Evolutionary Large-Scale Multi-Objective Optimization: A Survey

YE TIAN, LANGCHUN SI, and XINGYI ZHANG, Anhui University
RAN CHENG and CHENG HE, Southern University of Science and Technology
KAY CHEN TAN, The Hong Kong Polytechnic University
YAOCHU JIN, University of Surrey

Multi-objective evolutionary algorithms (MOEAs) have shown promising performance in solving various optimization problems, but their performance may deteriorate drastically when tackling problems containing a large number of decision variables. In recent years, much effort been devoted to addressing the challenges brought by large-scale multi-objective optimization problems. This article presents a comprehensive survey of stat-of-the-art MOEAs for solving large-scale multi-objective optimization problems. We start with a categorization of these MOEAs into decision variable grouping based, decision space reduction based, and novel search strategy based MOEAs, discussing their strengths and weaknesses. Then, we review the benchmark problems for performance assessment and a few important and emerging applications of MOEAs for large-scale multi-objective optimization. Last, we discuss some remaining challenges and future research directions of evolutionary large-scale multi-objective optimization.

CCS Concepts: • **General and reference** → **Surveys and overviews**; • **Theory of computation** → **Theory of randomized search heuristics**; • **Computing methodologies** → **Genetic algorithms;**

Additional Key Words and Phrases: Multi-objective optimization, large-scale optimization, evolutionary computation

This work was supported in part by the National Key R&D Program of China under grant 2018AAA0100100; in part by the National Natural Science Foundation of China under grants 61822301, 61876123, 61906001, 61906081, 61903178, and U20A20306; in part by the Hong Kong Scholars Program under grant XJ2019035; in part by the Anhui Provincial Natural Science Foundation under grant 1908085QF271; in part by the Collaborative Innovation Program of Anhui under grant GXXT-2020-051; in part by the Research Grants Council of the Hong Kong Special Administrative Region under grants PolyU11202418 and PolyU11209219; and in part by a Royal Society International Exchanges Program under grant IEC\NSFC\170279.

Authors' addresses: Y. Tian, Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Institutes of Physical Science and Information Technology, Anhui University, Hefei, 230601, China; email: field910921@gmail.com; L. Si, Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, Anhui University, Hefei, 230601, China; email: spring_slc@163.com; X. Zhang (corresponding author), Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Artificial Intelligence, Anhui University, Hefei, 230601, China; email: xyzhanghust@gmail.com; R. Cheng and C. He, Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, 518055, China; emails: {ranchengcn, chenghehust}@gmail.com; K. C. Tan, Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR; email: kaychen.tan@polyu.edu.hk; Y. Jin, Department of Computer Science, University of Surrey, Surrey, GU2 7XH, Guildford, U. K.; email: yaochu.jin@surrey.ac.uk.

0360-0300/2021/10-ART174 $15.00
https://doi.org/10.1145/3470971

ACM Reference format:
Ye Tian, Langchun Si, Xingyi Zhang, Ran Cheng, Cheng He, Kay Chen Tan, and Yaochu Jin. 2021. Evolutionary Large-Scale Multi-Objective Optimization: A Survey. *ACM Comput. Surv.* 54, 8, Article 174 (October 2021), 34 pages.
https://doi.org/10.1145/3470971

1 INTRODUCTION

In many scientific and engineering areas such as artificial intelligence [77], data mining [167], software engineering [187], scheduling [80], bioinformatics [166], and economics [12], there exist a variety of optimization problems containing multiple objectives and a large number of decision variables to be optimized. Although mathematical programming methods can quickly converge to a single optimum in the high-dimensional decision space, the population based evolutionary algorithms are better at obtaining a set of optimal solutions trading off between multiple conflicting objectives. However, conventional evolutionary algorithms converge more slowly than mathematical programming methods since the decision space grows exponentially with the number of decision variables, which is known as the *curse of dimensionality* [165].

To address this issue, many reproduction operators [171, 177] and local search strategies [109, 168] have been customized in evolutionary algorithms for solving the **large-scale multi-objective optimization problems (LSMOPs)** in specific areas. Moreover, a number of generic **multi-objective evolutionary algorithms (MOEAs)** have also been developed for solving LSMOPs since 2013 [3]. These MOEAs are high-level methodologies that do not rely on the specific characteristics of problems, most of which tackle the high-dimensional decision space via decision variable grouping, decision space reduction, and novel search strategies as illustrated in Figure 1. To empirically study the algorithmic performance of different MOEAs in solving LSMOPs, some benchmark test suites have also been developed in recent years [26, 171].

Due to the broad application scenarios of evolutionary large-scale multi-objective optimization, research on this topic is attracting increasing attention in the evolutionary computation community. Although some work has briefly analyzed the performance of some MOEAs on LSMOPs [220, 224], no comprehensive survey of evolutionary large-scale multi-objective optimization has been presented so far. Although the MOEAs for many other optimization problems such as many-objective optimization problems [87], dynamic multi-objective optimization problems [66], constrained multi-objective optimization problems [46], computationally expensive multi-objective optimization problems [28], and multimodal multi-objective optimization problems [159] have been well surveyed, this article focuses on the survey of evolutionary large-scale multi-objective optimization in terms of the following aspects:

- Based on the taxonomy shown in Figure 1, the three categories of MOEAs are introduced separately. In addition, the advantages and limitations of each category of MOEAs are discussed. The detailed introduction to existing MOEAs for large-scale multi-objective optimization aims at giving researchers a deep understanding of the state-of-the-art techniques for solving LSMOPs, as well as providing practitioners with a tutorial guidance when solving different LSMOPs.
- Benchmark LSMOPs for assessing the algorithmic performance of MOEAs are introduced, which can help researchers empirically compare the performance of different MOEAs and develop new MOEAs. Moreover, the introduced benchmark LSMOPs and many existing MOEAs have been implemented in the evolutionary multi-objective optimization platform[1]

[1]https://github.com/BIMK/PlatEMO.

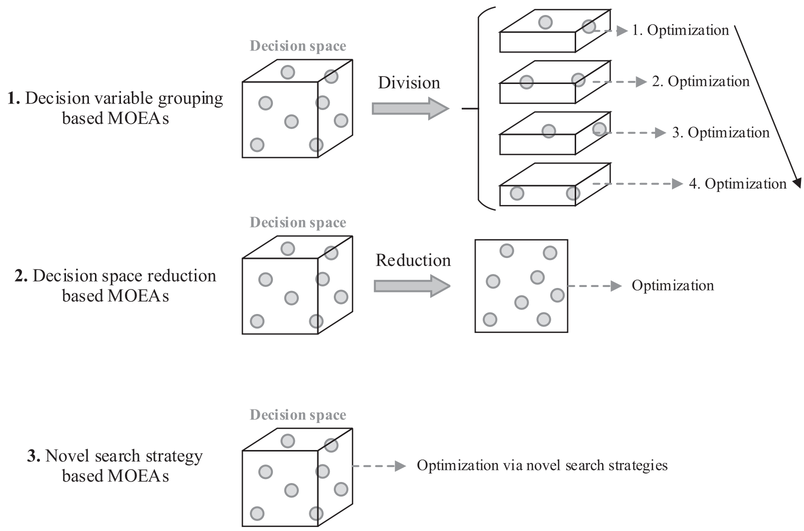

Fig. 1. Core techniques for solving LSMOPs, including decision variable grouping, decision space reduction, and novel search strategies.

[161] built by us, with which users can easily conduct comparative studies via one-click operation.

- The applications of evolutionary large-scale multi-objective optimization in some popular and emerging areas are introduced, covering the fields of machine learning, network science, vehicle routing, and economics. This part makes practitioners know whether their LSMOPs have been successfully handled by MOEAs and inspires practitioners to solve LSMOPs in other similar areas via MOEAs.
- Although many general LSMOPs have already been well solved by existing MOEAs, many challenges in research on LSMOPs still remain, such as the LSMOPs with constrained, sparse, computationally expensive, multimodal, or uncertain objective functions. Thus, this article also outlines some promising topics for future research, to promote the development of MOEAs on more types of challenging LSMOPs.

The rest of this article is organized as follows. The basic concepts of evolutionary large-scale multi-objective optimization are presented in Section 2. Then, the three categories of MOEAs for solving LSMOPs are introduced in Sections 3, 4, and 5, respectively. Afterward, the benchmark LSMOPs are introduced in Section 6, followed by some applications introduced in Section 7 and future research directions outlined in Section 8. Conclusions are drawn in Section 9.

2 BACKGROUND

The definition of multi-objective optimization problems is mathematically described as

$$\min\ \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})), \tag{1}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ denotes the d-dimensional decision vector of a solution in the decision space Ω, and $\mathbf{f}(\mathbf{x})$ denotes the objective vector containing m conflicting functions to be minimized. In particular, (1) is generally called an *LSMOP* if the number of decision variables d is equal to or larger than 100 [23, 26]. Since evolutionary algorithms are metaheuristics that do not need the specific characteristics of functions, the functions $\mathbf{f}(\mathbf{x})$ can be convex or nonconvex, differentiable or nondifferentiable, continuous or discrete, and unimodal or multimodal. In addition, the LSMOPs defined in (1) do not contain any constraint by default.

Definition 2.1 (Pareto Dominance). For any two solutions $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{x}$ is said to Pareto dominate $\mathbf{y}$, denoted as $\mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{y})$, if and only if $f_i(\mathbf{x}) \leq f_i(\mathbf{y})$ for all $i = 1, 2, \ldots, m$ and $f_j(\mathbf{x}) < f_j(\mathbf{y})$ for at least one $j = 1, 2, \ldots, m$.

Definition 2.2 (Nondominance). For any two solutions $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{x}$ is said to be nondominated with $\mathbf{y}$ if and only if $\mathbf{x}$ does not dominate $\mathbf{y}$ and $\mathbf{y}$ does not dominate $\mathbf{x}$.

Definition 2.3 (Pareto Optimality). A solution $\mathbf{x}$ is said to be Pareto optimal if and only if there does not exist any solution $\mathbf{y}$ in the decision space Ω dominating $\mathbf{x}$. Obviously, all Pareto optimal solutions are nondominated with each other.

Definition 2.4 (Pareto Optimal Set). All Pareto optimal solutions for an LSMOP constitute the Pareto optimal set PS (i.e., $PS = \{\mathbf{x} \in \Omega | \nexists \mathbf{y} \in \Omega \rightarrow \mathbf{f}(\mathbf{y}) \prec \mathbf{f}(\mathbf{x})\}$).

Definition 2.5 (Pareto Front). The objective vectors of all Pareto optimal solutions for an LSMOP constitute the Pareto front PF (i.e., $PF = \{\mathbf{f}(\mathbf{x}) | \mathbf{x} \in PS\}$).

Since the objectives of a multi-objective optimization problems are usually conflicting with each other to some extent, there does not exist a single solution minimizing all of the objectives. In other words, the Pareto optimal set contains multiple solutions trading off between all of the objectives, where the improvement of one objective cannot be achieved without the deterioration of at least one of the other objectives. Thus, the goal of multi-objective optimization is to find a set of solutions approximating the Pareto front with good convergence and diversity, where a good convergence indicates that the solutions are close to the Pareto front and a good diversity indicates that the solutions have good spread and evenness in the objective space. For this aim, many MOEAs have been proposed in the past two decades [219]. These MOEAs search for well-converged solutions by using conventional reproduction operators, such as the genetic algorithm [67], differential evolution [149], and particle swarm optimization [41]. In addition, they diversify the solutions for better diversity by using different selection strategies, which mainly include density estimation based selection [162], decomposition based selection [178], and indicator based selection [45]. However, these MOEAs are developed for solving small-scale multi-objective optimization problems, which are inefficient for solving LSMOPs with high-dimensional decision spaces [110, 213].

However, the evolutionary algorithms for large-scale single-objective optimization have been developed for many years, which address the curse of dimensionality by using decomposition approaches and nondecomposition approaches [75]. The decomposition approaches divide the decision variables into several groups and optimize each group of decision variables alternately, where several variable grouping strategies such as random grouping [195] and differential grouping [126] are adopted. The nondecomposition approaches search for well-converged solutions by suggesting efficient reproduction operators, which are represented by parameter adaptation based differential evolution [158] and new learning strategy based particle swarm optimization [24]. Nevertheless, these approaches cannot be directly transferred to LSMOPs, since a set of optimal solutions rather than a single one needs to be found for LSMOPs, and the solutions need to spread uniformly along the Pareto front. First, the variable grouping strategies are likely to drive the population toward a single optimal region but not the whole Pareto front, where the population diversity cannot be preserved in the objective space [110]. Second, the reproduction operators for large-scale single-objective optimization can efficiently find a single optimal solution in a high-dimensional decision

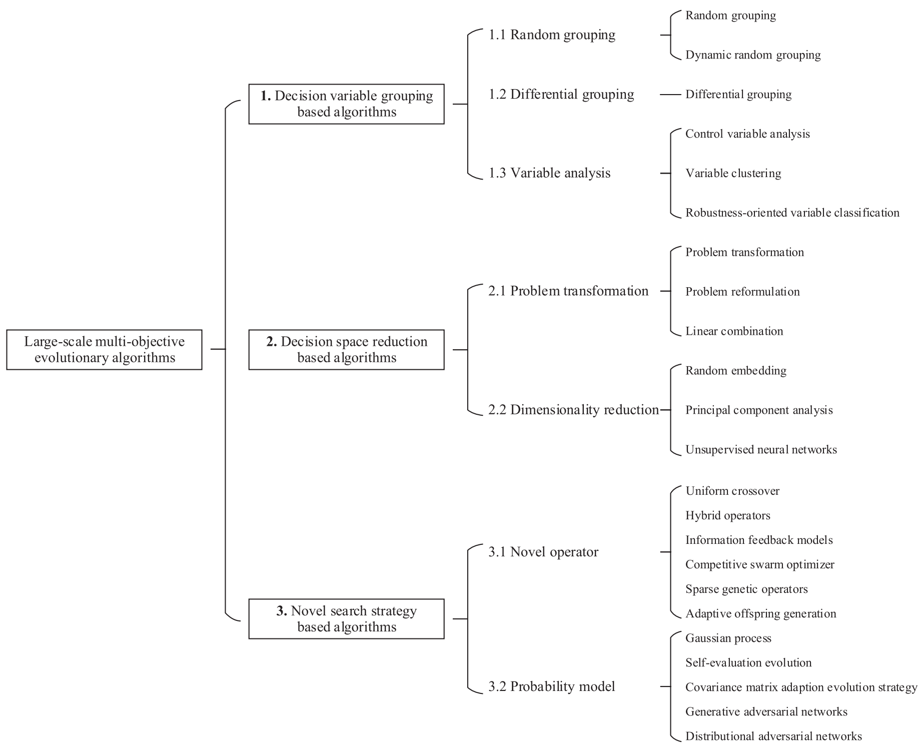

Fig. 2. Taxonomy of existing MOEAs for solving LSMOPs.

space, but the particle swarm optimization operators are inefficient to find multiple optimal solutions for LSMOPs [172], and the differential evolution operators cannot quantify the fitness improvement of solutions with multiple objectives to guide the parameter adaptation [143, 158].

As a consequence, LSMOPs are much more difficult than small-scale multi-objective optimization problems and large-scale single-objective optimization problems, and hence a variety of new techniques have been tailored for solving LSMOPs. As presented in Figure 2, the MOEAs for solving LSMOPs can be divided into three categories according to how the large number of decision variables are handled, including decision variable grouping based MOEAs, decision space reduction based MOEAs, and novel search strategy based MOEAs. The decision variable grouping based MOEAs suggest new variable grouping strategies, in which both the convergence and diversity are considered [213]. The decision space reduction based MOEAs facilitate the solving of LSMOPs by reducing the volume of the high-dimensional decision space. Based on problem transformation or dimensionality reduction techniques, LSMOPs can be converted into small-scale optimization problems and solved by conventional evolutionary algorithms [223]. In contrast to the preceding MOEAs that reduce the difficulties of LSMOPs before the evolution, the last category of MOEAs directly solve LSMOPs by suggesting novel search strategies, which mainly include novel reproduction operators [172] and probability models [19]. The details of these MOEAs are introduced in the next three sections.

