Improved Regularity Model-based EDA for Many-objective Optimization

Yanan Sun, *Member, IEEE*, Gary G. Yen, *Fellow, IEEE*, and Zhang Yi, *Fellow, IEEE*

Abstract—The performance of multi-objective evolutionary algorithms deteriorates appreciably in solving many-objective optimization problems which encompass more than three objectives. One of the known rationales is the loss of selection pressure which leads to the selected parents not generating promising offspring towards Pareto-optimal front with diversity. Estimation of distribution algorithms sample new solutions with a probabilistic model built from the statistics extracting over the existing solutions so as to mitigate the adverse impact of genetic operators. In this paper, an improved regularity-based estimation of distribution algorithm is proposed to effectively tackle unconstrained many-objective optimization problems. In the proposed algorithm, *diversity repairing mechanism* is utilized to mend the areas where need non-dominated solutions with a closer proximity to the Pareto-optimal front. Then *favorable solutions* are generated by the model built from the regularity of the solutions surrounding a group of representatives. These two steps collectively enhance the selection pressure which gives rise to the superior convergence of the proposed algorithm. In addition, dimension reduction technique is employed in the decision space to speed up the estimation search of the proposed algorithm. Finally, by assigning the Pareto-optimal solutions to the uniformly distributed reference vectors, a set of solutions with excellent diversity and convergence is obtained. To measure the performance, NSGA-III, GrEA, MOEA/D, HypE, MBN-EDA, and RM-MEDA are selected to perform comparison experiments over DTLZ and DTLZ$^-$ test suites with 3-, 5-, 8-, 10-, and 15-objective. Experimental results quantified by the selected performance metrics reveal that the proposed algorithm shows considerable competitiveness in addressing unconstrained many-objective optimization problems.

Index Terms—Estimation distribution algorithm (EDA), many-objective evolutionary algorithm (MaOEA), regularity-based EDA, diversity repairing, decision space dimension reduction.

I. INTRODUCTION

MANY-OBJECTIVE optimization problems (MaOPs) concern solving M conflicting objectives simultane-

Yanan Sun is with the College of Computer Science, Sichuan University, Chengdu 610065 CHINA and with the School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6140 NEW ZEALAND (e-mail:yanan.sun@ecs.vuw.ac.nz).

Gary G. Yen is with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078 USA (e-mail:gyen@okstate.edu).

Zhang Yi is with the College of Computer Science, Sichuan University, Chengdu 610065 CHINA (e-mail:zhangyi@scu.edu.cn).

ously where M is greater than three [1]. Generally, an MaOP has the following formulation described by Equation (1)

$$\begin{cases} f(x) = [f_1(x), \cdots, f_M(x)] \\ \quad s.t. \quad x \in \Lambda \end{cases} \quad (1)$$

where $\Lambda \in \mathbb{R}^n$ is the decision space, $f : \Lambda \to \Omega \in \mathbb{R}^M$ is the objective space. Without loss of generality, it is assumed that $f(x)$ is a minimization problem in which $f_1(x), \cdots, f_M(x)$ are to be minimized. Because of MaOPs widely existing in many real-world applications, such as management in land exploitation with 14-objective [2], calibration problems of automotive engine with 10-objective [3], to name a few, there is a strong incentive for efficiently and effectively solving MaOPs.

In MaOPs, there is no single perfect solution that optimizes all of the objectives at the same time but a set of Pareto-optimal solutions in which each individual is non-dominated with respect to each other. In addition, all the Pareto-optimal solutions constitute the Pareto-optimal set (PS) in the decision space while the image of PS produces a Pareto-optimal front (PF) in the objective space. Commonly, the goal in solving MaOPs is to obtain a limit number of Pareto-optimal solutions, which are uniformly distributed in PF, where a decision-maker can delegate a solution based on his or her preference. Among all the approaches for handing MaOPs, evolutionary algorithms are considered preferable because of the searching power exerted in these population-based meta-heuristics. During the past several decades, numerous multi-objective evolutionary algorithms (MOEAs), such as elitist non-dominated sorting genetic algorithm (NSGA-II) [4], advanced version of strength Pareto evolutionary algorithm (SPEA2) [5], have been developed for dealing with multi-objective optimization problems (MOPs) in which at most three objectives are to be optimized simultaneously. However, their performance degraded drastically in addressing MaOPs [6]. The main reason is the loss of selection pressure which is caused by the dominance resistance (DR) [7] and the curse of dimensionality [8] phenomena. To be specific, DR refers to a large proportion of solutions in which individuals are the best in one or very few objectives but far worse in others, and these solutions cannot be discriminated by the original Pareto domination principle. Then density-based secondary measurement is activated to decide which solutions are allowed to survive in the next generation [9]. Because of the behavior influenced by DR pointed out in [10], the selected solutions do not necessarily converge to the PF [8]. To this end, various many-objective evolutionary algorithms

(MaOEAs) for tackling MaOPs have been developed[1], such as multi-objective evolutionary algorithm based on decomposition (MOEA/D) [11], hypervolume-based many-objective optimization algorithm (HypE) [12], grid-based evolutionary algorithm for many-objective optimization (GrEA) [13], many-objective optimization algorithm using reference point based non-dominated sorting (NSGA-III) [14], and etc. [9], [15]–[17][2]. More precisely, MOEA/D employs the decomposition-based approach to construct a set of single objective problems by aggregating objectives considered in the original MaOP with different predefined weight vectors. New solutions are generated within a sub-region and diversity is maintained by the uniformly distributed weight vectors. The promising solutions are selected in HypE based on their fitness which is assigned by the corresponding contribution in hypervolume measure. As the computation of exact hypervolume is prohibitive, Monte Carlo simulation is employed to address this limitation. GrEA utilizes grid-based approach to show better performance in solving MaOPs by introducing the grid-based fitness comparison to relax the Pareto-based dominance relationship and grid-metrics to improve the diversity. Compared to NSGA-II, the improvement of NSGA-III is in the diversity mechanism by assigning the solutions to a set of uniformly distributed reference vectors. In summary, these state-of-the-art MaOEAs mentioned above mainly contemplate on two distinct issues: 1) reform the comparison manner in the traditional dominance relationship, such as HypE and GrEA and 2) apply new designs to reinforce the diversity, such as MOEA/D and NSGA-III.

Algorithm 1: Framework of An EDA

---

1 $t \leftarrow 0$;
2 $P_t \leftarrow$ Randomly initialize the population;
3 **while** *termination is not satisfied* **do**
4     $M \leftarrow$ Built probabilistic models from P_t;
5     $t \leftarrow t + 1$;
6     $U_t \leftarrow$ Generate offspring from M;
7     $P_t \leftarrow$ Select promising solutions from $U_t \cup P_{t-1}$;
8 **end**
9 **Return** P_t.

It is highly expected that individuals generated by selected parents with the crossover and mutation operators would march towards the PF in a MOP. However, this will not be the case in MaOPs due to the DR phenomenon. Specifically, if the parents are neighbors of DR solutions their offspring are also DR solutions. Otherwise, the newly generated solutions are not necessarily better than their parents who stand at a large space of many-objectives with a remote distance. This can be seen as the inefficiency of existing genetic operators for MaOPs [8]. Moreover, Deb *et al.* in [18] concluded that the

performances of MOEAs are significantly influenced by the genetic operators which cannot ensure to generate promising offspring. Furthermore, the parameters in genetic operators need empirically configured. For example, the distribution index of SBX in NSGA-III needs to be set at a larger number. For this purpose, researchers have developed estimation of distribution algorithms (EDAs) to tackle optimization problems [19]–[21] by generating new solutions without involving the traditional genetic operators, but probabilistic models which are built based on the statistics of the visited solutions. A general framework of EDAs is illustrated in Algorithm 1. Typically, the EDAs-based MOEAs are broadly classified into two categories based on their estimation models.

The first category covers the Bayesian network-based EDAs. For example, multi-objective Bayesian optimization algorithm [22] utilized the Bayesian optimization algorithm (BOA) to build a Bayesian network as its model for generating offspring. A related work was investigated in [23] to predict the model by strengthening Pareto ranking approach [24] and BOA. Furthermore, Laumanns *et al.* in [25] proposed a Bayesian multi-objective optimization algorithm whose model is built over the solutions selected by ϵ-Pareto ranking method [26]. In addition, an improved non-dominated sorting approach was employed by decision tree-based multi-objective EDA [27] to select a subset of solutions serving for a regression decision tree to learn the model. Recently, the multi-dimensional Bayesian network (MBN-EDA) was proposed in [21] specifically for addressing MaOPs.

The other category is often known as the mixture probability model-based EDAs. Examples include the multi-objective mixture-based iterated density estimation evolutionary algorithm [28] employing the mixed probability distributions to sample well-distributed solutions; and the multi-objective Parzen-based EDA [29] learning from the Gaussian and Cauchy kernels to build its models. In [30], the multi-objective hierarchical Bayesian optimization algorithm was designed by the mixture Bayesian network-based probabilistic model for discrete multi-objective optimization problems. In addition, the multi-objective extended compact genetic algorithm [31] took a marginal product model as the mixture of probability model. Furthermore, a regularity-based model EDA (RM-MEDA) was proposed in [20] in which the model is built based on the mixture normal distribution over the regularity. Zhou *et al.* [32] proposed a regularity-based method for solving the MOPs requiring the objective spaces to be $(m-1)$ dimensions.

It is believed that EDAs are capable of solving MaOPs without suffering the disadvantages of MOEAs with traditional genetic operators. Although, MBN-EDA has shown the promise in solving MaOPs, the development of many-objective optimization EDAs (MaOEDA) is still in infancy. Especially, probability models based on regularity have been extensively investigated in the discipline of statistical learning [33], [34], and regularity-based model are easier to build, and fairly effective. Based on our recent research achievements on this topic [35], [36] and motivated by the success of regularity-based EDAs for MOPs [20], [37], [38], an improved regularity-based EDA for MaOPs, in short named MaOEDA-IR, is proposed in this paper. To be specific, models employed to

---

[1] The algorithms which was designed originally for MOPs while is extended for MaOPs are also categorized to MaOEAs in this paper.

[2] Typically, these MaOEAs can be classified into three basic categories: Dominance-based, Decomposition-based and Hypervolume-based. MOEA/D and HypE are from the second and third categories, respectively. NSGA-III and GrEA are the hybridization of the first and the second categories.

generate new solutions in the proposed algorithm are built based on a group of neighbors selected by a uniformly distributed *reference vectors*. In order to improve the selection pressure, *diversity repairing* mechanism is developed to prevent the adverse DR phenomenon in each generation and push the new solutions toward having a closer proximity to the PF. Furthermore, *dimension reduction* technique is employed priori to the evolution to reduce the cost for exploration search. Specifically, convergence in the proposed algorithm is guaranteed by repairing diversity and sampling solutions based on the reference vectors, while diversity is facilitated by selecting solutions with the nearest perpendicular distances to the reference vectors. Compared to traditional MaOEAs and EDAs, the contributions of the proposed algorithm are summarized as follows:

1) Extend the uses of regularity model-based EDAs to MaOPs. In addition, reference vectors-based diversity mechanism is incorporated into the proposed algorithm to enhance the selection pressure.
2) Large search space poses a challenge for regularity model-based EDAs as do to all MaOEAs. To this end, dimension reduction technique is utilized in the decision space to speed up the exploration search for sampling promising solutions.
3) Convergence and diversity are considered equally important in the design of a quality MOEA or MaOEA. In the proposed algorithm, convergence is mainly treated in the first stage (the phase of dimension reduction), while diversity is focused in the second stage.

The remainder of this paper is organized as follows. Related reference vectors-based MaOEA, evolutionary algorithms based on dimension reduction, and the seminal work of regularity-based EDA are reviewed in Section II, respectively. In Section III, the framework of the proposed algorithm is outlined, and the respective steps are detailed. In addition, the complexity of the proposed algorithm are analyzed, and two crucial sub-components of the proposed algorithm as well as the principles for selecting neighbor solutions for building the model are discussed. In Section IV, a series of experiments are performed over widely used test suites against chosen peer competitors and the results measured by the selected performance metrics are statistically compared, in addition to experiments on investigating the effect of neighbor size, diversity repairing, and dimension reduction. Finally, conclusion and future work are given in Section V.

II. RELATED WORKS

Because the proposed algorithm are mainly concerned on the reference vectors, dimension reduction, and regularity-based EDA model, works related to these aspects are discussed. To be specific, state-of-the-art MaOEA, NSGA-III which is based on the reference vectors, is introduced first. Then the evolutionary algorithms employing dimension reduction are reviewed. Next, RM-MEDA which is a regularity-based EDA is analyzed. Finally, the disadvantages of RM-MEDA for solving MaOPs, and the differences to the proposed algorithm are highlighted.

