaptive distributed algorithms that cater to the specific characteristics of subproblems is necessary.

5.0.6   Multi-agent system vs MOEA/D

In theory, the decomposed subproblems position MOEA/D akin to multi-agent systems, where each subproblem functions as an autonomous agent engaging in a cooperative search, within a pre-defined

neighborhood, for its optimal solution through population-based meta-heuristics. This analogy has been explored by the matching-based selection mechanism [95–97], while the full potential of this concept has unfortunately yet been fully exploited in literature. Another light touch in [136] applied probability collectives [135]—originally proposed for distributed optimization—to MOEA/D in a multi-agent system manner. Given the wealth of theoretical and methodological advancements in multi-agent systems and game theory, there lies a promising avenue to harness these insights for crafting more effective and principled EMO algorithms.

5.0.7 Theoretical studies

In our EMO community, rigorous theoretical analysis has often lagged behind methodological developments. There are two primary types of theoretical studies within the MOEA/D framework. One mainly focused on the relationships among classic subproblem formulations and the impacts of their respective hyperparameters. For instance, [288] theoretically explored the geometric properties of TCH, uncovering asymptotic equivalences with the Hypervolume indicator. Similarly, [289] investigated the equivalence between PBI and AASF under various hyperparameter settings. Another line of research concerns the runtime analysis of MOEA/D [290, 291] and NSGA-III [292]. However, these studies are often limited to 'overly simplified' synthetic benchmark problems with known Pareto-optimal solutions.

6 Concluding Remarks

MOEA/D bridges the gap between population-based meta-heuristics, EMO in particular, and conventional MCDM. This article has provided a comprehensive survey of the developments of MOEA/D to date. We began with a gentle tutorial on MOEA/D's basic workings to ensure the survey is self-contained. Our survey then elaborated on the four core design components: weight vector settings, subproblem formulations, selection mechanisms, and reproduction operators. We also covered selected advanced topics, such as constraint handling, optimization in computationally expensive scenarios, and preference incorporation. Finally, we highlighted emerging directions for future development. Note that these areas are beyond the scope of MOEA/D, but are of universal significance for our entire EMO community.

> ✇ RESOURCES: There are a series of technical activities themed on the decomposition multi-objective optimization. For instance, there is a dedicated website gathering various resources of MOEA/D, including papers, source codes and active researchers. Sponsored by the IEEE Computational Intelligence Society, there has been a Task Force themed on decomposition-based techniques in EC since 2017. This is an international consortium that brings together global researchers to promote an active state of this area. Since 2018, there have been tutorials, workshops and special sessions regularly organized within major EC conferences such as PPSN, GECCO, CEC and SSCI.

Acknowledgment

References

[1] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," in *NeurIPS'18: Proc. of the 2018 Annual Conference on Neural Information Processing Systems*, 2018, pp. 525–536. 1

[2] M. T. Ribeiro, N. Ziviani, E. S. de Moura, I. Hata, A. Lacerda, and A. Veloso, "Multiobjective pareto-efficient approaches for recommender systems," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 4, pp. 53:1–53:20, 2014. 1

[3] P. M. Attia, A. Grover, N. Jin, K. A. Severson, T. M. Markov, Y. Liao, M. H. Chen, B. Cheong, N. Perkins, Z. Yang, P. K. Herring, M. Aykol, S. J. Harris, R. D. Braatz, S. Ermon, and W. C. Chueh, "Closed-loop optimization of fast-charging protocols for batteries with machine learning," *Nat.*, vol. 578, no. 7795, pp. 397–402, 2020. 1

[4] Y. Li, J. Hu, Y. Wang, J. Zhou, L. Zhang, and Z. Liu, "Deepscaffold: A comprehensive tool for scaffold-based de novo drug discovery using deep learning," *J. Chem. Inf. Model.*, vol. 60, no. 1, pp. 77–91, 2020. 1

[5] G. S. Hornby, J. D. Lohn, and D. S. Linden, "Computer-automated evolution of an x-band antenna for nasa's space technology 5 mission," *Evol. Comput.*, vol. 19, no. 1, pp. 1–23, 2011. 1

[6] T. Chen, K. Li, R. Bahsoon, and X. Yao, "FEMOSAA: feature-guided and knee-driven multi-objective optimization for self-adaptive software," *ACM Trans. Softw. Eng. Methodol.*, vol. 27, no. 2, pp. 5:1–5:50, 2018. 1

[7] A. Ponsich, A. L. Jaimes, and C. A. C. Coello, "A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications," *IEEE Trans. Evol. Comput.*, vol. 17, no. 3, pp. 321–344, 2013. 1

[8] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002. 1, 15

[9] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *PPSN'04: Proc. of 8th International Conference on Parallel Problem Solving from Nature*, 2004, pp. 832–842. 1

[10] 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, 2007. 1, 2, 11, 12, 15, 16

[11] M. P. Hansen, "Tabu search for multiobjective optimization: MOTS," in *MCDM'97: Proc. of 13th International Conference on Multiple Criteria Decision Making*, 1997, pp. 6–10. 2

[12] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Trans. Syst. Man Cybern. Part C*, vol. 28, no. 3, pp. 392–403, 1998. 2, 5

[13] E. L. Ulungu, J. Teghem, P. Fortemps, and D. Tuyttens, "Mosa method: a tool for solving multiobjective combinatorial optimization problems," *J. Multi-Crit. Decis. Anal.*, vol. 8, pp. 221–236, 1999. 2

[14] T. Murata, H. Ishibuchi, and M. Gen, "Specification of genetic search directions in cellular multi-objective genetic algorithms," in *EMO'01: Proc. the 1st International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 1993. Springer, 2001, pp. 82–95. 2

[15] Y. Jin, T. Okabe, and B. Sendhoff, "Adapting weighted aggregation for multiobjective evolution strategies," in *EMO'01: Proc. of the 1st International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 1993. Springer, 2001, pp. 96–110. 2, 6

[16] A. Jaszkiewicz, "Genetic local search for multi-objective combinatorial optimization," *Eur. J. Oper. Res.*, vol. 137, no. 1, pp. 50–71, 2002. 2, 5

[17] L. Paquete and T. Stützle, "A two-phase local search for the biobjective traveling salesman problem," in *EMO'03: Proc. of the 2nd International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 2632. Springer, 2003, pp. 479–493. 2, 14

[18] E. J. Hughes, "Evolutionary many-objective optimisation: many once or one many?" in *CEC'05: Proc. of the 2005 IEEE Congress on Evolutionary Computation*. IEEE, 2005, pp. 222–227. 2

[19] J. D. Knowles, "ParEGO: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 50–66, 2006. 2, 16

[20] M. J. Alves and M. Almeida, "MOTGA: A multiobjective Tchebycheff based genetic algorithm for the multidimensional knapsack problem," *Comput. Oper. Res.*, vol. 34, no. 11, pp. 3458–3470, 2007. 2

[21] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston, MA, USA: Kluwer Academic Publishers, 1999. 1, 4, 8

[22] I. Giagkiozis, R. C. Purshouse, and P. J. Fleming, "Towards understanding the cost of adaptation in decomposition-based optimization algorithms," in *SMC'13: Proc. of the 2013 IEEE Conference on Systems, Mans and Cybernetics*. IEEE, 2013, pp. 615–620. 2

[23] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, 2009. 2, 11, 12

[24] Q. Xu, Z. Xu, and T. Ma, "A survey of multiobjective evolutionary algorithms based on decomposition: Variants, challenges and future directions," *IEEE Access*, vol. 8, pp. 41 588–41 614, 2020. 2

[25] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 440–462, 2017. 2

[26] A. Santiago Pineda, H. J. Fraire Huacuja, B. Dorronsoro, J. E. Pecero, C. G. Santillán, J. J. G. Barbosa, and J. C. S. Monterrubio, "A survey of decomposition methods for multi-objective optimization," in *Recent Advances on Hybrid Approaches for Designing Intelligent Systems*, ser. Studies in Computational Intelligence. Springer, 2014, vol. 547, pp. 453–465. 2

[27] J. Wang, Y. Su, Q. Lin, L. Ma, D. Gong, J. Li, and Z. Ming, "A survey of decomposition approaches in multiobjective evolutionary algorithms," *Neurocomputing*, vol. 408, pp. 308–330, 2020. 2

[28] X. Ma, Y. Yu, X. Li, Y. Qi, and Z. Zhu, "A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 24, no. 4, pp. 634–649, 2020. 2

[29] X. Ma, J. Yang, N. Wu, Z. Ji, and Z. Zhu, "A comparative study on decomposition-based multi-objective evolutionary algorithms for many-objective optimization," in *CEC'16: Proc. of 2016 IEEE Congress on Evolutionary Computation*. IEEE, 2016, pp. 2477–2483. 2