3 DECISION VARIABLE GROUPING BASED EVOLUTIONARY ALGORITHMS

A straightforward idea for solving LSMOPs is to use the divide-and-conquer strategy, which divides the decision variables into several groups randomly or heuristically, then optimizes each

Table 1. Decision Variable Grouping Based MOEAs for Solving LSMOPs

Subcategory	Name	Year	Core Technique
Random grouping based MOEAs	CCGDE3 [3]	2013	Random grouping
	MOEA/D^2 [4]	2016	Random grouping
	MOEA/D-RDG [148]	2016	Dynamic random grouping
	OD-NSGA [5]	2019	Random grouping
Differential grouping based MOEAs	TS [144]	2018	Differential grouping
	CCLSM [94]	2018	Differential grouping
Variable analysis based MOEAs	MOEA/DVA [110]	2016	Control variable analysis
	DPCCMOEA [18]	2017	Control variable analysis
	LMEA [213]	2018	Variable clustering
	PEA [20]	2018	Variable clustering
	FR [39]	2018	Robustness-oriented variable classification
	CNSDE/DVC [40]	2018	Robustness-oriented variable classification
	mogDG-shift [17]	2020	Control variable analysis

group of decision variables alternately. This strategy has been widely used in solving large-scale single-objective optimization problems, where evolutionary algorithms randomly divide the decision variables into several groups with fixed size [195] or divide the decision variables according to the interactions between them [126], then alternately optimize each group of decision variables while fixing the rest. However, an LSMOP contains multiple conflicting objectives on which the interactions between decision variables may be different, and both the convergence and diversity of the population should be considered when optimizing each group of decision variables. Hence, the use of a divide-and-conquer strategy on LSMOPs is much more difficult, and some MOEAs with more delicate decision variable grouping techniques have been proposed to address this issue. As listed in Table 1, the decision variable grouping techniques in existing MOEAs mainly contain random grouping, differential grouping, and variable analysis, which are introduced in the following three sections in detail.

3.1 Random Grouping Based MOEAs

The first MOEA for solving LSMOPs suggested a random grouping technique and a cooperative coevolutionary framework, called *CCGDE3* [3]. Based on a generalized differential evolution algorithm GDE3 [85], CCGDE3 randomly divides the decision variables into several groups with equal size, where the random division of variables might increase the probability of optimizing interacting variables simultaneously. Although the first work on evolutionary large-scale multi-objective optimization is relatively naive, it obtained satisfactory performance on some LSMOPs with up to 5,000 decision variables, in comparison to conventional MOEAs whose performance was verified on problems with only 30 decision variables [34, 85]. Later, the random grouping technique was adopted in a decomposition based algorithm MOEA/D [210], gaining better results than conventional MOEAs on LSMOPs with up to 1,200 decision variables [4]. In 2019, the random grouping technique was also adopted in a nondominated sorting genetic algorithm NSGA-III [33] and exhibited good performance [5].

In the preceding random grouping based MOEAs, the group size is a parameter that should be given in advance, which is difficult to be predefined for different LSMOPs. To solve this issue,

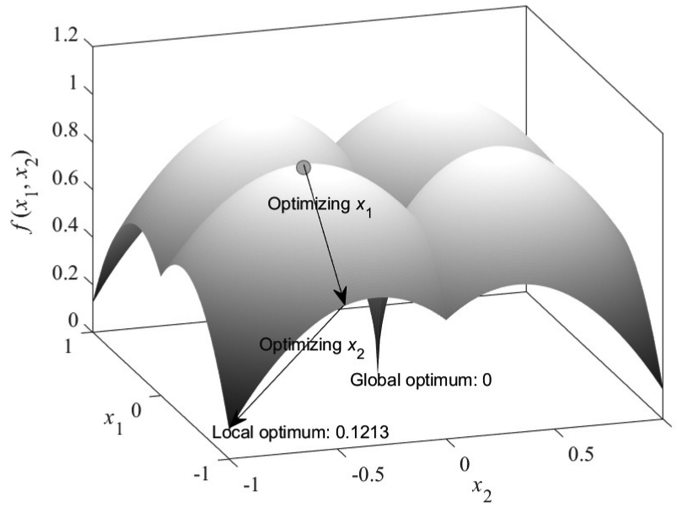

Fig. 3. Landscape of the function $f(x_1, x_2) = \frac{3}{2}\sqrt{|x_1| + |x_2|} - x_1^2 - x_2^2$. The solution is likely to get trapped in local optima when optimizing x_1 and x_2 alternately.

a random-based dynamic grouping technique was suggested in MOEA/D-RDG [148] to automatically adjust the group size during the evolutionary procedure. To be specific, a group size pool is established before the evolution, which contains multiple different group sizes according to the number of decision variables of the LSMOP—for instance, the group size pool is $\{5, 10, 25, 50, 100, 200, 500\}$ when dealing with 1,000 decision variables. In each generation, a group size is first selected from the pool by using the roulette wheel selection method, then the decision variables are divided according to this group size and optimized alternately. The probability of each group size to be selected is dynamically updated according to the performance improvement brought by this group size in previous generations, hence better group sizes are expected to be selected with higher probabilities to make the population converge faster.

3.2 Differential Grouping Based MOEAs

The random grouping technique is very efficient and easy to be implemented, but it does not consider the interactions between decision variables at all, which may drive the population toward a local optimal region of the decision space. Taking the following function with two decision variables as an example:

$$\begin{aligned} \min \quad & f(x_1, x_2) = \frac{3}{2}\sqrt{|x_1| + |x_2|} - x_1^2 - x_2^2 \\ \text{s.t.} \quad & x_1, x_2 \in [-1, 1] \end{aligned}, \tag{2}$$

where the global optimum of this function is 0 when $(x_1, x_2) = (0, 0)$. For a candidate solution of this function, optimizing only a single variable will push the solution toward the boundary of the landscape as illustrated in Figure 3. Thus, the alternate optimization of x_1 and x_2 is likely to obtain four local optima $(x_1, x_2) = (-1, -1), (-1, 1), (1, -1), (1, 1)$, whereas the global optimum can hardly be reached.

To address this problem, the differential grouping technique was proposed to identify the interactions between decision variables, where the variables interacting with each other are assigned to the same group and optimized simultaneously [126]. If two decision variables x_i and x_j of function $f(\mathbf{x})$ interact with each other, there must exist a solution $\mathbf{x}$ and four variable values a_1, a_2, b_1, and

b_2 such that [21]

$$\begin{cases} f(\mathbf{x})|_{x_i=a_1,x_j=b_1} > f(\mathbf{x})|_{x_i=a_2,x_j=b_1} \\ f(\mathbf{x})|_{x_i=a_1,x_j=b_2} < f(\mathbf{x})|_{x_i=a_2,x_j=b_2} \end{cases}. \tag{3}$$

In other words, for a continuously differential function $f(\mathbf{x})$, the sign of $\frac{\partial f(\mathbf{x})}{\partial x_i}$ is dependent on x_j such that the optimal value of x_i varies for different values of x_j. Hence, the global optimal value of x_i is hard to be found without considering x_j. It can be observed that the function given in (2) satisfies the preceding condition when $a_1 = 0$, $a_2 = 1$, $b_1 = 0$, and $b_2 = 1$, and thus the two decision variables cannot be optimized separately.

In the work of Sander et al. [144], the differential grouping technique was integrated into some existing MOEAs for solving LSMOPs. Since the interactions between decision variables may be different on the multiple objectives of an LSMOP, the algorithm considers two strategies to combine the interactions of all of the objectives. The first strategy divides two decision variables into the same group if they interact on at least one objective, and the second strategy divides two decision variables into the same group if they interact on all of the objectives. The differential grouping technique was empirically verified to be effective in solving complex LSMOPs, and it can eliminate the group size that should be predefined in the random grouping technique. However, it should be noted that the performance is improved at the expense of computational cost, since a large number of solutions should be evaluated to check whether there exist four variable values satisfying the condition in (3) for each pair of decision variables. To alleviate the computational burden of differential grouping, the algorithm in the work fo Li and Wei [94] adopted a fast interdependency identification approach to ignore some unnecessary checks between decision variables.

3.3 Variable Analysis Based MOEAs

Both the random grouping and differential grouping were originally proposed for solving large-scale single-objective optimization problems, which focus on dividing the decision variables in the decision space but ignore the population diversity in the objective space. Therefore, the MOEAs based on random grouping or differential grouping can easily find some local or global optimal solutions but may not be able to diversify the population along the whole Pareto front. In the work of Ma et al. [110], an MOEA was proposed to divide the decision variables by analyzing their control property in terms of their relations to the objective functions, called *MOEA/DVA*. For each decision variable, it is first perturbed for several times on a randomly sampled solution. Then, if all of the perturbed solutions are nondominated with each other, the decision variable is regarded as a position variable, whereas if each perturbed solution is dominated by or dominates all of the others, the decision variable is regarded as a distance variable; otherwise, it is regarded as a mixed variable. Obviously, the position variables influence the population diversity but do not change the population convergence, hence they need only to be slightly adjusted for maintaining the population diversity. On the contrary, the distance variables influence the population convergence but do not change the population diversity, which need to be deeply optimized for the best convergence. Therefore, MOEA/DVA further uses differential grouping to divide all distance variables and optimizes them alternately. After the distance variables are well optimized, MOEA/DVA optimizes all position variables, distance variables, and mixed variables together for fine-tuning the convergence and diversity of the population. Later, the variable analysis technique in MOEA/DVA was modified and parallelized in the work of Cao et al. [18], and the differential grouping performed on the distance variables was enhanced by a graph based grouping with shift in another work by Cao et al. [17].

It is worth noting that although MOEA/DVA was verified to be significantly better than many other MOEAs on benchmark LSMOPs, the objective functions in many real-world LSMOPs are so complex that few decision variables can meet the strict conditions of position variable and distance variable. In other words, most decision variables in real-world LSMOPs are regarded as mixed variables by MOEA/DVA, which cannot be divided and optimized alternately. To address this issue, LMEA suggested a decision variable clustering method to divide the decision variables more generically [213]. Based on a number of perturbed solutions for each decision variable, LMEA fits a line for the solutions and calculates the angle between the fitted line and the convergence direction. Then, all decision variables are divided into two groups by k-means according to their angles. Afterward, the decision variables with smaller angles are regarded as convergence-related variables since the perturbations on them make the solutions move mainly along the convergence direction, and the decision variables with larger angles are regarded as diversity-related variables since the perturbations on them diversify the solutions on the Pareto front. By alternately optimizing the convergence-related variables and diversity-related variables with different search strategies, LMEA can obtain competitive performance to MOEA/DVA on general LSMOPs and better performance than MOEA/DVA on complex LSMOPs. Later, LMEA was parallelized in the work of Chen et al. [20].

The idea of dividing the decision variables into multiple groups has also been adopted to solve other types of problems. In the work of Du et al. [39, 40], the decision variables are classified into highly robustness-related variables and weakly robustness-related variables for solving robust LSMOPs. In the work of Huang and Wang [70], the decision variables are classified into three types according to their relations to the upper-level and lower-level functions for solving bi-level optimization problems.

3.4 Discussion

Random grouping is the first and also one of the simplest techniques for solving large-scale optimization problems, which can directly convert an LSMOP into a number of small-scale optimization problems to be solved in sequence. Although the global optima may be missed if the interacting solutions are divided into different groups, different grouping can generally ensure that the interacting solutions are divided into the same group. In addition, the variable analysis techniques can further maintain the population diversity. On the contrary, random grouping does not need any additional function evaluations to divide the decision variables, whereas different grouping needs $O(d^2)$ function evaluations to detect whether each pair of decision variables interacting with each other and variable analysis techniques need $O(d)$ function evaluations to divide the decision variables into convergence-related and diversity-related ones. As a consequence, if an LSMOP has few interactions between decision variables, it can be handled by the random grouping technique with a small number of function evaluations. However, if an LSMOP has complex variable interactions and computationally efficient objective functions, it can be handled by both the differential grouping and a variable analysis technique with a large number of function evaluations.

4 DECISION SPACE REDUCTION BASED EVOLUTIONARY ALGORITHMS

The idea of dimensionality reduction has been widely adopted for handling big data in machine learning tasks [139], hence it is desirable to reduce the dimensionality of LSMOPs with the same idea. Briefly, the decision vectors of parents are shortened and used to generate offsprings, then the shortened vectors of offsprings are recovered to the original decision space for function evaluations. Thus, the algorithm only needs to find the optimal values of a short vector instead of searching in the high-dimensional decision space. As listed in Table 2, various decision space reduction techniques have been suggested for solving LSMOPs, including problem transformation, problem reformulation, random embedding, principal component analysis, and unsupervised neural networks. It should be noted that although a large number of dimensionality reduction techniques

Table 2. Decision Space Reduction Based MOEAs for Solving LSMOPs

Subcategory	Name	Year	Core Technique
Problem transformation based MOEAs	WOF [222, 223]	2016	Problem transformation
	LSMOF [64]	2019	Problem reformulation
	xNSGA-II [225]	2019	Linear combination
	WOF-MMOPSO-RDG [105]	2020	Problem transformation
	iLSMOA [60]	2020	Problem reformulation
	LMOEA-DS [137]	2021	Problem reformulation
Dimensionality reduction based MOEAs	ReMO [136]	2017	Random embedding
	PCA-MOEA [106]	2020	Principal component analysis
	MOEA/PSL [165]	2020	Unsupervised neural networks
	PM-MOEA [164]	2020	Pattern mining

have been developed in machine learning, most of them cannot be applied to solving LSMOPs since the vectors shortened by them are not recoverable.

4.1 Problem Transformation Based MOEAs

The main idea of problem transformation is to convert the LSMOP into a small-scale optimization problem, where the small-scale optimization problem is to find the optimal variant of a given solution. In the work of Zille et al. [222, 223], a weighted optimization framework (WOF) was proposed to solve LSMOPs via optimizing the weight vector of each solution in the population. Specifically, for a given solution $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ and its weight vector $\mathbf{w} = (w_1, w_2, \ldots, w_k)$, WOF uses conventional MOEAs to optimize the weight vector for the best objective values of the weighted solution $\mathbf{x}^*$ on the original LSMOP, where

$$\mathbf{x}^* = \Big(\underbrace{w_1 x'_1, \ldots, w_1 x'_{\frac{d}{k}}}, \underbrace{w_2 x'_{\frac{d}{k}+1}, \ldots, w_2 x'_{\frac{2d}{k}}}, \ldots, \underbrace{w_k x'_{d-\frac{d}{k}+1}, \ldots, w_k x'_d} \Big) \tag{4}$$

and $x'_1, \ldots, x'_d$ is a permutation of $x_1, x_2, \ldots, x_d$ determined by one of four grouping techniques, including random grouping, linear grouping, ordered grouping, and differential grouping. In other words, WOF divides the d decision variables into k groups and optimizes the weight w_i of each group $x'_{\frac{(i-1)d}{k}+1}, \ldots, x'_{\frac{id}{k}}$, hence the decision space is reduced from d dimensions to k dimensions by optimizing $\mathbf{w}$ instead of $\mathbf{x}$. At the beginning of each generation of WOF, several solutions having the best qualities are picked up from the population, and an optimal weight vector is obtained for each solution. Then, each weight vector is applied to all of the solutions in the population, and the population is combined with all of the weighted solutions and truncated by the environmental selection. In the experiments, three MOEAs (i.e., the differential evolution based GDE3 [85], the genetic algorithm based NSGA-II [34], and the particle swarm optimization based SMPSO [122]) were adopted to optimize the weight vectors, which converge much faster than conventional MOEAs on many challenging LSMOPs. In the work of Liu et al. [105], WOF was integrated with dynamic random grouping and a multi-objective particle swarm optimization algorithm with multiple search strategies [101].

