



# An easy-to-use real-world multi-objective optimization problem suite

Ryoji Tanabe, Hisao Ishibuchi \*

Shenzhen Key Laboratory of Computational Intelligence, University Key Laboratory of Evolving Intelligent Systems of Guangdong Province,  
Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China

## ARTICLE INFO

### Article history:

Received 29 March 2019  
Received in revised form 26 December 2019  
Accepted 4 January 2020  
Available online 16 January 2020

### Keywords:

Evolutionary multi-objective optimization  
Test problems  
Real-world problems

## ABSTRACT

Although synthetic test problems are widely used for the performance assessment of evolutionary multi-objective optimization algorithms, they are likely to include unrealistic properties which may lead to overestimation/underestimation. To address this issue, we present a multi-objective optimization problem suite consisting of 16 bound-constrained real-world problems. The problem suite includes various problems in terms of the number of objectives, the shape of the Pareto front, and the type of design variables. 4 out of the 16 problems are multi-objective mixed-integer optimization problems. We provide Java, C, and Matlab source codes of the 16 problems so that they are available in an off-the-shelf manner. We examine an approximated Pareto front of each test problem. We also analyze the performance of six representative evolutionary multi-objective optimization algorithms on the 16 problems. In addition to the 16 problems, we present 8 constrained multi-objective real-world problems.

© 2020 Elsevier B.V. All rights reserved.

## 1. Introduction

A bound-constrained multi-objective optimization problem (MOP) is to find a solution  $\mathbf{x} \in \mathbb{S} \subseteq \mathbb{R}^D$  that minimizes an objective function vector  $\mathbf{f} : \mathbb{S} \rightarrow \mathbb{R}^M$ . Here,  $\mathbb{S}$  is the  $D$ -dimensional solution space, and  $\mathbb{R}^M$  is the  $M$ -dimensional objective space. In general, the goal of MOPs is to find a set of non-dominated solutions that approximates the Pareto front in the objective space.

Evolutionary multi-objective optimization algorithms (EMOAs) are effective approaches for MOPs [1]. It is expected that an EMOA is able to find a set of non-dominated solutions with acceptable quality for a decision maker in a single run. A number of EMOAs have been proposed, including NSGA-II [2], SPEA2 [3], IBEA [4], SMS-EMOA [5], MOEA/D [6], and NSGA-III [7]. In general, it is difficult to theoretically evaluate the performance of EMOAs mainly due to their stochastic nature. Thus, the performance assessment of EMOAs is empirically conducted on benchmark problems in most cases.

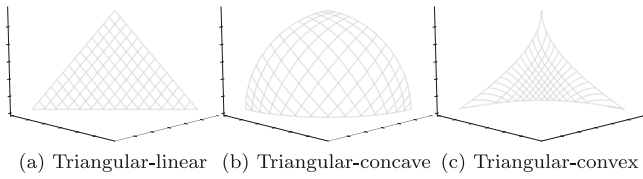
Test problems play a crucial role in the empirical performance evaluation. Synthetic test problems are usually used in the EMO community. Representative examples include the ZDT [8], DTLZ [9], WFG [10], and LZ [11] problem suites. The main advantages of synthetic test problems are threefold. First, they are easy to use. Since synthetic test problems are expressed by simple

equations, they can be easily implemented in any programming language. The calculation of each objective function is fast. Second, the properties of synthetic test problems are (almost) clear. Their Pareto fronts are usually known. Thus, some general knowledge can be obtained from experimental results on them. For example, an EMOA that works well on the DTLZ1 problem [9] is likely to have a good performance on separable multi-modal problems whose Pareto fronts are linear and triangular. In contrast, the properties of real-world problems are generally unclear. Thus, results on most real-world problems cannot be generalized. Third, some synthetic problems are scalable to the number of objectives  $M$  and design variables  $D$ . The scalability of EMOAs to  $M$  and  $D$  can be evaluated on such scalable problems.

Despite these advantages, most synthetic test problems have difficulties. They have unusual properties which are unlikely to appear in real-world applications [12,13]. The performance of an EMOA exploiting those unrealistic properties could be overrated on such problems. For example, some decomposition-based EMOAs (e.g., MOEA/D [6] and NSGA-III [7]) work well on the DTLZ and WFG problems even with many objectives [12]. This is because most DTLZ and WFG problems have triangular shape Pareto fronts. The term “triangular” means that the shape of the Pareto front looks like a distorted/bended triangle in a three-dimensional objective space [12]. Many existing scalable test problems have such triangular shaped Pareto fronts. Fig. 1(a)–(c) show linear, concave, and convex triangular shape Pareto fronts under three objectives, respectively. The Pareto fronts shown in Fig. 1(a)–(c) are identical to those of DTLZ1, DTLZ2 (DTLZ3, DTLZ4, WFG4–9), and convex DTLZ2 [7]. Any objective vector

\* Corresponding author.

E-mail addresses: [rt.ryoji.tanabe@gmail.com](mailto:rt.ryoji.tanabe@gmail.com) (R. Tanabe), [hisao@sustech.edu.cn](mailto:hisao@sustech.edu.cn) (H. Ishibuchi).



**Fig. 1.** Triangular shape Pareto fronts. The  $x$ ,  $y$ , and  $z$  axis represent  $f_1$ ,  $f_2$ , and  $f_3$ , respectively.

$f(\mathbf{x})$  in the Pareto front satisfies the following condition in the normalized objective space  $[0, 1]^M$  [14]:  $\sum_{i=1}^M (f_i(\mathbf{x}))^p = 1$ , where  $0 \leq (f_i(\mathbf{x})) \leq 1$  for  $i \in \{1, \dots, M\}$ ,  $p = 1$  for the linear Pareto front,  $1 < p$  for the concave Pareto front, and  $0 < p < 1$  for the convex Pareto front. The shape of the distribution of weight vectors in decomposition-based EMOAs is highly consistent with the shape of the triangular Pareto front [12]. For this reason, some decomposition-based EMOAs perform well on most DTLZ and WFG test problems even with many objectives. In contrast, some decomposition-based EMOAs perform poorly on problems with non-triangular Pareto fronts (e.g., the inverted DTLZ problems [15]). Test problems designed with the bottom-up approach (e.g., the DTLZ and WFG problems) include the distance and position variables. Here, the distance variables control the distance between the objective vector and the Pareto front. The position variables determine the position of the objective vector on the Pareto front. Since the distance and position variables play different roles, it is relatively easy to improve the convergence and diversity of a set of solutions by individually changing the distance and position variables [12].

Since synthetic test problems often exhibit such undesirable features, a new EMOA should be benchmarked on some real-world problems for more reliable evaluation. However, there does not exist an easy-to-use real-world multi-objective optimization problem suite. Although EMOAs have been applied to a number of real-world problems [16–19], it is difficult to use real-world problems to evaluate the performance of EMOAs.