[30] Q. Xu, Z. Xu, and T. Ma, "A short survey and challenges for multiobjective evolutionary algorithms based on decomposition," in *CITS'19: Proc. of 2019 International Conference on Computer, Information and Telecommunication Systems*. IEEE, 2019, pp. 1–5. 2

[31] X. Guo, "A survey of decomposition based evolutionary algorithms for many-objective optimization problems," *IEEE Access*, vol. 10, pp. 72 825–72 838, 2022. 2

[32] 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, pp. 631–657, 1998. 4

[33] 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, 2017. 5

[34] A. Jaszkiewicz, "On the performance of multiple-objective genetic local search on the 0/1 knapsack problem - a comparative experiment," *IEEE Trans. Evol. Comput.*, vol. 6, no. 4, pp. 402–412, 2002. 5

[35] H. Li, M. Ding, J. Deng, and Q. Zhang, "On the use of random weights in MOEA/D," in *CEC'15: Proc. of the 2015 IEEE Congress on Evolutionary Computation*. IEEE, 2015, pp. 978–985. 6

[36] Y. Tan, Y. Jiao, H. Li, and X. Wang, "A modification to MOEA/D-DE for multiobjective optimization problems with complicated pareto sets," *Inf. Sci.*, vol. 213, pp. 14–38, 2012. 6

[37] ——, "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives," *Comput. Oper. Res.*, vol. 40, no. 6, pp. 1648–1660, 2013. 6

[38] Y. Tan and Y. Jiao, "MOEA/D with uniform design for solving multiobjective knapsack problems," *J. Comput.*, vol. 8, no. 2, pp. 302–307, 2013. 6

[39] 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, 2014. 6

[40] 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, 2015. 6, 12, 15

[41] J. Blank, K. Deb, Y. D. Dhebar, S. Bandaru, and H. Seada, "Generating well-spaced points on a unit simplex for evolutionary many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 48–60, 2021. 6

[42] H.-L. Liu, F.-Q. Gu, and Y. ming Cheung, "T-MOEA/D: MOEA/D with objective transform in multi-objective problems," in *ICISME'10: Proc. of 2010 International Conference of Information Science and Management Engineering*. IEEE, 2010, pp. 1–6. 6

[43] S. Jiang, Z. Cai, J. Zhang, and Y. Ong, "Multiobjective optimization by decomposition with pareto-adaptive weight vectors," in *ICNC'11: Proc. of the 7th International Conference on Natural Computation*. IEEE, 2011, pp. 1260–1264. 6

[44] S. Jiang, J. Zhang, and Y. Ong, "Asymmetric pareto-adaptive scheme for multiobjective optimization," in *AI'11: Proc. of the 24th Australasian Joint Conference on Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, vol. 7106. Springer, 2011, pp. 351–360. 6

[45] F. qing Gu, H.-L. Liu, and K.-C. Tan, "A multiobjective evolutionary algorithm using dynamic weight design method," *Int. J. Innov. Comput. Inf. Control.*, vol. 8, no. 5, pp. 3677–3688, 2012. 6

[46] M. Wu, S. Kwong, Y. Jia, K. Li, and Q. Zhang, "Adaptive weights generation for decomposition-based multi-objective optimization using gaussian process regression," in *GECCO'17: Proc. of the 2017 Genetic and Evolutionary Computation Conference*. ACM, 2017, pp. 641–648. 6

[47] M. Wu, K. Li, S. Kwong, Q. Zhang, and J. Zhang, "Learning to decompose: A paradigm for decomposition-based multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 376–390, 2019. 6, 9, 10

[48] Q. Wang, Q. Gu, L. Chen, Y. Guo, and N. Xiong, "A MOEA/D with global and local cooperative optimization for complicated bi-objective optimization problems," *Appl. Soft Comput.*, vol. 137, p. 110162, 2023. 6

[49] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Cybern.*, vol. 43, pp. 59–69, 1982. 6

[50] F. Gu and Y. Cheung, "Self-organizing map-based weight design for decomposition-based many-objective evolutionary algorithm," *IEEE Trans. Evol. Comput.*, vol. 22, no. 2, pp. 211–225, 2018. 6

[51] B. Fritzke, "A growing neural gas network learns topologies," in *NIPS'94: Proc. of the 1994 Advances in Neural Information Processing Systems Conference.* MIT Press, 1994, pp. 625–632. 6

[52] Y. Liu, H. Ishibuchi, N. Masuyama, and Y. Nojima, "Adapting reference vectors and scalarizing functions by growing neural gas to handle irregular Pareto fronts," *IEEE Trans. Evol. Comput.*, vol. 24, no. 3, pp. 439–453, 2020. 6

[53] Q. Liu, Y. Jin, M. Heiderich, T. Rodemann, and G. Yu, "An adaptive reference vector-guided evolutionary algorithm using growing neural gas for many-objective optimization of irregular problems," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 2698–2711, 2022. 6

[54] C. Wang, H. Huang, and H. Pan, "A constrained many-objective evolutionary algorithm with learning vector quantization-based reference point adaptation," *Swarm Evol. Comput.*, vol. 82, p. 101359, 2023. 6

[55] Y. Qi, X. Ma, F. Liu, L. Jiao, J. Sun, and J. Wu, "MOEA/D with adaptive weight adjustment," *Evol. Comput.*, vol. 22, no. 2, pp. 231–264, 2014. 7

[56] L. R. C. de Farias, P. H. M. Braga, H. de F. Bassani, and A. F. R. Araújo, "MOEA/D with uniformly randomly adaptive weights," in *GECCO'18: Proc. of the Genetic and Evolutionary Computation Conference.* ACM, 2018, pp. 641–648. 7

[57] L. R. C. de Farias and A. F. R. Araújo, "Many-objective evolutionary algorithm based on decomposition with random and adaptive weights," in *SMC'19: Proc. of the 2019 IEEE International Conference on Systems, Man and Cybernetics.* IEEE, 2019, pp. 3746–3751. 7

[58] M. Li and X. Yao, "What weights work for you? adapting weights for any pareto front shape in decomposition-based evolutionary multiobjective optimisation," *Evol. Comput.*, vol. 28, no. 2, pp. 227–253, 2020. 7

[59] M. Wang, Y. Wang, and X. Wang, "A space division multiobjective evolutionary algorithm based on adaptive multiple fitness functions," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 30, no. 3, pp. 1 659 005:1–1 659 005:25, 2016. 7

[60] 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, 2016. 7, 16

[61] 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, 2014. 7, 15

[62] ——, "An improved adaptive approach for elitist nondominated sorting genetic algorithm for many-objective optimization," in *EMO'13: Proc. of the 7th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 7811. Springer, 2013, pp. 307–321. 7

[63] A. Camacho, G. T. Pulido, R. L. Becerra, and H. Ishibuchi, "Indicator-based weight adaptation for solving many-objective optimization problems," in *EMO'19: Proc. of the 10th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 11411. Springer, 2019, pp. 216–228. 7

[64] X. Guo, X. Wang, and Z. Wei, "MOEA/D with adaptive weight vector design," in *CIS'15: Proc. of the 11th International Conference on Computational Intelligence and Security*. IEEE Computer Society, 2015, pp. 291–294. 7

[65] S. Jiang, L. Feng, C. K. Heng, Q. C. Nguyen, Y. Ong, N. A. Zhang, and P. S. Tan, "Adaptive indicator-based evolutionary algorithm for multiobjective optimization problems," in *CEC'16: Proc. of the 2016 IEEE Congress on Evolutionary Computation*. IEEE, 2016, pp. 492–499. 7

[66] 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, 2018. 7

[67] I. Giagkiozis and P. J. Fleming, "Methods for multi-objective optimization: An analysis," *Inf. Sci.*, vol. 293, pp. 338–350, 2015. 8

[68] M. Wu, K. Li, S. Kwong, and Q. Zhang, "Evolutionary many-objective optimization based on adversarial decomposition," *IEEE Trans. Cybern.*, vol. 50, no. 2, pp. 753–764, 2020. 8

[69] R. Wang, Q. Zhang, and T. Zhang, "Pareto adaptive scalarising functions for decomposition based algorithms," in *EMO'15: Proc. of the 8th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 9018. Springer, 2015, pp. 248–262. 8, 10

[70] ——, "Decomposition-based algorithms using pareto adaptive scalarizing methods," *IEEE Trans. Evol. Comput.*, vol. 20, no. 6, pp. 821–837, 2016. 8, 10

[71] S. Jiang, S. Yang, Y. Wang, and X. Liu, "Scalarizing functions in decomposition-based multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 22, no. 2, pp. 296–313, 2018. 8

[72] S. Yang, S. Jiang, and Y. Jiang, "Improving the multiobjective evolutionary algorithm based on decomposition with new penalty schemes," *Soft Comput.*, vol. 21, no. 16, pp. 4677–4691, 2017. 9, 10