In the LSMOF framework [64], a problem reformulation approach was suggested to accelerate the computational efficiency of MOEAs on LSMOPs, where two weight variables are specified to move a solution along two search directions for finding better solutions. In contrast to WOF optimizing the weight vector of a single solution on the original LSMOP, LSMOF optimizes the weight variables of k solutions simultaneously for maximizing their hypervolume [183]:

$$\max\ f\left(w_1^l, w_1^u, w_2^l, w_2^u, \dots, w_k^l, w_k^u\right) = H\left(\mathbf{x}_1^l, \mathbf{x}_1^u, \mathbf{x}_2^l, \mathbf{x}_2^u, \dots, \mathbf{x}_k^l, \mathbf{x}_k^u\right), \tag{5}$$

where $H(\dots)$ denotes the hypervolume value of a set of solutions in the objective space, and $\mathbf{x}_i^l$ and $\mathbf{x}_i^u$ are two variants of solution $\mathbf{x}_i$ in terms of weight variables w_i^l and w_i^u, respectively:

$$\begin{aligned} \mathbf{x}_i^l &= \mathbf{o} + w_i^l \frac{\mathbf{x}_i - \mathbf{o}}{\|\mathbf{x}_i - \mathbf{o}\|}\|\mathbf{t} - \mathbf{o}\| \\ \mathbf{x}_i^u &= \mathbf{t} - w_i^u \frac{\mathbf{t} - \mathbf{x}_i}{\|\mathbf{t} - \mathbf{x}_i\|}\|\mathbf{t} - \mathbf{o}\| \end{aligned}, \tag{6}$$

where $\mathbf{o}$ is the lower boundary point of the decision space and $\mathbf{t}$ is the upper boundary point of the decision space. Obviously, LSMOF can search for the optimal variants of each solution along the directions $\mathbf{x}_i - \mathbf{o}$ and $\mathbf{t} - \mathbf{x}_i$, and the decision space is reduced from d dimensions to $2k$ dimensions. In addition, it does not need to divide the decision variables. At the beginning of each generation of LSMOF, k solutions having the best qualities are picked up from the population, and the differential evolution algorithm is used to optimize the weight variables of each solution. Then, the population is combined with all of the variants and truncated by the environmental selection of an MOEA. The experimental results demonstrated that LSMOF can considerably accelerate the convergence speed of four popular MOEAs. More recently, the LSMOF framework was alternated with a decomposition based MOEA for solving LSMOPs in the work of He et al. [60], where the population diversity can be significantly improved. In addition, LMOEA-DS [137] shares a similar idea to LSMOF, whereas the solutions on the search directions are directly sampled but not generated by genetic operators.

In the work of Zille and Mostaghim [225], a linear search mechanism was proposed for finding the optimal linear combinations of the solutions in the current population. Specifically, it optimizes the weight vector $\mathbf{w} = (w_1, w_2, \dots, w_k)$ for the best combination $\mathbf{x}^*$ of multiple solutions $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k$, where

$$\mathbf{x}^* = w_1\mathbf{x}_1 + w_2\mathbf{x}_2 + \cdots + w_k\mathbf{x}_k. \tag{7}$$

Hence, the decision space can be reduced from d dimensions to k dimensions. In practice, an existing MOEA is adopted to alternately optimize the population and the weight vector until the termination criteria is met. It was empirically verified that this linear search mechanism can improve the performance of existing MOEAs in solving LSMOPs.

4.2 Dimensionality Reduction Based MOEAs

In contrast to the problem transformation based MOEAs that reduce the decision space by searching for the optimal variants of existing solutions, some dimensionality reduction techniques in unsupervised learning have been applied to decrease the number of decision variables by capturing the correlations between them. In ReMO [136], the random embedding technique was adopted to solve the LSMOPs with low intrinsic dimensions, whose objective functions are only affected by a small proportion of all decision variables. By multiplying each solution $\mathbf{x} \in \mathbb{R}^{1\times d}$ by a random matrix $A \in \mathbb{R}^{d\times k}$ with $k \ll d$ and $A_{ij} \sim \mathcal{N}(0,1)$, the number of decision variables can be directly decreased from d to k. For the LSMOPs with low intrinsic dimensions, it was theoretically proved in the same work that the global optima of the original objective functions still exist in the k-dimensional space, hence the convergence speed can be highly improved without the deterioration of performance. Similarly, principal component analysis has also been employed to solve general multi-objective optimization problems [154, 211] and LSMOPs [106], where the matrix for decreasing the number of decision variables consists of the eigenvectors of the covariance matrix of all decision variables. In PCA-MOEA [106], the decision variables are first divided into convergence-related variables and diversity-related variables by the decision variable clustering method in LMEA, then the convergence-related variables are decreased by principal component analysis and further divided by differential grouping.

Although random embedding and principal component analysis shorten the decision vector linearly, unsupervised neural networks consider the nonlinear correlations between decision variables in dimensionality reduction. To solve the LSMOPs with sparse optimal solutions, MOEA/PSL [165] represents each solution $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ with a binary vector $\mathbf{xb} = (xb_1, xb_2, \ldots, xb_d)$ and a real vector $\mathbf{xr} = (xr_1, xr_2, \ldots, xr_d)$, where $x_i = xb_i \times xr_i$. At each generation, MOEA/PSL trains a restricted Boltzmann machine [49] according to the binary vectors in the population and a denoising autoencoder [175] according to the real vectors in the population in an unsupervised manner, then shortens the binary and real vectors of solutions via the trained restricted Boltzmann machine and denoising autoencoder. Experimental results demonstrated that MOEA/PSL is effective for the sparse LSMOPs in machine learning, network science, signal processing, data mining, and economics. In contrast to the neural network based dimensionality reduction in MOEA/PSL, an evolutionary pattern mining approach was suggested in PM-MOEA [164] for dimensionality reduction, which was shown to be more efficient, less greedy, and parameterless.

4.3 Discussion

The problem transformation based MOEAs reduce the decision space by optimizing the weights for the optimal variants of existing solutions, where the weight vector has much fewer variables than the decision vector. These MOEAs can quickly find some local optimal solutions for LSMOPs, but they might not be able to find the global optimal solutions even if much more function evaluations are consumed. This is because the transformed problem is likely to lose the global optimal solutions of the original LSMOP—the global optimal solutions probably correspond to a different weight for each decision variable of a solution, whereas the weights for the decision variables in the same group are always the same in WOF and the weights for all decision variables of a solution are always the same in the linear search mechanism; for the two search directions for each solution in LSMOF, they could hardly intersect the $(m-1)$-dimensional Pareto optimal set in the d-dimensional decision space since $m \ll d$. However, the dimensionality reduction based MOEAs may retain the global optimal solutions in the reduced decision space, but it is difficult to estimate the Pareto optimal set for eliminating the redundant decision variables. Hence, the dimensionality reduction based MOEAs are usually employed to solve LSMOPs whose Pareto optimal sets are easy to be estimated. As a consequence, if an LSMOP has computationally expensive objective functions, the problem transformation based MOEAs can be used to find local optimal solutions with a tiny number of function evaluations. However, if an LSMOP has low intrinsic dimensions or sparse optimal solutions, ReMO or MOEA/PSL can be used to approximate the global optimal solutions.

5 NOVEL SEARCH STRATEGY BASED EVOLUTIONARY ALGORITHMS

The decision variable grouping based MOEAs and decision space reduction based MOEAs facilitate the solving of LSMOPs by dividing and reducing the decision space, respectively, where the search strategies of conventional MOEAs are adopted to evolve the population in the reduced decision space. On the contrary, some recently proposed MOEAs directly solve LSMOPs by suggesting novel search strategies for generating offsprings in the original decision space, including new reproduction operators and probability models as listed in Table 3. These MOEAs do not need to handle the decision space with complex operations, holding simple procedures and balanced performance on different LSMOPs.

Table 3. Novel Search Strategy Based MOEAs for Solving LSMOPs

Subcategory	Name	Year	Core Technique
Novel reproduction operator based MOEAs	NSGA-III-UC [199]	2020	Uniform crossover
	MOSM [1]	2020	Hybrid reproduction operators
	MOEA/D-IFM [215]	2020	Information feedback models
	LMOCSO [172]	2020	Competitive swarm optimizer
	SparseEA [171]	2020	Sparse genetic operators
	DGEA [61]	2020	Adaptive offspring generation
Probability model based MOEAs	IM-MOEA [25]	2015	Gaussian process based inverse model
	DLS-MOEA [68]	2018	Self-evaluation evolution
	S^3-CMA-ES [19]	2021	Covariance matrix adaptation evolution strategy
	GMOEA [63]	2020	Generative adversarial networks
	MOEA-CSOD [99]	2020	Distributional adversarial networks

5.1 Novel Reproduction Operator Based MOEAs

Although most decision variable grouping and decision space reduction based MOEAs use the general reproduction operators in the genetic algorithm (i.e., simulated binary crossover [31] and polynomial mutation [32]), differential evolution [149], and particle swarm optimization [41] to generate offsprings in the reduced decision space, some work adopts other existing reproduction operators or proposes new reproduction operators to generate offsprings in the original decision space. In the work of Zille et al. [221], three mutation operators (i.e., polynomial mutation, linked polynomial mutation, and grouped polynomial mutation) making use of differential grouping were compared, where the linked and grouped polynomial mutation can significantly improve the performance of existing MOEAs on LSMOPs. In the work of Yi et al. [199], three crossover operators (i.e., simulated binary crossover, uniform crossover, and a modified uniform crossover) and their variants (i.e., one of the parents is always the best solution in the population) were embedded in NSGA-III [33], where the experimental results indicated that the uniform crossover can lead to better performance on LSMOPs than the other two operators. In the work of Abdi and Feizi-Derakhshi [1], many reproduction operators with dynamically adjusted weights were hybridized in a single MOEA for solving LSMOPs, including those in the genetic algorithm, particle swarm optimization, bat algorithm [74], sine cosine algorithm [117], imperialist competitive algorithm [69], whale optimization algorithm [118], trader optimization algorithm [114], heat transfer search [130], and moth-flame optimization algorithm [116].

In addition to the preceding MOEAs utilizing existing reproduction operators, several new reproduction operators have also been tailored for solving LSMOPs. In MOEA/D-IFM [215], six information feedback models were designed and integrated in MOEA/D, which take advantage of the historical information by updating the offsprings generated by general genetic operators according to the solutions in previous generations. In LMOCSO [172], a new reproduction operator based on the competitive swarm optimizer [24] was suggested. The competitive swarm optimizer is an efficient algorithm for solving large-scale single-objective optimization problems, which divides the solutions into losers and winners by using a pairwise competition mechanism and lets the losers learn from the winners. To accelerate the convergence speed of competitive swarm optimizer on LSMOPs, a novel strategy considering both the velocity and acceleration was proposed for updating the losers according to the winners, which was verified to be more efficient in driving the losers to move toward better positions. In SparseEA [171], a new crossover operator and a new mutation operator were developed for solving sparse LSMOPs. SparseEA also represents each solution with a binary vector and a real vector, and it performs simulated binary crossover and polynomial mutation on the real vectors and new operators on the binary vectors. Although general binary operators are likely to make the decision variables of the offsprings having the same number of 0 and 1, the new binary operators in SparseEA can control the sparsity of the offsprings (i.e., most decision variables are 0), hence the sparse optimal solutions of the LSMOPs can be found more easily. In DGEA [61], an adaptive offspring generation method was proposed for solving LSMOPs, which generates solutions along the directions between nondominated solutions and dominated solutions in a diverse population. Experimental results demonstrated the superiority of the adaptive offspring generation method over existing reproduction operators.

5.2 Probability Model Based MOEAs

According to the Karush-Kuhn-Tucker condition [42], the Pareto optimal set of an m-objective optimization problem is an $(m-1)$-dimensional piecewise continuous manifold. To explicitly exploit this regularity property, some MOEAs generate offsprings by using probability models instead of reproduction operators, where the probability models are used for characterizing promising solutions and approximating the Pareto optimal set [89, 211]. In addition, a variety of probability models have been suggested for solving LSMOPs in some recent studies. IM-MOEA [25] constructs Gaussian process based inverse models to map the solutions from the objective space to the decision space, and uses the inverse models to generate offsprings by sampling the objective space. DLS-MOEA [68] inherits the fast convergence speed of a hypervolume based steady-state algorithm SMS-EMOA [9], and enhances the diversification ability by using a self-evaluation evolution [192] based dual local search mechanism, where the self-evaluation evolution is an efficient algorithm for high-dimensional optimization that generates offsprings by Gaussian or Cauchy mutation and evaluates offsprings by a meta-model. S^3-CMA-ES [19] first uses the variable analysis technique in MOEA/DVA to divide the decision variables, then evolves a subpopulation for each group of distance variables by using the covariance matrix adaptation evolution strategy [59], which is a classical and efficient evolutionary algorithm that generates offsprings by an iteratively updated Gaussian model. In addition, the generative adversarial networks [55] and distributional adversarial networks [88] trained on the solutions in the current population have also been adopted to generate offsprings in GMOEA [63] and MOEA-CSOD [99], respectively.

5.3 Discussion

The competitive performance of the preceding MOEAs indicates that novel reproduction operators and probability models can effectively solve LSMOPs without the help of decision variable grouping or decision space reduction. For the novel reproduction operators inspired by the genetic algorithm, particle swarm optimization, and other evolutionary paradigms, they can strike a balance between exploitation and exploration in the high-dimensional decision space and accelerate the convergence speed of the population. As for the probability models including Gaussian process, covariance matrix, generative adversarial network, and so on, they can approximate the Pareto optimal set by learning the distributions of promising solutions and generate better offsprings. Since most reproduction operators and probability models are independent of the selection strategies, they can be easily embedded in many MOEAs to improve their performance on LSMOPs. More importantly, they are quite efficient since no operation needs to be performed to handle the decision space. As a consequence, if an LSMOP is so complex that the decision space is difficult to be characterized, the novel search strategy based MOEAs can be adopted for a balanced performance.

6 BENCHMARK PROBLEMS FOR EVOLUTIONARY LARGE-SCALE MULTI-OBJECTIVE OPTIMIZATION

Over the past two decades, a number of multi-objective test suites have been designed to efficiently assess the performance of MOEAs [202]. Most of these test suites contain multiple benchmark problems with various difficulties and a scalable number of decision variables, hence they can be used as benchmark LSMOPs by setting the number of decision variables to large values. Although these test suites do not reflect many complex characteristics of LSMOPs, several specialized large-scale multi-objective test suites have also been designed. In this section, some popular test suites in evolutionary large-scale multi-objective optimization are presented.

6.1 Artificial Benchmark Problems

Table 4 summarizes the features of the benchmark LSMOPs in five widely used artificial test suites, including ZDT [226], DTLZ [35], WFG [71], LSMOP [26], and SMOP [171]. First, the Pareto fronts in these benchmark LSMOPs have various geometrical shapes, where MOEAs can easily spread the population along the Pareto fronts with convex, linear, and concave shapes, but it is difficult to find the whole Pareto fronts with disconnected and degenerate shapes. In other words, the MOEAs without delicate diversity preservation strategies may be able to find well-converged solutions for LSMOPs, but the population will have a relatively bad diversity on complex Pareto fronts. Second, the landscapes in benchmark LSMOPs can be unimodal, multimodal, deceptive, or biased, where MOEAs can easily converge to the global optimum of unimodal landscapes but may be trapped into the local optima of multimodal or deceptive landscapes; in addition, the biased landscapes challenge MOEAs in converging to the whole Pareto front. Third, the separable objective functions indicate no interaction between decision variables, which means that the global optimum can be obtained by optimizing the decision variables one by one; by contrast, the nonseparable objective functions indicate that each decision variable interacts with all of the others, which means that all decision variables should be optimized simultaneously. In short, the features of Pareto front are related to the diversity performance of MOEAs, the features of landscape are related to the convergence performance of MOEAs, and the separability of objective functions determines whether the decision variables are separable in terms of differential grouping.