A. NSGA-III

The major difference of NSGA-III compared to its predecessor is the diversity improving mechanism which is performed by the reference vectors, and it begins to take effect when j solutions need to be chosen from the non-dominated sorting front F_i where $i > 0$, $0 < j < |F_i|$, and $|\cdot|$ is a cardinality operator. To be specific, each reference vector is assigned to solutions in F_k where $0 < k < i - 1$ by calculating which solution has the nearest perpendicular distance to it. Then the reference vector v who is assigned the smallest number of solutions is marked and the solution p in F_i that has the smallest perpendicular distance to v is selected. Next, solution p is removed from F_i and the assigned number of v is increased by 1. These steps are iterated until j solutions are selected from F_i. Because these reference vectors are uniformly distributed, the selected solutions are hopefully evenly distributed in the objective space. Specifically, the reference vectors are uniformly generated in the R_+^M space, and the sum of elements in one reference vector is equal to 1. However, the problem to be optimized is not necessary in this unit hyperplane. For this purpose, all the objectives are to be normalized priori to calculating the perpendicular distances. This mechanism of uniformly distributed reference vector assisting to select solutions is more and more preferred by MaOEAs due to its explicitly diversity preserving nature.

B. Evolutionary Algorithms Based on Dimension Reduction

The notion that state-of-the-art MOEAs are capable of solving the MOPs naturally leads to an intent of reducing the number of objective in a MaOP (i.e., dimension reduction in the objective space), and then applying these powerful MOEAs in solving them. Specifically, dimension reduction in the objective space refers to removing the redundant objectives, while the same solutions are obtained as with all objectives involved [39]. With the utilization of dimension reduction, the computational complexity is reduced due to a smaller number of objectives. In summary, the dimension reduction schemes considered in literatures can be sorted into two categories. The first category is often known as the correlation-based methods, such as the works in [40]–[42]. In addition, the correntropy principal component analysis (C-PCA), maximum variance unfolding PCA (MVU-PCA) [43], and PCA-based algorithm [44] were proposed to analyze the correlation between the solutions generated in each generation, while Saxena *et al.* [45] proposed the linear PCA and nonlinear MVC over a set of Pareto-optimal solutions to check the correlation. In addition, Guo *et al.* [46] employed the interdependence factors to identify the correlation for dimension reduction. Recently, Wang and Yao proposed a novel approach to reduce the objective dimension by measuring the linear and nonlinear correlation between objectives using nonlinear correlation information entropy [47]. The second category covers the algorithms which employ the dimension reduction based on dominance structure, such as the work [41] in which the ϵ-dominance was employed to identify the redundant objectives. In addition, the Pareto Corner Search Evolutionary Algorithm [48] utilized the corner sorting technique to find the corner solutions in which the di-

mension reduction was performed. In summary, the dimension reduction techniques in these algorithms are performed in the objective space, while the proposed improved regularity-based MaOEDA builds its model in the decision space. Moreover, it is common that the number of decision variables is much greater than that of the objectives. As a consequence, it is reasonable to reduce the dimension of the decision space for the purpose of computational efficiency and effectiveness.

C. RM-MEDA

The probability model of RM-MEDA is built based on the regularity of decision space. To be specific, all the solutions are clustered into multiple disjoint groups by local PCA [49] first, and then models are constructed to generate new solutions for each group. Specifically, local PCA is employed for manifold dimension reduction by performing multiple PCA operations in each piecewise linear segment over the entire given data. Compared to PCA, local PCA is considered better to collectively capture the global structure. For intuitively comparing the effects of local PCA and PCA, an example is plotted in Fig. 1 in which it is clearly shown that the local PCA works better in estimating the entire structure of the data. Supposed that $\lambda_1, \cdots, \lambda_N$ are the eigenvalues of the covariance matrix of one group, $\lambda_1 \geq \cdots \geq \lambda_N$, and the corresponding eigenvectors are $v_1, \cdots, v_N$. Then the model is formulated with the assumption that the PS is a $(M-1)$-dimensional piecewise manifold in a continuous problem. In further, solutions are sampled from this model with the number which is proportional to the volume of the model to that of all the models. Combined with non-dominated sorting, a set of solutions with diversity approximating the PF are generated.

EDAs generate new solutions with probabilistic model to reject the detrimental consequence lead by genetic operator. Especially, regularity-based model is much preferable than most Bayesian network-based models in EDAs because of the simplicity yet effectiveness [50]. For example, EDAs based on Bayesian network need the procedure of training while regularity-based methods are analytic. However, because RM-MEDA is originally developed for solving MOPs with variables linkage [20], it may not be suitable for solving MaOPs. For example, the diversity in RM-MEDA is maintained by uniformly sampling new solutions in the decision space and this cannot give rise to the corresponding diversity in the objective space especially in MaOPs such as all the test problems in DTLZ [51]. Furthermore, local PCA makes sense when the PS is in full rank in the decision space, which is not necessary the case in MaOPs. To this end, an improved regularity-based EDA for effectively addressing MaOPs is proposed in this paper. In addition, a new diversity facilitating mechanism which employs the uniformly distributed reference vectors is also incorporated to improve the selection pressure. Furthermore, the dimension reduction technique based on the correlation scheme is utilized in the decision space over a set of Pareto-optimal solutions to save the computational complexity. Compared to RM-MEDA, the main contributions of the proposed algorithm are listed as follows.

1) The diversity improving mechanism of the proposed algorithm is compatible to most problems while RM-MEDA is only suitable to the problems in which the PS has the same image of PF.
2) The PS in RM-MEDA is not allowed to lie on any subspaces of the decision space, while the proposed algorithm has no such a requirement of it.
3) RM-MEDA is implemented with the manifold assumption that the PS is a piece-wise manifold with $(M-1)$-dimension. In this paper, the proposed algorithm is developed without such an assumption.

III. PROPOSED ALGORITHM

In this section, the framework of the proposed algorithm, i.e., improved regularity model-based EDA for many-objective optimization (MaOEDA-IR), is given first. Then the details of each step in the framework are presented. This is followed by the analysis of the computational complexity. Finally, significant sub-components of the proposed algorithm, and the principles of selecting neighbor solutions are discussed, respectively. It is noted here that the proposed algorithm is given in the context of problem described by Equation (1).

A. Framework of the Proposed Algorithm

The proposed algorithm begins with reducing the dimension of the decision space (Subsection III-B). Then new promising solutions are generated in the reduced decision space to speed up the evolutionary progress and lower the computational cost of exploitation search in the proposed algorithm, and their fitness are evaluated (Subsection III-C). Next, a set of generated reference vectors in R_+^M (it refers to the sub-part of R^M, where all points in this sub-part are with element values no less than 0) is mapped (Subsection III-D), and regularity-based model is built (Subsection III-E) for repairing the diversity of the proposed algorithm (Subsection III-F) in which a set of solutions R_t are generated. Based on R_t, new offspring are generated (Subsection III-G). With the help of environmental selection operator (Subsection III-H), a set of solutions with a better quality in convergence and diversity are obtained. Repairing the diversity, model updating, sampling new solutions, and environmental selection are performed one by one in a limit number of generations. At last, final selection is utilized to choose the representative solutions for the available slots (Subsection III-H). In addition, maximum generation number, a set of uniformly distributed reference vectors, neighbor size, and respective threshold for dimension reduction and model building need to be made available prior to the proposed algorithm running. In summary, the framework of the proposed algorithm is listed in Algorithm 2.

B. Reducing the Dimension of Decision Space

Dimension reduction technique is used to reduce the volume of exploration space to speed up the search of sampling new solutions. Ideally, the subspace of PS is desirable. To this end, a set of Pareto-optimal solutions is suitable to be the training data and then exploitation is performed in the subspace. Noted here that, the training data only require the solutions with convergence and the diversity is not necessary. Consequently,
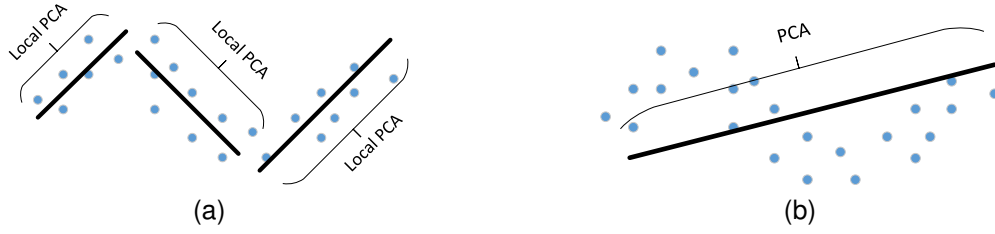
Fig. 1. An example comparison of local PCA and PCA on the same data. Specifically, Fig. 1a denotes the utilization of local PCA, while Fig. 1b denotes that of PCA. The solid lines on both figures denote the main directions of the principal components. Obviously, local PCA is better to capture the global structure of the give data.

Algorithm 2: Main Framework of MaOEDA-IR

Input: the maximum number of generations t_{max}, a set of unit reference vectors $\mathbf{r}_0 = \{r_{0,1}, \cdots, r_{0,N}\}$, neighbor size T, dimension reduction threshold α, regularity-based model threshold β;

Output: final population P;

1 Reduce the dimension of decision space from R^n to R^k;
2 Create population P_0 with TN individuals from R^k;
3 Evaluate the fitness of P_0;
4 Map reference vectors $\mathbf{r}_0$ to $\mathbf{v}_0 = \{v_{0,1}, \cdots, v_{0,N}\}$;
5 Build the regularity-based model Φ;
6 $t \leftarrow 0$;
7 **while** $t < t_{max}$ **do**
8     $R_t \leftarrow$ Repair the diversity;
9     $S \leftarrow$ Non-dominated selection from $P_t \cup R_t$;
10   Map reference vectors $\mathbf{v}_{t-1}$ to $\mathbf{v}_t = \{v_{t,1} \cdots, v_{t,N}\}$;
11    $Q_t \leftarrow$ Generate offspring;
12    $S \leftarrow$ Non-dominated selection from $P_t \cup R_t \cup Q_t$;
13   Update reference vectors $\mathbf{v}_t = \{v_{t,1} \cdots, v_{t,N}\}$;
14    $P_{t+1} \leftarrow$ Environmental selection from $Q_t \cup R_t \cup P_t$;
15    $t \leftarrow t + 1$;
16 **end**
17 $P \leftarrow$ Final selection from $P_{t_{max}}$;
18 **Return** P.

---

algorithms such as the conventional weighted aggregation method [52] for problems with convex PF, and the evolutionary dynamic weighted aggregation methods [53] for problems with non-convex PF are suitable for this. In this paper, the Pareto Corner Search Evolutionary Algorithm (PCSEA) [48] is employed because 1) a set of corner Pareto-optimal solutions which can be used as the training data as well as the extreme points (extreme points are employed for building the regularity-based model in Subsection III-E) are obtained simultaneously when the PCSEA completes, 2) the source code is available, and 3) the computational cost is promising compared to the algorithms selected for generating the extreme points and the training data. Moreover, the PCSEA has been successfully employed to generate solutions for dimension reduction in the objective space in its seminal paper. In each generation of PCSEA, $2M$ lists are increasingly ordered, where M is the number of objectives. Specifically, the first M lists are about the M objectives, while the last M lists are with the exclusive L_2 norm square. Especially, the exclusive L_2 norm square of

the i-th objective is with the form $\sum_{j=1, j\neq i}^{M} f_i^2$. By assigning the solutions in the top of the lists with a smaller rank value, the corner solutions are highlighted (more details can be found in [48]). When the evolution process of PCSEA is completed, the Pareto-optimal solutions, which are denoted by S_p, are selected from the population.

For convenience of the development, a matrix X is used to represent S_p. Specifically, each row in X denotes one solution while the columns refer to the different dimension of decision variables. Then the process of dimension reduction is illustrated in Algorithm 3 in which it can be divided into two parts. The first part is to employ the principal component analysis (PCA) to find the projected space in which X keeps its $(\alpha \times 100\%)$ information where α is a predefined threshold (lines 1-3). After the transformed data is projected back to its original space (in line 4), the values at reduced dimensions become 0. Thereafter, the index of the reduced dimensions I are selected, which are implemented in lines 5-11. Then, I is saved together with the mean value μ of X for evaluating the fitness of solutions generated in the reduced decision variable space.

Algorithm 3: Dimension Reduction

Input: the matrix X representing S_p, threshold α;

Output: the index of the removed dimension, the mean value of X;

1 $\mu \leftarrow$ Compute the column mean value of X;
2 $X' \leftarrow$ Subtract μ from X;
3 $U \leftarrow$ Select principal components with threshold α;
4 $\hat{X} \leftarrow U'UX'$;
5 $I \leftarrow \emptyset; V \leftarrow \emptyset$;
6 **for** $j \leftarrow 1$ **to** n **do**
7     $s \leftarrow$ compute the j-th column sum of $\hat{X}$;
8   **if** $s == 0$ **then**
9       $I \leftarrow I \cup j$;
10   **end**
11 **end**
12 **Return** I, μ.

In particular, the main reason for not using U but the original space is for reducing the computational complexity. Specifically, U is not the original decision space, and the solutions sampled from U cannot be directly used for fitness evaluation. There are two ways to solve this problem. The first one is to sample solutions from U and then transform

solutions into the original space when they are evaluated for the fitness. The other one is to transform U to the original space in advance, and then sample solutions from the original space. If we use the first method, the transformation operation needs to be performed for each solution in each generation. However, if we use the second method, we only need to do this transformation once. Obviously, the second method is with less computational complexity. Next, we will explain the mechanism of lines 5-11 in Algorithm 3.