In most cases, a new method is generally benchmarked only on some synthetic test problems. For example, the previous study [6] evaluated the performance of MOEA/D on ZDT and DTLZ. In some cases, a new method is benchmarked on a few real-world problems in addition to synthetic test problems. For example, the previous study [2] evaluated the performance of NSGA-II on the water resource planning problem [20]. However, only a few real-world problems are generally used for benchmarking EMOAs in almost all previous studies. Thus, the general performance of EMOAs on various real-world problems has not been examined in the literature. On the one hand, as mentioned above, EMOAs have been applied to a number of real-world problems, including supersonic wing design problems [16], finance/economics application problems [17], oil production planning problems [18], and electric vehicle control problems [19]. On the other hand, comparisons of multiple EMOAs were not performed in most application-oriented studies. An application-oriented study usually reports experimental results of a single EMOA in order to demonstrate that a good approximation of the Pareto front of a given real-world problem can be obtained (i.e., a better approximation can be obtained by the EMO approach than other approaches). Thus, the performance of multiple EMOAs on a given real-world problem was not investigated in such an application-oriented study. The RE problem set makes the performance comparison on real-world problems easy.

Let us consider the performance comparison of EMOAs on supersonic wing design problems [16]. In the first place, researchers who have no knowledge of aerodynamics cannot tackle

**Table 1**

Advantages and disadvantages of each problem type.

Problems	Non-artificiality	Easiness	Clearness	Scalability
Synthetic problems		✓	✓	✓
RE problems	✓	✓		

this real-world problem. Numerical simulations of complex fluid flow require specific software tools, high-performance computers, and long computation time. According to [16], it took about 100 hours to evaluate 4480 solutions on the supersonic wing design problems using 32 processing elements of an NEC SX-4 computer. If six EMOAs are carried out on this problem for 30 times, it takes about 2 years in total. While benchmarking EMOAs on real-world problems is needed, it is difficult due to those obstacles.

To fill this gap between the necessity and the availability of real-world problems, we present 16 bound-constrained real-world multi-objective optimization problems where definitions are given by simple equations or surrogate models. We refer to the set of the 16 problems as the REal world problem suite RE. The RE problem suite provides a wide variety of computationally cheap problems regarding the number of objectives, the shape of the Pareto front, and the type of design variables. The 16 RE problems do not perform a computationally expensive simulation. As shown in Section S.1 in the supplementary file of this paper, the objective functions in the 16 RE problems are represented by relatively simple equations. For this reason, the objective functions in the 16 RE problems are computationally cheap. Table 1 shows a summary of the advantages and disadvantages of synthetic problems and the RE problems. In contrast to synthetic problems, we believe that the RE problems do not have the above-mentioned unrealistic properties (e.g., the triangular Pareto front). We do not claim that real-world problems (including the RE problems) must always be better than synthetic test problems in terms of the performance evaluation of evolutionary algorithms. As mentioned above, most synthetic test functions have at least three advantages. Although the RE problems are as easy to use as synthetic test functions, they have two disadvantages compared to synthetic test functions. Similar to other real-world problems, the properties of the RE problems are unclear. All the RE problems are not scalable with respect to the number of objectives  $M$  and design variables  $D$ . However, we try to address the two disadvantages of the RE problems in this paper. We analyze the shape of the Pareto front of each RE problem in order to clarify its property. We have selected a wide variety of real-world problems with respect to  $M$  and  $D$  so that the scalability of EMOAs can be evaluated.

Our main contributions in this paper are as follows:

1. We present the 16 RE problems. Despite their importance, most of the original 16 RE problems have not received much attention in the EMO community. Four RE problems are multi-objective mixed-integer optimization problems. As pointed out in [32], there are only a few (multi-objective) mixed-integer optimization problems for benchmarking evolutionary algorithms. For this reason, we believe that the four mixed-integer RE problems are helpful in investigating the performance of evolutionary algorithms for multi-objective mixed-integer optimization.
2. We present details of the 16 RE problems in the supplementary file of this paper. The supplementary file is also available at the supplementary website (<https://ryojitnab.github.io/reproblems/>). Since this paper is self-contained, readers do not need to refer to each original article.

**Table 2**

Properties of the 16 test problems (RE2-4-1, ..., RE9-7-1).  $M$  and  $D$  denote the number of objectives and the number of decision variables, respectively. The type of design variables, the shape of the Pareto front, and other information are also described.

Name	Original name	$M$	$D$	Variables	Pareto front
RE2-4-1	Four bar truss design [21]	2	4	Continuous	Convex
RE2-3-2	Reinforced concrete beam design [22]	2	3	Mixed	Mixed
RE2-4-3	Pressure vessel design [23]	2	4	Mixed	Mixed, Disconnected
RE2-2-4	Hatch cover design [22]	2	2	Continuous	Convex
RE2-3-5	Coil compression spring design [24]	2	3	Mixed	Mixed, Disconnected
RE3-3-1	Two bar truss design [25]	3	3	Continuous	Unknown
RE3-4-2	Welded beam design [26]	3	4	Continuous	Unknown
RE3-4-3	Disk brake design [26]	3	4	Continuous	Unknown
RE3-5-4	Vehicle crashworthiness design [27]	3	5	Continuous	Unknown
RE3-7-5	Speed reducer design [28]	3	7	Mixed	Unknown
RE3-4-6	Gear train design [29]	3	4	Integer	Concave, Disconnected
RE3-4-7	Rocket injector design [30]	3	4	Continuous	Unknown
RE4-7-1	Car side impact design [15]	4	7	Continuous	Unknown
RE4-6-2	Conceptual marine design [31]	4	6	Continuous	Unknown
RE6-3-1	Water resource planning [20]	6	3	Continuous	Unknown
RE9-7-1	Car cab design [7]	9	7	Continuous	Unknown

3. We uploaded Java, C, and Matlab source codes of the RE problems to the supplementary website. Researchers can easily examine the performance of their EMOAs on the RE problems in an off-the-shelf manner.
4. We analyze the performance of six representative EMOAs on the RE problems.
5. In addition to the 16 RE problems, we present 8 Constrained multi-objective REal-world problems (CRE).

The rest of this paper is organized as follows. Section 2 presents the 16 RE problems. Section 3 describes experimental settings for the comparisons of EMOAs on the RE problem set. Section 4 reports experimental results. Section 5 concludes this paper.