ZDT is one of the first test suites for multi-objective optimization, which contains six bi-objective optimization problems with different Pareto fronts and landscapes. These benchmark problems are scalable with respect to the number of decision variables, but they are easy to be solved even if the number of decision variables is large, since there is no interaction between decision variables in these problems. For the seven scalable benchmark problems in the DTLZ test suite, although their decision variables also do not interact with any others, some of them are relatively difficult to solve due to the highly multimodal landscapes (i.e., DTLZ1 and DTLZ3) and extremely irregular Pareto fronts (i.e., DTLZ5–DTLZ7). As for the nine scalable benchmark problems in the WFG test suite, they have simpler landscapes and Pareto fronts than the DTLZ problems; by contrast, some interactions between decision variables exist in the WFG problems and pose challenges for decision variable grouping based MOEAs to divide the decision variables properly.

It is worth noting that the ZDT, DTLZ, and WFG test suites were designed for assessing the performance of MOEAs on small-scale multi-objective optimization problems, whereas the LSMOP test suite is tailored for large-scale multi-objective optimization. The nine scalable benchmark problems in the LSMOP test suite characterize various difficulties in real-world LSMOPs, including nonuniform grouping of decision variables, different correlations between variables and objective functions, mixed separability, and difficult landscape functions taken from large-scale single-objective optimization [95]. In addition, they introduce linkages between the position variables and

Table 4. Features of Widely Used Artificial Benchmark LSMOPs

Test Suite (year)	Benchmark Problem	Features of Pareto Front	Features of Landscape	Separability of Objective Functions
ZDT (2000) [226]	ZDT1	Convex	Unimodal	Separable
	ZDT2	Concave	Unimodal	Separable
	ZDT3	Convex, disconnected	Multimodal	Separable
	ZDT4	Convex	Multimodal	Separable
	ZDT5	Convex	Multimodal	Separable
	ZDT6	Concave	Multimodal	Separable
DTLZ (2005) [35]	DTLZ1	Linear	Multimodal	Separable
	DTLZ2	Concave	Unimodal	Separable
	DTLZ3	Concave	Multimodal	Separable
	DTLZ4	Concave	Unimodal, biased	Separable
	DTLZ5	Concave, degenerate	Unimodal	Separable
	DTLZ6	Concave, degenerate	Unimodal, biased	Separable
	DTLZ7	Disconnected	Multimodal	Separable
WFG (2006) [71]	WFG1	Convex	Unimodal, biased	Separable
	WFG2	Convex, disconnected	Multimodal	Nonseparable
	WFG3	Linear, degenerate	Unimodal	Nonseparable
	WFG4	Concave	Multimodal	Separable
	WFG5	Concave	Multimodal, deceptive	Separable
	WFG6	Concave	Unimodal	Nonseparable
	WFG7	Concave	Unimodal, biased	Separable
	WFG8	Concave	Unimodal, biased	Nonseparable
	WFG9	Concave	Multimodal, deceptive, biased	Nonseparable
LSMOP (2017) [26]	LSMOP1	Linear	Unimodal	Separable
	LSMOP2	Linear	Mixed	Partially separable
	LSMOP3	Linear	Multimodal	Mixed
	LSMOP4	Linear	Mixed	Mixed
	LSMOP5	Concave	Unimodal	Separable
	LSMOP6	Concave	Mixed	Partially separable
	LSMOP7	Concave	Multimodal	Mixed
	LSMOP8	Concave	Mixed	Mixed
	LSMOP9	Disconnected	Mixed	Separable
SMOP (2020) [171]	SMOP1	Linear	Mixed	Separable
	SMOP2	Linear	Multimodal, deceptive	Separable
	SMOP3	Linear	Unimodal, deceptive	Separable
	SMOP4	Convex	Deceptive	Separable
	SMOP5	Convex	Unimodal	Separable
	SMOP6	Convex	Multimodal	Separable
	SMOP7	Concave	Multimodal	Mixed
	SMOP8	Concave	Deceptive	Nonseparable

distance variables, which makes them much more difficult to solve than the benchmark problems in the ZDT, DTLZ, and WFG test suites. More specifically, once a single solution on the Pareto front is obtained, the solution can be easily spread along the whole Pareto fronts of the ZDT, DTLZ, and WFG test suites by perturbing the position variables, but it cannot be spread along the whole Pareto fronts of the LSMOP test suite since the position variables interact the distance variables and hence influence the convergence quality of solutions. In short, the LSMOP test suite is very challenging since multiple solutions should converge to different regions of the Pareto front simultaneously.

SMOP is a test suite for assessing the performance of MOEAs on LSMOPs with sparse optimal solutions (i.e., most decision variables of the global optimal solutions are zero). Sparse LSMOPs widely exist in real-world applications, such as the feature selection aiming to select a small subset of features from many candidate features [91], the sparse reconstruction aiming to find the most accurate and sparest signal [90], and the critical node detection aiming to select the fewest nodes for the largest destruction to the graph [86], but few benchmark problems consider the sparsity nature of LSMOPs. Hence, nine benchmark problems with sparse optimal solutions were designed in the SMOP test suite, which are characterized by low intrinsic dimensionality, variable interactions, deception, and multimodality. These difficulties make the SMOP test suite difficult to solve by general MOEAs; by contrast, the global optimal solutions of the SMOP test suite can be found with a small number of function evaluations if the sparse nature of solutions is considered in the generation of offsprings.

In addition to the preceding five test suites, there are some other benchmark problems that can be extended to LSMOPs in the literature [25, 89, 128, 211]. Most of these benchmark problems are with complex landscapes, weak interactions between decision variables, and strong linkages between the position variables and distance variables.

6.2 Real-World Benchmark Problems

In contrast to the artificial benchmark test suites, some other benchmark LSMOPs are directly taken from real-world applications. For example, six benchmark LSMOPs were proposed based on the time-varying ratio error estimation task in the work of He et al. [62]. The estimation of time-varying ratio error is an important task for obtaining the true primary voltage such that the failure rate of voltage transformers can be detected and corrected [82]. Based on the physical and statistical rules in the power delivery system, the time-varying ratio error estimation problem was formulated as an LSMOP, and six benchmark LSMOPs were established based on different datasets from reality. These LSMOPs are very challenging since they have up to 300,000 decision variables, three objectives, and six constraints. In addition, it was investigated that these LSMOPs involve different types of complex interactions between decision variables. Hence, these benchmark LSMOPs can measure the performance of MOEAs in handling a large number of interacting decision variables with constraints.

All of the preceding benchmark LSMOPs are continuous optimization problems; however, five types of combinatorial benchmark LSMOPs and three types of continuous benchmark LSMOPs were established in the work of Tian et al. [165], covering the areas of feature selection, instance selection, neural network training, community detection, critical node detection, signal reconstruction, pattern mining, and portfolio optimization. For each of the eight types of problems, three benchmark LSMOPs were established based on different datasets, hence there were 24 benchmark LSMOPs in total. These benchmark LSMOPs have sparse optimal solutions with 1,000 to 10,000 decision variables, and they have complex landscapes and computationally expensive objective functions. Similar to the SMOP test suite, these benchmark LSMOPs can measure the efficiency of MOEAs in finding sparse optimal solutions in both continuous and discrete decision spaces.

6.3 Discussion

The benchmark LSMOPs have a variety of characteristics with adjustable difficulties, which can be used to quickly assess the performance of MOEAs on LSMOPs due to the computationally efficient objective functions. However, most of them can be easily handled since their difficulties are clear and regular. For example, the ZDT, DTLZ, and WFG test suites contain few interactions between decision variables, where random grouping based MOEAs can efficiently solve them since the alternate optimization of each single variable leads to the global optimum. The LSMOP test suite contains strong linkages between position variables and distance variables, which can be detected and tackled by variable analysis based MOEAs. Since the optimal solutions of the SMOP test suite are very sparse, they can be easily approximated by the MOEAs considering the sparse nature of solutions, such as SparseEA [171], MOEA/PSL [165], and PM-MOEA [164].

On the contrary, the real-world benchmark LSMOPs are with complex Pareto fronts and chaotic interactions between decision variables, and each decision variable cannot be regarded as a pure position variable or distance variable, hence it is difficult to conclude that which category of MOEAs performs the best on a specific real-world LSMOP. The next section introduces the applications of MOEAs on more scientific and engineering areas, where each application has been well developed and has its own benchmark datasets. Therefore, these applications can also be adopted as benchmarks for the performance assessment of MOEAs. These applications have distinct objective functions, landscapes, decision spaces, and interactions between decision variables. In addition to the real encoding and binary encoding, they also include some novel encodings such as the categorial encoding in data clustering and the permutation based encoding in vehicle routing.

7 APPLICATIONS OF EVOLUTIONARY LARGE-SCALE MULTI-OBJECTIVE OPTIMIZATION

Evolutionary algorithms have shown competitiveness on many real-world LSMOPs, especially on those with complex landscapes or discrete decision spaces. Although most MOEAs are high-level methodologies that can find multiple diverse solutions for different black-box optimization problems by using the same search strategy, customized local search strategies and reproduction operators are also required for solving specific types of LSMOPs. In this section, the applications of evolutionary large-scale multi-objective optimization in some popular and emerging areas are introduced.

7.1 Applications in Machine Learning

Most machine learning tasks are essentially optimization problems—for example, supervised learning aims to find the features, instances, and parameters of models for the highest accuracy, and data clustering aims to divide a number of instances into multiple groups for the highest cluster compactness [77]. Since these tasks usually handle large datasets, they are large-scale optimization problems that cannot be tackled easily. Therefore, they have been solved by many MOEAs with the assistance of the characteristics of the datasets.

Feature selection is a basic machine learning task that aims at selecting the relevant features and eliminating the irrelevant or redundant features from a dataset, to maximize the performance of the model [91]. Feature selection is a typical subset selection problem, and it is very difficult due to the complex interactions between features. Various MOEAs have been proposed for the feature selection problem, which can maximize the classification accuracy of the model and minimize the number of selected features simultaneously [124, 169, 190]. Although most work focused on the feature selection task for classification, the feature selection task for clustering has also been tackled by MOEAs [57].

Instance selection is also a subset selection problem, which aims to select the fewest training instances for the highest performance of the model [174]. Generally, instance selection is more difficult than feature selection since the instances are much more than the features in a dataset. MOEAs have been applied to the feature selection problem with thousands of features; however, they have successfully solved the instance selection problem with approximately 30,000 instances [22, 131, 141].

Ensemble learning aims to construct a powerful model by combing multiple naive models [30], which is a challenging subset selection problem since each variable is a real value rather than a binary value. In the past decade, some MOEAs have been developed to construct ensemble models by optimizing the weights of ensemble members, which can maximize multiple performance indicators (e.g., precision, recall, and F-measure) of the ensemble model and minimize the number of selected ensemble members simultaneously [10, 127, 135].

Neural network training has been handled by evolutionary algorithms as early as in the 1990s [197], but general evolutionary algorithms are ineffective for optimizing the millions of weights in a **deep neural network (DNN)**. Although gradient based approaches (e.g. stochastic gradient descent [140] and Adam [84]) have shown to be promising in training DNNs, these approaches are easily trapped into local optima and sensitive to the parameter settings [77, 198]. To address these issues, some MOEAs have been proposed for training DNNs, which can maximize the learning performance and minimize the network complexity simultaneously [54, 185, 194]. To tackle the curse of dimensionality, some work further reduced the decision space by dimensionality reduction techniques [104, 155].

Neural architecture search is currently one of the hottest topics in deep learning, which is mainly handled by the evolutionary algorithm [153], reinforcement learning [8], and gradient descent [102]. The architecture of a DNN is usually represented by a graph [8] or a hypernetwork [102], which forms a large-scale optimization problem since the graph or hypernetwork includes many candidate edges to be selected. More recently, some MOEAs have been adopted for the neural architecture search of convolutional neural networks and recurrent neural networks [108, 176, 196].

Adversarial attack is an emerging topic in deep learning, which aims at perturbing an instance (e.g. changing the pixels of an image) so that it is misjudged by well-trained DNNs. It is a large-scale optimization problem since a few pixels are expected to be perturbed among all pixels of an image, hence the perturbation is imperceptible to human eyes [151]. Some MOEAs have been adopted to generate adversarial images in recent years, which can minimize both the prediction probability of the correct class and the distortion of the perturbation [37, 156, 203]. In addition, some MOEAs were adopted for the adversarial attack on speech signals [83].

Data clustering is a core task of unsupervised learning, which is a difficult large-scale optimization problem due to the highly redundant and discrete decision space formed by the labels of instances [73]. A number of MOEAs have been used for data clustering, which can automatically determine the number of clusters and optimize both the locations of cluster centers and the labels of instances [51, 58, 129]. In addition to the MOEAs for crisp clustering, fuzzy clustering has also been tackled by some MOEAs via optimizing the locations of cluster centers [142, 184] or membership matrix between instances and cluster centers [44].

7.2 Applications in Network Science

Many real-world complex systems can be represented by networks, in which the nodes denote the objects (e.g. individuals in social networks [52] and genes in biological networks [103]) and the edges denote the connections between objects. The goals of many tasks in network science are to mine useful information from networks, which are challenging optimization problems due to the high-dimensional and highly discrete decision space. Owing to the superiorities in solving large-scale combinatorial optimization problems, many MOEAs have been applied to various tasks in network science.

Community detection is one of the most important issues in network science, which aims to divide the nodes in a network into several communities such that the nodes in the same community have dense connections and the nodes in different communities have sparse connections [15]. In comparison to the data clustering tasks in Euclidean space, community detection is much more challenging due to the difficulty in representing the cluster centers and measuring the distances between nodes. To solve this problem, some MOEAs have been tailored for community detection by maximizing the intra-link density in all communities and minimizing the inter-link density between different communities [53, 132, 205]. Since general MOEAs are effective for the networks with hundreds or thousands of nodes, an MOEA that can iteratively reduce the network size during the evolutionary process was designed for handling networks with more nodes [214]. In addition, some MOEAs have been proposed for detecting overlapping communities (i.e., a node may belong to multiple communities) [96, 168], and some MOEAs have been proposed for the community detection of attributed networks (i.e., both the topological structure and node attributes should be considered) [97, 133].

Module identification aims to find a locally dense module from a network rather than divide all of the nodes, which is useful for identifying disease-related genes from protein-protein interaction networks [65]. By maximizing both the association of the module with the disease and intra-link density of the model, some MOEAs were developed for finding modules with functionally correlated genes, which not only enhances the molecular understanding of disease mechanisms but also can be effective in the classification of disease and control samples [138, 166, 191].

Critical node detection is an important task for many critical infrastructure networks, which aims to find the critical nodes (i.e., vulnerabilities) in the networks. The deletion of a few critical nodes has a very large destruction to the pairwise connectivity to the network—that is, the network will be split into many disconnected components if the critical nodes are deleted [86]. Since the deletion of many general nodes will not split the network at all, the critical node detection problem has a flat landscape and thus is very challenging. To properly solve the critical node detection problem, some MOEAs have been adopted to address the discrete decision space and flat landscape, which can find a number of critical node sets having different sizes and destructions to the network in a single run [173, 181, 208].

Influence maximization aims to select a number of seed nodes from a social network for the largest influence spread under cascade model [81]. By maximizing the influence and decreasing the diffusion time simultaneously, some MOEAs can find a number of seed sets having different sizes and influences [13, 119]. In addition, a recently proposed MOEA considered the diversity of the selected seeds based on the detected overlapping communities [206].

7.3 Applications in Routing Problems

Routing problems are encountered in many areas, including logistics, transportation, scheduling, communications, and so on, which are practical but very challenging [14]. On the one hand, routing problems aim to find one or more routes for the shortest distance and minimum consumption, where novel reproduction operators should be designed for the permutation based encoding instead of existing operators for the binary or real encodings. On the other hand, routing problems have various constraints such as the capacity and time window, posing stiff challenges for heuristics to find feasible and optimal solutions. Hence, some MOEAs have been tailored for different routing problems with the assistance of customized reproduction operators and local search strategies.