By using PCA, the generated principal space U is with less number of dimensions compared to that of the original space. When U is transformed to the original space, the values of the dimensions reduced by PCA are all zeros. Therefore, we use lines 5-11 in Algorithm 3 to find these dimensions by checking where their values are zeros. Once we find these dimensions, we remove them and store their indexes. In Algorithm 3 we use I to denote the space without the reduced dimensions, and sample solutions from I. When these solutions are used for fitness evaluation, just padding their corresponding mean value (stored in μ) to the corresponding dimension based on the stored index, and using them to do fitness evaluation.

In most PCA-based methods, none of solutions sampled from the reduced space needs to be operated in the original space. However, in the proposed design, the solutions must be transformed back to the original space for fitness evaluation. Naturally, the dimension reduction technique used here is quite different from a general PCA-based method.

C. Creating Population and Evaluating Fitness

Assuming the dimension of the reduced decision variable space is k (obviously $k = n - |I|$). Based on the design principles of the proposed algorithm, each final solution of the proposed algorithm is desirable to be associated with one reference vector, and the model is built on the neighbor solutions of one particular reference vector. Therefore, the population is randomly initialized in R^k with size TN. In order to evaluate the fitness, the created population needs to be translated back to R^n, which is demonstrated by Algorithm 4. Specifically, the translated population P_0' has the same number to that of the initialized population P_0, and each individual in P_0' is with n-dimension. Furthermore, the values in reduced dimension are equal to that of the elements in the mean value vector μ. To this end, P_0' is initialized in $R^{TN \times n}$ (line 1) with zeros first. Then, the element values in each row of P_0' is set to be as μ (line 2). Finally, each column of P_0 is added to the corresponding column of P_0', which is implemented by lines 4-7. Once the translation is performed, the fitness is evaluated by introducing the population to the problem to be optimized.

D. Mapping Reference Vectors

Conventionally, Das and Dennis's method [54] is employed to generate the uniformly distributed reference vectors $\mathbf{r}_0$ which is constructed in R_+^M. However, the PF of the problem to be optimized does not necessary span the entire R_+^M space. In order to keep the same number of reference vectors in the PF, $\mathbf{r}_0$ needs to be mapped. For illustrating this motivation,

---

Algorithm 4: Translate the Population

Input: population $P_0 \in R^k$, index of reduced dimension I, mean value μ;
Output: population $P_0' \in R^n$;
1 Initialize a matrix $P_0' \in R^{TN \times n}$ with zeros;
2 Copy μ to each row of P_0';
3 $l \leftarrow 1$;
4 **for** $i \leftarrow 1$ **to** n and i **not in** I **do**
5   Add the l-th column of P_0 to the i-th column of P_0';
6     $l \leftarrow l + 1$;
7 **end**
8 **Return** P_0'.

Algorithm 5: Mapping Reference Vectors

Input: reference vectors $\mathbf{r}_0$, data S_p from Algorithm 3;
Output: mapped reference vector $\mathbf{v}_0$;
1 $F \leftarrow$ Calculate the objective values of S_p;
2 $\mathbf{z^u} \leftarrow \emptyset$;
3 **for** $i \leftarrow 1$ **to** M **do**
4     $\mathbf{v} \leftarrow [0, \cdots, 0]$;
5   **for each** f **in** F **do**
6   **if** $f_i > v_i$ **then**
7           $\mathbf{v} \leftarrow f$;
8   **end**
9   **end**
10     $\mathbf{z^u} \leftarrow \mathbf{z^u} \cup \mathbf{v}$;
11 **end**
12 $\mathbf{z}^* \leftarrow \emptyset$;
13 **for each** f **in** F **do**
14     $v \leftarrow +\infty$;
15   **for** $i \leftarrow 1$ **to** M **do**
16   **if** $f_i < v$ **then**
17           $v \leftarrow f_i$;
18   **end**
19   **end**
20     $\mathbf{z}^* \leftarrow \mathbf{z}^* \cup v$;
21 **end**
22 Update each z in $\mathbf{z^u}$ by $z \leftarrow z - \mathbf{z}^*$;
23 $\mathbf{a} = [a_1 \cdots, a_M] \leftarrow$ Find the intercepts of the hyperplane constructed by $\mathbf{z^u}$;
24 **for** $i \leftarrow 1$ **to** N **do**
25     $v_{0,i} \leftarrow = r_{0,i} \times \mathbf{a} + \mathbf{z}^*$;
26 **end**
27 **Return** $\mathbf{v}_0 = \{v_{0,1}, \cdots, v_{0,N}\}$.

---

an example in the 2-dimensional space is plotted in Fig. 2 in which nine blue lines denote the reference vectors and line AB denotes the PF. Specifically, Fig. 2a describes the reference vectors generated by Das and Dennis's method and only four reference vectors intersect AB, while Fig. 2b depicts the nine reference vectors intersect AB after they are mapped. Because the final solutions in the proposed algorithm are selected by their corresponding reference vector, it is obvious that the design in Fig. 2b is preferred. In addition, Algorithm 5 presents the details of mapping the reference vectors which

can be divided into four steps. First, the objective values of the training data for dimension reduction in Algorithm 3 and the extreme points (denoted by $\mathbf{z^u}$) are calculated (lines 1-11). Second, the ideal point is derived by selecting the minimum values in all objectives, which are implemented in lines 12-21. Noted here that, this approach to calculate the ideal point is also utilized in [55]–[59]. Third, the extreme points are updated by subtracting the ideal point and the intercepts are calculated (lines 22-23) by solving the equation[3]

E. Building the Regularity-based Model

Generally, the regularity-based model is composed of multiple sub-models due to the complexity of regularity on which a unified model is difficult to exactly capture the global intrinsic relation [20], [61]. In this proposed algorithm, each sub-model is built based on one reference vector with its neighbor solutions and Algorithm 6 presents the details.

Algorithm 6: Build the j-th Regularity-based Sub-model

Input: neighbor solutions S_n of the j-th reference vector, threshold β, enlargement factor γ;

Output: model Φ;

1 Let matrix X denote S_n;
2 $\mu \leftarrow$ Compute the column mean value of X;
3 $X' \leftarrow$ Subtract μ from each rom in X;
4 $[\lambda_1, \cdots, \lambda_M], [v_1, \cdots, v_M] \leftarrow$ Eigen-factorize the covariance matrix of X and descend the eigenvalues and eigenvectors;
5 $i \leftarrow 0$;
6 **while** $(\lambda_1 + \cdots + \lambda_i)/(\lambda_1 + \cdots + \lambda_M) < \beta$ **do**
7     $i \leftarrow i + 1$;
8 **end**
9 $y \leftarrow X' \times [v_1, \cdots, v_i]$;
10 $l, u \leftarrow$ Find the minimum and the maximum values in each row of y;
11 $\Omega \leftarrow \left\{ \mu + \sum_{j=1}^{i} \tau v_i, l - \gamma(u - l) \leq \tau \leq u + \gamma(u - l) \right\}$;
12 $\epsilon \leftarrow \frac{1}{M-i+1} \sum_{j=i}^{M} \lambda_j$;
13 **Return** $\Phi = \Omega + \epsilon$.

To be specific, given the neighbor solutions S_n of the j-th reference vector, the threshold β, and the enlargement factor γ. The steps of building the j-th sub-model are illustrated as follows. First, S_n is represented by the matrix X and centered (lines 1-3). Then the eigenvalues as well as the eigenvectors of X are obtained and ordered based on descending the eigenvalues (line 4). Lines 5-8 demonstrates the search of principal components on which the centered data is projected (line 9). Next, the projected space is constrained by find the minimum and the maximum values of projection (line 10),

and the latent space for generating new offspring is obtained with the enlargement factor and the principle components (line 11). Finally, the noise for the latent space is computed from the mean values of the eigenvalues regarding the non-principle components (line 12), and the regularity-based model is obtained.

F. Repairing the Diversity

Diversity repairing is employed by sampling new solutions to mitigate the adverse of the phenomenon that reference vectors lack associated solutions. To this end, all the non-dominated individuals in the current population are enumerated to find their respective nearest reference vectors first, which is implemented by lines 2-5. Then lines 8-10 demonstrate the selection of neighbor solutions. In addition, the model building and new solution sampling are presented in line 11 and lines 12-14, respectively. Noted in the phase of selecting neighbor solutions for reference vector $v_{t,i}$, its neighbor solutions are from the current population S_t and the non-dominated solutions in S_t, the motivation of which is discussed in Subsection III-J. In addition, it is obvious that the size of the neighbor solutions is not necessary with T. I.e., it is with $T + 1$ when the selected non-dominated neighbor solution has been included in line 8. The reasons that the neighbor solution size are not strictly kept with T are that 1) it does not affect the built model and 2) removing the extra solution gives more computational cost.

Algorithm 7: Repair the Diversity

Input: current population P_t, reference vectors $\mathbf{v}_t$, neighbour size T;

Output: new solutions R_t;

1 $S = \{s_1, \cdots, s_k\} \leftarrow$ Non-dominated selection from P_t;
2 $I \leftarrow \emptyset$;
3 **for each** s in S **do**
4     $I \leftarrow I \cup \arg\min_i ||s - v_{t,i} s v_{t,i}^T/(v_{t,i} v_{t,i}^T)||$;
5 **end**
6 $R_t \leftarrow \emptyset$;
7 **for** $i \leftarrow 1$ **to** N **and** i **not in** I **do**
8   neighbour$(i) \leftarrow$ Select T solutions from current population who have the smallest perpendicular distances to $v_{t,i}$;
9     $j \leftarrow \arg\min_j ||s_j - v_{t,i} s_j v_{t,i}^T/(v_{t,i} v_{t,i}^T)||$;
10   neighbour$(i) \leftarrow$ neighbour$(i) \cup s_j$;
11   Build model $\Phi_i = \Omega_i + \epsilon_i$ with neighbour(i);
12     $A \leftarrow$ Uniformly sample T points in $[l - \gamma(u - l), u + \gamma(u - l)]$ from Ω_i;
13     $B \leftarrow$ Sample points from the normal distribution with mean 0 and standard derivation $\sqrt{\epsilon_i}$;
14     $R \leftarrow (A + B)$;
15     $R_t \leftarrow R_t \cup R$;
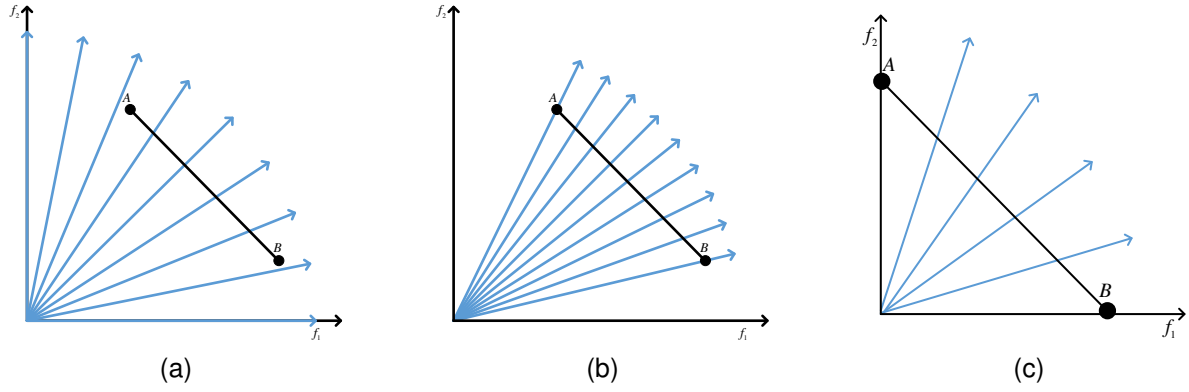16 **end**
17 **Return** R_t.

Fig. 2. A 2-dimensional example to illustrate the motivation of mapping the reference vectors. Specifically, there are nine reference vectors (blue lines) generated by the Das and Dennis's method, and line AB denotes the Pareto front. The reference vectors without mapping are plotted in Fig. 2a in which only four reference vectors intersect AB, while the reference vectors which have been mapped are plotted in Fig. 2b in which all the sampled reference vectors intersect AB (i.e., the situation that Algorithm 5 have been performed). Fig. 2c denotes the situation that only lines 1-22 in Algorithm 5 have been performed.

G. Generating Offspring

After repairing the diversity, the solution set R_t is generated. Then, non-dominated solutions in S are selected from R_t and the current population P_t (line 9 of Algorithm 2). Next, reference vectors $\mathbf{v}_t = \{v_{t,1}, \cdots, v_{t,N}\}$ are mapped (line 10 of Algorithm 2), which is performed by Algorithm 5 with the input parameters $\mathbf{v}_{t-1}$ and S. Finally, offspring Q_t are generated by Algorithm 8. In summary, generating offspring can be viewed as the diversity repairing in each reference vector, which could be investigated through the analogy between lines 4-10 of Algorithm 8 and lines 8-14 of Algorithm 7. However, the motivation of generating offspring is different to that of diversity repairing, which will be discussed in Subsection III-J. In addition, updating the reference vectors is motivated by achieving a better performance of the proposed algorithm, although it has been reported that PCSEA is capable of finding the approximated corner solutions from which the extreme points are derived [48].