## 2. RE problem suite

This section describes the RE problem suite. Table 2 shows the properties of the 16 RE problems. Section S.1 in the supplementary file provides details of the RE problems, including their definitions. In the first column of Table 2, we use the notation RE“ $M$ ”-“ $D$ ”-“ $Z$ ” in this paper, where  $M$  is the number of objectives,  $D$  is the number of decision variables, and  $Z$  is the problem ID. For example,  $M$  and  $D$  of RE3-4-2 are 3 and 4, respectively. The problem ID prevents that different RE problems share the same name (e.g., RE3-4-2 and RE3-4-3). If a new problem is proposed, it can be sequentially added to the RE problem set (e.g., a new three-objective and five-variable problem could be denoted as the RE3-5-8 problem). The current RE problem suite consists of problems with  $M \in \{2, 3, 4, 6, 9\}$ .

### 2.1. On four surrogate-based RE problems

Four out of the 16 RE problems (RE3-5-4, RE3-4-7, RE4-7-1, and RE9-7-1) include parameters obtained by the response surface method using data sampled from simulations. Thus, these four surrogate-based RE problems are not exactly the same as their corresponding original problems. Although the four RE problems are based on data taken from the real-world applications, they may be viewed as “artificial” problems. This fact should be kept in mind.

### 2.2. Reformulation of constrained single- and multi-objective real-world problems

The original car side-impact problem is a constrained single-objective problem with eight constraint functions [33]. A nine-objective version of it is designed in [7] by converting the constraint functions into eight new objective functions. Inspired

by [7], we reformulated constrained single- and multi-objective real-world problems as bound-constrained multi-objective problems in a similar manner (except for RE2-4-1, RE3-5-4, and RE3-4-7). However, our preliminary results indicated that the constraint functions of most original problems in Table 2 can be simultaneously minimized. Thus, we formulated the sum of constraint violation values as an additional objective function for all RE problems except for RE9-7-1.

For example, RE-2-3-2 is the reinforced concrete beam design problem [22]. The first objective  $f_1$  of the RE2-3-2 problem is to minimize the total cost of concrete and reinforcing steel of the beam:

$$f_1(\mathbf{x}) = 29.4x_1 + 0.6x_2x_3, \quad (1)$$

where  $x_2 \in [0, 20]$  and  $x_3 \in [0, 40]$ . In RE-2-3-2,  $x_1$  has a pre-defined discrete value from 0.2 to 15 (for details, see Section S.1.2 in the supplementary file). The three variables  $x_1$ ,  $x_2$ , and  $x_3$  represent the area of the reinforcement, the width of the beam, and the depth of the beam, respectively. The second objective  $f_2$  of RE2-3-2 is the sum of the two constraint violations:

$$f_2(\mathbf{x}) = \sum_{i=1}^2 \max\{g_i(\mathbf{x}), 0\}, \quad (2)$$

$$g_1(\mathbf{x}) = x_1x_3 - 7.735 \frac{x_1^2}{x_2} - 180 \geq 0, \quad (3)$$

$$g_2(\mathbf{x}) = 4 - \frac{x_3}{x_2} \geq 0. \quad (4)$$

Due to space constraint, we explain details of the other 15 RE problems in the supplementary file. Interested readers can refer to the supplementary file.

### 2.3. CRE problem suite

In addition to the 16 RE problems, we provide 8 constrained multi-objective real-world problems. We denote the 8 constrained problems as constrained RE (CRE) problems. Table 3 shows the properties of the eight CRE problems (CRE2-4-1, ..., CRE5-3-1). Each CRE problem is the original version of the corresponding RE problem (see Table 3). Recall that an additional objective function for some RE problems is the sum of constraint violation values. For the definition of each CRE problem, see the definition of the corresponding RE problem in Section S.1 in the supplementary file.

Studies on unconstrained multi-objective optimization problems can be the basis of studies on more challenging problems, including constrained multi-objective optimization problems [34], dynamic multi-objective optimization problems [35],

**Table 3**

Properties of the eight constrained test problems (CRE2-3-1, ..., RE5-3-1).

Name	Original name	RE ver.	M	D	N	Variables
CRE2-3-1	Two bar truss design [25]	RE3-3-1	2	3	3	Continuous
CRE2-4-2	Welded beam design [26]	RE3-4-2	2	4	4	Continuous
CRE2-4-3	Disk brake design [26]	RE3-4-3	2	4	4	Continuous
CRE2-7-4	Speed reducer design [28]	RE3-7-5	2	7	11	Mixed
CRE2-4-5	Gear train design [29]	RE3-4-6	2	4	1	Integer
CRE3-7-1	Car side impact design [15]	RE4-7-1	3	7	10	Continuous
CRE3-6-2	Conceptual marine design [31]	RE4-6-2	3	6	9	Continuous
CRE5-3-1	Water resource planning [20]	RE6-3-1	5	3	7	Continuous

and multi-modal multi-objective optimization problems [36]. For this reason, as the first step toward designing a benchmark set of easy-to-use real-world problems, we focus only on unconstrained multi-objective optimization problems in this paper. Although this paper does not analyze the eight CRE problems for this reason, we believe that the CRE problems could also be good benchmark problems for constrained multi-objective optimizers.

#### 2.4. Multi-objective mixed-integer RE problems

In Table 2, some (or all) design variables of the five RE problems take integer or discrete values. RE-2-3-2, RE2-4-3, RE2-3-5, and RE3-7-5 are multi-objective mixed-integer optimization problems, and RE3-4-6 is a multi-objective integer optimization problem. For example, as explained in Section 2.2, the first variable  $x_1$  in RE2-3-2 is the area of the reinforcement, which should be a pre-defined discrete value from 0.2 to 15. It should be noted that the 16 RE problems in Table 2 do not have categorical variables, whose ordering information cannot be exploited [37], such as a material of the designed product (e.g., copper, aluminum, and iron). Although multi-objective mixed-integer optimization problems appear in real-world applications [38,39], they have received less attention in the EMO community. One reason could be the unavailability of the proper test problems. The five RE problems would encourage researchers to develop new EMOAs for multi-objective mixed-integer optimization problems.

#### 2.5. Comparison to simulation-based real-world problems

All the RE problems are derived from the literature published in the 1980s–2000s. Also, the maximum number of decision variables in the RE problems is seven. These reasons are due to the progress of computer technology. Since some researchers could not use high-performance simulators in an easy manner twenty and thirty years ago, they needed to implement their real-world problems by human-understandable equations. It was also difficult to formulate a problem with many decision variables without degrading the understandability. Some RE problems are “secondary products” from these limitations. In contrast, thanks to technological progress, recent researchers can easily perform complex simulations on their personal workstations. As a result, almost all recently proposed real-world problems are based on computational simulations. However, such simulation-based real-world problems cannot be added to the RE problem set since their problem formulations are not given by human-understandable equations. These are the reasons why all the RE problems are derived from the old literature, and the maximum number of variables in the RE problems is seven.