The *traveling salesman problem* is a basic routing problem that aims to find the shortest Hamiltonian cycle containing all nodes in a weighted and undirected complete graph [193]. By using the delicate reproduction operators for the permutation based encoding [121], some MOEAs have been employed for solving the large-scale multi-objective traveling salesman problem, which can find the Hamiltonian cycles for multiple graphs simultaneously [16, 109, 120]. More recently, an MOEA was combined with deep reinforcement learning for solving the multi-objective traveling salesman problem [93]. In addition, some MOEAs have been applied to variants of the traveling salesman problem, including those with profits [78] and those with both profits and maximum capacity [11].

The *vehicle routing problem* is one of the most important routing problems, which aims to find the routes of multiple vehicles covering all nodes in a graph [80]. In general, all vehicles start from the same depot and have a maximum capacity, hence the vehicle routing problem is much more difficult than the traveling salesman problem due to the more complex encoding and strict constraints of capacities. Some MOEAs with novel reproduction operators have been proposed for the vehicle routing problem, which can minimize both the travel distance and duration time [7, 79]. However, more MOEAs have been tailored for the vehicle routing problem with time windows (i.e., each node should be visited within a given time window), where local search strategies were developed in these MOEAs for repairing infeasible solutions and enhancing feasible solutions [27, 134, 177]. In addition, some MOEAs have been applied to the vehicle routing problem with multiple types of depots [72, 179] or destinations [157].

The *pickup and delivery problem* simulates a more complex scenario, where the graph contains multiple pairs of pickup locations and delivery locations, and each pickup location should be visited before the corresponding delivery location in the same route within a given time window [56]. Moreover, this problem usually allows the violations of time windows and unvisited nodes, aiming to minimize many objectives including the number of vehicles, travel distance, travel time of longest route, waiting time, delay time, and uncollected profit. To tackle such a large-scale many-objective optimization problem with highly discrete decision space, some MOEAs have been employed with the help of many-objective optimization techniques [38, 50, 180].

The *location routing problem* is the combination of the vehicle routing problem and the facility location problem, which considers multiple warehouses, distribution centers, and clients [111]. On the one hand, the locations of multiple distribution centers should be selected from candidate locations, where each distribution center receives products from one or more warehouses and sends them to multiple clients. On the other hand, the route between each distribution center and its clients should be minimized. In short, it is a challenging bi-level optimization problem since the routes can be optimized once the locations of distribution centers are determined. In the past decade, some MOEAs have cooperated with other heuristics to solve the location routing problem [113, 123, 182].

7.4 Applications in Economics

The concept of Pareto dominance in MOEAs was taken from economics and named after an economist [2]. In general, many optimization problems in economics contain multiple conflicting objectives (e.g., profit and risk) and are pursued based on large datasets (e.g., price series), hence they are essentially LSMOPs and have been solved by many MOEAs.

Portfolio optimization aims to optimize the weights of a number of investments for the maximum profit and the minimum risk, which is a typical LSMOP since a higher profit usually requires taking on more risk. According to the modern portfolio theory [112], the optimal solutions for portfolio optimization constitute a curve termed efficient frontier, which is the same as the Pareto front in multi-objective optimization. Hence, many MOEAs have been applied to the portfolio optimization of a large number of investments [100, 125, 146], and some of them considered more complex models including the cardinality constraints [48], buy-in thresholds [12], and roundlots constraints [150]. In addition, some work has been dedicated to the portfolio optimization of special areas with different objectives, such as insurance investments [145] and bank financial products [188].

Goods recommendation is a pattern mining task for recommending the goods to the customer according to the historical shopping lists, which aims to maximize both the frequency and occupancy of the pattern (i.e., set of recommended goods) [160]. Generally, a short pattern has a high frequency in the lists but occurs in a small proportion of all of the goods, and vice versa. Hence, the frequency and occupancy are conflicting with each other, and some MOEAs have been proposed for the goods recommendation problem with a large number of candidate goods [167, 212]. In addition, some MOEAs additionally maximized the profits of the recommended goods [204, 209].

P2P lending provides a platform for fund turnover, on which lenders expect to obtain stable returns and borrowers expect to receive money from lenders [216]. The goal of the platform is to assign one or more borrowers to each lender. Taking the expected returns and default probabilities as two conflicting objectives, some MOEAs have been employed for recommending loans given by borrowers to lenders, where the expected returns and default probabilities are calculated by models [207] or estimated by machine learning techniques [6, 217]. In addition, some work maximized the borrowing limit and minimized the interest rate from the borrowers' perspective [98].

7.5 Discussion

Real-world LSMOPs usually contain chaotic interactions between decision variables, where the decision variable grouping based MOEAs can hardly detect the variable interactions and solve them efficiently. By contrast, the other categories of MOEAs exhibit good efficiency on real-world LSMOPs, such as the novel reproduction operator based MOEA on feature selection and neural network training [171, 172], the dimensionality reduction based MOEAs on critical node detection and portfolio optimization [164, 165], and the problem transformation based MOEAs on time-varying ratio error estimation [62].

However, due to the computationally expensive objective functions in many applications, some MOEAs have been equipped with customized search strategies to improve the convergence speed, such as the duplication analysis for feature selection [189], the gradient based local search for neural network training [77], and the transaction-oriented initialization strategy for goods recommendation [212]. Furthermore, to solve the combinatorial LSMOPs in real-world applications, novel crossover and mutation operators have been customized for each type of encoding, including the binary encoding (e.g., feature selection and module identification), categorial encoding (e.g., data clustering and community detection), and permutation based encoding (e.g., traveling salesman problem and vehicle routing). In addition, some compressive encodings (e.g. the orthogonal encoding for neural network training [155] and the generative encoding for neural architecture search [152]) have been suggested to reduce the high-dimensional decision spaces.

8 FUTURE RESEARCH DIRECTIONS OF EVOLUTIONARY LARGE-SCALE MULTI-OBJECTIVE OPTIMIZATION

Although a variety of LSMOPs have been tackled by evolutionary algorithms, many issues related to large-scale multi-objective optimization remain challenging and need more research effort. According to many emerging applications in the real world, some desirable research directions are introduced in this section.

8.1 Improving the Effectiveness and Efficiency of Solving LSMOPs

Many LSMOPs in real-world applications are pursued based on large datasets, leading to computationally expensive objective functions and massive decision variables, such as the large-scale feature selection with about 45,000 candidate features [147] and the DNN training with more than 150,000 weights [104]. However, existing MOEAs are usually effective for LSMOPs with less than 10,000 decision variables, and both the function evaluation and offspring generation of these LSMOPs are very time consuming. Therefore, the effectiveness and efficiency should be improved to scale up the applications of MOEAs. For this aim, it is important to develop novel problem transformation, dimensionality reduction, and other techniques to reduce the high-dimensional search space so that MOEAs can effectively approximate the optimal solutions by using fewer function evaluations. In addition, the information about the objective functions and datasets (e.g., gradients in neural network training [194] and shopping lists in goods recommendation [212]) can be adopted to further improve the effectiveness, and the operations in MOEAs can be parallelized and accelerated by GPUs to improve the efficiency. On the contrary, it is less important to focus on selection strategies since many developed selection strategies can be directly taken from existing MOEAs for small-scale multi- and many-objective optimization.

8.2 Solving Sparse LSMOPs

Sparse LSMOPs widely appear in many real-world applications as summarized in the work of Tian et al. [165], whose Pareto optimal solutions are so sparse that the number of nonzero variables d' in each optimal solution is much smaller than the total number of decision variables d. Due to the time-consuming function evaluations of many real-world sparse LSMOPs, only the function evaluations sufficient for solving a d'-variable problem are available for solving a d-variable sparse LSMOP, hence most existing MOEAs cannot find the optimal solutions of sparse LSMOPs efficiently. By contrast, if an MOEA can detect and optimize only the nonzero variables, the sparse LSMOPs can be highly eased and the available function evaluations are sufficient for convergence. Although some MOEAs have been adopted to solve specific sparse LSMOPs as introduced in Section 7 (e.g., many applications in machine learning, network science, and economics), the development of generic MOEAs for sparse LSMOPs is still in its infancy, where only two MOEAs have been proposed in the literature, namely SparseEA [171] and MOEA/PSL [165]. In addition, it is desirable to design a performance indicator to measure the convergence, diversity, and sparsity of the population simultaneously.

8.3 Solving Computationally Expensive LSMOPs

The function evaluations of some real-world optimization problems are extremely expensive—for example, the shape optimization of air intake ventilation system takes several minutes to perform a computational fluid dynamics simulation to evaluate the quality of a candidate design [29], and the hyperparameter optimization of DNNs takes a couple of days to perform a training and validation process to evaluate the accuracy of a candidate network [115]. To solve these LSMOPs by using very few function evaluations, some surrogate models have been adopted in MOEAs to estimate the objective values of offsprings such that many function evaluations can be saved [28]. Although many existing surrogate-assisted MOEAs are effective for handling a few decision variables, they are unable to solve many real-world optimization problems that have a large number of decision variables, such as task-oriented pattern mining [212] and neural architecture search [8]. This is mainly due to the adopted surrogate models (e.g. Kriging [43]), which are inefficient to learn the mapping between a large number of decision variables and multiple objective functions. Although some initial attempts have been dedicated to solving specific computationally expensive optimization problems with hundreds of decision variables [167, 186, 218], the development of more efficient surrogate models for computationally expensive LSMOPs is highly desirable.

8.4 Solving Constrained LSMOPs

Although most MOEAs tailored for LSMOPs assume no constraint in the problems, various constraints appear in many real-world LSMOPs, such as vehicle routing with time windows [27], software product configuration [187], and time-varying ratio error estimation [62]. The constraints in many LSMOPs are as complex as the objective functions and should be strictly satisfied, hence constrained LSMOPs are much more challenging than unconstrained LSMOPs. However, a number of MOEAs have been proposed to solve constrained multi-objective optimization problems with a few decision variables, most of which are based on coevolutionary frameworks or multi-stage frameworks [47, 92, 170]. In view of this, the techniques for solving LSMOPs can be embedded in the frameworks for handling constraints to solve constrained LSMOPs such that the population can efficiently converge to the feasible boundary in the high-dimensional decision space.

8.5 Solving Multimodal LSMOPs

A multimodal optimization problem has multiple Pareto optimal solutions corresponding to the same or similar objective values, where these solutions are considerably different in the decision space and all of them should be found [159]. Multimodal LSMOPs also exist in some emerging applications, such as the ensemble of feature selection [201] and neural network training [77], where multiple diverse learning models are expected to be found as the ensemble members. Although some MOEAs have solved multimodal multi-objective optimization problems with a few decision variables by preserving the population diversity in the decision space [36, 107, 200], they are difficult to be applied to multimodal LSMOPs. This is mainly due to the adopted distance metrics (e.g., crowding distance and Euclidean distance), which are ineffective to measure the similarities between solutions in the high-dimensional decision space of LSMOPs [163]. To properly solve multimodal LSMOPs, more effective distance metrics and diversity preservation strategies should be developed to find multiple equivalent Pareto optimal solutions simultaneously.

8.6 Solving Robust LSMOPs

The objective functions of some LSMOPs are influenced by uncertain factors, such as the uncertain daily production quantities in order scheduling [40] and the uncertain road networks in shelter location [186]. To solve such LSMOPs, the robust solutions are more preferred than the Pareto optimal solutions, where a robust solution indicates that its objective values change slightly in its neighborhood and thus can alleviate the performance deterioration brought by the uncertainty [76]. For this aim, CNSDE/DVC [40] measures the robustness of each solution by comparing it with its neighboring solutions sampled by Latin hypercube sampling, which can successfully solve robust LSMOPs with hundreds of decision variables. Nevertheless, more efficient search strategies are expected to be developed for determining the robust region in the high-dimensional decision space by using fewer function evaluations.

9 CONCLUSION

This article has presented a comprehensive survey of evolutionary large-scale multi-objective optimization, including the methodologies, assessment methods, applications, and future directions. First, existing MOEAs for solving LSMOPs have been divided into three categories (i.e., decision variable grouping based MOEAs, decision space reduction based MOEAs, and novel search strategy based MOEAs), where each category of MOEAs has been elaborated and its main advantages and disadvantages have been discussed. Second, the benchmark problems for the performance assessment of MOEAs on LSMOPs have been presented. Third, some popular and emerging applications of evolutionary large-scale multi-objective optimization have been presented, including a variety of LSMOPs in machine learning, network science, vehicle routing, and economics. Last, some future research directions of evolutionary large-scale multi-objective optimization have been discussed. It is our hope that the work presented in this article will help promote the future development of this exciting research topic.

REFERENCES

[1] Yousef Abdi and Mohammad-Reza Feizi-Derakhshi. 2020. Hybrid multi-objective evolutionary algorithm based on search manager framework for big data optimization problems. *Applied Soft Computing* 87 (2020), 105991.

[2] Luigi Amoroso. 1938. Vilfredo pareto. *Econometrica* 6, 1 (1938), 1–21.

[3] Luis Miguel Antonio and Carlos A. Coello Coello. 2013. Use of cooperative coevolution for solving large scale multiobjective optimization problems. In *Proceedings of the 2013 IEEE Congress on Evolutionary Computation*. IEEE, Los Alamitos, CA, 2758–2765.

[4] Luis Miguel Antonio and Carlos A. Coello Coello. 2016. Decomposition-based approach for solving large scale multiobjective problems. In *Proceedings of the International Conference on Parallel Problem Solving from Nature*. 525–534.

[5] Luis Miguel Antonio, Carlos A. Coello Coello, Silvia González Brambila, Josué Figueroa González, and Guadalupe Castillo Tapia. 2019. Operational decomposition for large scale multi-objective optimization problems. In *Proceedings of the 2019 Annual Genetic and Evolutionary Computation Conference*. ACM, New York, NY, 225–226.

[6] Golnoosh Babaei and Shahrooz Bamdad. 2020. A multi-objective instance-based decision support system for investment recommendation in peer-to-peer lending. *Expert Systems with Applications* 150, 15 (2020), 113278.

[7] Hiba Bederina and Mhand Hifi. 2018. A hybrid multi-objective evolutionary optimization approach for the robust vehicle routing problem. *Applied Soft Computing* 71 (2018), 980–993.

[8] Irwan Bello, Barret Zoph, Vijay Vasudevan, and Quoc V. Le. Neural architecture search with reinforcement learning. In *Proceedings of the 5th International Conference on Learning Representations*.

[9] Nicola Beume, Boris Naujoks, and Michael Emmerich. 2007. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research* 181, 3 (2007), 1653–1669.

[10] Urvesh Bhowan, Mark Johnston, and Mengjie Zhang. Ensemble learning and pruning in multi-objective genetic programming for classification with unbalanced data. In *Proceedings of the 24th Australasian Joint Conference on Artificial Intelligence*.

[11] Julian Blank, Kalyanmoy Deb, and Sanaz Mostaghim. Solving the bi-objective traveling thief problem with multiobjective evolutionary algorithms. In *Proceedings of the 2017 International Conference on Evolutionary Multi-Criterion Optimization*.

[12] J. Branke, B. Scheckenbach, M. Stein, K. Deb, and H. Schmeck. 2009. Portfolio optimization with an envelope-based multi-objective evolutionary algorithm. *European Journal of Operational Research* 199, 3 (2009), 684–693.

[13] Doina Bucur, Giovanni Iacca, Andrea Marcelli, Giovanni Squillero, and Alberto Tonda. Multi-objective evolutionary algorithms for influence maximization in social networks. In *Proceedings of the 2017 European Conference on the Applications of Evolutionary Computation*.

[14] Jose Caceres-Cruz, Pol Arias, Daniel Guimarans, Daniel Riera, and Angel A. Juan. 2015. Rich vehicle routing problem: Survey. *ACM Computing Surveys* 47, 2 (2015), 1–28.

[15] Qing Cai, Lijia Ma, and Maoguo Gong. 2014. A survey on network community detection based on evolutionary computation. *International Journal of Bio-Inspired Computation* 8, 2 (2014), 84–98.