Algorithm 8: Generate Offspring

Input: non-dominated solutions $S = \{s_1, \cdots, s_k\}$,
1 neighbour size T, reference vectors $\mathbf{v}_t$;
      Output: offspring Q_t;
2 $Q_t \leftarrow \emptyset$;
3 **for** $i \leftarrow 1$ **to** N **do**
4   neighbour$(i) \leftarrow$ Select T solutions from S who have the smallest perpendicular distances to $v_{t,i}$;
5     $j \leftarrow \arg\min_j ||s_j - v_{t,i}s_j v_{t,i}^T/(v_{t,i}v_{t,i}^T)||$;
6   neighbour$(i) \leftarrow$ neighbour$(i) \cup s_j$;
7   Build model $\Phi_i = \Omega_i + \epsilon_i$ with neighbour(i);
8     $A \leftarrow$ Uniformly sample T points in $[l - \gamma(u - l), u + \gamma(u - l)]$ from Ω_i;
9     $B \leftarrow$ Sample points from the normal distribution with mean 0 and standard derivation $\sqrt{\epsilon_i}$;
10     $Q \leftarrow (A + B)$;
11     $Q_t \leftarrow Q_t \cup Q$;
12 **end**
13 **Return** Q_t.

H. Environmental Selection and Final Selection

After offspring Q_t are generated, non-dominated solutions S are selected from the current population, i.e., $P_t \cup R_t \cup Q_t$ (line 12 of Algorithm 2). Then, reference vectors $\mathbf{v}_t = \{v_{t,1}, \cdots, v_{t,N}\}$ are updated (line 13 of Algorithm 2) by performing Algorithm 5 with the input parameters S and itself. Next, the environmental selection is performed. Specifically, the environmental selection aims at removing ill-fit solutions from the current population and maintaining a limit size of representative individuals to reduce the cost of computation in each generation. The final selection is to choose the best-fit solutions. Moreover, both selections are dependent in the proposed algorithm. In the framework of the proposed algorithm, it is assumed that N solutions are needed by the decision-maker when the algorithm is finished. Because the proposed algorithm is based on the statistics of regularity of the T neighbors regarding each solution in N. As a consequence, the number of solutions for building the model should be TN. In addition, extra solutions are incorporated from the phases of diversity repairing and offspring generating in each generation. To this end, the purpose of environmental selection is for maintaining a size of population with the same number of initialized population while the final selection is for selecting N solutions. It is hopeful that each reference point has T neighbor solutions after the environmental selection which is described by Algorithm 9. Specifically, final selection is implemented if the number T is replaced by 1 in the environmental selection.

In summary, the environmental selection covers two steps. The first step is to assign the reference vectors by selecting the non-dominated solutions which have smallest perpendicular distances to them (lines 2-6 of Algorithm 9). The other is to set T solutions for each reference vector. Specifically, the non-dominated sorting and truncated selection is employed when more than T solutions are assigned to one reference vector (lines 7-20 of Algorithm 9), otherwise necessary solutions are selected from the current population based on the smallest perpendicular distances (lines 22-30 of Algorithm 9). Noted that, selected solutions are removed from the current popula-

---

Algorithm 9: Environmental Selection

Input: non-dominated solutions $S = \{s_1, \cdots, s_k\}$,
1   neighbour size T, reference vectors $\mathbf{v}_t$; **Output:** P_{t+1};
2   $L_1, \cdots, L_N \leftarrow \emptyset$;
3   **for each** s **in** S **do**
4     $i \leftarrow \arg\min_i ||s - v_{t,i}sv_{t,i}^T/(v_{t,i}v_{t,i}^T)||$;
5     $L_i \leftarrow L_i \cup s$;
6   **end**
7   **for** $i \leftarrow 1$ **to** N **do**
8   **if** $|L_i| > T$ **then**
9       $\{F_1, \cdots, F_l\} \leftarrow$ Non-dominated sorting solutions in L_i;
10      $k \leftarrow 1, L_i \leftarrow \emptyset$;
11   **while** $|L_i| + |F_k| \leq T$ **do**
12        $L_i \leftarrow L_i \cup F_k$;
13        $k \leftarrow k + 1$;
14   **end**
15   **if** $|L_i| < T$ **then**
16        $D \leftarrow$ Select $T - |L_i|$ solutions from F_k who have the smallest perpendicular distances to $v_{t,i}$;
17        $L_i \leftarrow L_i \cup D$;
18   **end**
19   **end**
20   **end**
21   From current population removing solutions in $L_i \cup \cdots \cup L_N$;
22   **for** $i \leftarrow 1$ **to** N **do**
23   **if** $|L_i| < T$ **then**
24       $D \leftarrow$ Select $T - |L_i|$ solutions from current population who have the smallest perpendicular distances to $v_{t,i}$;
25       $L_i \leftarrow L_i \cup D$;
26   **else**
27       $D \leftarrow$ Select T solutions from L_i who have the smallest perpendicular distances to $v_{t,i}$;
28   **end**
29   From current population removing solutions in D;
30   **end**
31   **Return** $Q_t = L_i \cup \cdots \cup L_N$.

---

tion to avoid repetitions being re-selected (lines 21 and 29 of Algorithm 9).

I. Computational Complexity

Complexity of the proposed algorithm is analyzed with the context of problem defined by Equation (1). For convenience, the number of solutions for PCSEA to generate the training data for dimension reduction is set to be same to that in the final selection of the proposed algorithm. Consequently, the computational times of PCSEA is $O(MNlog(N))$. For the dimension reduction, the most cost is for computing the eigenvalues and eigenvectors which has $O((TN)^3)$ computations. In summary, the computational times of dimension reduction is $O((TN)^3)$. I addition, the creating population and fitness evaluation need $O(TNk)$ and $O(TNM)$ computation

times, respectively. The computational complexity of mapping reference vectors is $O(TNM)$. Moreover, the complexity of building the model is mainly contributed by the matrix factorization whose complexity is $O(T^3)$. Briefly, lines 1-5 in Algorithm 2 takes $O((TN)^3)$ computational times. Furthermore, the model building, non-dominated selection from $P_t \cup R_t$, non-dominated selection from $P_t \cup R_t \cup Q_t$ dominate the computational complexity of the diversity repairing, generating offspring, and environmental selection. As a consequence, the computational complexity of lines 8-14 is $O(MN^2)$ or $O(T^3)$ which is greater. In addition, the final selection takes $O(MN^2)$ computational times. Generally, the neighbor size T is generally set to be a number with order of magnitude 1, and the maximum generation is set to that of magnitude 1. In summary, the computational complexity of the proposed algorithm is $O((TN)^3)$, where T is the neighbor size, and N is the number of solutions the decision-makers require.

J. Discussion

In the proposed algorithm, two sub-components, dimension reduction, and diversity repairing are well-designed to guarantee the performance. In this section, their design motivations are discussed first, and then the experimental verification are presented in Sections IV-F and IV-G.

First, redundancy exists in the high dimensional data, and data from a small part of the dimensions is sufficient to represent these high dimensional data, such as in the feature selection discipline. In order to obtain these low dimensional data, dimension reduction technique is utilized. Employing these low dimensional data against the original high dimensional data can significantly benefit the utilization of data, such as lowering the computational cost, improving the precision by removing the interference from the elements in the redundant dimensions, and so on. Furthermore, a series of literatures [40]–[46], [48] have been proposed to reduce the objective number in MaOPs, thereafter state-of-the-art MOEAs can be utilized to solve them. Generally, the numbers of decision variables are greater than that of objectives in MaOPs, such as the M-dimensional DTLZ7 [51] with $19+M$ decision variables, and as well in MOPs, for example, 2-dimensional ZDT1 problem [62] with 30 decision variables. Actually, it is no surprise that problems are with the number of decision variables greater than that of objectives because it is difficult to determine which particular factors affect the response, and a better way for modeling is to select all the observed factors, which is always with a larger size. As a result, dimension reduction in the decision space is appropriate and well justified, specifically for the proposed algorithm which is based on the regularity of the decision space. Moreover, the Pareto-optimal solutions can be viewed as the features of the decision space in solving MaOPs, and if we obtain the subspace of the PS, subsequent operations can be constrained in this subspace to reduce the cost of exploration. Basically, this subspace is obtained by reducing the dimension on the Pareto-optimal solutions. Ideally, only the diversity is concerned if the exact subspace of the PS is obtained, which is not true in implementation. To this end, extra components are incorporated in

the proposed algorithm for improving the convergence, such as the reference vector updating, and non-dominated selection for building model, diversity repairing, generating offspring, environmental selection, and final selection.

To intuitively understand the diversity repairing mechanism in the proposed algorithm, an example of 3-objective DTLZ1 problem [51] is illustrated in Fig. 3. To be specific, twelve different markers in red are shown in Fig. 3a) identifying a set of uniformly distributed reference vectors in the objective space. The corresponding solutions (given the same markers) in the decision space with the reduced dimensions are shown in Fig. 3b) which are not necessarily uniformly distributed. For each given solution in the decision space, nine different neighbors are chosen and displayed in the same marker as shown in Fig. 3c). However, all the reference vectors are not necessarily assigned by the solutions in one generation due to the heuristic nature of evolution. This can be seen from Fig. 3d) in which blue markers denote all the solutions in this generation and the area constrained by the ellipse circle implies that there is no solution for assigning to the corresponding reference vector. To this end, diversity repairing mechanism is activated by using the neighbor solutions. The "cross" markers in Fig. 3e) are the solutions selected for generating new solutions. These solutions are then used for diversity repairing and generate solutions shown in "hollow circle" markers in Fig. 3f). Intuitively, diversity repairing is employed for improving the diversity by assigning the corresponding solution. In fact, the convergence is strengthened in the phase of diversity repairing by generating new solutions from which dominated solution is assigned to the corresponding reference vector which is short of diversity previously. In summary, both convergence and diversity are promoted by diversity repairing mechanism.

Normally, the neighbor solutions for one particular reference vector consist of one non-dominated solution and other nearest solutions to this reference vector from the current population. With this neighbor solution assignment, it is hopeful that solutions with convergence and diversity are sampled from the built model. Moreover, Fig. 4 highlights our motivation on this design. Especially, the blue line denotes the reference vector, black circles, A, B, C, D and E denotes the current population, rectangle area denotes the built model, and red circles denoted the sampled offspring. Fig. 4a plots the solutions which have the nearest perpendicular distances to the reference vectors, while Fig. 4b depicts the non-dominated solution included into the neighbor solutions. It is obviously that solutions with good diversity and convergence are generated in Fig. 4b.

IV. EXPERIMENTS

To demonstrate the quality of the proposed algorithm, a series of experiments are well-designed and performed on 8 test problems, which are from two benchmark test suits, DTLZ [51] and DTLZ^{-1} [63], with 3-, 5-, 8-, 10-, and 15-objective. Since the proposed MaOEDA-IR is an EDA-based algorithm for solving MaOPs, state-of-the-art algorithms covering two categories 1) traditional MaOEAs (NSGA-III [14], MOEA/D [11], GrEA [13], and HypE [12]) and
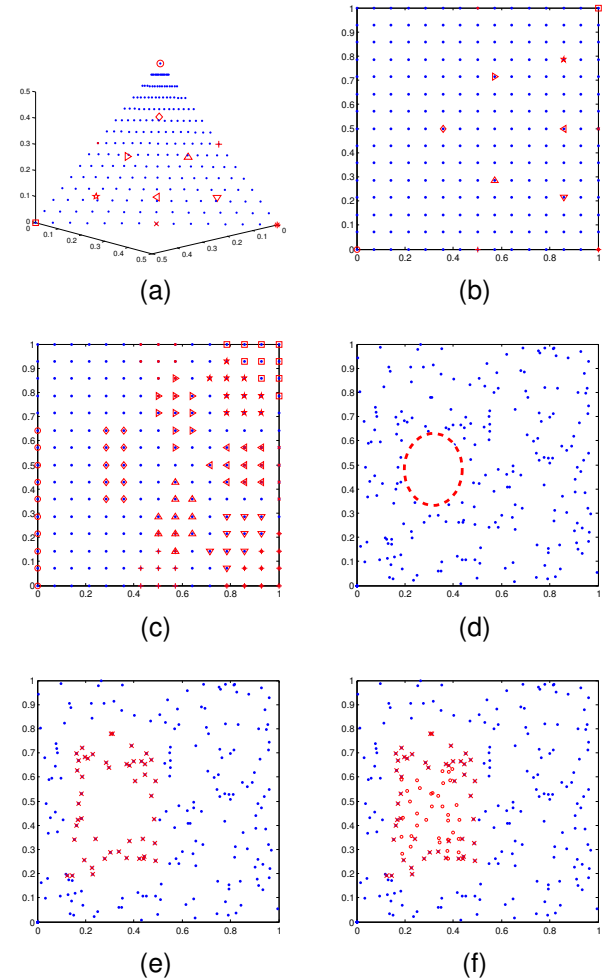

Fig. 3. A schematic diagram of the diversity repairing mechanism. In Figs. 3b)-3c), the blue markers denote the uniformly distributed solutions in the decision variable space with the reduced dimension, and their corresponding objectives are plotted with the same color in Fig. 3a). A set of uniformly distributed reference vectors is plotted in Fig. 3a) in red color and their corresponding solutions and neighbors are plotted with the same shape in Figs. 3b) and 3c), respectively. The blue markers in Figs. 3d)-3f) denote all the solutions in the reduced dimension decision variable space in one generation. Especially, the area constrained by the ellipse circle denotes there is no solution for assigning to the corresponding reference vector. The cross markers in Fig. 3e) are the solutions selected for generating new solutions which are plotted in Fig. 3f) with solid markers to repair the diversity of the corresponding reference vector.