Table 4 shows five simulation-based real-world problems that are available through the Internet. The five problems in Table 4 were proposed recently. The five problems have a larger number of decision variables than the RE problems. However, unlike the RE problems, the five problems need to perform computational simulations to calculate the objective function values of a solution. Thus, their problem formulations are not explicitly

**Table 4**

Five simulation-based real-world problems that are available online.

Test problems	M	D
Radar waveform design problem [40]	9	4, ..., 12
Heat exchanger design problem [41]	2	Any
Car structure design problem [42]	2	222
TopTrumps problem [43]	2	88, 128, 168, 208
MarioGAN problem [43]	2	10, 20, 30, 40

given by equations. Also, some of them require specific software tools. For example, the software for the radar waveform design problem [40] can be executed only by Matlab. For these reasons, we do not add the five problems into our RE problems. It should be recalled that one of the advantages of the RE problems is easy-to-use. This paper presents the RE problems so that researchers in the EMO community can use the 16 RE problems in an off-the-shelf manner.

#### 2.6. Approximated pareto fronts of the RE problems

Here, we discuss the shape of the approximated Pareto front of each RE problem. Since the Pareto optimal solution set and the Pareto front are unknown in each RE test problem, we used an approximate Pareto solution set  $\tilde{\mathbf{A}}$  for each problem.  $\tilde{\mathbf{A}}$  is available at the supplementary website. We selected  $500 \times M$  solutions in  $\tilde{\mathbf{A}}$  from a large number of non-dominated solutions  $\mathbf{A}$  found by various optimizers. We confirmed that the size of  $\mathbf{A}$  is more than  $500 \times M$  for all 16 RE problems, except for RE3-4-6. Due to the discrete property of RE3-4-6, the number of non-dominated solutions found in our experiments is limited to only 28. Only for RE3-4-6, we set  $\mathbf{A}$  to  $\tilde{\mathbf{A}}$  as is. For details of  $\mathbf{A}$ , see Section 3.1. We used a distance-based method to select well-distributed non-dominated solutions in the objective space [44]. Section S.2 in the supplementary file explains the distance-based selection method.

Fig. 2 shows approximated Pareto fronts of the five two-objective RE problems (RE2-4-1, RE-2-3-2, RE2-4-3, RE2-2-4, and RE2-3-5). As shown in Fig. 2, RE2-4-3 and RE2-3-5 have disconnected Pareto fronts whereas the other three problems have continuous Pareto fronts. It should be noted that Fig. 2 shows the approximated Pareto fronts, not the true Pareto fronts. In contrast to synthetic test problems, it is almost impossible to obtain the true Pareto optimal solution set in a real-world problem. While many synthetic test problems have regular shapes (except for some intentionally generated difficult problems [11]), the RE problems in Fig. 2 have somewhat irregular Pareto fronts. This is because a synthetic problem is usually designed to have a typical Pareto front shape such as linear and convex. However, real-world problems do not have such a nature. The objective functions of most RE problems are also differently scaled. For example, Fig. 2(a) shows that the first objective values in the approximated Pareto front are approximately in the range  $[1.2 \times 10^3, 2.1 \times 10^3]$ . In contrast, the second objective values in the approximated Pareto front are approximately in the range  $[2.8 \times$



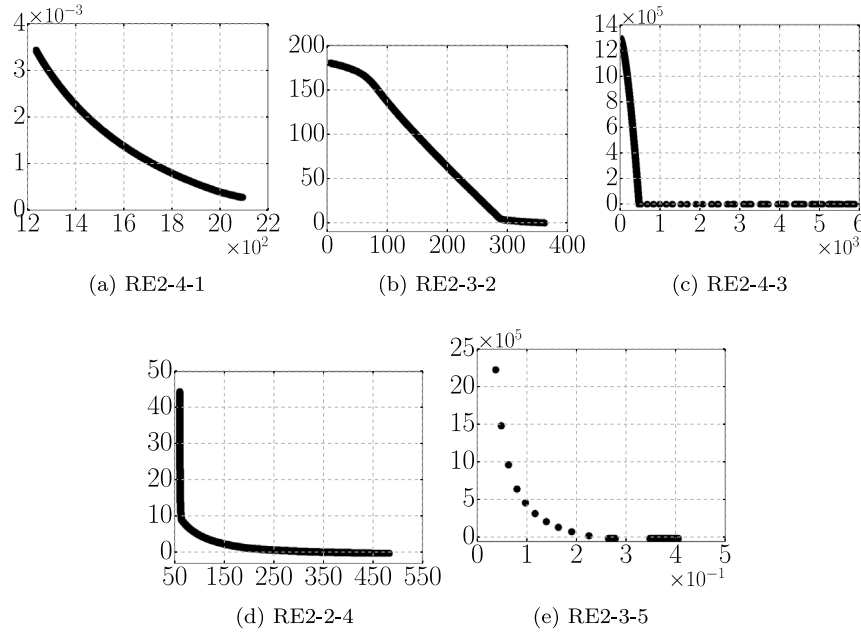


Fig. 2. Approximated Pareto fronts of the RE2-4-1, ..., RE2-3-5 problems. The x and y axis represent  $f_1$  and  $f_2$ , respectively.

$10^{-4}$ ,  $3.4 \times 10^{-3}$ ]. Thus, the scale of the first and second objective values in RE2-4-1 is totally different. We believe that most real-world problems have differently scaled objective values as in the RE problems.

The shape of the Pareto front of the RE2-4-1 problem is convex. In the RE-2-3-2 problem, the two extreme regions of the Pareto front are locally concave and convex, respectively. While the Pareto front shape of the RE2-4-3 problem seems to be convex at first glance, it is mixed and has disconnected regions (the top-left region close to the vertical axis is locally concave). The RE2-2-4 problem has a convex Pareto front with a sharp knee point. The RE2-3-5 problem has the mixed (linear and convex) and disconnected Pareto front. The bottom-right region of the Pareto front is linear in Fig. 2(e). Objective vectors on the disconnected region of the Pareto front are sparsely distributed.

Fig. 3 shows approximated Pareto fronts of the six three-objective RE problems. We do not show the approximated Pareto front of RE3-7-5, but it is similar to that of RE3-4-2. While the approximated Pareto fronts of the three-objective RE problems can be still visualized by 3D graphics as shown in Fig. 3, their shapes cannot be understood well. Figs. S.1–S.11 in the supplementary file show the scatter matrix of the approximated Pareto front of RE3-3-1, RE3-4-2, RE3-4-3, RE3-5-4, RE3-7-5, RE3-4-6, RE3-4-7, RE4-7-1, RE4-6-2, RE6-3-1, and RE9-7-1, respectively. Unfortunately, except for RE3-4-6, the scatter matrices do not provide clear shapes of the Pareto fronts. This is because most RE problems have complicated Pareto fronts. It is usually difficult to clearly visualize Pareto front shapes of problems with four or more objectives [45]. Although parallel coordinates are frequently used to discuss the distribution of the objective vectors for problems with more than four objectives, they do not provide sufficient information about the shape of the Pareto front [46].