[16] Xinye Cai, Yexing Li, Zhun Fan, and Qingfu Zhang. 2015. An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization. *IEEE Transactions on Evolutionary Computation* 19, 4 (2015), 508–523.

[17] Bin Cao, Jianwei Zhao, Yu Gu, Yingbiao Ling, and Xiaoliang Ma. 2020. Applying graph-based differential grouping for multiobjective large-scale optimization. *Swarm and Evolutionary Computation* 53 (2020), 100626.

[18] Bin Cao, Jianwei Zhao, Zhihan Lv, and Xin Liu. 2017. A distributed parallel cooperative coevolutionary multiobjective evolutionary algorithm for large-scale optimization. *IEEE Transactions on Industrial Informatics* 13, 4 (2017), 2030–2038.

[19] Huangke Chen, Ran Cheng, Jinming Wen, Haifeng Li, and Jian Weng. 2020. Solving large-scale many-objective optimization problems by covariance matrix adaptation evolution strategy with scalable small subpopulations. *Information Sciences* 509 (2020), 457–469.

[20] Huangke Chen, Xiaomin Zhu, Witold Pedrycz, Shu Yin, Guohua Wu, and Hui Yan. 2018. PEA: Parallel evolutionary algorithm by separating convergence and diversity for large-scale multi-objective optimization. In *Proceedings of the 2018 IEEE International Conference on Distributed Computing Systems*. IEEE, Los Alamitos, CA, 223–232.

[21] Wenxiang Chen, Thomas Weise, Zhenyu Yang, and Ke Tang. 2010. Large-scale global optimization using cooperative coevolution with variable interaction learning. In *Proceedings of the 2010 International Conference on Parallel Problem Solving from Nature*. 300–309.

[22] Fan Cheng, Jiabin Chen, Jianfeng Qiu, and Lei Zhang. 2020. A subregion division based multi-objective evolutionary algorithm for SVM training set selection. *Neurocomputing* 394 (2020), 70–83.

[23] Ran Cheng. 2016. *Nature Inspired Optimization of Large Problems*. Ph.D. Dissertation. University of Surrey.

[24] Ran Cheng and Yaochu Jin. 2015. A competitive swarm optimizer for large scale optimization. *IEEE Transactions on Cybernetics* 45, 2 (2015), 191–204.

[25] Ran Cheng, Yaochu Jin, Kaname Narukawa, and Bernhard Sendhoff. 2015. A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling. *IEEE Transactions on Evolutionary Computation* 19, 6 (2015), 838–856.

[26] Ran Cheng, Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. 2017. Test problems for large-scale multiobjective and many-objective optimization. *IEEE Transactions on Cybernetics* 47, 12 (2017), 4108–4121.

[27] Tsung-Che Chiang and Wei-Huai Hsu. 2014. A knowledge-based evolutionary algorithm for the multiobjective vehicle routing problem with time windows. *Computers & Operations Research* 45 (2014), 25–37.

[28] Tinkle Chugh, Karthik Sindhya, Jussi Hakanen, and Kaisa Miettinen. 2019. A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms. *Soft Computing* 23 (2019), 3137–3166.

[29] Tinkle Chugh, Karthik Sindhya, Kaisa Miettinen, Yaochu Jin, Tomas Kratky, and Pekka Makkonen. 2017. Surrogate-assisted evolutionary multiobjective shape optimization of an air intake ventilation system. In *Proceedings of the 2017 IEEE Congress on Evolutionary Computation.* IEEE, Los Alamitos, CA.

[30] Carlos Alberto de Araújo Padilha, Dante Augusto Couto Barone, and Adriao Duarte Dória Neto. 2016. A multi-level approach using genetic algorithms in an ensemble of least squares support vector machines. *Knowledge-Based Systems* 106 (2016), 85–95.

[31] Kalyanmoy Deb and Ram Bhusan Agrawal. 1995. Simulated binary crossover for continuous search space. *Complex Systems* 9, 4 (1995), 115–148.

[32] Kalyanmoy Deb and Mayank Goyal. 1996. A combined genetic adaptive search (GeneAS) for engineering design. *Computer Science and Informatics* 26, 4 (1996), 30–45.

[33] Kalyanmoy Deb and Himanshu Jain. 2013. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation* 18, 4 (2013), 577–601.

[34] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197.

[35] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. 2005. Scalable test problems for evolutionary multiobjective optimization. In *Evolutionary Multiobjective Optimization.* Springer, 105–145.

[36] Kalyanmoy Deb and Santosh Tiwari. 2008. Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization. *European Journal of Operational Research* 185 (2008), 1062–1087.

[37] Yepeng Deng, Chunkai Zhang, and Xuan Wang. A multi-objective examples generation approach to fool the deep neural networks in the black-box scenario. In *Proceedings of the 2019 IEEE International Conference on Data Science in Cyberspace.* IEEE, Los Alamitos, CA.

[38] Imen Harbaoui Dridi, Ryan Kammarti, Mekki Ksouri, and Pierre Borne. 2011. Multi-objective optimization for the m-PDPTW: Aggregation method with use of genetic algorithm and lower bounds. *International Journal of Computers, Communications & Control* 6, 2 (2011), 246–257.

[39] Wei Du, Le Tong, and Yang Tang. 2018. A framework for high-dimensional robust evolutionary multi-objective optimization. In *Proceedings of the 2018 Annual Genetic and Evolutionary Computation Conference.* ACM, New York, NY, 1791–1796.

[40] Wei Du, Weimin Zhong, Yang Tang, Wenli Du, and Yaochu Jin. 2018. High-dimensional robust multi-objective optimization for order scheduling: A decision variable classification approach. *IEEE Transactions on Industrial Informatics* 15, 1 (2018), 293–304.

[41] R. Eberhart and J. Kennedy. 1995. A new optimizer using particle swarm theory. In *Proceedings of the 6th International Symposium on Micro Machine and Human Science.* IEEE, Los Alamitos, CA, 39–43.

[42] Matthias Ehrgott. 2005. *Multicriteria Optimization.* Springer Science & Business Media.

[43] M. T. M. Emmerich, K. C. Giannakoglou, and B. Naujoks. 2006. Single- and multi-objective evolutionary optimization assisted by Gaussian random field metamodels. *IEEE Transactions on Evolutionary Computation* 10, 4 (2006), 421–439.

[44] P. Shahsamandi Esfahani and A. Saghaei. 2017. A multi-objective approach to fuzzy clustering using ITLBO algorithm. *Journal of AI and Data Mining* 5, 2 (2017), 307–317.

[45] Jesús Guillermo Falcón-Cardona and Carlos A. Coello Coello. 2020. Indicator-based multi-objective evolutionary algorithms: A comprehensive survey. *ACM Computing Surveys* 53, 2 (2020), 1–35.

[46] Zhun Fan, Yi Fang, Wenji Li, Jiewei Lu, and Xinye Cai. 2017. A comparative study of constrained multi-objective evolutionary algorithms on constrained multi-objective optimization problems. In *Proceedings of the 2017 IEEE Congress on Evolutionary Computation.* IEEE, Los Alamitos, CA.

[47] Zhun Fan, Wenji Li, Xinye Cai, Li Hui, and Erik D. Goodman. 2019. Push and pull search for solving constrained multi-objective optimization problems. *Swarm and Evolutionary Computation* 44 (2019), 665–679.

[48] Jonathan E. Fieldsend, John Matatko, and Ming Peng. Cardinality constrained portfolio optimisation. In *Proceedings of the 5th International Conference on Intelligent Data Engineering and Automated Learning.*

[49] Asja Fischer and Christian Igel. 2012. An introduction to restricted Boltzmann machines. In *Proceedings of the Iberoamerican Congress on Pattern Recognition.* 14–36.

[50] Abel García-Nájera and Antonio López-Jaimes. 2018. An investigation into many-objective optimization on combinatorial problems: Analyzing the pickup and delivery problem. *Swarm and Evolutionary Computation* 38 (2018), 218–230.
[51] Mario Garza-Fabre, Julia Handl, and Joshua Damian Knowles. 2018. An improved and more scalable evolutionary approach to multiobjective clustering. *IEEE Transactions on Evolutionary Computation* 22, 4 (2018), 515–535.
[52] Michelle Girvan and Mark E. J. Newman. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America* 99, 12 (2002), 7821–7826.
[53] Maoguo Gong, Qing Cai, Xiaowei Chen, and Lijia Ma. 2014. Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition. *IEEE Transactions on Evolutionary Computation* 18, 1 (2014), 82–97.
[54] Maoguo Gong, Jia Liu, Hao Li, Qing Cai, and Linzhi Su. 2015. A multiobjective sparse feature learning model for deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems* 26, 12 (2015), 3263–3277.
[55] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*. Curran Associates, 2672–2680.
[56] L. Grandinetti, F. Guerriero, F. Pezzella, and O. Pisacane. 2014. The multi-objective multi-vehicle pickup and delivery problem with time windows. *Procedia: Social and Behavioral Sciences* 111, 5 (2014), 203–212.
[57] Emrah Hancer, Bing Xue, and Mengjie Zhang. 2020. A survey on feature selection approaches for clustering. *Artificial Intelligence Review* 53 (2020), 4519–4545.
[58] J. Handl and Joshua Damian Knowles. 2007. An evolutionary approach to multiobjective clustering. *IEEE Transactions on Evolutionary Computation* 11, 1 (2007), 56–76.
[59] Nikolaus Hansen and Andreas Ostermeier. 2001. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation* 9, 2 (2001), 159–195.
[60] Cheng He, Ran Cheng, Ye Tian, and Xingyi Zhang. 2020. Iterated problem reformulation for evolutionary large-scale multiobjective optimization. In *Proceedings of the 2020 IEEE Congress on Evolutionary Computation*. IEEE, Los Alamitos, CA.
[61] Cheng He, Ran Cheng, and Danial Yazdani. 2020. Adaptive offspring generation for evolutionary large-scale multiobjective optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*. Early access, July 10, 2020.
[62] Cheng He, Ran Cheng, Chuanji Zhang, Ye Tian, Qin Chen, and Xin Yao. 2018. Evolutionary large-scale multiobjective optimization for ratio error estimation of voltage transformers. *IEEE Transactions on Evolutionary Computation* 24, 5 (2018), 868–881.
[63] Cheng He, Shihua Huang, Ran Cheng, Kay Chen Tan, and Yaochu Jin. 2021. Evolutionary multiobjective optimization driven by generative adversarial networks (GANs). *IEEE Transactions on Cybernetics* 51, 6 (2021), 3129–3142.
[64] Cheng He, Lianghao Li, Ye Tian, Xingyi Zhang, Ran Cheng, Yaochu Jin, and Xin Yao. 2019. Accelerating large-scale multiobjective optimization via problem reformulation. *IEEE Transactions on Evolutionary Computation* 23, 6 (2019), 949–961.
[65] Shan He, Guanbo Jia, Zexuan Zhu, Daniel A. Tennant, Qiang Huang, Ke Tang, Jing Liu, Mirco Musolesi, John K. Heath, and Xin Yao. 2016. Cooperative co-evolutionary module identification with application to cancer disease module discovery. *IEEE Transactions on Evolutionary Computation* 20, 6 (2016), 874–891.
[66] Mardé Helbig and Andries P. Engelbrecht. 2014. Benchmarks for dynamic multi-objective optimisation algorithms. *ACM Computing Surveys* 46, 3 (2014), 37.
[67] John H. Holland. 1992. *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge, MA.
[68] Wenjing Hong, Ke Tang, Aimin Zhou, Hisao Ishibuchi, and Xin Yao. 2018. A scalable indicator-based evolutionary algorithm for large-scale multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 23, 3 (2018), 525–537.
[69] Seyedmohsen Hosseini and Abdullah Al Khaled. 2014. A survey on the imperialist competitive algorithm metaheuristic: Implementation in engineering domain and directions for future research. *Applied Soft Computing* 24 (2014), 1078–1094.
[70] Peiqiu Huang and Yong Wang. 2020. A framework for scalable bilevel optimization: Identifying and utilizing the interactions between upper-level and lower-level variables. *IEEE Transactions on Evolutionary Computation* 24, 6 (2020), 1150–1163.
[71] Simon Huband, Philip Hingston, Luigi Barone, and Lyndon While. 2006. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation* 10, 5 (2006), 477–506.
[72] E. Jabir, Vinay V. Panicker, and R. Sridharan. 2015. Multi-objective optimization model for a green vehicle routing problem. *Procedia: Social and Behavioral Sciences* 189, 15 (2015), 33–39.
[73] Anil Kumar Jain, M. Narasimha Murty, and P. J. Flynn. 1999. Data clustering: A review. *ACM Computing Surveys* 31, 3 (1999), 264–323.

[74] T. Jayabarathi, T. Raghunathan, and A. H. Gandomi. 2018. The bat algorithm, variants and some practical engineering applications: A review. In *Nature-Inspired Algorithms and Applied Optimization*. Studies in Computational Intelligence, Vol. 744. Springer, 313–330.

[75] Jun-Rong Jian, Zhi-Hui Zhan, and Jun Zhang. 2020. Large-scale evolutionary optimization: A survey and experimental comparative study. *International Journal of Machine Learning and Cybernetics* 11 (2020), 729–745.

[76] Yaochu Jin and J. Branke. 2005. Evolutionary optimization in uncertain environments—A survey. *IEEE Transactions on Evolutionary Computation* 9, 3 (2005), 303–317.

[77] Yaochu Jin and Bernhard Sendhoff. 2008. Pareto-based multiobjective machine learning: An overview and case studies. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38, 3 (2008), 397–415.

[78] Nicolas Jozefowiez, Fred Glover, and Manuel Laguna. 2008. Multi-objective meta-heuristics for the traveling salesman problem with profits. *Journal of Mathematical Modelling and Algorithms* 7 (2008), 177–195.

[79] Nicolas Jozefowiez, Frédéric Semet, and El-Ghazali Talbi. 2002. Parallel and hybrid models for multi-objective optimization: Application to the vehicle routing problem. In *Proceedings of the 2002 International Conference on Parallel Problem Solving from Nature*.

[80] Nicolas Jozefowiez, Frédéric Semet, and El-Ghazali Talbi. 2008. Multi-objective vehicle routing problems. *European Journal of Operational Research* 189, 2 (2008), 293–309.

[81] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY.

[82] Kedar V. Khandeparkar, Shreevardhan A. Soman, and Gopal Gajjar. 2017. Detection and correction of systematic errors in instrument transformers along with line parameter estimation using PMU data. *IEEE Transactions on Power Systems* 32, 4 (2017), 3089–3098.

[83] Shreya Khare, Rahul Aralikatte, and Senthil Mani. 2018. Adversarial black-box attacks on automatic speech recognition systems using multi-objective evolutionary optimization. arXiv:1811.01312.

[84] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv:1412.6980.

[85] Saku Kukkonen and Jouni Lampinen. 2005. GDE3: The third evolution step of generalized differential evolution. In *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*. IEEE, Los Alamitos, CA, 443–450.

[86] Mohammed Lalou, Mohammed Amin Tahraoui, and Hamamache Kheddouci. 2018. The critical node detection problem in networks: A survey. *Computer Science Review* 28 (2018), 92–117.

[87] Bingdong Li, Jinlong Li, Ke Tang, and Xin Yao. 2015. Many-objective evolutionary algorithms: A survey. *ACM Computing Surveys* 48, 1 (2015), 13.

[88] Chengtao Li, David Alvarez-Melis, Keyulu Xu, Stefanie Jegelka, and Suvrit Sra. 2017. Distributional adversarial networks. arXiv:1706.09549.

[89] Hui Li, Qingfu Zhang, and Jingda Deng. 2016. Biased multiobjective optimization and decomposition algorithm. *IEEE Transactions on Cybernetics* 47, 1 (2016), 52–66.

[90] Hui Li, Qingfu Zhang, Jingda Deng, and Zong-Ben Xu. 2018. A preference-based multiobjective evolutionary approach for sparse optimization. *IEEE Transactions on Neural Networks and Learning Systems* 29, 5 (2018), 1716–1731.