[73] M. Ming, R. Wang, Y. Zha, and T. Zhang, "Pareto adaptive penalty-based boundary intersection method for multi-objective optimization," *Inf. Sci.*, vol. 414, pp. 158–174, 2017. 9, 10

[74] H. Sato, "Inverted PBI in MOEA/D and its impact on the search performance on multi and many-objective optimization," in *GECCO'14: Proc. of the 2014 Genetic and Evolutionary Computation Conference*. ACM, 2014, pp. 645–652. 9

[75] ——, "Analysis of inverted PBI and comparison with other scalarizing functions in decomposition based moeas," *J. Heuristics*, vol. 21, no. 6, pp. 819–849, 2015. 9

[76] L. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Constrained subproblems in a decomposition-based multiobjective evolutionary algorithm," *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 475–480, 2016. 9, 12

[77] 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, 2018. 10

[78] H. 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, 2014. 10

[79] H. Liu, L. Chen, K. Deb, and E. D. Goodman, "Investigating the effect of imbalance between convergence and diversity in evolutionary multiobjective algorithms," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 408–425, 2017. 10

[80] H. Liu, L. Chen, Q. Zhang, and K. Deb, "Adaptively allocating search effort in challenging many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 3, pp. 433–448, 2018. 10

[81] S. Jiang and S. Yang, "An improved multiobjective optimization evolutionary algorithm based on decomposition for complex pareto fronts," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 421–437, 2016. 11

[82] X. Li, H. Zhang, and S. Song, "A decomposition based multiobjective evolutionary algorithm with self-adaptive mating restriction strategy," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 11, pp. 3017–3030, 2019. 11

[83] S. Zhao, P. N. Suganthan, and Q. Zhang, "Decomposition-based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes," *IEEE Trans. Evol. Comput.*, vol. 16, no. 3, pp. 442–446, 2012. 11

[84] S. X. Zhang, L. Zheng, L. Liu, S. Y. Zheng, and Y. M. Pan, "Decomposition-based multi-objective evolutionary algorithm with mating neighborhood sizes and reproduction operators adaptation," *Soft Comput.*, vol. 21, no. 21, pp. 6381–6392, 2017. 11

[85] I. Gholaminezhad and G. Iacca, "A multi-objective relative clustering genetic algorithm with adaptive local/global search based on genetic relatedness," in *EvoApplications'14: Proc. of the 17th European Conference on Applications of Evolutionary Computation*, ser. Lecture Notes in Computer Science, vol. 8602. Springer, 2014, pp. 591–602. 11

[86] X. Li, S. Song, and H. Zhang, "Evolutionary multiobjective optimization with clustering-based self-adaptive mating restriction strategy," *Soft Comput.*, vol. 23, no. 10, pp. 3303–3325, 2019. 11

[87] F. Wang, H. Zhang, Y. Li, Y. Zhao, and Q. Rao, "External archive matching strategy for MOEA/D," *Soft Comput.*, vol. 22, no. 23, pp. 7833–7846, 2018. 11

[88] Y. Yuan, H. Xu, and B. Wang, "Evolutionary many-objective optimization using ensemble fitness ranking," in *GECCO'14: Proc. of the 2014 Genetic and Evolutionary Computation Conference*. ACM, 2014, pp. 669–676. 12

[89] M. Elarbi, S. Bechikh, and L. B. Said, "On the importance of isolated solutions in constrained decomposition-based many-objective optimization," in *GECCO'17: Proc. of the 2017 Genetic and Evolutionary Computation Conference*. ACM, 2017, pp. 561–568. 12

[90] 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, 2017. 12

[91] 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, 2018. 12, 16

[92] 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, 2019. 12

[93] Y. Zhou, Z. Chen, Z. Huang, and Y. Xiang, "A multiobjective evolutionary algorithm based on objective-space localization selection," *IEEE Trans. Cybern.*, pp. 1–14, 2020, accepted for publication. 12

[94] C. Liu, Q. Zhao, B. Yan, S. M. Elsayed, T. Ray, and R. A. Sarker, "Adaptive sorting-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 247–257, 2019. 12

[95] K. Li, Q. Zhang, S. Kwong, M. Li, and R. Wang, "Stable matching-based selection in evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 6, pp. 909–923, 2014. 12, 20

[96] M. Wu, K. Li, S. Kwong, Y. Zhou, and Q. Zhang, "Matching-based selection with incomplete lists for decomposition multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 4, pp. 554–568, 2017. 12, 20

[97] 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, 2015. 12, 20

[98] Z. Wang, Q. Zhang, M. Gong, and A. Zhou, "A replacement strategy for balancing convergence and diversity in MOEA/D," in *CEC'14: Proc. of the 2014 IEEE Congress on Evolutionary Computation.* IEEE, 2014, pp. 2132–2139. 12

[99] M. Wu, S. Kwong, Q. Zhang, K. Li, R. Wang, and B. Liu, "Two-level stable matching-based selection in MOEA/D," in *SMC'15: Proc. of 2015 IEEE International Conference on Systems, Man, and Cybernetics.* IEEE, 2015, pp. 1720–1725. 12

[100] Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Adaptive replacement strategies for MOEA/D," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 474–486, 2016. 12

[101] X. Chen, C. Shi, A. Zhou, B. Wu, and P. Sheng, "On balancing neighborhood and global replacement strategies in MOEA/D," *IEEE Access*, vol. 7, pp. 45 274–45 290, 2019. 12

[102] 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, 2016. 12

[103] 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, 2016. 12

[104] R. Liu, R. Wang, R. Bian, J. Liu, and L. Jiao, "A decomposition-based evolutionary algorithm with correlative selection mechanism for many-objective optimization," *Evol. Comput.*, vol. 29, no. 2, pp. 269–304, 2021. 12

[105] S. Zhou, K. Li, and G. Min, "Attention-based genetic algorithm for adversarial attack in natural language processing," in *PPSN'22: Proc. of the 17th International Conference on Parallel Problem Solving from Nature*, ser. Lecture Notes in Computer Science, vol. 13398. Springer, 2022, pp. 341–355. 12

[106] ——, "Adversarial example generation via genetic algorithm: a preliminary result," in *GECCO'22: Companion of the 2022 Genetic and Evolutionary Computation Conference.* ACM, 2022, pp. 469–470. 12

[107] P. N. Williams, K. Li, and G. Min, "Black-box adversarial attack via overlapped shapes," in *GECCO'22: Companion of the 2022 Genetic and Evolutionary Computation Conference*, J. E. Fieldsend and M. Wagner, Eds. ACM, 2022, pp. 467–468. 12

[108] P. N. Williams and K. Li, "Black-box sparse adversarial attack via multi-objective optimisation CVPR proceedings," in *CVPR'23: Proc. of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2023, pp. 12 291–12 301. 12

[109] P. N. Williams, K. Li, and G. Min, "Sparse adversarial attack via bi-objective optimization," in *EMO'23: Proc. of the 12th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 13970. Springer, 2023, pp. 118–133. 12

[110] ——, "A surrogate assisted evolutionary strategy for image approximation by density-ratio estimation," in *CEC'23: Proc. of 2023 IEEE Congress on Evolutionary Computation*. IEEE, 2023, pp. 1–8. 12

[111] W. Peng and Q. Zhang, "A decomposition-based multi-objective particle swarm optimization algorithm for continuous optimization problems," in *GrC'08: Proc. of the 2008 IEEE International Conference on Granular Computing*. IEEE, 2008, pp. 534–537. 13

[112] N. A. Moubayed, A. Petrovski, and J. A. W. McCall, "A novel smart multi-objective particle swarm optimisation using decomposition," in *PPSN XI: Proc. of the 11th International Conference on Parallel Problem Solving from Nature*, ser. Lecture Notes in Computer Science, vol. 6239. Springer, 2010, pp. 1–10. 13

[113] S. Z. Martínez and C. A. C. Coello, "A multi-objective particle swarm optimizer based on decomposition," in *GECCO'13: Proc. of the 13th Annual Genetic and Evolutionary Computation Conference*. ACM, 2011, pp. 69–76. 13

[114] N. A. Moubayed, A. Petrovski, and J. A. W. McCall, "D^{2}mopso: MOPSO based on decomposition and dominance with archiving using crowding distance in objective and solution spaces," *Evol. Comput.*, vol. 22, no. 1, pp. 47–77, 2014. 13

[115] Q. Lin, J. Li, Z. Du, J. Chen, and Z. Ming, "A novel multi-objective particle swarm optimization with multiple search strategies," *Eur. J. Oper. Res.*, vol. 247, no. 3, pp. 732–744, 2015. 13

[116] A. Pan, L. Wang, W. Guo, and Q. Wu, "A diversity enhanced multiobjective particle swarm optimization," *Inf. Sci.*, vol. 436-437, pp. 441–465, 2018. 13