2) EDA-based evolutionary algorithm (MBN-EDA [21], and RM-MEDA [20]) are considered as the peer competitors to compare the performance against the proposed MaOEDA-IR.

In the following subsections, the selected benchmark test problems are introduced first. Then, the performance indicators chosen to measure the results generated by these compared algorithms are documented. Next, the parameter settings utilized by compared algorithms are declared. Finally, experiments on compared algorithms are performed and their results measured by the selected performance indicators are analyzed. In addition, empirical experiments on investigating the diversity repairing, dimension reduction, and neighbor size are performed to highlight their superiority and promote
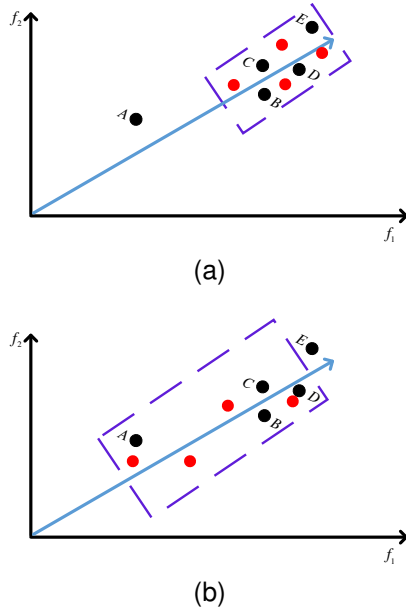
Fig. 4. An example to highlight the quality of generated solutions that the non-dominated solution A is included in the neighbor solutions (see Fig. 4b) or not (see Fig. 4a). Especially, the blue line denotes the reference vector, A, B, C, D and E denote the current population, rectangle area denotes the built model based on the selected neighbor solutions, and red circles are the generated solutions.

efficacy in addressing real-word problems.

A. Benchmark Test Problems

DTLZ1-DTLZ4 problems which are from the scalable benchmark test suit DTLZ are considered as the test instances in these experiments. Specifically, each M-objective test problem is with $n = M + k - 1$ decision variables where k is specified as 5 for DTLZ1 and 10 for DTLZ2-DTLZ4. Furthermore, the Pareto-optimal solutions of DTLZ1-DTLZ4 in the normalized M-dimensional objective space have the form formulated by Equation 2.

$$\sum_{i=1}^{M} f_i(x)^p = 1 \qquad (2)$$

where $p = 1$ for DTLZ1 and $p = 2$ for DTLZ2-DTLZ4. Because the employed reference vectors $r_{0,i} = [r_{0,i}^1, \cdots, r_{0,i}^M]$ generated by the systematical Das and Dennis's method in the proposed algorithm are with the form $\sum_{j=1}^{M} r_{0,i}^j = 1$ that is similar to Equation 2, DTLZ test problems are considered less challengeable. To this end, the DTLZ1^{-1}-DTLZ4^{-1} problems from the DTLZ^{-1} test suit, which is a variant of the DTLZ by multiplying the negative sign to each test problem in DTLZ, are included into the considered benchmark test problems for their more complicated PF shapes which especially challenges algorithms based on reference vectors.

B. Performance Metrics

Two widely used performance metrics, Inverted Generational Distance (IGD) [64] and Hypervolume (HV) [24] which can simultaneously quantify the performance in convergence and diversity of the algorithms, are adopted in these experiments. The results generated by these compared algorithms are normalized to $[0, 1]$ priori to employing the performance indicators, which is in the same manner [65]. In addition, $100,000$ reference points are uniformly sampled from Equation 2 for the calculation of IGD, and $[1.1, \cdots, 1.1]$ is specified as the reference point for the calculation of HV. Furthermore, Monte Carlo simulation [12] is applied for the calculation of HV when $M \geq 10$, otherwise the exact approach proposed in [66] is utilized due to the computation cost dramatically increasing as the number of objectives grows.

C. Parameter Settings

In this subsection, the parameter settings are presented. First, the general settings for most compared algorithms are listed. Thereafter, special settings for partial algorithms are specified.

1) Crossover and Mutation: SBX [67] and polynomial mutations [68] are employed as the crossover operator and mutation operator, respectively. Furthermore, the probabilities for SBX and polynomial mutation, and the crossover distribution index are set to be 1.0, $1/n$ (n is the number of decision variables), and 20, respectively. In addition, the distribution index of NSGA-III is set to be 30 according to the suggestions in [14] while others are set to be 20.

2) Population Size: The population size can be set arbitrarily for executing experiments. However, reference vectors assisted algorithms, such as NSGA-III, require the same number of the population size to that of reference vectors, other peer algorithms adopt the same population size for a fair comparison. Furthermore, only boundary reference vectors are generated when the division numbers is less than M in the phase of sampling of reference vectors. To this end, the two-layer approach [14] is employed for generating the reference vectors. In addition, the implementation of GrEA and NSGA-III require the population size to be a multiple of 4. In summary, the settings for reference vector and population size are listed in Table I.

TABLE I
SETTINGS FOR REFERENCE VECTORS AND POPULATION SIZE.

M	# of division		# of reference vectors	population size of GrEA and NSGA-III
	H_1	H_2		
3	14	-	120	120
5	5	-	126	128
8	3	2	156	156
10	2	2	110	112
15	2	1	135	136

3) Special Settings: The grid sizes of GrEA varying in $\{6, 7, 8, 9, 10, 11, 12\}$ are tested individually and the best scores selected based on the corresponding performance indicators are picked up for comparisons. Because RM-MEDA is originally designed for MOEAs, the default configurations are not suitable for solving MaOPs in this experiments. Consequently, the parameter setting in RM-MEDA is slightly modified for maximizing its performance to deal with MaOPs. Specifically, the number of clusters in local PCA varies in $\{10, 20, 30, 50\}$; the maximum iterations of local PCA with

$\{50, 100\}$ are tested individually with the maximum generations at 500 for selecting the best mean value indicated by performance metrics, while the parameters settings in MBN-EDA are set based on the developers' suggestion in [21]. In addition, both the thresholds for dimension reduction and model building are set to be 0.96 according to the convention of the community. Furthermore the neighbor size is specified as 25 for the investigation in Subsection IV-E, and the enlargement factor is set to be 0.5. In the settings of the proposed algorithm, the generation number of PCSEA for obtaining training data is specified as 50, and the population size is set to be 100.

The proposed MaOEDA-IR is based on the training data obtained by PCSEA. For a fair comparison, the termination criterion of MaOEDA-IR should be set to the total function evaluation number of the peer competitors minus that of PCSEA. Table II[4] shows these particular settings for each considered number of objective.

TABLE II
THE SETTINGS FOR THE MAXIMAL FUNCTION EVALUATION NUMBERS.

M	total evaluation numbers	evaluation numbers for PCSEA	evaluation numbers for MaOEDA-IR
3	7.2E+4	1.5E+4	5.7E+4
5	1.3E+5	2.5E+4	1.0E+5
8	2.5E+5	4.0E+4	2.1E+5
10	2.2E+5	5.0E+4	1.7E+5
15	4.1E+5	7.5E+4	3.3E+5

D. Performance on DTLZ and DTLZ^{-1}

The HV results of the proposed MaOEDA-IR against its peer competitors (NSGA-III, MOEA/D, GrEA, HypE, MBN-EDA, and RM-MEDA) on DTLZ1-DTLZ4 and DTLZ1$^-$-DTLZ4$^-$ test problems with 3-, 5-, 8-, 10-, and 15-objective are presented in Table III. Furthermore, each compared algorithm is independently performed 30 runs and the best median HV results are highlighted in bold face. Moreover, the Mann-Whitney-Wilcoxon rank-sum test [69] with a 5% significance level is used to conduct the HV results due to the heuristic nature of peer algorithms, and the symbols "+," "=," and "-" denote whether the HV results of the proposed MaOEDA-IR are statistically better than, equal to, or worse than those of the corresponding peer competitors. In addition, the last row in Table III summarizes how many times the proposed MaOEDA-IR is better than, equal to, or worse than its respective peer competitor.

The results in Table III indicate that the proposed MaOEDA-IR obtains the best performances on the DTLZ4, DTLZ1$^-$, and DTLZ3$^-$ test problems with all considered objective numbers except for 5-objective DTLZ1$^-$ and 8-objective DTLZ3$^-$ which are worse than GrEA. Moreover, MaOEDA-IR is superior to others over DTLZ3 with 8- and 10-objective while is inferior to GrEA with 3-objective, RM-MEDA with 5-objective, and NSGA-III with 15-objective. Although the

---

[4]These settings apply only to the experimental results in Subsection IV-D. For other experiments, the terminated criteria for MaOEDA-IR and PCSEA are specified as 200 generations, and the population size for PCSEA is set to be 100 for $M \leq 10$ and 200 for $M > 10$.
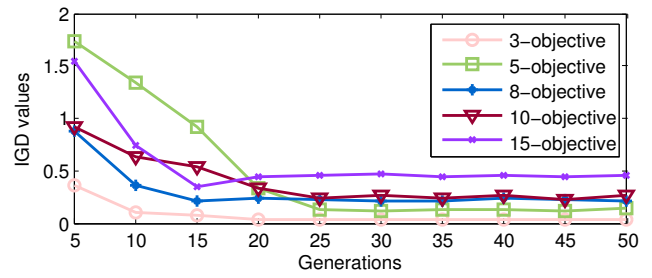

Fig. 5. IGD values of the results generated by 3-, 5-, 8-, 10-, and 15-objective DTLZ1 test problems with different neighbor sizes varying in $\{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$.

performance of MaOEDA-IR in DTLZ2 and DTLZ4$^-$ are worse than those of NSGA-III and GrEA on the 3-, and 5-objective, respectively, the proposed MaOEDA-IR performs better than others on the 10-objective. In addition, the proposed MaOEDA-IR outperforms peer competitors on DTLZ1 and DTLZ2$^-$ test problems with 8-, and 15-objective. In summary, the HV results of the proposed MaOEDA-IR against selected compared algorithms over eight test problems with 3-, 5-, 8-, 10-, and 15-objective indicate that MaOEDA-IR has a comparable performance by winning 181 test scores out of 240 comparisons, and performing equally well to 10 comparisons.

E. Investigation on Neighbor Size

To investigate how the neighbor size T affecting the performance of the proposed MaOEDA-IR, a series of experiments is performed by varying T in $[5, 50]$ with an interval of 5. Specifically, the results measured by IGD on DTLZ1 test problems with considered objective numbers are plotted in Fig. 5 in which it is clearly shown that appreciable changes have been taken place in T with smaller numbers and gradually remain steady as T increases. It is interpreted that the neighbor solutions with one particular reference vector for building the model are from other reference vectors when T is with a smaller number, which causes the inaccuracy of the built model based on which new solutions are generated that led to deteriorating IGD values. Especially, most IGD values remain level when $T > 25$ in Fig. 5 (it is actually applicable to other tested benchmark problems based on the investigations.), and the T with larger size will increase the computational cost by introducing more initialized solutions. As a consequence, T is specified as 25 in our experiments.

F. Investigation on Diversity Repairing Mechanism

As discussed in Subsection III-J, both the diversity and convergence have been improved with the diversity repairing mechanism. To this end, experimental comparisons on the test problems with and without the diversity repairing mechanism are performed. Specifically, the IGD values and the HV values of the evolution trajectory results generated by 10- and 15-objective DTLZ2 test problems over 200 generations are illustrated in Figs. 6a, 6b, 7a, and 7b, respectively. In these figures, the red lines denote the results without the diversity repairing mechanism, while the blue lines refer to

TABLE III
HV RESULTS OF MaOEDA-IR AGAINST NSGA-III, MOEA/D, GrEA, HypE, MBN-EDA, AND RM-MEDA OVER THE DTLZ1, DTLZ2, DTLZ3, DTLZ4, DTLZ1⁻, DTLZ2⁻, DTLZ3⁻, AND DTLZ4⁻ TEST PROBLEMS WITH 3-, 5-, 8-, 10-, AND 15-OBJECTIVE. EACH COMPARED ALGORITHM IS INDEPENDENTLY PERFORMED 30 RUNS, AND THE BEST MEDIAN HV RESULTS ARE HIGHLIGHTED IN BOLD FACE. THE SYMBOLS "+," "=," AND "-" DENOTE WHETHER THE HV RESULTS OF THE PROPOSED MaOEDA-IR ARE STATISTICALLY BETTER THAN, EQUAL TO, OR WORSE THAN THAT OF THE CORRESPONDING PEER COMPETITORS WITH A SIGNIFICANT LEVEL 5%, RESPECTIVELY.