[91] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. 2018. Feature selection: A data perspective. *ACM Computing Surveys* 50, 6 (2018), 1–45.

[92] Ke Li, Renzhi Chen, Guangtao Fu, and Xin Yao. 2018. Two-archive evolutionary algorithm for constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 23, 2 (2018), 303–315.

[93] Kaiwen Li, Tao Zhang, and Rui Wang. 2021. Deep reinforcement learning for multiobjective optimization. *IEEE Transactions on Cybernetics* 51, 6 (2021), 3103–3114.

[94] Minghan Li and Jingxuan Wei. 2018. A cooperative co-evolutionary algorithm for large-scale multi-objective optimization problems. In *Proceedings of the 2018 Annual Genetic and Evolutionary Computation Conference*. ACM, New York, NY, 1716–1721.

[95] Xiaodong Li, Ke Tang, Mohammmad Nabi Omidvar, Zhenyu Yang, and Kai Qin. 2013. *Benchmark Functions for the CEC'2013 Special Session and Competition on Large-Scale Global Optimization*. Technical Report. RMIT University.

[96] Yangyang Li, Yang Wang, Jing Chen, Licheng Jiao, and Ronghua Shang. 2015. Overlapping community detection through an improved multi-objective quantum-behaved particle swarm optimization. *Journal of Heuristics* 21, 4 (2015), 549–575.

[97] Zhangtao Li, Jing Liu, and Kai Wu. 2017. A multiobjective evolutionary algorithm based on structural and attribute similarities for community detection in attributed networks. *IEEE Transactions on Cybernetics* 48, 7 (2017), 1963–1976.

[98] Zhihong Li, Lanteng Wu, and Hongting Tang. 2018. Optimizing the borrowing limit and interest rate in P2P system: From borrowers' perspective. *Scientific Programming* 2018 (2018), 2613739.

[99] Zhenyu Liang, Yunfan Li, and Zhongwei Wan. 2020. Large scale many-objective optimization driven by distributional adversarial networks. arXiv:2003.07013.

[100] D. Lin, S. Wang, and H. Yan. A multiobjective genetic algorithm for portfolio selection. In *Proceedings of the 5th International Conference on Optimization: Techniques and Applications*.

[101] Qiuzhen Lin, Jianqiang Li, Zhihua Du, Jianyong Chen, and Zhong Ming. 2015. A novel multi-objective particle swarm optimization with multiple search strategies. *IEEE Transactions on Evolutionary Computation* 247, 3 (2015), 732–744.

[102] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *Proceedings of the 7th International Conference on Learning Representations*.

[103] Jing Liu, Yaxiong Chi, Chen Zhu, and Yaochu Jin. 2017. A time series driven decomposed evolutionary optimization approach for reconstructing large-scale gene regulatory networks based on fuzzy cognitive maps. *BMC Bioinformatics* 18 (2017), 241.

[104] Jia Liu, Maoguo Gong, Qiguang Miao, Xiaogang Wang, and Hao Li. 2017. Structure learning for deep neural networks based on multiobjective optimization. *IEEE Transactions on Neural Networks and Learning Systems* 29, 6 (2017), 2450–2463.

[105] Ruochen Liu, Jin Liu, Yifan Li, and Jing Liu. 2020. A random dynamic grouping based weight optimization framework for large-scale multi-objective optimization problems. *Swarm and Evolutionary Computation* 55 (2020), 100684.

[106] Ruochen Liu, Rui Ren, Jin Liu, and Jing Liu. 2020. A clustering and dimensionality reduction based evolutionary algorithm for large-scale multi-objective problems. *Applied Soft Computing* 89 (2020), 106120.

[107] Yiping Liu, Gary G. Yen, and Dunwei Gong. 2019. A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies. *IEEE Transactions on Evolutionary Computation* 23, 4 (2019), 660–674.

[108] Zhichao Lu, Ian Whalen, Vishnu Boddeti, Yashesh Dhebar, Kalyanmoy Deb, Erik Goodman, and Wolfgang Banzhaf. NSGA-Net: Neural architecture search using multiobjective genetic algorithm. In *Proceedings of the 2019 Annual Genetic and Evolutionary Computation Conference*. ACM, New York, NY.

[109] Thibaut Lust and Jacques Teghem. 2009. Two-phase Pareto local search for the biobjective traveling salesman problem. *Journal of Heuristics* 16 (2009), 475–510.

[110] Xiaoliang Ma, Fang Liu, Yutao Qi, Xiaodong Wang, Lingling Li, Licheng Jiao, Minglei Yin, and Maoguo Gong. 2016. A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables. *IEEE Transactions on Evolutionary Computation* 20, 2 (2016), 275–298.

[111] Yannis Marinakis and Magdalene Marinaki. 2008. A particle swarm optimization algorithm with path relinking for the location routing problem. *Journal of Mathematical Modelling and Algorithm* 7 (2008), 59–78.

[112] Harry Markowitz. 1952. Portfolio selection. *Journal of Finance* 7, 1 (1952), 77–91.

[113] Iris Abril Martínez-Salazar, Julian Molina, Francisco Ángel Bello, Trinidad Gómez, and Rafael Caballero. 2014. Solving a bi-objective transportation location routing problem by metaheuristic algorithms. *European Journal of Operational Research* 234 (2014), 25–36.

[114] Yosef Masoudi-Sobhanzadeh, Yadollah Omidi, Massoud Amanlou, and Ali Masoudi-Nejad. 2019. Trader as a new optimization algorithm predicts drug-target interactions efficiently. *Scientific Reports* 9 (2019), 9348.

[115] Robert Miikkulainen, Cesare Alippi, Yoonsuck Choe, Francesco, and Carlo Morabito. 2019. Evolving deep neural networks. In *Artificial Intelligence in the Age of Neural Networks and Brain Computing*. Academic Press, 293–312.

[116] Seyedali Mirjalili. 2015. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-Based Systems* 89 (2015), 228–249.

[117] Seyedali Mirjalili. 2016. SCA: A sine cosine algorithm for solving optimization problems. *Knowledge-Based Systems* 96 (2016), 120–133.

[118] Seyedali Mirjalili and Andrew Lewis. 2016. The whale optimization algorithm. *Advances in Engineering Software* 95 (2016), 51–67.

[119] Azadeh Mohammadi and Mohamad Saraee. 2018. Finding influential users for different time bounds in social networks using multi-objective optimization. *Swarm and Evolutionary Computation* 40 (2018), 158–165.

[120] Deyvid Heric Moraes, Danilo Sipoli Sanches, Josimar da Silva Rocha, Jader Maikol, Caldonazzo Garbelini, and Marcelo Favoretto Castoldi. 2019. A novel multi-objective evolutionary algorithm based on subpopulations for the bi-objective traveling salesman problem. *Soft Computing* 23 (2019), 6157–6168.

[121] P. Larra Naga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic. 1999. Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review* 13, 2 (1999), 129–170.

[122] Antonio J. Nebro, Juan José Durillo, Jose Garcia-Nieto, C. A. Coello Coello, Francisco Luna, and Enrique Alba. 2009. SMPSO: A new PSO-based metaheuristic for multi-objective optimization. In *Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making*. IEEE, Los Alamitos, CA, 66–73.

[123] N. Nekooghadirli, R. Tavakkoli-Moghaddam, V. R. Ghezavati and S. Javanmard. 2014. Solving a new bi-objective location-routing-inventory problem in a distribution network by meta-heuristics. *Computers & Industrial Engineering* 76 (2014), 204–221.

[124] Bach Hoai Nguyen, Bing Xue, Peter Andreae, Hisao Ishibuchi, and Mengjie Zhang. 2020. Multiple reference points-based decomposition for multiobjective feature selection in classification: Static and dynamic mechanisms. *IEEE Transactions on Evolutionary Computation* 24, 1 (2020), 170–184.

[125] Kyong Joo Oh, Tae Yoon Kim, and Sungky Min. 2005. Using genetic algorithm to support portfolio optimization for index fund management. *Expert Systems with Applications* 28 (2005), 371–379.

[126] Mohammad Nabi Omidvar, Xiaodong Li, Yi Mei, and Xin Yao. 2014. Cooperative co-evolution with differential grouping for large scale optimization. *IEEE Transactions on Evolutionary Computation* 18, 3 (2014), 378–393.

[127] Aytug Onan, Serdar Korukoğlu, and Hasan Bulut. 2016. A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification. *Expert Systems with Applications* 62 (2016), 1–16.

[128] Qingfu Zhang, Aimin Zhou, Shizheng Zhao, Ponnuthurai Nagaratnam Suganthan, Wudong Liu, and Santosh Tiwari. 2008. *Multiobjective Optimization Test Instances for the CEC 2009 Special Session and Competition.* Technical Report. University of Essex, Colchester, UK.

[129] Andrea Paoli, Farid Melgani, and Edoardo Pasolli. 2009. Clustering of hyperspectral images based on multiobjective particle swarm optimization. *IEEE Transactions on Geoscience and Remote Sensing* 47, 12 (2009), 4175–4188.

[130] Vivek K. Patel and Vimal J. Savsani. 2015. Heat transfer search (HTS): A novel optimization algorithm. *Information Sciences* 324 (2015), 217–246.

[131] Romaric Pighetti, Denis Pallez, and Frédéric Precioso. Improving SVM training sample selection using multi-objective evolutionary algorithm and LSH. In *Proceedings of the 2015 IEEE Symposium on Computational Intelligence and Data Mining.* IEEE, Los Alamitos, CA.

[132] Clara Pizzuti. 2012. A multiobjective genetic algorithm to find communities in complex networks. *IEEE Transactions on Evolutionary Computation* 16, 3 (2012), 418–430.

[133] Clara Pizzuti and Annalisa Socievole. 2020. Multiobjective optimization and local merge for clustering attributed graphs. *IEEE Transactions on Cybernetics* 50, 12 (2020), 4997–5009.

[134] Yutao Qi, Zhanting Hou, He Li, Jianbin Huang, and Xiaodong Li. 2015. A decomposition based memetic algorithm for multi-objective vehicle routing problem with time windows. *Computers & Operations Research* 62 (2015), 61–77.

[135] Chao Qian, Yang Yu, and Zhihua Zhou. Pareto ensemble pruning. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence.*

[136] Hong Qian and Yang Yu. 2017. Solving high-dimensional multi-objective optimization problems with low effective dimensions. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence.* 875–881.

[137] Shufen Qin, Chaoli Sun, Yaochu Jin, Ying Tan, and Jonathan Fieldsend. 2021. Large-scale evolutionary multi-objective optimization assisted by directed sampling. *IEEE Transactions on Evolutionary Computation.* Early access, March 3, 2021.

[138] Sumanta Ray and Ujjwal Maulik. 2017. Identifying differentially coexpressed module during HIV disease progression: A multiobjective approach. *Scientific Reports* 7 (2017), 86.

[139] G. Thippa Reddy, M. Praveen Kumar Reddy, Kuruva Lakshmanna, Rajesh Kaluri, Dharmendra Singh Rajput, Gautam Srivastava, and Thar Baker. 2020. Analysis of dimensionality reduction techniques on big data. *IEEE Access* 8 (2020), 54776–54788.

[140] Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *Annals of Mathematical Statistics* 22, 3 (1951), 400–407.

[141] Alejandro Rosales-Pérez, Salvador García, Jesus A. Gonzalez, Carlos A. Coello Coello, and Francisco Herrera. 2017. An evolutionary multiobjective model and instance selection for support vector machines with Pareto-based ensembles. *IEEE Transactions on Evolutionary Computation* 21, 6 (2017), 863–877.

[142] Indrajit Saha, Ujjwal Maulik, and Dariusz Plewczynski. 2011. A new multi-objective technique for differential fuzzy clustering. *Applied Soft Computing* 11, 2 (2011), 2765–2776.

[143] Karam M. Sallam, Saber M. Elsayed, Ripon K. Chakrabortty, and Michael J. Ryan. 2020. Improved multi-operator differential evolution algorithm for solving unconstrained problems. In *Proceedings of the IEEE Congress on Evolutionary Computation.*

[144] Frederick Sander, Heiner Zille, and Sanaz Mostaghim. 2018. Transfer strategies from single- to multi-objective grouping mechanisms. In *Proceedings of the 2018 Annual Genetic and Evolutionary Computation Conference.* ACM, New York, NY, 729–736.

[145] Wen Shi, Wei-Neng Chen, Ying Lin, Tianlong Gu, Sam Kwong, and Jun Zhang. 2019. An adaptive estimation of distribution algorithm for multipolicy insurance investment planning. *IEEE Transactions on Evolutionary Computation* 23, 1 (2019), 1–14.

[146] J. Shoaf and J. A. Foster. The efficient set GA for stock portfolios. In *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation.* IEEE, Los Alamitos, CA.

[147] Sameer Singh, Jeremy Kubica, Scott Larsen, and Daria Sorokina. 2009. Parallel large scale feature selection for logistic regression. In *Proceedings of the 2009 SIAM International Conference on Data Mining.*

[148] An Song, Qiang Yang, Wei-Neng Chen, and Jun Zhang. 2016. A random-based dynamic grouping strategy for large scale multi-objective optimization. In *Proceedings of the 2016 IEEE Congress on Evolutionary Computation*. IEEE, Los Alamitos, CA, 468–475.

[149] Rainer Storn and Kenneth Price. 1997. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* 11, 4 (1997), 341–359.

[150] Felix Streichert, Holger Ulmer, and Andreas Zell. Comparing discrete and continuous genotypes on the constrained portfolio selection problem. In *Proceedings of the 2004 Genetic and Evolutionary Computation Conference*.

[151] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. 2019. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation* 23, 5 (2019), 828–841.

[152] Marcin Suchorzewski and Jeff Clune. 2011. A novel generative encoding for evolving modular, regular and scalable networks. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*. ACM, New York, NY.

[153] Masanori Suganuma, Mete Ozay, and Takayuki Okatani. 2018. Exploiting the potential of standard convolutional autoencoders for image restoration by evolutionary search. *Proceedings of Machine Learning Research* 80 (2018), 4771–4780.

[154] Yanan Sun, Bing Xue, Mengjie Zhang, and Gary G. Yen. 2018. A new two-stage evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation* 23, 5 (2018), 748–761.

[155] Yanan Sun, Gary G. Yen, and Zhang Yi. 2019. Evolving unsupervised deep neural networks for learning meaningful representations. *IEEE Transactions on Evolutionary Computation* 23, 1 (2019), 89–103.

[156] Takahiro Suzuki, Shingo Takeshita, and Satoshi Ono. Adversarial example generation using evolutionary multi-objective optimization. In *Proceedings of the 2019 IEEE Congress on Evolutionary Computation*. IEEE, Los Alamitos, CA.

[157] K. C. Tan, Y. H. Chew, and L. H. Lee. 2006. A hybrid multi-objective evolutionary algorithm for solving truck and trailer vehicle routing problems. *European Journal of Operational Research* 172, 3 (2006), 855–885.

[158] Ryoji Tanabe and Alex Fukunaga. 2013. Success-history based parameter adaptation for differential evolution. In *Proceedings of the 2013 IEEE Congress on Evolutionary Computation*. IEEE, Los Alamitos, CA.

[159] Ryoji Tanabe and Hisao Ishibuchi. 2020. A review of evolutionary multi-modal multi-objective optimization. *IEEE Transactions on Evolutionary Computation* 24, 1 (2020), 193–200.

[160] Linpeng Tang, Lei Zhang, Ping Luo, and Min Wang. Incorporating occupancy into frequent pattern mining for high quality pattern recommendation. In *Proceedings of the 21th ACM International Conference on Information and Knowledge Management*. ACM, New York, NY.

[161] Ye Tian, Ran Cheng, Xingyi Zhang, and Yaochu Jin. 2017. PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]. *IEEE Computational Intelligence Magazine* 12, 4 (2017), 73–87.