While recently proposed EMOAs perform well on problems with many objectives, they lack the ability to handle problems with irregular Pareto fronts [12,47]. To address this issue, some EMOAs were designed to handle complex problems (e.g., [47–49]). The RE problems will be used as a proper problem suite to evaluate the performance of such EMOAs.

### 3. Experimental settings

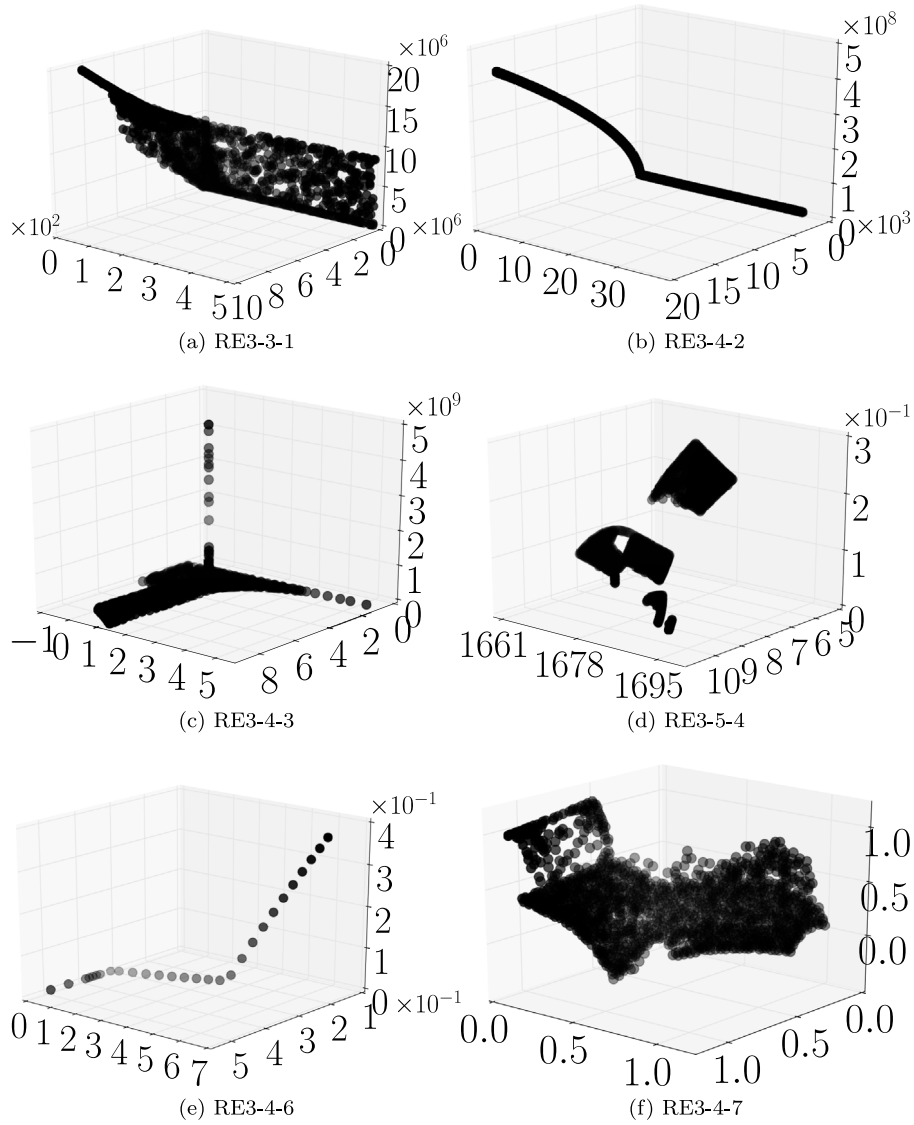
We investigate the performance of representative EMOAs on the RE problem suite. This section describes our experimental settings. Results are reported in Section 4.

#### 3.1. Performance indicators

We use the hypervolume indicator (HV) [50] for the performance assessment. HV is the only Pareto-compliant quality indicator known so far in the literature [51]. A large HV value indicates that a given solution set approximates the Pareto front well in terms of both convergence and diversity in the objective space. In general, the Pareto optimal solutions and the Pareto front of a real-world problem are unavailable. For this reason, we do not use quality indicators which require reference objective vectors (e.g., IGD [52] and the  $\epsilon$ -indicator [53]). As in [12,54], we set the  $M$ -dimensional reference vector for the HV calculation to  $(1.1, \dots, 1.1)^T$  in the normalized objective space  $[0, 1]^M$ . Before calculating the HV value, we normalize the objective vector  $\mathbf{f}(\mathbf{x})$  using the approximated ideal point  $\mathbf{z}^{\text{ideal}}$  and the approximated nadir point  $\mathbf{z}^{\text{nadir}}$  of each RE problem. For each  $i \in \{1, \dots, M\}$ , the  $i$ th normalized objective value  $f_i^{[0,1]}(\mathbf{x})$  of a solution  $\mathbf{x}$  is obtained as follows:  $f_i^{[0,1]}(\mathbf{x}) = (f_i(\mathbf{x}) - z_i^{\text{nadir}}) / (z_i^{\text{ideal}} - z_i^{\text{nadir}})$ .

To obtain  $\mathbf{z}^{\text{ideal}}$  and  $\mathbf{z}^{\text{nadir}}$ , we used L-SHADE [55] and five EMOAs (NSGA-II [2], MOEA/D-TCH [6], MOEA/D-PBI [6], IBEA [4], and NSGA-III [7]). Only for these experiments, the number  $n^{\text{max}}$  of function evaluations was set to 100 000, and 31 independent runs were performed. L-SHADE was the winner of the IEEE CEC2014 competition on single-objective real-parameter optimization. We used the Java source code of L-SHADE provided by the authors of [55]. The default parameter setting was used for L-SHADE. For each RE problem, we individually applied L-SHADE to each objective function. For the five EMOAs, we used the same control parameter setting described in Section 3.3. We applied the five EMOAs to each RE problem. For L-SHADE, we maintained only the best-so-far solution for each run. For the five EMOAs, we maintained only the final population for each run.