Problem	M	MaOEDA-IR	NSGA-III	MOEA/D	GrEA	HypE	MBN-EDA	RM-MEDA
DTLZ1	3	0.838(2.1E-2)	0.915(5.6E-2)(-)	0.816(1.6E-3)(+)	0.925(8.2E-2)(-)	0.906(6.3E-2)(-)	**0.999(9.8E-4)(-)**	0.835(2.5E-2)(=)
	5	0.959(3.3E-3)	0.986(8.9E-3)(-)	0.936(2.6E-3)(+)	**0.989(8.0E-3)(-)**	0.973(3.0E-2)(-)	0.970(6.9E-3)(-)	0.980(9.5E-5)(-)
	8	**0.997(6.1E-4)**	0.996(3.1E-3)(=)	0.202(1.1E-1)(+)	0.988(5.0E-2)(+)	0.982(1.8E-2)(+)	0.984(3.5E-3)(+)	0.997(2.4E-3)(=)
	10	0.991(4.2E-4)	0.990(4.0E-3)(+)	0.047(6.7E-2)(+)	0.971(2.4E-2)(+)	0.937(5.1E-2)(+)	0.984(4.1E-3)(+)	**0.992(5.2E-3)(-)**
	15	**0.997(2.2E-4)**	0.994(6.1E-3)(+)	0.153(1.5E-1)(+)	0.968(3.0E-2)(+)	0.930(4.3E-2)(+)	0.992(2.3E-3)(+)	0.993(4.2E-3)(+)
DTLZ2	3	0.533(1.2E-3)	**0.645(4.7E-2)(-)**	0.540(2.1E-3)(-)	0.543(1.9E-3)(-)	0.361(2.9E-2)(+)	0.563(4.5E-2)(-)	0.566(1.6E-3)(-)
	5	0.780(5.5E-3)	**0.956(1.1E-2)(-)**	0.710(1.8E-3)(+)	0.795(1.2E-3)(-)	0.494(6.4E-2)(+)	0.739(4.0E-2)(+)	0.795(1.0E-2)(-)
	8	**0.925(6.7E-4)**	0.924(4.3E-3)(+)	0.925(8.7E-3)(=)	0.904(2.8E-3)(+)	0.543(8.6E-2)(+)	0.842(3.9E-2)(+)	0.020(1.1E-2)(+)
	10	**0.932(9.3E-3)**	0.931(1.0E-2)(+)	0.002(4.1E-3)(+)	0.922(3.7E-3)(+)	0.443(7.2E-2)(+)	0.879(3.1E-2)(+)	0.880(1.8E-2)(+)
	15	0.979(4.2E-3)	**0.982(4.4E-4)(-)**	0.066(1.1E-1)(+)	0.888(1.8E-2)(+)	0.404(7.2E-2)(+)	0.922(1.9E-2)(+)	0.876(1.6E-2)(+)
DTLZ3	3	0.513(2.7E-3)	0.986(3.0E-2)(-)	0.574(7.3E-2)(-)	**0.979(2.2E-4)(-)**	0.816(1.7E-1)(-)	0.680(4.6E-2)(-)	0.977(7.1E-3)(-)
	5	0.989(3.6E-4)	0.987(1.8E-2)(+)	0.704(2.8E-2)(+)	0.794(1.7E-3)(+)	0.995(3.7E-3)(-)	0.862(3.1E-2)(+)	**0.996(1.4E-3)(-)**
	8	**0.997(2.6E-5)**	0.994(6.5E-3)(+)	0.313(1.9E-1)(+)	0.923(5.6E-4)(+)	0.995(6.2E-3)(+)	0.921(1.5E-2)(+)	0.949(8.9E-3)(+)
	10	**0.989(2.0E-5)**	0.984(1.4E-2)(+)	0.096(1.4E-1)(+)	0.943(8.2E-4)(+)	0.988(1.0E-2)(+)	0.929(1.4E-2)(+)	0.898(1.9E-2)(+)
	15	0.975(4.9E-4)	**0.997(3.0E-3)(-)**	0.079(9.3E-2)(+)	0.902(2.2E-2)(+)	0.985(1.2E-2)(-)	0.963(4.1E-3)(+)	0.887(1.6E-2)(+)
DTLZ4	3	**0.733(2.9E-4)**	0.521(9.2E-2)(+)	0.445(1.0E-1)(+)	0.476(1.7E-1)(+)	0.471(1.0E-1)(+)	0.565(2.1E-3)(+)	0.538(3.2E-3)(+)
	5	**0.920(6.9E-3)**	0.798(9.0E-3)(+)	0.614(5.9E-2)(+)	0.785(3.0E-2)(+)	0.585(6.6E-2)(+)	0.795(1.1E-3)(+)	0.726(2.8E-2)(+)
	8	**0.938(5.2E-3)**	0.923(2.3E-3)(+)	0.065(5.7E-2)(+)	0.916(2.1E-3)(+)	0.521(9.2E-2)(+)	0.928(9.5E-4)(+)	0.859(1.4E-2)(+)
	10	**0.953(6.2E-5)**	0.943(2.3E-3)(+)	0.011(2.1E-2)(+)	0.936(1.3E-3)(+)	0.449(9.4E-2)(+)	0.949(9.7E-4)(+)	0.897(2.1E-2)(+)
	15	**0.997(3.2E-5)**	0.991(5.5E-4)(+)	0.009(1.9E-2)(+)	0.936(9.6E-3)(+)	0.546(9.1E-2)(+)	0.991(2.3E-4)(+)	0.985(9.4E-3)(+)
DTLZ1⁻	3	**0.289(3.1E-3)**	0.217(3.2E-3)(+)	0.208(3.2E-3)(+)	0.227(1.3E-3)(+)	0.121(1.0E-2)(+)	0.118(1.6E-2)(+)	0.255(1.6E-2)(+)
	5	0.010(2.7E-3)	0.019(3.8E-3)(-)	0.007(4.1E-4)(+)	**0.199(9.0E-3)(-)**	0.001(1.0E-4)(+)	0.019(3.7E-3)(-)	0.011(1.0E-3)(=)
	8	**0.112(2.9E-2)**	0.001(5.2E-4)(+)	0.000(1.2E-5)(+)	0.000(1.4E-5)(+)	0.000(3.1E-6)(+)	0.001(0.0E+0)(+)	0.000(2.6E-5)(+)
	10	**0.099(3.6E-4)**	0.000(1.7E-4)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(6.6E-6)(+)
	15	**0.102(6.6E-3)**	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)
DTLZ2⁻	3	0.533(7.2E-2)	0.541(1.0E-2)(-)	0.527(1.8E-3)(+)	**0.587(2.3E-2)(-)**	0.465(2.5E-2)(+)	0.340(2.3E-2)(+)	0.531(6.1E-3)(+)
	5	0.067(3.3E-5)	0.140(2.0E-2)(-)	0.078(2.0E-3)(-)	**0.258(3.1E-2)(-)**	0.008(1.3E-3)(+)	0.144(2.2E-2)(-)	0.067(9.7E-3)(=)
	8	**0.102(5.2E-3)**	0.012(4.5E-3)(+)	0.000(4.5E-5)(+	0.001(8.4E-5)(+)	0.000(1.6E-5)(+)	0.017(3.4E-3)(+)	0.001(2.1E-4)(+)
	10	**0.033(3.6E-2)**	0.004(2.4E-3)(+)	0.000(0.0E+0)(+)	0.000(1.5E-5)(+)	0.000(0.0E+0)(+)	0.002(5.7E-5)(+)	0.000(2.4E-5)(+)
	15	**0.003(3.2E-3)**	0.000(4.2E-5)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)
DTLZ3⁻	3	**0.553(1.9E-2)**	0.539(1.1E-2)(+)	0.535(1.7E-2)(+)	0.540(1.7E-3)(+)	0.483(1.5E-2)(+)	0.242(2.9E-2)(+)	0.547(2.4E-2)(+)
	5	**0.230(3.3E-2)**	0.131(1.9E-2)(+)	0.119(1.1E-2)(+)	0.071(9.0E-4)(+)	0.074(3.6E-3)(+)	0.105(1.2E-2)(+)	0.087(8.6E-3)(+)
	8	0.001(3.2E-3)	0.012(3.0E-3)(-)	0.001(1.8E-4)(=)	**0.102(1.2E-2)(-)**	0.001(2.3E-4)(=)	0.014(4.0E-3)(-)	0.001(2.2E-4)(=)
	10	**0.030(3.0E-3)**	0.003(2.0E-3)(+)	0.000(6.9E-6)(+)	0.000(1.4E-5)(+)	0.000(2.4E-5)(+)	0.003(1.1E-4)(+)	0.000(2.8E-5)(+)
	15	**0.003(2.1E-5)**	0.000(1.6E-4)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)
DTLZ4⁻	3	0.531(2.8E-3)	0.533(6.4E-3)(-)	0.528(2.0E-3)(+)	**0.621(3.2E-2)(-)**	0.492(1.4E-2)(+)	0.170(1.2E-2)(+)	0.511(1.2E-2)(+)
	5	0.060(3.9E-2)	0.076(1.5E-2)(-)	0.071(2.1E-3)(-)	**0.329(3.4E-2)(-)**	0.007(1.4E-3)(+)	0.005(1.1E-3)(+)	0.014(2.7E-3)(+)
	8	0.001(6.3E-3)	0.002(1.0E-3)(-)	0.000(2.4E-5)(+)	**0.124(1.1E-2)(-)**	0.002(7.9E-6)(-)	0.000(0.0E+0)(+)	0.001(1.5E-5)(=)
	10	**0.033(3.2E-3)**	0.000(2.9E-4)(+)	0.000(0.0E+0)(+)	0.000(3.1E-6)(+)	0.000(2.2E-6)(+)	0.000(0.0E+0)(+)	0.001(2.2E-6)(+)
	15	**0.005(6.2E-2)**	0.003(3.8E-5)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)	0.000(0.0E+0)(+)
+/=/-			25/1/14	34/2/4	28/0/12	33/1/6	33/0/7	28/6/6

those with the diversity repairing. To be specific, both the IGD values of 10-objective DTLZ2 with and without the diversity repairing mechanism sharply decrease during the first 20 generations then gradually remain stable, while those of 15-objective DTLZ2 smoothly decline throughout the entire evolution. For both the HV values of 10-objective DTLZ2 with and without the diversity repairing mechanism, they grow substantially before the 40-th generation then go up moderately as the evolution continues, while the ones resulted by the proposed algorithm without the diversity repairing mechanism stay lower than those with the diversity repairing mechanism during the entire evolution. In summary, both the best IGD results in Figs. 6a and 6b and the HV results in Figs. 7a and 7b demonstrate the promising performance of the proposed MaOEDA-IR when the diversity repairing mechanism is employed.

Furthermore, Table IV shows the extensive experimental comparisons between the proposed algorithm with and without the diversity repairing mechanism. Specifically, these experiments are independently performed 30 runs over each test problem, and their results are measured by HV. Moreover, the best median HV results are highlighted in bold face, and the symbols "+," "=," and "-" denote whether the HV results of the proposed algorithm with the diversity repairing mechanism are statistically better than, equal to, or worse than those of the proposed algorithm without the diversity repairing mechanism with a significant level 5%, respectively. In addition, the last row in Table 3 summarizes how many times the proposed algorithm with the diversity repairing mechanism are better than, equal to, or worse than itself without this mechanism. It is clearly shown in Table IV that when the diversity repairing mechanism is employed, the proposed algorithm obtains all the best median HV results and most best statistical results against its competitors over DTLZ1-DTLZ4, and DTLZ1⁻-
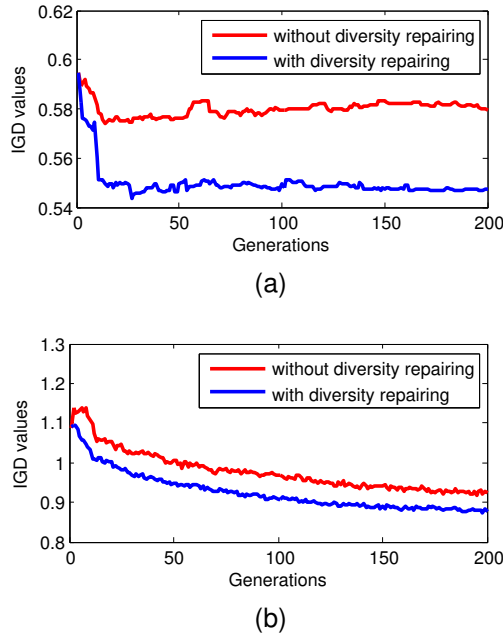
Fig. 6. IGD values of the results generated by 10-(Fig. 6a) and 15-objective (Fig. 6b) DTLZ2 with and without diversity repairing over 200 generations.
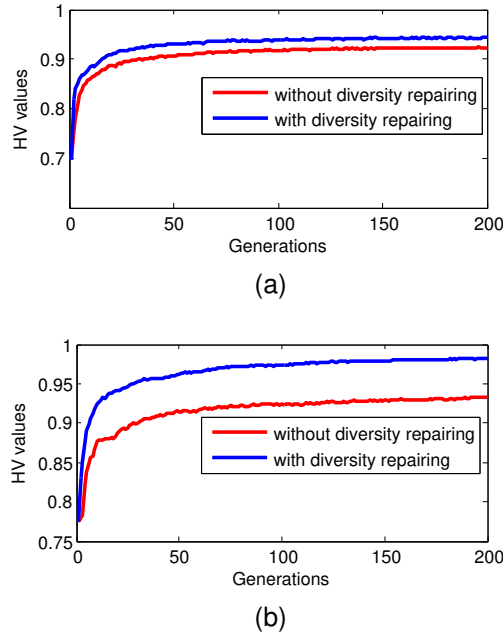

Fig. 7. HV values of the results generated by 10-(Fig. 7a) and 15-objective (Fig. 7b) DTLZ2 with and without diversity repairing over 200 generations.