[117] X. Yu, W. Chen, T. Gu, H. Zhang, H. Yuan, S. Kwong, and J. Zhang, "Set-based discrete particle swarm optimization based on decomposition for permutation-based multiobjective combinatorial optimization problems," *IEEE Trans. Cybern.*, vol. 48, no. 7, pp. 2139–2153, 2018. 13

[118] Y. Xiang, Y. Zhou, Z. Chen, and J. Zhang, "A many-objective particle swarm optimizer with leaders selected from historical solutions by using scalar projections," *IEEE Trans. Cybern.*, vol. 50, no. 5, pp. 2209–2222, 2020. 13

[119] L. Ke, Q. Zhang, and R. Battiti, "MOEA/D-ACO: A multiobjective evolutionary algorithm using decomposition and antcolony," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1845–1859, 2013. 13

[120] Z. Zhang, Z. Xu, S. Luan, X. Li, and Y. Sun, "Opposition-based ant colony optimization algorithm for the traveling salesman problem," *Mathematics*, vol. 8, no. 10, pp. 1–16, 2020. 13

[121] J. Ning, Q. Zhao, P. Sun, and Y. Feng, "A multi-objective decomposition-based ant colony optimisation algorithm with negative pheromone," *Journal of Experimental & Theoretical Artificial Intelligence*, 2020, accepted for publication. 13

[122] Y. Wu, Z. Qi, L. Jiang, C. Zhang, and J. Xu, "A multi-objective ant colony optimization algorithm with local optimum avoidance strategy," in *ML4CS'20: Proc. of the 3rd International Conference Machine Learning for Cyber Security*, ser. Lecture Notes in Computer Science, vol. 12487. Springer, 2020, pp. 427–435. 13

[123] Y. Zhong, Y. Xiang, and H. Liu, "A multi-objective artificial bee colony algorithm based on division of the searching space," *Appl. Intell.*, vol. 41, no. 4, pp. 987–1011, 2014. 13

[124] L. Chen, W. Gan, H. Li, X. Xu, L. Cao, and Y. Feng, "A multi-objective cuckoo search algorithm based on decomposition," in *ICACI'19: Proc. of the 11th International Conference on Advanced Computational Intelligence.* IEEE, 2019, pp. 229–233. 13

[125] S. Mirjalili, S. Saremi, S. M. Mirjalili, and L. dos Santos Coelho, "Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization," *Expert Syst. Appl.*, vol. 47, pp. 106–119, 2016. 13

[126] M. Abdel-Basset, R. Mohamed, and S. Mirjalili, "A novel whale optimization algorithm integrated with nelder-mead simplex for multi-objective optimization problems," *Knowl. Based Syst.*, vol. 212, p. 106619, 2021. 13

[127] F. Gao, A. Zhou, and G. Zhang, "An estimation of distribution algorithm based on decomposition for the multiobjective TSP," in *ICNC'12: Proc. of the 8th International Conference on Natural Computation.* IEEE, 2012, pp. 817–821. 13

[128] A. Zhou, F. Gao, and G. Zhang, "A decomposition based estimation of distribution algorithm for multiobjective traveling salesman problems," *Comput. Math. Appl.*, vol. 66, no. 10, pp. 1857–1868, 2013. 13

[129] M. Z. de Souza, R. Santana, A. T. R. Pozo, and A. Mendiburu, "Moea/d-gm: Using probabilistic graphical models in moea/d for solving combinatorial optimization problems," 2015. 13

[130] H. Xing, Z. Wang, T. Li, H. Li, and R. Qu, "An improved MOEA/D algorithm for multi-objective multicast routing with network coding," *Appl. Soft Comput.*, vol. 59, pp. 88–103, 2017. 13

[131] I. Giagkiozis, R. C. Purshouse, and P. J. Fleming, "Generalized decomposition and cross entropy methods for many-objective optimization," *Inf. Sci.*, vol. 282, pp. 363–387, 2014. 13

[132] S. Park and J. Lee, "A multi-objective covariance matrix adaptation evolutionary strategy based on decomposition for analog circuit design," in *IECON'13: Proc. of the 39th Annual Conference of the IEEE Industrial Electronics Society.* IEEE, 2013, pp. 6365–6370. 13

[133] S. Z. Martínez, B. Derbel, A. Liefooghe, D. Brockhoff, H. E. Aguirre, and K. Tanaka, "Injecting CMA-ES into MOEA/D," in *GECCO'15: Proc. of the 2015 Genetic and Evolutionary Computation Conference.* ACM, 2015, pp. 783–790. 13

[134] H. Li, Q. Zhang, and J. Deng, "Biased multiobjective optimization and decomposition algorithm," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 52–66, 2017. 13

[135] D. H. Wolpert, C. E. M. Strauss, and D. G. Rajnarayan, "Advances in distributed optimization using probability collectives," *Adv. Complex Syst.*, vol. 9, no. 4, pp. 383–436, 2006. 13, 20

[136] D. Morgan, A. Waldock, and D. Corne, "MOPC/D: A new probability collectives algorithm for multiobjective optimisation," in *MCDM'13: Proc. of the 2013 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making.* IEEE, 2013, pp. 17–24. 13, 20

[137] A. Zhou, H. Zhao, H. Zhang, and G. Zhang, "Pareto optimal set approximation by models: A linear case," in *EMO'19: Proc. of the 10th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 11411. Springer, 2019, pp. 451–462. 14

[138] S. Wang, A. Zhou, G. Zhang, and F. Fang, "Learning regularity for evolutionary multiobjective search: A generative model-based approach," *IEEE Comput. Intell. Mag.*, vol. 18, no. 4, pp. 29–42, 2023. 14

[139] K. Li, Á. Fialho, S. Kwong, and Q. Zhang, "Adaptive operator selection with bandits for a multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 114–130, 2014. 14

[140] R. A. Gonçalves, C. P. de Almeida, and A. T. R. Pozo, "Upper confidence bound (UCB) algorithms for adaptive operator selection in MOEA/D," in *EMO'15: Proc. of the 8th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 9018.   Springer, 2015, pp. 411–425. 14

[141] Y. Qi, L. Bao, X. Ma, Q. Miao, and X. Li, "Self-adaptive multi-objective evolutionary algorithm based on decomposition for large-scale problems: A case study on reservoir flood control operation," *Inf. Sci.*, vol. 367-368, pp. 529–549, 2016. 14, 18

[142] R. A. Gonçalves, L. M. Pavelski, C. P. de Almeida, J. N. Kuk, S. M. Venske, and M. R. Delgado, "Adaptive operator selection for many-objective optimization with NSGA-III," in *EMO'17: Proc. of the 9th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 10173.   Springer, 2017, pp. 267–281. 14

[143] L. Sun and K. Li, "Adaptive operator selection based on dynamic thompson sampling for MOEA/D," in *PPSN'XVI: Proc. of the 16th International Conference Parallel Problem Solving from Nature*, ser. Lecture Notes in Computer Science, vol. 12270.   Springer, 2020, pp. 271–284. 14

[144] K. Li, Á. Fialho, and S. Kwong, "Multi-objective differential evolution with adaptive control of parameters and operators," in *LION5: Proc. of the 5th International Conference on Learning and Intelligent Optimization*, 2011, pp. 473–487. 14

[145] K. Li, R. Wang, S. Kwong, and J. Cao, "Evolving extreme learning machine paradigm with adaptive operator selection and parameter control," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 21, no. supp02, pp. 143–154, 2013. 14

[146] Q. Lin, Z. Liu, Q. Yan, Z. Du, C. A. C. Coello, Z. Liang, W. Wang, and J. Chen, "Adaptive composite operator selection and parameter control for multiobjective evolutionary algorithm," *Inf. Sci.*, vol. 339, pp. 332–352, 2016. 14

[147] L. Cui, G. Li, Q. Lin, J. Chen, and N. Lu, "Adaptive differential evolution algorithm with novel mutation strategies in multiple sub-populations," *Comput. Oper. Res.*, vol. 67, pp. 155–173, 2016. 14

[148] Q. Lin, C. Tang, Y. Ma, Z. Du, J. Li, J. Chen, and Z. Ming, "A novel adaptive control strategy for decomposition-based multiobjective algorithm," *Comput. Oper. Res.*, vol. 78, pp. 94–107, 2017. 14

[149] W. Wang, S. Yang, Q. Lin, Q. Zhang, K. Wong, C. A. C. Coello, and J. Chen, "An effective ensemble framework for multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 4, pp. 645–659, 2019. 14

[150] Y. Xie, J. Qiao, D. Wang, and B. Yin, "A novel decomposition-based multiobjective evolutionary algorithm using improved multiple adaptive dynamic selection strategies," *Inf. Sci.*, 2020, accepted for publicaiton. 14