[162] Ye Tian, Cheng He, Ran Cheng, and Xingyi Zhang. 2019. A multistage evolutionary algorithm for better diversity preservation in multiobjective optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2019).

[163] Ye Tian, Ruchen Liu, Xingyi Zhang, Haiping Ma, Kay Chen Tan, and Yaochu Jin. 2021. A multi-population evolutionary algorithm for solving large-scale multi-modal multi-objective optimization problems. *IEEE Transactions on Evolutionary Computation* 25, 3 (2021), 405–418.

[164] Ye Tian, Chang Lu, Xingyi Zhang, Fan Cheng, and Yaochu Jin. 2020. A pattern mining based evolutionary algorithm for large-scale sparse multi-objective optimization problems. *IEEE Transactions on Cybernetics*. Early access, December 30, 2020.

[165] Ye Tian, Chang Lu, Xingyi Zhang, Kay Chen Tan, and Yaochu Jin. 2021. Solving large-scale multiobjective optimization problems with sparse optimal solutions via unsupervised neural networks. *IEEE Transactions on Cybernetics* 51, 6 (2021), 3115–3128.

[166] Ye Tian, Xiaochun Su, Yansen Su, and Xingyi Zhang. 2020. EMODMI: A multi-objective optimization based method to identify disease modules. *IEEE Transactions on Emerging Topics in Computational Intelligence*. Early access, August 21, 2020.

[167] Ye Tian, Shangshang Yang, Lei Zhang, Fuchen Duan, and Xingyi Zhang. 2019. A surrogate-assisted multiobjective evolutionary algorithm for large-scale task-oriented pattern mining. *IEEE Transactions on Emerging Topics in Computational Intelligence* 3, 2 (2019), 106–116.

[168] Ye Tian, Shangshang Yang, and Xingyi Zhang. 2020. An evolutionary multiobjective optimization based fuzzy method for overlapping community detection. *IEEE Transactions on Fuzzy Systems* 28, 11 (2020), 2841–2855.

[169] Ye Tian, Shangshang Yang, Xingyi Zhang, and Yaochu Jin. Using PlatEMO to solve multi-objective optimization problems. In *Proceedings of the 2019 IEEE Congress on Evolutionary Computation*. IEEE, Los Alamitos, CA.

[170] Ye Tian, Tao Zhang, Jianhua Xiao, Xingyi Zhang, and Yaochu Jin. 2021. A coevolutionary framework for constrained multi-objective optimization problems. *IEEE Transactions on Evolutionary Computation* 25, 1 (2021), 102–116.

[171] Ye Tian, Xingyi Zhang, Chao Wang, and Yaochu Jin. 2020. An evolutionary algorithm for large-scale sparse multi-objective optimization problems. *IEEE Transactions on Evolutionary Computation* 24, 2 (2020), 380–393.
[172] Ye Tian, Xiutao Zheng, Xingyi Zhang, and Yaochu Jin. 2020. Efficient large-scale multiobjective optimization based on a competitive swarm optimizer. *IEEE Transactions on Cybernetics* 50, 8 (2020), 3696–3708.
[173] Mario Ventresca, Kyle Robert Harrison, and Beatrice M. Ombuki-Berman. An experimental evaluation of multi-objective evolutionary algorithms for detecting critical nodes in complex networks. In *Proceedings of the 2015 European Conference on the Applications of Evolutionary Computation*.
[174] Nele Verbiest, Joaquín Derrac, Chris Cornelis, Salvador García, and Francisco Herrera. 2016. Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis. *Applied Soft Computing* 38 (2016), 10–22.
[175] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*. ACM, New York, NY, 1096–1103.
[176] Bin Wang, Yanan Sun, Bing Xue, and Mengjie Zhang. Evolving deep neural networks by multi-objective particle swarm optimization for image classification. In *Proceedings of the 2019 Annual Genetic and Evolutionary Computation Conference*. ACM, New York, NY.
[177] Jiahai Wang, Wenbin Ren, Zizhen Zhang, Han Huang, and Yuren Zhou. 2020. A hybrid multiobjective memetic algorithm for multiobjective periodic vehicle routing problem with time windows. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50, 11 (2020), 4732–4745.
[178] Jia Wang, Yuchao Su, Qiuzhen Lin, Lijia Ma, Dunwei Gong, Jianqiang Li, and Zhong Ming. 2020. A survey of decomposition approaches in multiobjective evolutionary algorithms. *Neurocomputing* 408 (2020), 308–330.
[179] Jiahai Wang, Taiyao Weng, and Qingfu Zhang. 2019. A two-stage multiobjective evolutionary algorithm for multiobjective multidepot vehicle routing problem with time windows. *IEEE Transactions on Cybernetics* 49, 7 (2019), 2467–2478.
[180] Jiahai Wang, Ying Zhou, Yong Wang, Jun Zhang, C. L. Philip Chen, and Zibin Zheng. 2016. Multiobjective vehicle routing problems with simultaneous delivery and pickup and time windows: Formulation, instances, and algorithms. *IEEE Transactions on Cybernetics* 46, 3 (2016), 582–594.
[181] Shuai Wang, Jing Liu, and Yaochu Jin. 2020. Surrogate-assisted robust optimization of large-scale networks based on graph embedding. *IEEE Transactions on Evolutionary Computation* 24, 4 (2020), 735–749.
[182] Yong Wang, Kevin Assogba, Yong Liu, Xiaolei Ma, Maozeng Xu, and Yinhai Wang. 2018. Two-echelon location-routing optimization with time windows based on customer clustering. *Expert Systems with Applications* 104 (2018), 244–260.
[183] Lyndon While, Philip Hingston, Luigi Barone, and Simon Huband. 2006. A faster algorithm for calculating hypervolume. *IEEE Transactions on Evolutionary Computation* 10, 1 (2006), 29–38.
[184] Siripen Wikaisuksakul. 2014. A multi-objective genetic algorithm with fuzzy c-means for automatic data clustering. *Applied Soft Computing* 24 (2014), 679–691.
[185] Yu Wu, Yongshan Zhang, Xiaobo Liu, Zhihua Cai, and Yaoming Cai. 2018. A multiobjective optimization-based sparse extreme learning machine algorithm. *Neurocomputing* 317 (2018), 88–100.
[186] Xiaoshu Xiang, Ye Tian, Jianhua Xiao, and Xingyi Zhang. 2020. A clustering-based surrogate-assisted multi-objective evolutionary algorithm for shelter location under uncertainty of road networks. *IEEE Transactions on Industrial Informatics* 16, 12 (2020), 7544–7555.
[187] Yi Xiang, Yuren Zhou, Zibin Zheng, and Miqing Li. 2018. Configuring software product lines by combining many-objective optimization and SAT solvers. *ACM Transactions on Software Engineering and Methodology* 26, 4 (2018), 14.
[188] Jian Xiong, Chao Zhang, Gang Kou, Rui Wang, Hisao Ishibuchi, and Fawaz E. Alsaadi. 2020. Optimizing long-term bank financial products portfolio problems with a multiobjective evolutionary approach. *Complexity* 2020 (2020), 3106097.
[189] Hang Xu, Bin Xue, and Mengjie Zhang. 2021. A duplication analysis based evolutionary algorithm for bi-objective feature selection. *IEEE Transactions on Evolutionary Computation* 25, 2 (2021), 205–218.
[190] Bing Xue, Mengjie Zhang, and Will N. Browne. 2013. Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE Transactions on Cybernetics* 43, 6 (2013), 1656–1671.
[191] Cheng-Hong Yang, Li-Yeh Chuang, and Yu-Da Lin. 2017. Multiobjective differential evolution-based multifactor dimensionality reduction for detecting genegene interactions. *Scientific Reports* 7 (2017), 12869.
[192] Peng Yang, Ke Tang, and Xin Yao. 2018. Turning high-dimensional optimization into computationally expensive optimization. *IEEE Transactions on Evolutionary Computation* 22, 1 (2018), 143–156.
[193] Shengxiang Yang, Miqing Li, Xiaohui Liu, and Jinhua Zheng. 2013. A grid-based evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation* 17, 5 (2013), 721–736.

[194] Shangshang Yang, Ye Tian, Cheng He, Xingyi Zhang, Kay Chen Tan, and Yaochu Jin. 2021. Gradient guided evolutionary approach to training deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems.* Early access, March 4, 2021.

[195] Zhenyu Yang, Ke Tang, and Xin Yao. 2008. Large scale evolutionary optimization using cooperative coevolution. *Information Sciences* 178, 15 (2008), 2985–2999.

[196] Zhaohui Yang, Yunhe Wang, Xinghao Chen, Boxin Shi, Chao Xu, Chunjing Xu, Qi Tian, and Chang Xu. CARS: Continuous evolution for efficient neural architecture search. In *Proceedings of the 2020 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* IEEE, Los Alamitos, CA.

[197] Xin Yao. 1993. A review of evolutionary artificial neural networks. *International Journal of Intelligent Systems* 8, 4 (1993), 539–567.

[198] Xin Yao. 1999. Evolving artificial neural networks. *Proceedings of the IEEE* 87, 9 (1999), 1423–1447.

[199] Jiao-Hong Yi, Li-Ning Xing, Gai-Ge Wang, Junyu Dong, Athanasios V. Vasilakos, Amir H. Alavi, and Ling Wang. 2020. Behavior of crossover operators in NSGA-III for large-scale optimization problems. *Information Sciences* 509 (2020), 470–487.

[200] Caitong Yue, Boyang Qu, and Jing Liang. 2018. A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems. *IEEE Transactions on Evolutionary Computation* 22, 5 (2018), 805–817.

[201] C. T. Yue, J. J. Liang, B. Y. Qu, K. J. Yu, and H. Song. 2019. Multimodal multiobjective optimization in feature selection. In *Proceedings of the 2019 IEEE Congress on Evolutionary Computation.* IEEE, Los Alamitos, CA.

[202] Saúl Zapotecas-Martínez, Carlos A. Coello Coello, Hernán E. Aguirre, and Kiyoshi Tanaka. 2019. A review of features and limitations of existing scalable multiobjective test suites. *IEEE Transactions on Evolutionary Computation* 23, 1 (2019), 130–142.

[203] Chunkai Zhang, Yepeng Deng, Xin Guo, Xuan Wang, and Chuanyi Liu. An adversarial attack based on multi-objective optimization in the black-box scenario: MOEA-APGA II. In *Proceedings of the 2019 International Conference on Information and Communications Security.*

[204] Lei Zhang, Guanglong Fu, Fan Cheng, Jianfeng Qiu, and Yansen Su. 2018. A multi-objective evolutionary approach for mining frequent and high utility itemsets. *Applied Soft Computing* 62 (2018), 974–986.

[205] Lei Zhang, Hebin Pan, Yansen Su, and Xingyi Zhang. 2014. A mixed representation based multi-objective evolutionary algorithm for overlapping community detection. *IEEE Transactions on Cybernetics* 47, 9 (2014), 2703–2716.

[206] Lei Zhang, Fengjiao Sun, Fan Cheng, Haiping Ma, and Xiaoyan Sun. An overlapping community detection based multi-objective evolutionary algorithm for diversified social influence maximization. In *Proceedings of the 2020 IEEE Congress on Evolutionary Computation.* IEEE, Los Alamitos, CA.

[207] Lei Zhang, Xinpeng Wu, Hongke Zhao, Fan Chen, and Qi Liu. 2020. Personalized recommendation in P2P lending based on risk-return management: A multi-objective perspective. *IEEE Transactions on Big Data.* Early access, May 8, 2020.

[208] Lei Zhang, Jiajun Xia, Fan Cheng, Jianfeng Qiu, and Xingyi Zhang. 2020. Multi-objective optimization of critical node detection based on cascade model in complex networks. *IEEE Transactions on Network Science and Engineering* 7, 3 (2020), 2052–2066.

[209] Lei Zhang, Shangshang Yang, Xinpeng Wu, Fan Cheng, Ying Xie, and Zhiting Lin. 2019. An indexed set representation based multi-objective evolutionary approach for mining diversified top-k high utility patterns. *Engineering Applications of Artificial Intelligence* 77 (2019), 9–20.

[210] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multi-objective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation* 11, 6 (2007), 712–731.

[211] Qingfu Zhang, Aimin Zhou, and Yaochu Jin. 2008. RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm. *IEEE Transactions on Evolutionary Computation* 12, 1 (2008), 41–63.

[212] Xingyi Zhang, Fuchen Duan, Lei Zhang, Fan Cheng, Yaochu Jin, and Ke Tang. 2017. Pattern recommendation in task-oriented applications: A multi-objective perspective [application notes]. *IEEE Computational Intelligence Magazine* 12, 3 (2017), 43–53.

[213] Xingyi Zhang, Ye Tian, Ran Cheng, and Yaochu Jin. 2018. A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization. *IEEE Transactions on Evolutionary Computation* 22, 1 (2018), 97–112.

[214] Xingyi Zhang, Kefei Zhou, Hebin Pan, Lei Zhang, Xiangxiang Zeng, and Yaochu Jin. 2020. A network reduction-based multiobjective evolutionary algorithm for community detection in large-scale complex networks. *IEEE Transactions on Cybernetics* 50, 2 (2020), 703–716.

[215] Yin Zhang, Gai-Ge Wang, Keqin Li, Wei-Chang Yeh, Muwei Jian, and Junyu Dong. 2020. Enhancing MOEA/D with information feedback models for large-scale many-objective optimization. *Information Sciences* 522 (2020), 1–16.

[216] Hongke Zhao, Yong Ge, Qi Liu, Guifeng Wang, Enhong Chen, and Hefu Zhang. 2017. P2P lending survey: Platforms, recent advances and prospects. *ACM Transactions on Intelligent Systems and Technology* 8, 6 (2017), 1–28.

[217] Hongke Zhao, Qi Liu, Guifeng Wang, Yong Ge, and Enhong Chen. Portfolio selections in P2P lending: A multi-objective perspective. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY.

[218] Mengjie Zhao, Kai Zhang, Guodong Chen, Xinggang Zhao, Chuanjin Yao, Hai Sun, Zhaoqin Huang, and Jun Yao. 2020. A surrogate-assisted multi-objective evolutionary algorithm with dimension-reduction for production optimization. *Journal of Petroleum Science and Engineering* 192 (2020), 107192.

[219] Aimin Zhou, Bo-Yang Qu, Hui Li, Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, and Qingfu Zhang. 2011. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation* 1 (2011), 32–49.

[220] Heiner Zille. 2019. *Large-Scale Multi-Objective Optimisation: New Approaches and a Classification of the State-of-the-Art*. Ph.D. Dissertation. Otto-von-Guericke-Universität Magdeburg, Fakultät für Informatik.

[221] Heiner Zille, Hisao Ishibuchi, Sanaz Mostaghim, and Yusuke Nojima. 2016. Mutation operators based on variable grouping for multi-objective large-scale optimization. In *Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence*. IEEE, Los Alamitos, CA.

[222] Heiner Zille, Hisao Ishibuchi, Sanaz Mostaghim, and Yusuke Nojima. 2016. Weighted optimization framework for large-scale multi-objective optimization. In *Proceedings of the 2016 Annual Genetic and Evolutionary Computation Conference Companion*. ACM, New York, NY, 83–84.

[223] Heiner Zille, Hisao Ishibuchi, Sanaz Mostaghim, and Yusuke Nojima. 2018. A framework for large-scale multiobjective optimization based on problem transformation. *IEEE Transactions on Evolutionary Computation* 22, 2 (2018), 260–275.

[224] Heiner Zille and Sanaz Mostaghim. Comparison study of large-scale optimisation techniques on the LSMOP benchmark functions. In *Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence*.

[225] Heiner Zille and Sanaz Mostaghim. 2019. Linear search mechanism for multi- and many-objective optimisation. In *Proceedings of the 2019 International Conference on Evolutionary Multi-Criterion Optimization*. 399–410.

[226] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. 2000. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation* 8, 2 (2000), 173–195.

Received August 2020; revised May 2021; accepted June 2021