For each RE problem, let  $\mathbf{A}$  be a set of non-dominated solutions obtained by the above-mentioned procedure. We set the  $i$ th objective value  $z_i^{\text{ideal}}$  of  $\mathbf{z}^{\text{ideal}}$  as follows:  $z_i^{\text{ideal}} = \min_{\mathbf{x} \in \mathbf{A}} f_i(\mathbf{x})$ . We



**Fig. 3.** Approximated Pareto fronts of the RE3-3-1, ..., RE3-4-7 problems. The x, y, and z axis represent  $f_1$ ,  $f_2$ , and  $f_3$ , respectively.

also set the  $i$ th objective value  $z_i^{\text{nadir}}$  of  $\mathbf{z}^{\text{nadir}}$  as follows:  $z_i^{\text{nadir}} = \max_{\mathbf{x} \in A} f_i(\mathbf{x})$ .

### 3.2. Performance score

We used the performance score [56] to rank EMOAs on each RE problem. Suppose that  $n$  algorithms  $A_1, \dots, A_n$  are compared for a problem based on any performance indicator values obtained in multiple runs. For each  $i \in \{1, \dots, n\}$  and  $j \in \{1, \dots, n\} \setminus \{i\}$ , let  $\delta_{i,j} = 1$  if  $A_j$  significantly outperforms  $A_i$  based on the Wilcoxon rank-sum test with  $p < 0.05$ , otherwise  $\delta_{i,j} = 0$ . The performance score  $P(A_i)$  is defined as follows:  $P(A_i) = \sum_{j \in \{1, \dots, n\} \setminus \{i\}} \delta_{i,j}$ . The score  $P(A_i)$  represents the number of algorithms outperforming  $A_i$ . A small  $P(A_i)$  value indicates that  $A_i$  is not worse than many other algorithms.

### 3.3. EMOAs

We examine the performance of NSGA-II [2], MOEA/D [6], IBEA [4], SMS-EMOA [5], and NSGA-III [7], which are representative EMOAs based on dominance, decomposition, indicator, and reference vector, respectively. We used the jMetal 4.5 framework [57] for algorithm implementation. For NSGA-III, we used

the source code implemented by the authors of [54]. As analyzed in [58], better performance of NSGA-III can be obtained by using the source code implemented by the authors of [54] than the jMetal 5.0 source code [59]. We conducted all experiments on a workstation with an Intel(R) Xeon(R) CPU E5-2620 v4@2.10 GHz and 8GB RAM using the Ubuntu 16.04 OS. The Java version that was used was OpenJDK version 1.8.0\_232. The number  $n^{\text{max}}$  of function evaluations was set to 10 000. We performed 31 independent runs.

For all EMOAs, the population size  $\mu$  was set to 100, 105, 120, 210, 182, and 210 for  $M = 2, 3, 4, 6$ , and 9, respectively. Since  $\mu$  in NSGA-II must be a multiple of two,  $\mu = 106$  for  $M = 3$ . SBX crossover [60] and polynomial mutation [60] were used in all EMOAs. Their control parameters were set as follows:  $p_c = 1$ ,  $\eta_c = 20$ ,  $p_m = 1/D$ , and  $\eta_m = 20$ . A real-coded value was rounded to the nearest feasible value for discrete and integer variables in RE22, RE23, RE25, RE35, and RE36. The additive epsilon indicator  $I_{\epsilon+}$  [4] was used for IBEA. According to [6], the neighborhood size of MOEA/D was set to 20. MOEA/D decomposes  $M$ -objective problems into  $N$  single-objective problems by using a set of weight vectors  $\mathbf{W} = \{\mathbf{w}_1, \dots, \mathbf{w}_N\}$  and a scalarizing function  $g : \mathbb{R}^M \rightarrow \mathbb{R}$ . A set of weight/reference vectors of MOEA/D and NSGA-III were generated by the simplex-lattice design [61] for

**Table 5**

Results of the six EMOAs on the 16 RE problems. The table shows the mean HV values over 31 runs. The numbers in parenthesis indicate the ranks of the six EMOAs based on the performance score value (see Section 3.2). The first ranked results are represented by the bold font.

	NSGA-II	MOEA/D-TCH	MOEA/D-PBI	NSGA-III	IBEA	SMS-EMOA
RE2-4-1	8.53e-01 (5)	8.54e-01 (3)	8.53e-01 (5)	8.55e-01 (3)	<b>8.56e-01</b> (1)	8.55e-01 (2)
RE2-3-2	7.54e-01 (2)	7.53e-01 (3)	7.41e-01 (6)	7.49e-01 (5)	7.51e-01 (3)	<b>7.59e-01</b> (1)
RE2-4-3	1.16e+00 (3)	<b>1.16e+00</b> (1)	1.12e+00 (6)	1.16e+00 (4)	1.16e+00 (2)	1.15e+00 (5)
RE2-2-4	1.17e+00 (2)	1.17e+00 (3)	1.09e+00 (6)	1.17e+00 (3)	1.16e+00 (5)	<b>1.17e+00</b> (1)
RE2-3-5	<b>1.08e+00</b> (1)	1.08e+00 (3)	1.04e+00 (6)	1.08e+00 (5)	1.08e+00 (3)	1.08e+00 (2)
RE3-3-1	1.33e+00 (3)	<b>1.33e+00</b> (1)	1.32e+00 (6)	1.33e+00 (2)	1.33e+00 (4)	1.33e+00 (5)
RE3-4-2	<b>1.33e+00</b> (1)	1.31e+00 (4)	1.28e+00 (6)	1.33e+00 (2)	1.33e+00 (3)	1.30e+00 (5)
RE3-4-3	9.05e-01 (5)	9.55e-01 (3)	8.81e-01 (5)	9.78e-01 (2)	9.28e-01 (4)	<b>1.00e+00</b> (1)
RE3-5-4	1.03e+00 (3)	1.00e+00 (5)	9.60e-01 (6)	1.02e+00 (4)	<b>1.04e+00</b> (1)	<b>1.04e+00</b> (1)
RE3-7-5	<b>1.30e+00</b> (1)	1.30e+00 (3)	1.28e+00 (6)	1.29e+00 (4)	1.30e+00 (5)	<b>1.30e+00</b> (1)
RE3-4-6	<b>1.01e+00</b> (1)	9.94e-01 (5)	9.92e-01 (5)	1.00e+00 (4)	1.00e+00 (2)	1.00e+00 (2)
RE3-4-7	7.89e-01 (5)	8.07e-01 (2)	8.02e-01 (3)	7.92e-01 (4)	<b>8.14e-01</b> (1)	7.89e-01 (5)
RE4-7-1	6.79e-01 (5)	7.19e-01 (2)	7.20e-01 (2)	7.16e-01 (4)	<b>7.64e-01</b> (1)	-
RE4-6-2	7.49e-01 (3)	7.66e-01 (2)	7.33e-01 (3)	7.41e-01 (5)	<b>7.86e-01</b> (1)	-
RE6-3-1	1.05e+00 (4)	1.09e+00 (3)	9.62e-01 (5)	1.12e+00 (2)	<b>1.18e+00</b> (1)	-
RE9-7-1	4.39e-02 (3)	5.58e-02 (2)	2.11e-02 (5)	4.17e-02 (4)	<b>6.61e-02</b> (1)	-