[151] N. Luo, W. Lin, G. Jin, C. Jiang, and J. Chen, "Decomposition-based multiobjective evolutionary algorithm with genetically hybrid differential evolution strategy," *IEEE Access*, vol. 9, pp. 2428–2442, 2021. 14

[152] R. A. Gonçalves, J. N. Kuk, C. P. de Almeida, and S. M. Venske, "MOEA/D-HH: A hyper-heuristic for multi-objective problems," in *EMO'15: Proc. of the 8th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 9018. Springer, 2015, pp. 94–108. 14

[153] ——, "Decomposition based multiobjective hyper heuristic with differential evolution," in *IC-CCI'15: Proc. of the 7th International Conference on Computational Collective Intelligence*, ser. Lecture Notes in Computer Science, vol. 9330. Springer, 2015, pp. 129–138. 14

[154] L. Prestes, M. R. de Biase da Silva Delgado, R. A. Gonçalves, C. P. de Almeida, and A. T. R. Pozo, "A hyper-heuristic in MOEA/D-DRA using the upper confidence bound technique," in *BRACIS'17: Proc. of the 2017 Brazilian Conference on Intelligent Systems*. IEEE Computer Society, 2017, pp. 396–401. 14

[155] A. S. Ferreira, R. A. Gonçalves, and A. T. R. Pozo, "A multi-armed bandit selection strategy for hyper-heuristics," in *CEC'17: Proc. of the 2017 IEEE Congress on Evolutionary Computation*. IEEE, 2017, pp. 525–532. 14

[156] L. Prestes, M. R. Delgado, R. Lüders, R. A. Gonçalves, and C. P. de Almeida, "Boosting the performance of MOEA/D-DRA with a multi-objective hyper-heuristic based on irace and UCB method for heuristic selection," in *CEC'18: Proc. of the 2018 IEEE Congress on Evolutionary Computation*. IEEE, 2018, pp. 1–8. 14

[157] R. A. Gonçalves, C. P. de Almeida, R. Lüders, and M. R. Delgado, "A new hyper-heuristic based on a contextual multi-armed bandit for many-objective optimization," in *CEC'18: Proc. of the 2018 IEEE Congress on Evolutionary Computation*. IEEE, 2018, pp. 1–8. 14

[158] C. P. de Almeida, R. A. Gonçalves, S. M. Venske, R. Lüders, and M. R. Delgado, "Hyper-heuristics using multi-armed bandit models for multi-objective optimization," *Appl. Soft Comput.*, vol. 95, p. 106520, 2020. 14

[159] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Comput. J.*, vol. 7, no. 4, pp. 308–313, 1965. 14

[160] S. Z. Martínez, A. A. Montaño, and C. A. C. Coello, "A nonlinear simplex search approach for multi-objective optimization," in *CEC'11: Proc. of the 2011 IEEE Congress on Evolutionary Computation*. IEEE, 2011, pp. 2367–2374. 14

[161] S. Z. Martínez and C. A. C. Coello, "A direct local search mechanism for decomposition-based multi-objective evolutionary algorithms," in *CEC'12: Proc. of the 2012 IEEE Congress on Evolutionary Computation*. IEEE, 2012, pp. 1–8. 14

[162] ——, "MOEA/D assisted by rbf networks for expensive multi-objective optimization problems," in *GECCO'13: Proc. of the 2013 Genetic and Evolutionary Computation Conference*. ACM, 2013, pp. 1405–1412. 14, 17

[163] ——, "Monss: A multi-objective nonlinear simplex search approach," *Eng. Optimiz.*, vol. 48, no. 1, pp. 16–38, 2016. 14

[164] L. Prestes, C. P. de Almeida, and R. A. Gonçalves, "Implementation of quasi-simplex local search on moea/d," in *LA-CCI'15: Proc. of the 2015 Latin America Congress on Computational Intelligence*, 2015, pp. 1–6. 14

[165] H. Zhang, A. Zhou, G. Zhang, and H. K. Singh, "Accelerating MOEA/D by nelder-mead method," in *CEC'17: Proc. of the 2017 IEEE Congress on Evolutionary Computation*. IEEE, 2017, pp. 976–983. 14

[166] K. Deb and M. Abouhawwash, "An optimality theory-based proximity measure for set-based multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 515–528, 2016. 14

[167] K. Deb, M. Abouhawwash, and H. Seada, "A computationally fast convergence measure and implementation for single-, multiple-, and many-objective optimization," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 1, no. 4, pp. 280–293, 2017. 14

[168] M. Abouhawwash, H. Seada, and K. Deb, "Towards faster convergence of evolutionary multi-criterion optimization algorithms using karush kuhn tucker optimality based local search," *Comput. Oper. Res.*, vol. 79, pp. 331–346, 2017. 14

[169] 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, 2019. 14

[170] P. Moscato and M. G. Norman, "A "memetic" approach for the traveling salesman problem implementation of a computational ecology for combinatorial optimization on message-passing systems," in *PCTA'92: Proc. of the 1992 International Conference on Parallel Computing and Transputer Applications.* IOS Press, 1992, pp. 177–186. 14

[171] T. Lust and J. Teghem, "Two-phase pareto local search for the biobjective traveling salesman problem," *J. Heuristics*, vol. 16, no. 3, pp. 475–510, 2010. 14

[172] J. Shi, Q. Zhang, and J. Sun, "PPLS/D: parallel pareto local search based on decomposition," *IEEE Trans. Cybern.*, vol. 50, no. 3, pp. 1060–1071, 2020. 14

[173] S. Hu, X. Wu, H. Liu, Y. Wang, R. Li, and M. Yin, "Multi-objective neighborhood search algorithm based on decomposition for multi-objective minimum weighted vertex cover problem," *Sustainability*, vol. 11, no. 13, p. 3634, 2019. 14

[174] X. Cai, M. Hu, D. Gong, Y. Guo, Y. Zhang, Z. Fan, and Y. Huang, "A decomposition-based co-evolutionary multiobjective local search for combinatorial multiobjective optimization," *Swarm Evol. Comput.*, vol. 49, pp. 178–193, 2019. 14

[175] Y. Zhou, L. Kong, Y. Cai, Z. Wu, S. Liu, J. Hong, and K. Wu, "A decomposition-based local search for large-scale many-objective vehicle routing problems with simultaneous delivery and pickup and time windows," *IEEE Syst. J.*, vol. 14, no. 4, pp. 5253–5264, 2020. 14

[176] J. Wang, L. Yuan, Z. Zhang, S. Gao, Y. Sun, and Y. Zhou, "Multiobjective multiple neighborhood search algorithms for multiobjective fleet size and mix location-routing problem with time windows," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 51, no. 4, pp. 2284–2298, 2021. 14

[177] A. Alhindi and Q. Zhang, "MOEA/D with tabu search for multiobjective permutation flow shop scheduling problems," in *CEC'14: Proc. of the 2014 IEEE Congress on Evolutionary Computation.* IEEE, 2014, pp. 1155–1164. 15

[178] Y. Zhou, J. Wang, Z. Wu, and K. Wu, "A multi-objective tabu search algorithm based on decomposition for multi-objective unconstrained binary quadratic programming problem," *Knowl. Based Syst.*, vol. 141, pp. 18–30, 2018. 15

[179] Y. Du, L. Xing, J. Zhang, Y. Chen, and Y. He, "MOEA based memetic algorithms for multi-objective satellite range scheduling problem," *Swarm Evol. Comput.*, vol. 50, 2019. 15

[180] A. Alhindi, A. Alhindi, A. Alhejali, A. Alsheddy, N. Tairan, and H. Alhakami, "MOEA/D-GLS: a multiobjective memetic algorithm using decomposition and guided local search," *Soft Comput.*, vol. 23, no. 19, pp. 9605–9615, 2019. 15

[181] S. Kirkpatrick, C. D. G. Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983. 15

[182] H. Li and D. Landa-Silva, "Evolutionary multi-objective simulated annealing with adaptive and competitive search direction," in *CEC'08: Proc of the 2008 IEEE Congress on Evolutionary Computation.* IEEE, 2008, pp. 3311–3318. 15

[183] ——, "An adaptive evolutionary multi-objective approach based on simulated annealing," *Evol. Comput.*, vol. 19, no. 4, pp. 561–595, 2011. 15

[184] X. Cai, X. Cheng, Z. Fan, E. D. Goodman, and L. Wang, "An adaptive memetic framework for multi-objective combinatorial optimization problems: studies on software next release and travelling salesman problems," *Soft Comput.*, vol. 21, no. 9, pp. 2215–2236, 2017. 15

[185] S. Jiang and L. Zhang, "Energy-oriented scheduling for hybrid flow shop with limited buffers through efficient multi-objective optimization," *IEEE Access*, vol. 7, pp. 34 477–34 487, 2019. 15