TABLE IV
HV RESULTS OF MaOEDA-IR WITH AND WITHOUT THE DIVERSITY REPAIRING MECHANISM OVER THE DTLZ1, DTLZ2, DTLZ3, DTLZ4, DTLZ1$^-$, DTLZ2$^-$, DTLZ3$^-$, AND DTLZ4$^-$ TEST PROBLEMS WITH 3-, 5-, 8-, 10-, AND 15-OBJECTIVE. EACH COMPARED ALGORITHM IS INDEPENDENTLY PERFORMED 30 RUNS, AND THE BEST MEDIAN HV RESULTS ARE HIGHLIGHTED IN BOLD FACE. THE SYMBOLS "+," "=," AND "-" DENOTE WHETHER THE HV RESULTS OF THE MaOEDA-IR WITH DR MECHANISM ARE STATISTICALLY BETTER THAN, EQUAL TO, OR WORSE THAN THAT OF THE CORRESPONDING MaOEDA-IR WITHOUT THE DIVERSITY REPAIRING MECHANISM WITH A SIGNIFICANT LEVEL 5%, RESPECTIVELY.

Problem	M	Diversity Repairing	
		With	Without
DTLZ1	3	**0.847(1.2E-3)**	0.843(1.2E-3)(=)
	5	**0.975(4.5E-4)**	0.933(6.6E-2)(+)
	8	**0.997(1.7E-4)**	0.844(5.6E-2)(+)
	10	**0.997(1.5E-4)**	0.590(7.5E-2)(+)
	15	**0.999(1.2E-4)**	0.727(9.7E-3)(+)
DTLZ2	3	**0.567(2.8E-3)**	0.503(4.3E-2)(+)
	5	**0.796(4.0E-3)**	0.709(5.0E-2)(+)
	8	**0.923(9.6E-4)**	0.811(9.2E-3)(+)
	10	**0.943(7.2E-4)**	0.924(1.5E-2)(+)
	15	**0.987(1.0E-3)**	0.933(1.2E-3)(+)
DTLZ3	3	**0.566(1.3E-3)**	0.558(1.0E-3)(=)
	5	**0.999(4.1E-6)**	0.777(3.1E-2)(+)
	8	**0.997(5.5E-6)**	0.785(2.0E-2)(+)
	10	**0.998(2.8E-3)**	0.554(6.3E-2)(+)
	15	**0.982(6.1E-4)**	0.652(1.7E-4)(+)
DTLZ4	3	**0.750(8.3E-2)**	0.745(3.9E-3)(=)
	5	**0.937(1.7E-2)**	0.925(3.4E-2)(+)
	8	**0.991(2.9E-3)**	0.881(5.5E-2)(+)
	10	**0.992(3.8E-3)**	0.813(4.9E-2)(+)
	15	**0.999(6.1E-4)**	0.900(3.7E-5)(+)
DTLZ1$^-$	3	**0.338(3.9E-3)**	0.285(3.1E-2)(+)
	5	**0.011(3.3E-4)**	0.002(5.3E-2)(+)
	8	**0.142(5.1E-3)**	0.129(3.0E-3)(+)
	10	**0.124(7.0E-3)**	0.042(5.1E-2)(+)
	15	**0.102(6.7E-3)**	0.010(2.8E-3)(+)
DTLZ2$^-$	3	**0.541(1.4E-3)**	0.443(9.0E-2)(+)
	5	**0.071(9.6E-4)**	0.063(3.0E-2)(+)
	8	**0.103(1.5E-2)**	0.014(9.6E-2)(+)
	10	**0.038(7.7E-3)**	0.006(2.9E-2)(+)
	15	**0.003(8.9E-4)**	0.000(0.0E-0)(=)
DTLZ3$^-$	3	**0.595(2.1E-2)**	0.518(9.1E-2)(+)
	5	**0.255(3.0E-2)**	0.240(6.1E-2)(+)
	8	**0.001(6.5E-5)**	0.000(0.0E-0)(=)
	10	**0.036(7.2E-3)**	0.013(1.1E-4)(+)
	15	**0.003(9.0E-4)**	0.000(0.0E-0)(=)
DTLZ4$^-$	3	**0.540(1.2E-3)**	0.472(9.7E-4)(+)
	5	**0.069(7.7E-4)**	0.031(9.7E-2)(+)
	8	**0.001(5.9E-5)**	0.000(0.0E-0)(=)
	10	**0.043(8.0E-3)**	0.010(5.5E-2)(+)
	15	**0.005(1.5E-3)**	0.000(0.0E-0)(=)
+/=/-			33/7/0

DTLZ4$^-$ with 3-, 5-, 8-, 10-, and 15-objective, while the proposed algorithm without the diversity repairing mechanism could not even obtain effective solutions over DTLZ2$^-$ with 15-objective, DTLZ3$^-$ with 8- and 15-objective, and DTLZ4$^-$ with 8-objective (i.e., the HV results upon these generated solutions are approximately zeros, which is caused by the domination by the employed reference points for the calculation of HV). In addition, it also can be observed that the diversity repairing mechanism may not significantly improve the performance in solving MOPs where the phenomenon of diversity losing is not severe. For example, the proposed algorithm obtains the same statistical results over 3-objective DTLZ1, DTLZ3, and DTLZ4 test problems no matter if the diversity repairing mechanism is employed. In summary, the diversity repairing mechanism can significantly improve the performance of the proposed algorithm especially in solving MaOPs.

G. Investigation on Dimension Reduction

It is expected that dimension reduction in the decision variable space is capable of reducing the computational complexity of the proposed MaOEDA-IR. In this situation, two

types of experiments are to be performed in order to draw any meaningful conclusion. The first one is to measure the performance of the solutions generated by the proposed algorithm with and without the dimension reduction within the same generation numbers. The other one is to compare the generation numbers when the same performance is achieved by the proposed algorithm with and without the dimension reduction. In the following, the first experimental results would be shown, while the second experimental comparisons are presented in Supplemental Materials. Specifically, the generation number of the first experiment is adopted from the parameter settings in Subsection IV-C (i.e., 200). The experimental comparisons are independently performed 30 runs by the proposed algorithm with and without the dimension reduction over DTLZ1-DTLZ4 and DTLZ1$^-$-DTLZ4$^-$ with 3-, 5-, 8-, 10-, and 15-objective. Then their results are measured by HV and shown in Table V where the best median HV results are highlighted in bold face, and the symbols "+," "=," and "-" denote whether the HV results of the proposed algorithm with the dimension reduction are statistically better than, equal to, or worse than that of the proposed algorithm without the dimension reduction with a significant level 5%, respectively. Furthermore, the last row in Table V summarizes how many times the proposed algorithm with the dimension reduction are better than, equal to, or worse than itself without this technique. It is obvious from Table V that the proposed algorithm obtains the significant performance improvement when the dimension reduction is employed. Furthermore, without the dimension reduction, the proposed algorithm cannot perform well over several test problems, such as the DTLZ1$^-$ with 5-objective, DTLZ2$^-$ with 10- and 5-objective, and DTLZ3$^-$ as well as DTLZ4$^-$ with 8-, 10-, and 15-objective (their HV results are zeros). In summary, the proposed algorithm shows its superiority when the dimension reduction is utilized.

V. CONCLUSION

In solving many-objective optimization problems, the performance of most multi-objective evolutionary algorithms often deteriorate appreciably because of the loss of selection pressure during the evolution process. This is largely due to the selected parents solutions not generating promising individuals with the conventional genetic operators to direct the search towards the Pareto-optimal front. An improved regularity-based estimation of distribution algorithm, which generates new solutions with a probabilistic model built from the solutions the algorithm has visited, is proposed in this paper. To be specific, the proposed algorithm made an innovation in the following aspects: 1) devising a diversity repairing mechanism to reduce the risk of dominance resistant solutions and 2) generating promising solutions with the statistics of regularity learnt from the neighboring solutions with respect to the representatives which are uniformly distributed in the objective space. These two steps works in conjunction with each other to direct the search towards Pareto-optimal front. In addition, dimension reduction technique is utilized to reduce the cost of exploitation and exploration. Furthermore, in addition to the investigations are performed on the diversity repairing and

TABLE V
HV RESULTS OF MAOEDA-IR WITH AND WITHOUT THE DIMENSION REDUCTION OVER THE DTLZ1, DTLZ2, DTLZ3, DTLZ4, DTLZ1$^-$, DTLZ2$^-$, DTLZ3$^-$, AND DTLZ4$^-$ TEST PROBLEMS WITH 3-, 5-, 8-, 10-, AND 15-OBJECTIVE. EACH COMPARED ALGORITHM IS INDEPENDENTLY PERFORMED 30 RUNS, AND THE BEST MEDIAN HV RESULTS ARE HIGHLIGHTED IN BOLD FACE. THE SYMBOLS "+," "=," AND "-" DENOTE WHETHER THE HV RESULTS OF THE MAOEDA-IR WITH THE DIMENSION REDUCTION ARE STATISTICALLY BETTER THAN, EQUAL TO, OR WORSE THAN THAT OF THE CORRESPONDING MAOEDA-IR WITHOUT THE DIMENSION REDUCTION WITH A SIGNIFICANT LEVEL 5%, RESPECTIVELY.

Problem	M	Dimension Reduction	
		With	Without
DTLZ1	3	**0.847(1.2E-3)**	0.776(3.2E-3)(+)
	5	**0.975(4.5E-4)**	0.969(3.4E-4)(+)
	8	**0.997(1.7E-4)**	0.902(4.4E-3)(+)
	10	**0.997(1.5E-4)**	0.948(6.6E-2)(+)
	15	**0.999(1.2E-4)**	0.954(1.6E-2)(+)
DTLZ2	3	**0.567(2.8E-3)**	0.555(2.6E-2)(+)
	5	**0.796(4.0E-3)**	0.746(5.1E-2)(+)
	8	**0.923(9.6E-4)**	0.907(7.0E-3)(+)
	10	**0.943(7.2E-4)**	0.909(8.9E-2)(+)
	15	**0.987(1.0E-3)**	0.828(9.6E-3)(+)
DTLZ3	3	**0.566(1.3E-3)**	0.511(2.4E-5)(+)
	5	**0.999(4.1E-6)**	0.985(9.3E-3)(+)
	8	**0.997(5.5E-6)**	0.912(3.5E-4)(+)
	10	**0.998(2.8E-3)**	0.328(2.0E-2)(+)
	15	**0.982(6.1E-4)**	0.928(2.5E-4)(+)
DTLZ4	3	**0.750(8.3E-2)**	0.688(2.9E-4)(+)
	5	**0.937(1.7E-2)**	0.890(7.6E-2)(+)
	8	**0.991(2.9E-3)**	0.956(7.5E-3)(+)
	10	**0.992(3.8E-3)**	0.109(3.8E-4)(+)
	15	**0.999(6.1E-4)**	0.940(5.7E-4)(+)
DTLZ1$^-$	3	**0.338(3.9E-3)**	0.330(4.7E-2)(+)
	5	**0.011(3.3E-4)**	0.000(0.0E-0)(=)
	8	**0.142(5.1E-3)**	0.089(3.4E-2)(+)
	10	**0.124(7.0E-3)**	0.046(1.6E-4)(+)
	15	**0.102(6.7E-3)**	0.009(7.9E-3)(+)
DTLZ2$^-$	3	**0.541(1.4E-3)**	0.510(7.5E-4)(+)
	5	**0.071(9.6E-4)**	0.018(4.5E-2)(+)
	8	**0.103(1.5E-2)**	0.086(8.4E-3)(+)
	10	**0.038(7.7E-3)**	0.000(0.0E-0)(=)
	15	**0.003(8.9E-4)**	0.000(0.0E-0)(=)
DTLZ3$^-$	3	**0.595(2.1E-2)**	0.580(9.6E-3)(+)
	5	**0.255(3.0E-2)**	0.172(4.6E-5)(+)
	8	**0.001(6.5E-5)**	0.000(0.0E-0)(=)
	10	**0.036(7.2E-3)**	0.000(0.0E-0)(=)
	15	**0.003(9.0E-4)**	0.000(0.0E-0)(=)
DTLZ4$^-$	3	**0.540(1.2E-3)**	0.532(2.6E-3)(+)
	5	**0.069(7.7E-4)**	0.029(1.5E-3)(+)
	8	**0.001(5.9E-5)**	0.000(0.0E-0)(=)
	10	**0.043(8.0E-3)**	0.000(0.0E-0)(=)
	15	**0.005(1.5E-3)**	0.000(0.0E-0)(=)
+/=/-			31/9/0

dimension reduction, investigation is also performed based on the size of neighbors affecting the performance of the proposed algorithm to give the guideline for decision-marker. Extensive experiments are performed and the results measured by the chosen performance metrics indicate that the proposed algorithm shows superiority in tackling many-objective optimization problems. In our future research, we will extend the proposed algorithm to deal with highly constrained many-objective optimization problems in which complicated regularity of Pareto fronts often exists.