$M \leq 5$  and its two-layered version [7] for  $M \geq 6$ . We used the Tchebycheff function  $g^{\text{tch}}$  [6] and the PBI function  $g^{\text{pbi}}$  [6] in MOEA/D since they were used in many previous studies [62]. The resulting algorithm versions are denoted as MOEA/D-TCH and MOEA/D-PBI, respectively. The Tchebycheff function  $g^{\text{tch}}$  and the PBI function  $g^{\text{pbi}}$  are given as follows:

$$g^{\text{tch}}(\mathbf{x}|\mathbf{w}) = \max_{i \in \{1, \dots, M\}} \{w_i |f_i(\mathbf{x}) - z_i^{\text{ideal}}|\}, \quad (5)$$

$$g^{\text{pbi}}(\mathbf{x}|\mathbf{w}) = d_1 + \theta d_2, \quad (6)$$

$$d_1 = \frac{\|(\mathbf{f}(\mathbf{x}) - \mathbf{z}^{\text{ideal}})^T \mathbf{w}\|}{\|\mathbf{w}\|}, \quad (7)$$

$$d_2 = \left\| \mathbf{f}(\mathbf{x}) - \left( \mathbf{z}^{\text{ideal}} + d_1 \frac{\mathbf{w}}{\|\mathbf{w}\|} \right) \right\|, \quad (8)$$

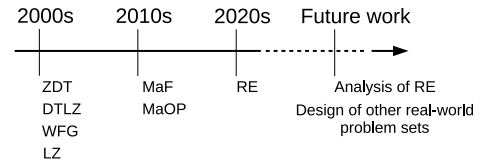
where  $\|\mathbf{a}\|$  is the Euclidean norm of a given vector  $\mathbf{a}$ . The distance  $d_1$  measures how well  $\mathbf{f}(\mathbf{x})$  converges to the Pareto front. The distance  $d_2$  is the perpendicular distance between  $\mathbf{f}(\mathbf{x})$  and  $\mathbf{w}$ . Thus,  $d_2$  measures how close  $\mathbf{f}(\mathbf{x})$  and  $\mathbf{w}$  are. The penalty parameter  $\theta$  strikes the balance between the convergence ( $d_1$ ) and the diversity ( $d_2$ ). According to [6], we set  $\theta$  to 5. The simple normalization strategy described in [6] was introduced into MOEA/D-TCH and MOEA/D-PBI in order to handle the RE problems with totally different objective values. As explained in Section 2.6, the objective values of most RE problems are differently scaled. As demonstrated in [7], the use of the normalization strategy significantly improves the performance of EMOAs on problems with differently scaled objective values.

## 4. Experimental results

### 4.1. Performance comparison

Table 5 shows results of the six EMOAs on the 16 RE problems. It should be noted that the rank of each EMOA in Table 5 is based on the performance score value (see Section 3.2), not the raw HV value. SMS-EMOA was applied only to the RE problems with two and three objectives due to its high computational cost.

Table 5 indicates that there is no clear winner on all the two-objective RE problems. IBEA, MOEA/D-TCH, and NSGA-II obtain the best HV value for RE2-4-1, RE2-4-3, and RE2-3-5, respectively. SMS-EMOA performs the best on RE2-3-2 and RE2-2-4 in terms of HV. Unlike the results of the two-objective RE problems, the best HV values are obtained by IBEA for 6 out of the 10 RE problems with three or more objectives. As explained in Section 3.3, our IBEA uses the additive  $\epsilon$  indicator  $I_{\epsilon+}$ , not the hypervolume difference indicator ( $I_{HD}$ ) [4]. Thus, our IBEA does not directly optimize

**Fig. 4.** Progress on multi-objective continuous test problem sets.

HV. NSGA-II and SMS-EMOA are competitive with each other on RE3-7-5 regarding the performance score with the Wilcoxon rank-sum test. MOEA/D-PBI shows the worst performance on 10 out of the 16 RE problems.

### 4.2. Discussion

#### 4.2.1. Inconsistency of the performance of NSGA-III and MOEA/d-PBI

Table 5 shows that NSGA-II outperforms NSGA-III on 10 out of the 16 RE problems in terms of the rank based on the performance score value. While MOEA/D-PBI showed a good performance on many-objective problems in previous studies [7,63], it does not work well in our study.

The reason is twofold. First, NSGA-II and MOEA/D-PBI seemingly cannot handle irregular shape Pareto fronts well as described in Section 1. It was shown in [47] that NSGA-III fails to find well-distributed solutions on some problems with convex or disconnected Pareto fronts. As pointed out in [64], solutions obtained by MOEA/D-PBI with an intermediate  $\theta$  value (e.g.,  $\theta = 5$ ) are biasedly distributed on the center of the convex Pareto front. For details of  $\theta$ , see Section 3.3. The second reason is due to the setting of the maximum number of evaluations  $n^{\text{max}}$ . Previous studies [44,65] show that some EMOAs (including NSGA-III and MOEA/D-PBI) do not work well for a small value of  $n^{\text{max}}$ . While  $n^{\text{max}} = 10,000$  in our study, it was set to a very large value in most recent studies [44]. For example,  $n^{\text{max}} = 550,000$  in the original paper of NSGA-III [7]. However, it is difficult to set  $n^{\text{max}}$  to a large value in some real-world problems. This is because they require a computationally expensive simulation to evaluate a solution [16]. Since the RE problems are computationally cheap,  $n^{\text{max}}$  can be set to a large value (e.g.,  $n^{\text{max}} = 550,000$ ). However, it would be more practical to set  $n^{\text{max}}$  to a small value even on the RE problems so that the performance of EMOAs on computationally expensive problems can be estimated.

## 5. Conclusion

While real-world problems are important in evaluating the performance of EMOAs, they are difficult to use due to some obstacles. To fill this gap, we presented the RE problem suite, which consists of 16 real-world multi-objective optimization problems. We provided Java, C, and Matlab source codes of the RE problem suite as the supplementary materials so that researchers can use them as standard benchmark problems. We also presented the comparison of six EMOAs on the RE problems.