[186] A. Kafafy, A. Bounekkar, and S. Bonnevay, "A hybrid evolutionary metaheuristics (HEMH) applied on 0/1 multiobjective knapsack problems," in *GECCO'11: Proc. of the 13th Annual Genetic and Evolutionary Computation Conference.* ACM, 2011, pp. 497–504. 15

[187] A. Alhindi, Q. Zhang, and E. P. K. Tsang, "Hybridisation of decomposition and GRASP for combinatorial multiobjective optimisation," in *UKCI'14: Proc. of the 14th UK Workshop on Computational Intelligence.* IEEE, 2014, pp. 1–7. 15

[188] Y. Lei, J. Shi, and Z. Yan, "A memetic algorithm based on MOEA/D for the examination timetabling problem," *Soft Comput.*, vol. 22, no. 5, pp. 1511–1523, 2018. 15

[189] A. Gharaei and F. Jolai, "A multi-agent approach to the integrated production scheduling and distribution problem in multi-factory supply chain," *Appl. Soft Comput.*, vol. 65, pp. 577–589, 2018. 15

[190] 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, 2019. 15, 16

[191] M. A. Jan and Q. Zhang, "Moea/d for constrained multiobjective optimization: Some preliminary experimental results," in *UKCI'10: Proc. of the 2010 UK Workshop on Computational Intelligence.* IEEE, 2010, pp. 1–6. 15

[192] M. Asafuddoula, T. Ray, R. A. Sarker, and K. Alam, "An adaptive constraint handling approach embedded MOEA/D," in *CEC'12: Proc. of the 2012 IEEE Congress on Evolutionary Computation.* IEEE, 2012, pp. 1–8. 15

[193] M. Asafuddoula, T. Ray, and R. A. Sarker, "A decomposition-based evolutionary algorithm for many objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 3, pp. 445–460, 2015. 15

[194] M. A. Jan and R. A. Khanum, "A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D," *Appl. Soft Comput.*, vol. 13, no. 1, pp. 128–148, 2013. 15

[195] Z. Yang, X. Cai, and Z. Fan, "Epsilon constrained method for constrained multiobjective optimization problems: some preliminary results," in *GECCO'14: Proc. of the 2014 Genetic and Evolutionary Computation Conference.* ACM, 2014, pp. 1181–1186. 15

[196] Z. Fan, W. Li, X. Cai, H. Huang, Y. Fang, Y. You, J. Mo, C. Wei, and E. D. Goodman, "An improved epsilon constraint-handling method in MOEA/D for CMOPs with large infeasible regions," *Soft Comput.*, vol. 23, no. 23, pp. 12 491–12 510, 2019. 15

[197] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, and E. D. Goodman, "Push and pull search for solving constrained multi-objective optimization problems," *Swarm Evol. Comput.*, vol. 44, pp. 665–679, 2019. 15

[198] Z. Fan, Z. Wang, W. Li, Y. Yuan, Y. You, Z. Yang, F. Sun, and J. Ruan, "Push and pull search embedded in an M2M framework for solving constrained multi-objective optimization problems," *Swarm Evol. Comput.*, vol. 54, p. 100651, 2020. 15

[199] C. Peng, H. Liu, and F. Gu, "An evolutionary algorithm with directed weights for constrained multi-objective optimization," *Appl. Soft Comput.*, vol. 60, pp. 613–622, 2017. 16

[200] Y. Tian, T. Zhang, J. Xiao, X. Zhang, and Y. Jin, "A coevolutionary framework for constrained multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 102–116, 2021. 16

[201] R. Jiao, S. Zeng, C. Li, S. Yang, and Y. Ong, "Handling constrained many-objective optimization problems via problem transformation," *IEEE Trans. Cybern.*, vol. 51, no. 10, pp. 4834–4847, 2021. 16

[202] S. Li, K. Li, and W. Li, "Do we really need to use constraint violation in constrained evolutionary multi-objective optimization?" in *PPSN'22: Proc. of 17th International Conference on Parallel Problem Solving from Nature*, ser. Lecture Notes in Computer Science, vol. 13399. Springer, 2022, pp. 124–137. 16

[203] J. J. Liang, K. Qiao, K. Yu, B. Qu, C. Yue, W. Guo, and L. Wang, "Utilizing the relationship between unconstrained and constrained pareto fronts for constrained multiobjective optimization," *IEEE Trans. Cybern.*, vol. 53, no. 6, pp. 3873–3886, 2023. 16

[204] S. Xie, K. Li, W. Wang, H. Wang, C. Peng, and H. Jalil, "A tractive population assisted dual-population and two-phase evolutionary algorithm for constrained multi-objective optimization," *IEEE Trans. Evol. Comput.*, 2023, accepted for publication. 16

[205] J. Zou, R. Sun, Y. Liu, Y. Hu, S. Yang, J. Zheng, and K. Li, "A multipopulation evolutionary algorithm using new cooperative mechanism for solving multiobjective problems with multiconstraint," *IEEE Trans. Evol. Comput.*, vol. 28, no. 1, pp. 267–280, 2024. 16

[206] A. Muruganantham, K. C. Tan, and P. Vadakkepat, "Evolutionary dynamic multiobjective optimization via Kalman filter prediction," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2862–2873, 2016. 16

[207] S. B. Gee, K. C. Tan, and C. Alippi, "Solving multiobjective optimization problems in unknown dynamic environments: An inverse modeling approach," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4223–4234, 2017. 16

[208] L. Cao, L. Xu, E. D. Goodman, and H. Li, "Decomposition-based evolutionary dynamic multiobjective optimization using a difference model," *Appl. Soft Comput.*, vol. 76, pp. 473–490, 2019. 16

[209] L. Cao, L. Xu, E. D. Goodman, C. Bao, and S. Zhu, "Evolutionary dynamic multiobjective optimization assisted by a support vector regression predictor," *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 305–319, 2020. 16

[210] R. Rambabu, P. Vadakkepat, K. C. Tan, and M. Jiang, "A mixture-of-experts prediction framework for evolutionary dynamic multiobjective optimization," *IEEE Trans. Cybern.*, vol. 50, no. 12, pp. 5099–5112, 2020. 16

[211] R. Chen, K. Li, and X. Yao, "Dynamic multiobjectives optimization with a changing number of objectives," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 157–171, 2018. 16

[212] Y. Hu, J. Zheng, J. Zou, S. Yang, J. Ou, and R. Wang, "A dynamic multi-objective evolutionary algorithm based on intensity of environmental change," *Inf. Sci.*, vol. 523, pp. 49–62, 2020. 16

[213] A. Ahrari, S. M. Elsayed, R. A. Sarker, D. Essam, and C. A. C. Coello, "Weighted pointwise prediction method for dynamic multiobjective optimization," *Inf. Sci.*, vol. 546, pp. 349–367, 2021. 16

[214] H. Peng, C. Mei, S. Zhang, Z. Luo, Q. Zhang, and Z. Wu, "Multi-strategy dynamic multi-objective evolutionary algorithm with hybrid environmental change responses," *Swarm Evol. Comput.*, vol. 82, p. 101356, 2023. 16

[215] M. Jiang, Z. Wang, L. Qiu, S. Guo, X. Gao, and K. C. Tan, "A fast dynamic evolutionary multiobjective algorithm via manifold transfer learning," *IEEE Trans. Cybern.*, vol. 51, no. 7, pp. 3417–3428, 2021. 16

[216] M. Jiang, Z. Wang, H. Hong, and G. G. Yen, "Knee point-based imbalanced transfer learning for dynamic multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 117–129, 2021. 16

[217] Q. Lin, Y. Ye, L. Ma, M. Jiang, and K. C. Tan, "Dynamic multiobjective evolutionary optimization via knowledge transfer and maintenance," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 54, no. 2, pp. 936–949, 2024. 16

[218] Z. Liang, S. Zheng, Z. Zhu, and S. Yang, "Hybrid of memory and prediction strategies for dynamic multiobjective optimization," *Inf. Sci.*, vol. 485, pp. 200–218, 2019. 16

[219] R. Liu, J. Li, Y. Jin, and L. Jiao, "A self-adaptive response strategy for dynamic multiobjective evolutionary optimization based on objective space decomposition," *Evol. Comput.*, vol. 29, no. 4, pp. 491–519, 2021. 16

[220] C. Jiang, F. Ge, D. Chen, and H. Liu, "Ensemble methods based on characterization of dynamism for dynamic multi-objective optimization," *Appl. Soft Comput.*, vol. 129, p. 109605, 2022. 16

[221] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Glob. Optim.*, vol. 13, no. 4, pp. 455–492, 1998. 16

[222] Q. Zhang, W. Liu, E. P. K. Tsang, and B. Virginas, "Expensive multiobjective optimization by MOEA/D with gaussian process model," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 456–474, 2010. 17

[223] T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, and K. Sindhya, "A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 129–142, 2018. 17