REFERENCES

[1] M. Farina and P. Amato, "On the optimal solution definition for many-criteria optimization problems," in *Proceedings of the NAFIPS-FLINT international conference*, 2002, pp. 233–238.

[2] O. Chikumbo, E. Goodman, and K. Deb, "Approximating a multi-dimensional pareto front for a land use management problem: A modified moea with an epigenetic silencing metaphor," in *Evolutionary Computation (CEC), 2012 IEEE Congress on*. IEEE, 2012, pp. 1–9.

[3] R. J. Lygoe, M. Cary, and P. J. Fleming, "A real-world application of a many-objective optimisation complexity reduction process," in *Evolutionary Multi-Criterion Optimization*. Springer, 2013, pp. 641–655.

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[5] E. Zitzler, L. Marco, and T. Lothar, "Spea2: Improving the strength pareto evolutionary algorithm," 2001.

[6] J. Knowles and D. Corne, "Quantifying the effects of objective space dimension in evolutionary multiobjective optimization," in *Evolutionary multi-criterion optimization*. Springer, 2007, pp. 757–771.

[7] C. M. Fonseca and P. J. Fleming, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms. i. a unified formulation," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 28, no. 1, pp. 26–37, 1998.

[8] R. C. Purshouse and P. J. Fleming, "On the evolutionary optimization of many conflicting objectives," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 770–784, 2007.

[9] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, p. 13, 2015.

[10] T. Wagner, N. Beume, and B. Naujoks, "Pareto-, aggregation-, and indicator-based methods in many-objective optimization," in *Evolutionary multi-criterion optimization*. Springer, 2007, pp. 742–756.

[11] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[12] J. Bader and E. Zitzler, "Hype: An algorithm for fast hypervolume-based many-objective optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, 2011.

[13] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, 2013.

[14] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[15] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 3, pp. 440–462, 2017.

[16] Y. Sun, G. G. Yen, and Z. Yi, "Global view-based selection mechanism for many-objective evolutionary algorithms," in *2017 IEEE Congress on Evolutionary Computation*. IEEE, 2017, pp. 427–434.

[17] ——, "Reference line-based estimation of distribution algorithm for many-objective optimization," *Knowledge-Based Systems*, vol. 132, pp. 129–143, 2017.

[18] K. Deb, A. Sinha, and S. Kukkonen, "Multi-objective test problems, linkages, and evolutionary methodologies," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006, pp. 1141–1148.

[19] M. Pelikan, K. Sastry, and D. E. Goldberg, "Multiobjective estimation of distribution algorithms," in *Scalable optimization via probabilistic modeling*. Springer, 2006, pp. 223–248.

[20] Q. Zhang, A. Zhou, and Y. Jin, "Rm-meda: A regularity model-based multiobjective estimation of distribution algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 41–63, 2008.

[21] H. Karshenas, R. Santana, C. Bielza, and P. Larranaga, "Multiobjective estimation of distribution algorithm based on joint modeling of objectives and variables," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 519–542, 2014.

[22] N. Khan, D. E. Goldberg, and M. Pelikan, "Multi-objective bayesian optimization algorithm," *Urbana*, vol. 51, p. 61801, 2002.

[23] J. Schwarz and J. Ocenasek, "Multiobjective bayesian optimization algorithm for combinatorial problems: Theory and practice," *Neural Network World*, vol. 11, no. 5, pp. 423–442, 2001.

[24] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE Transactions on evolutionary computation*, vol. 3, no. 4, pp. 257–271, 1999.

[25] M. Laumanns and J. Ocenasek, "Bayesian optimization algorithms for multi-objective optimization," in *Parallel Problem Solving from NaturePPSN VII*. Springer, 2002, pp. 298–307.

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

[27] X. Zhong and W. Li, "A decision-tree-based multi-objective estimation of distribution algorithm," in *Computational Intelligence and Security, 2007 International Conference on*. IEEE, 2007, pp. 114–11.

[28] P. A. Bosman and D. Thierens, "Multi-objective optimization with diversity preserving mixture-based iterated density estimation evolutionary algorithms," *International Journal of Approximate Reasoning*, vol. 31, no. 3, pp. 259–289, 2002.

[29] M. Costa and E. Minisci, "Moped: a multi-objective parzen-based estimation of distribution algorithm for continuous problems," in *Evolutionary Multi-Criterion Optimization*. Springer, 2003, pp. 282–294.

[30] M. Pelikan, K. Sastry, and D. E. Goldberg, "Multiobjective hboa, clustering, and scalability," in *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. ACM, 2005, pp. 663–670.

[31] K. Sastry, D. E. Goldberg, and M. Pelikan, "Limits of scalability of multiobjective estimation of distribution algorithms," in *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, vol. 3. IEEE, 2005, pp. 2217–2224.

[32] A. Zhou, Q. Zhang, and Y. Jin, "Approximating the set of pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1167–1189, 2009.

[33] V. Cherkassky and F. M. Mulier, *Learning from data: concepts, theory, and methods*. John Wiley & Sons, 2007.

[34] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, "The elements of statistical learning: data mining, inference and prediction," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005.

[35] Z. He and G. G. Yen, "Visualization and performance metric in many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 386–402, 2016.

[36] J. Cheng, G. G. Yen, and G. Zhang, "A many-objective evolutionary algorithm with enhanced mating and environmental selections," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 4, pp. 592–605, 2015.

[37] A. Zhou, Q. Zhang, Y. Jin, E. Tsang, and T. Okabe, "A model-based evolutionary algorithm for bi-objective optimization," in *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, vol. 3. IEEE, 2005, pp. 2568–2575.

[38] A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, and E. Tsang, "Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion," in *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*. IEEE, 2006, pp. 892–899.

[39] T. Gal and T. Hanne, "Consequences of dropping nonessential objectives for the application of mcdm methods," *European Journal of Operational Research*, vol. 119, no. 2, pp. 373–378, 1999.

[40] D. Brockhoff and E. Zitzler, "Automated aggregation and omission of objectives for tackling many-objective problems," in *New Developments in Multiple Objective and Goal Programming*. Springer, 2010, pp. 81–102.

[41] ——, "Objective reduction in evolutionary multiobjective optimization: Theory and applications," *Evolutionary Computation*, vol. 17, no. 2, pp. 135–166, 2009.

[42] A. López Jaimes, C. A. Coello Coello, and D. Chakraborty, "Objective reduction using a feature selection technique," in *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*. ACM, 2008, pp. 673–680.

[43] D. K. Saxena and K. Deb, "Non-linear dimensionality reduction procedures for certain large-dimensional multi-objective optimization problems: employing correntropy and a novel maximum variance unfolding," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2007, pp. 772–787.

[44] D. Brockhoff, D. K. Saxena, K. Deb, and E. Zitzler, "On handling a large number of objectives a posteriori and during optimization," in *Multiobjective Problem Solving from Nature*. Springer, 2008, pp. 377–403.

[45] D. K. Saxena, J. A. Duro, A. Tiwari, K. Deb, and Q. Zhang, "Objective reduction in many-objective optimization: Linear and nonlinear algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 1, pp. 77–99, 2013.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2018.2794319, IEEE Transactions on Evolutionary Computation

17

[46] X. Guo, X. Wang, M. Wang, and Y. Wang, "A new objective reduction algorithm for many-objective problems: employing mutual information and clustering algorithm," in *2012 Eighth International Conference on Computational Intelligence and Security*. IEEE, 2012, pp. 11–16.

[47] H. Wang and X. Yao, "Objective reduction based on nonlinear correlation information entropy," *Soft Computing*, vol. 20, no. 6, pp. 2393–2407, 2016.

[48] H. K. Singh, A. Isaacs, and T. Ray, "A pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 4, pp. 539–556, 2011.

[49] N. Kambhatla and T. K. Leen, "Dimension reduction by local principal component analysis," *Neural Computation*, vol. 9, no. 7, pp. 1493–1516, 1997.

[50] H. Wang, Q. Zhang, L. Jiao, and X. Yao, "Regularity model for noisy multiobjective optimization," *IEEE Transactions on Cybernetics*, vol. 46, no. 9, pp. 1997–2009, 2016.

[51] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, *Scalable test problems for evolutionary multiobjective optimization*. Springer, 2005.

[52] P. Fleming, "Computer aided control systems using a multiobjective optimization approach," in *Proc. IEEE Control85 Conference*, 1985, pp. 174–179.

[53] Y. Jin, T. Okabe, and B. Sendho, "Adapting weighted aggregation for multiobjective evolution strategies," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2001, pp. 96–110.

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

[55] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 180–198, 2015.

[56] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, 2016.

[57] K. Deb, K. Miettinen, and D. Sharma, "A hybrid integrated multi-objective optimization procedure for estimating nadir point," in *Evolutionary Multi-Criterion Optimization*. Springer, 2009, pp. 569–583.

[58] K. Deb, S. Chaudhuri, and K. Miettinen, "Towards estimating nadir objective vector using evolutionary approaches," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006, pp. 643–650.

[59] H. Wang, S. He, and X. Yao, "Nadir point estimation for many-objective optimization problems based on emphasized critical regions," *Soft Computing*, pp. 1–13, 2015.

[60] Y. Yuan, H. Xu, and B. Wang, "Evolutionary many-objective optimization using ensemble fitness ranking," in *Proceedings of the 2014 conference on Genetic and evolutionary computation*. ACM, 2014, pp. 669–676.

[61] Q. Zhang, A. Zhou, Y. Jin *et al.*, "Modelling the regularity in estimation of distribution algorithm for continuous multi-objective evolutionary optimization with variable linkages," *IEEE Transactions on Evolutionary Computation*, vol. 5, pp. 1–40, 2006.

[62] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, 2000.

[63] H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of decomposition-based many-objective algorithms strongly depends on pareto front shapes," *IEEE Transactions on Evolutionary Computation*, 2016.

[64] P. A. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 174–188, 2003.

[65] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 16–37, 2016.

[66] L. While, L. Bradstreet, and L. Barone, "A fast way of calculating exact hypervolumes," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 1, pp. 86–95, 2012.

[67] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex systems*, vol. 9, no. 2, pp. 115–148, 1995.

[68] K. Deb, *Multi-objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, 2001, vol. 16.

[69] S. Robert, J. Torrie, and D. Dickey, "Principles and procedures of statistics: a biometrical approach," 1997.

Yanan Sun (S'15-M'18) received a Ph.D. degree in engineering from the Sichuan University, Chengdu, China, in 2017. From 2015.08-2017.02, he is a jointly Ph.D. student financed by the China Scholarship Council in the School of Electrical and Computer Engineering, Oklahoma State University (OSU), USA. He is currently a Postdoc Research Fellow in the School of Engineering and Computer Science, Victoria University of Wellington, Wellington, New Zealand. His research topics are many-objective optimization and deep learning.

Gary G. Yen (S'87-M'88-SM'97-F'09) received a Ph.D. degree in electrical and computer engineering from the University of Notre Dame in 1992. Currently he is a Regents Professor in the School of Electrical and Computer Engineering, Oklahoma State University (OSU). Before joined OSU in 1997, he was with the Structure Control Division, U.S. Air Force Research Laboratory in Albuquerque. His research interest includes intelligent control, computational intelligence, conditional health monitoring, signal processing and their industrial/defense applications.

Dr. Yen was an associate editor of the *IEEE Control Systems Magazine, IEEE Transactions on Control Systems Technology, Automatica, Mechantronics, IEEE Transactions on Systems, Man and Cybernetics, Parts A and B* and I*EEE Transactions on Neural Networks*. He is currently serving as an associate editor for the *IEEE Transactions on Evolutionary Computation* and the *IEEE Transactions on Cybernetics*. He served as the General Chair for the *2003 IEEE International Symposium on Intelligent Control* held in Houston, TX and *2006 IEEE World Congress on Computational Intelligence* held in Vancouver, Canada. Dr. Yen served as Vice President for the Technical Activities in 2005-2006 and then President in 2010-2011 of the IEEE Computational intelligence Society. He was the founding editor-in-chief of the *IEEE Computational Intelligence Magazine*, 2006-2009. In 2011, he received Andrew P Sage Best Transactions Paper award from *IEEE Systems, Man and Cybernetics Society* and in 2014, he received Meritorious Service award from *IEEE Computational Intelligence Society*.

Zhang Yi (F'16) received a Ph.D. degree in mathematics from the Institute of Mathematics, The Chinese Academy of Science, Beijing, China, in 1994. Currently, he is a Professor at the Machine Intelligence Laboratory, College of Computer Science, Sichuan University, Chengdu, China. He is the co-author of three books: *Convergence Analysis of Recurrent Neural Networks* (Kluwer Academic Publishers, 2004), *Neural Networks: Computational Models and Applications* (Springer, 2007), and *Subspace Learning of Neural Networks* (CRC Press, 2010). He was an Associate Editor of *IEEE Transactions on Neural Networks and Learning Systems* (2009 2012), and He is an Associate Editor of *IEEE Transactions on Cybernetics* (2014). His current research interests include Neural Networks and Big Data. He is a fellow of IEEE.