[224] K. Li and R. Chen, "Batched data-driven evolutionary multiobjective optimization based on manifold interpolation," *IEEE Trans. Evol. Comput.*, vol. 27, no. 1, pp. 126–140, 2023. 17

[225] R. Chen and K. Li, "Surrogate-assisted evolutionary multi-objective optimization for hardware design space exploration," in *GPSMDS'22: Proc. of the NeurIPS Workshop on Gaussian Processes, Spatiotemporal Modeling, and Decision-making Systems*, 2022, pp. 1–10. 17

[226] ——, "Data-driven evolutionary multi-objective optimization based on multiple-gradient descent for disconnected pareto fronts," in *EMO'23: Proc. of the 12th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 13970. Springer, 2023, pp. 56–70. 17

[227] S. Z. Martínez and C. A. C. Coello, "Combining surrogate models and local search for dealing with expensive multi-objective optimization problems," in *CEC'13: Proc. of the 2013 IEEE Congress on Evolutionary Computation.* IEEE, 2013, pp. 2572–2579. 17

[228] L. M. Pavelski, M. R. Delgado, C. P. de Almeida, R. A. Gonçalves, and S. M. Venske, "ELMOEA/D-DE: extreme learning surrogate models in multi-objective optimization based on decomposition and differential evolution," in *BRACIS'14: Proc. of the 2014 Brazilian Conference on Intelligent Systems.* IEEE Computer Society, 2014, pp. 318–323. 17

[229] ——, "Extreme learning surrogate models in multi-objective optimization based on decomposition," *Neurocomputing*, vol. 180, pp. 55–67, 2016. 17

[230] D. Zhang, J. Liao, X. Huang, and J. Jin, "A multifractal-guided multilevel surrogate model-based evolutionary algorithm for expensive multiobjective problems," *J. Circuits Syst. Comput.*, vol. 26, no. 7, pp. 1 750 109:1–1 750 109:18, 2017. 17

[231] C. He, R. Cheng, Y. Jin, and X. Yao, "Surrogate-assisted expensive many-objective optimization by model fusion," in *CEC'19: Proc. of the 2019 IEEE Congress on Evolutionary Computation.* IEEE, 2019, pp. 1672–1679. 17

[232] T. Sonoda and M. Nakata, "Moea/d-s³: MOEA/D using svm-based surrogates adjusted to subproblems for many objective optimization," in *CEC'20: Proc. of the 2020 IEEE Congress on Evolutionary Computation.* IEEE, 2020, pp. 1–8. 17

[233] G. Pruvost, B. Derbel, A. Liefooghe, S. Vérel, and Q. Zhang, "Surrogate-assisted multi-objective combinatorial optimization based on decomposition and Walsh basis," in *GECCO'20: Proc. of the 2020 Genetic and Evolutionary Computation Conference.* ACM, 2020, pp. 542–550. 17

[234] Z. Feng, Q. Zhang, Q. Zhang, Q. Tang, T. Yang, and Y. Ma, "A multiobjective optimization based framework to balance the global exploration and local exploitation in expensive optimization," *J. Glob. Optim.*, vol. 61, no. 4, pp. 677–694, 2015. 17

[235] X. Wang, Y. Jin, S. Schmitt, and M. Olhofer, "An adaptive bayesian approach to surrogate-assisted evolutionary multi-objective optimization," *Inf. Sci.*, vol. 519, pp. 317–331, 2020. 17

[236] N. Namura, K. Shimoyama, and S. Obayashi, "Expected improvement of penalty-based boundary intersection for expensive multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 6, pp. 898–913, 2017. 17

[237] F. Li, L. Gao, A. Garg, W. Shen, and S. Huang, "Two infill criteria driven surrogate-assisted multi-objective evolutionary algorithms for computationally expensive problems with medium dimensions," *Swarm Evol. Comput.*, vol. 60, p. 100774, 2021. 17

[238] 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, 2019. 17

[239] M. Zhao, K. Zhang, G. Chen, X. Zhao, C. Yao, H. Sun, Z. Huang, and J. Yao, "A surrogate-assisted multi-objective evolutionary algorithm with dimension-reduction for production optimization," *Journal of Petroleum Science and Engineering*, vol. 192, p. 107192, 2020. 17

[240] Z. Tan and H. Wang, "A kriging-assisted evolutionary algorithm using feature selection for expensive sparse multi-objective optimization," in *CEC'20: Proc. of the 2020 IEEE Congress on Evolutionary Computation.* IEEE, 2020, pp. 1–8. 17

[241] A. T. W. Min, Y. Ong, A. Gupta, and C. K. Goh, "Multiproblem surrogates: Transfer evolutionary multiobjective optimization of computationally expensive problems," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 15–28, 2019. 17

[242] C. Yang, J. Ding, Y. Jin, and T. Chai, "Offline data-driven multiobjective optimization: Knowledge transfer between surrogates and generation of final solutions," *IEEE Trans. Evol. Comput.*, vol. 24, no. 3, pp. 409–423, 2020. 17

[243] J. Luo, A. Gupta, Y. Ong, and Z. Wang, "Evolutionary optimization of expensive multiobjective problems with co-sub-pareto front gaussian process surrogates," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1708–1721, 2019. 17

[244] X. Fan, K. Li, and K. C. Tan, "Surrogate assisted evolutionary algorithm based on transfer learning for dynamic expensive multi-objective optimisation problems," in *IEEE Congress on Evolutionary Computation, CEC 2020, Glasgow, United Kingdom, July 19-24, 2020.* IEEE, 2020, pp. 1–8. 17

[245] R. Chen and K. Li, "Multioutput surrogate assisted evolutionary algorithm for expensive multi-modal optimization problems," in *SMC'23: Proc. of 2023 IEEE International Conference on Systems, Man, and Cybernetics.* IEEE, 2023, pp. 5176–5181. 17

[246] K. Li, R. Chen, and X. Yao, "A data-driven evolutionary transfer optimization for expensive problems in dynamic environments," *IEEE Trans. Evol. Comput.*, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10225543 17

[247] K. Li, M. Liao, K. Deb, G. Min, and X. Yao, "Does preference always help? A holistic study on preference-based evolutionary multiobjective optimization using reference points," *IEEE Trans. Evol. Comput.*, vol. 24, no. 6, pp. 1078–1096, 2020. 17

[248] H. Wang, M. Olhofer, and Y. Jin, "A mini-review on preference modeling and articulation in multi-objective optimization: current status and challenges," *Complex & Intelligent Systems*, vol. 3, no. 4, pp. 233–245, 2017. 17

[249] B. Xin, L. Chen, J. Chen, H. Ishibuchi, K. Hirota, and B. Liu, "Interactive multiobjective optimization: A review of the state-of-the-art," *IEEE Access*, vol. 6, pp. 41 256–41 279, 2018. 17

[250] K. Li, R. Chen, G. Min, and X. Yao, "Integration of preferences in decomposition multiobjective optimization," *IEEE Trans. Cybern.*, vol. 48, no. 12, pp. 3359–3370, 2018. 17

[251] A. Mohammadi, M. N. Omidvar, and X. Li, "Reference point based multi-objective optimization through decomposition," in *CEC'12: Proc. of the IEEE Congress on Evolutionary Computation.* IEEE, 2012, pp. 1–8. 17

[252] A. Mohammadi, M. N. Omidvar, X. Li, and K. Deb, "Integrating user preferences and decomposition methods for many-objective optimization," in *CEC'14: Proc. of the IEEE Congress on Evolutionary Computation.* IEEE, 2014, pp. 421–428. 17

[253] H. Zhu, Z. He, and Y. Jia, "An improved reference point based multi-objective optimization by decomposition," *Int. J. Mach. Learn. Cybern.*, vol. 7, no. 4, pp. 581–595, 2016. 17

[254] R. Wang, R. C. Purshouse, and P. J. Fleming, "On finding well-spread pareto optimal solutions by preference-inspired co-evolutionary algorithm," in *GECCO'13: Proc. of the 2013 Genetic and Evolutionary Computation Conference.* ACM, 2013, pp. 695–702. 17

[255] M. Pilát and R. Neruda, "Incorporating user preferences in MOEA/D through the coevolution of weights," in *GECCO'15: Proc. of the Genetic and Evolutionary Computation Conference.* ACM, 2015, pp. 727–734. 17

[256] I. R. Meneghini and F. G. Guimarães, "Evolutionary method for weight vector generation in multi-objective evolutionary algorithms based on decomposition and aggregation," in *CEC'17: Proc. of 2017 IEEE Congress on Evolutionary Computation.* IEEE, 2017, pp. 1900–1907. 17

[257] X. Ma, F. Liu, Y. Qi, L. Li, L. Jiao, X. Deng, X. Wang, B. Dong, Z. Hou, Y. Zhang, and J. Wu, "MOEA/D with biased weight adjustment inspired by user preference and its application on multi-objective reservoir flood control problem," *Soft Comput.*, vol. 20, no. 12, pp. 4999–5023, 2016. 17

[258] G. Yu, J. Zheng, R. Shen, and M. Li, "Decomposing the user-preference in multiobjective optimization," *Soft Comput.*, vol. 20, no. 10, pp. 4005–4021, 2016. 17

[259] A. Jaszkiewicz, "Interactive multiobjective optimization with the pareto memetic algorithm," in *Proc. of the 4th EU/ME Workshop: Design and Evaluation of Advanced Hybrid Meta-heuristics*, 2004, pp. 15–20. 18

[260] M. Gong, F. Liu, W. Zhang, L. Jiao, and Q. Zhang, "Interactive MOEA/D for multi-objective decision making," in *GECCO'11: Proc. of the 13th Annual Genetic and Evolutionary Computation Conference.* ACM, 2011, pp. 721–728. 18

[261] K. Li, "Progressive preference learning: Proof-of-principle results in MOEA/D," in *EMO'19: Proc. of the 10th International Conference on Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, vol. 11411. Springer, 2019, pp. 631–643. 18

[262] K. Li, R. Chen, D. A. Savic, and X. Yao, "Interactive decomposition multiobjective optimization via progressively learned value functions," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 5, pp. 849–860, 2019. 18

[263] G. Lai, M. Liao, and K. Li, "Empirical studies on the role of the decision maker in interactive evolutionary multi-objective optimization," in *CEC'21: Proc. of 2021 IEEE Congress on Evolutionary Computation.* IEEE, 2021, pp. 185–192. 18

[264] K. Li, G. Lai, and X. Yao, "Interactive evolutionary multiobjective optimization via learning to rank," *IEEE Trans. Evol. Comput.*, vol. 27, no. 4, pp. 749–763, 2023. 18

[265] M. K. Tomczyk and M. Kadzinski, "Decomposition-based interactive evolutionary algorithm for multiple objective optimization," *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 320–334, 2020. 18

[266] X. Ma, F. Liu, Y. Qi, X. Wang, L. Li, L. Jiao, M. Yin, and M. Gong, "A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 275–298, 2016. 18

[267] B. Cao, J. Zhao, Z. Lv, and X. Liu, "A distributed parallel cooperative coevolutionary multi-objective evolutionary algorithm for large-scale optimization," *IEEE Trans. Ind. Informatics*, vol. 13, no. 4, pp. 2030–2038, 2017. 18

[268] B. Cao, J. Zhao, Y. Gu, Y. Ling, and X. Ma, "Applying graph-based differential grouping for multiobjective large-scale optimization," *Swarm Evol. Comput.*, vol. 53, p. 100626, 2020. 18

[269] L. Ma, M. Huang, S. Yang, R. Wang, and X. Wang, "An adaptive localized decision variable analysis approach to large-scale multiobjective and many-objective optimization," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 6684–6696, 2022. 18

[270] O. Schütze, M. Vasile, and C. A. C. Coello, "Computing the set of epsilon-efficient solutions in multiobjective space mission design," *J. Aerosp. Comput. Inf. Commun.*, vol. 8, no. 3, pp. 53–70, 2011. 18

[271] T. Hiroyasu, S. Nakayama, and M. Miki, "Comparison study of SPEA2+, SPEA2, and NSGA-II in diesel engine emissions and fuel economy problem," in *CEC'05: Proc. of the 2005 IEEE Congress on Evolutionary Computation.* IEEE, 2005, pp. 236–242. 18

[272] F. Kudo, T. Yoshikawa, and T. Furuhashi, "A study on analysis of design variables in pareto solutions for conceptual design optimization problem of hybrid rocket engine," in *CEC'11: Proc. of the 2011 IEEE Congress on Evolutionary Computation.* IEEE, 2011, pp. 2558–2562. 18

[273] R. Tanabe and H. Ishibuchi, "A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization," in *PPSN XV: Proc. of the 15th International Conference on Parallel Problem Solving from Nature*, ser. Lecture Notes in Computer Science, vol. 11101. Springer, 2018, pp. 249–261. 18

[274] ——, "A framework to handle multimodal multiobjective optimization in decomposition-based evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 24, no. 4, pp. 720–734, 2020. 18

[275] M. Pal and S. Bandyopadhyay, "Decomposition in decision and objective space for multi-modal multi-objective optimization," *Swarm Evol. Comput.*, vol. 62, p. 100842, 2021. 18

[276] Z. Wei, W. Gao, M. Gong, and G. G. Yen, "A bi-objective evolutionary algorithm for multimodal multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 28, no. 1, pp. 168–177, 2024. 19

[277] G. Soremekun, Z. Gürdal, R. Haftka, and L. Watson, "Composite laminate design optimization by genetic algorithm with generalized elitist selection," *Computers and Structures*, vol. 79, no. 2, pp. 131–143, 2001. 19

[278] C. Ting, C. Lee, H. Chang, and J. Wu, "Wireless heterogeneous transmitter placement using multiobjective variable-length genetic algorithm," *IEEE Trans. Syst. Man Cybern. Part B*, vol. 39, no. 4, pp. 945–958, 2009. 19

[279] J. F. Herbert-Acero, O. Probst, P.-E. Réthoré, G. C. Larsen, and K. K. Castillo-Villar, "A review of methodological approaches for the design and optimization of wind farms," *Energies*, vol. 7, no. 11, pp. 6930–7016, 2014. 19

[280] H. Li and K. Deb, "Challenges for evolutionary multiobjective optimization algorithms in solving variable-length problems," in *CEC'17: Proc. of the 2017 IEEE Congress on Evolutionary Computation*. IEEE, 2017, pp. 2217–2224. 19

[281] H. Li, K. Deb, and Q. Zhang, "Variable-length pareto optimization via decomposition-based evolutionary multiobjective algorithm," *IEEE Trans. Evol. Comput.*, vol. 23, no. 6, pp. 987–999, 2019. 19

[282] L. Ma, Y. Liu, G. Yu, X. Wang, H. Mo, G. Wang, Y. Jin, and Y. Tan, "Decomposition-based multiobjective optimization for variable-length mixed-variable pareto optimization and its application in cloud service allocation," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 53, no. 11, pp. 7138–7151, 2023. 19

[283] H. Li, X. Su, Z. Xu, and Q. Zhang, "MOEA/D with iterative thresholding algorithm for sparse optimization problems," in *PPSN'12: Proc. of 2012 International Conference on Parallel Problem Solving from Nature*, vol. 7492. Springer, 2012, pp. 93–101. 19

[284] H. Li, J. Sun, M. Wang, and Q. Zhang, "MOEA/D with chain-based random local search for sparse optimization," *Soft Comput.*, vol. 22, no. 21, pp. 7087–7102, 2018. 19

[285] H. Li, Q. Zhang, J. Deng, and Z. Xu, "A preference-based multiobjective evolutionary approach for sparse optimization," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 29, no. 5, pp. 1716–1731, 2018. 19

[286] W. Liao, H. Ishibuchi, L. M. Pang, and K. Shang, "Parallel implementation of MOEA/D with parallel weight vectors for feature selection," in *SMC'20: Proc. of the 2020 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, 2020, pp. 1524–1531. 19

[287] J. Yu, L. Li, and Y. Qi, "Parallel MOEA/D for real-time multi-objective optimization problems," in *Edutainment'18: Proc. of the 2018 International Conference on E-Learning and Games*, ser. Lecture Notes in Computer Science, vol. 11462. Springer, 2018, pp. 236–240. 19

[288] X. Ma, Q. Zhang, G. Tian, J. Yang, and Z. Zhu, "On Tchebycheff decomposition approaches for multiobjective evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 2, pp. 226–244, 2018. 20

[289] H. K. Singh and K. Deb, "Investigating the equivalence between PBI and AASF scalarization for multi-objective optimization," *Swarm Evol. Comput.*, vol. 53, p. 100630, 2020. 20

[290] Z. Huang, Y. Zhou, Z. Chen, X. He, X. Lai, and X. Xia, "Running time analysis of MOEA/D on pseudo-boolean functions," *IEEE Trans. Cybern.*, vol. 51, no. 10, pp. 5130–5141, 2021. 20

[291] Z. Huang, Y. Zhou, C. Luo, and Q. Lin, "A runtime analysis of typical decomposition approaches in MOEA/D framework for many-objective optimization problems," in *IJCAI'21: Proc. of the Thirtieth International Joint Conference on Artificial Intelligence*, 2021, pp. 1682–1688. 20

[292] S. Wietheger and B. Doerr, "A mathematical runtime analysis of the non-dominated sorting genetic algorithm III (NSGA-III)," in *IJCAI'23: Proc. of the Thirty-Second International Joint Conference on Artificial Intelligence.* ijcai.org, 2023, pp. 5657–5665. 20