Fig. 4 shows the progress on multi-objective continuous test problem sets. For details, refer to [10,13]. In the 2000s, various test problem sets have been proposed in the literature such as ZDT [8], DTLZ [9], WFG [10], and LZ [11] problem suites. In the 2010s, more sophisticated problem sets have been proposed, including MaF [66] and MaOP [67]. However, most synthetic test problems have some undesirable features as discussed in Section 1. We believe that the RE problem set can address these issues. Future research topics include the performance evaluation of other state-of-the-art EMOAs and further analysis of each RE problem (e.g., shape of the feasible region of each problem in the objective space, distance from initial solutions to the Pareto front, difficulty of convergence and diversification). Designing other benchmark sets (e.g., a set of dynamic multi-objective real-world problems) is also an avenue for future work. We also hope that the RE problem set is helpful in developing efficient EMOAs for a general computational intelligence aided design framework in the smart design process.

## Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to <https://doi.org/10.1016/j.asoc.2020.106078>.

## Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant No. 61876075), the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (Grant No. 2017ZT07X386), Shenzhen Peacock Plan (Grant No. KQTD2016112514355531), the Science and Technology Innovation Committee Foundation of Shenzhen (Grant No. ZDSYS201703031748284), the Program for University Key Laboratory of Guangdong Province (Grant No. 2017KSYS008).

## Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.asoc.2020.106078>.

## References

- [1] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms, John Wiley & Sons, 2001.
- [2] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE TEVC* 6 (2) (2002) 182–197.
- [3] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the Strength Pareto Evolutionary Algorithm, Tech. Rep. 103, ETHZ, 2001.
- [4] E. Zitzler, S. Künzli, Indicator-based selection in multiobjective search, in: *PPSN*, 2004, pp. 832–842.
- [5] N. Beume, B. Naujoks, M.T.M. Emmerich, SMS-EMOA: multiobjective selection based on dominated hypervolume, *EJOR* 181 (3) (2007) 1653–1669.
- [6] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, *IEEE TEVC* 11 (6) (2007) 712–731.
- [7] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints, *IEEE TEVC* 18 (4) (2014) 577–601.
- [8] E. Zitzler, K. Deb, L. Thiele, Comparison of multiobjective evolutionary algorithms: Empirical results, *Evol. Comput.* 8 (2) (2000) 173–195.
- [9] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable test problems for evolutionary multi-objective optimization, in: *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*, Springer, 2005, pp. 105–145.
- [10] S. Huband, P. Hingston, L. Barone, R.L. While, A review of multiobjective test problems and a scalable test problem toolkit, *IEEE TEVC* 10 (5) (2006) 477–506.
- [11] H. Li, Q. Zhang, Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II, *IEEE TEVC* 13 (2) (2009) 284–302.
- [12] H. Ishibuchi, Y. Setoguchi, H. Masuda, Y. Nojima, Performance of decomposition-based many-objective algorithms strongly depends on pareto front shapes, *IEEE TEVC* 21 (2) (2017) 169–190.
- [13] S.Z. Martínez, C.A.C. Coello, H.E. Aguirre, K. Tanaka, A review of features and limitations of existing scalable multiobjective test suites, *IEEE TEVC* 23 (1) (2019) 130–142.
- [14] H. Ishibuchi, L. He, K. Shang, Regular pareto front shape is not realistic, in: *IEEE CEC*, 2019, pp. 2034–2041.
- [15] H. Jain, K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: handling constraints and extending to an adaptive approach, *IEEE TEVC* 18 (4) (2014) 602–622.
- [16] S. Obayashi, D. Sasaki, Y. Takeguchi, N. Hirose, Multiobjective evolutionary computation for supersonic wing-shape optimization, *IEEE TEVC* 4 (2) (2000) 182–187.
- [17] A. Ponsich, A.L. Jaimes, C.A.C. Coello, A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications, *IEEE TEVC* 17 (3) (2013) 321–344.
- [18] H.K. Singh, T. Ray, R.A. Sarker, Optimum oil production planning using infeasibility driven evolutionary algorithm, *Evol. Comput.* 21 (1) (2013) 65–82.
- [19] R. Cheng, T. Rodemann, M. Fischer, M. Olhofer, Y. Jin, Evolutionary many-objective optimization of hybrid electric vehicle control: From general optimization to preference articulation, *IEEE Trans. Emerg. Top. Comput. Intell.* 1 (2) (2017) 97–111.
- [20] T. Ray, K. Tai, K.C. Seow, Multiobjective design optimization by an evolutionary algorithm, *Eng. Opt.* 33 (4) (2001) 399–424.
- [21] F.Y. Cheng, X.S. Li, Generalized center method for multiobjective engineering optimization, *Eng. Opt.* 31 (5) (1999) 641–661.
- [22] H.M. Amir, T. Hasegawa, Nonlinear mixed-discrete structural optimization, *J. Struct. Eng.* 115 (3) (1989) 626–646.
- [23] B.K. Kannan, S.N. Kramer, An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design, *J. Mech. Des.* 116 (2) (1994) 405–411.
- [24] J. Lampinen, I. Zelinka, Mixed integer-discrete-continuous optimization by differential evolution, part 2: a practical example, in: *Int. Conf. on Soft Comput.*, 1999, pp. 77–81.
- [25] C.A.C. Coello, G.T. Pulido, Multiobjective structural optimization using a microgenetic algorithm, *Struct. Multidiscip. Optim.* 30 (5) (2005) 388–403.
- [26] T. Ray, K.M. Liew, A swarm metaphor for multiobjective design optimization, *Eng. Opt.* 34 (2) (2002) 141–153.
- [27] X. Liao, Q. Li, X. Yang, W. Zhang, W. Li, Multiobjective optimization for crash safety design of vehicles using stepwise regression model, *Struct. Multidiscip. Optim.* 35 (6) (2008) 561–569.
- [28] A. Farhang-Mehr, S. Azarm, Entropy-based multi-objective genetic algorithm for design optimization, *Struct. Multidiscip. Optim.* 24 (5) (2002) 351–361.
- [29] K. Deb, A. Srinivasan, Innovization: Innovative Design Principles Through Optimization, Tech. Rep. KanGAL-2005007, Indian Institute of Technology, 2005.
- [30] R. Vaidyanathan, K. Tucker, N. Papila, W. Shyy, Cfd-based design optimization for single element rocket injector, in: *AIAA Aerospace Sciences Meeting*, 2003, pp. 1–21.
- [31] M.G. Parsons, R.L. Scott, Formulation of multicriterion design optimization problems for solution with scalar numerical optimization methods, *J. Ship Res.* 48 (1) (2004) 61–76.
- [32] T. Tusar, D. Brockhoff, N. Hansen, Mixed-integer benchmark problems for single- and bi-objective optimization, in: *GECCO*, 2019, pp. 718–726.
- [33] L. Gu, R.J. Yang, C.H. Tho, M. Makowskit, O. Faruquet, Y. Li, Optimisation and robustness for crashworthiness of side impact, *Int. J. Veh. Des.* 26 (4) (2001) 348–360.
- [34] E. Mezura-Montes, C.A.C. Coello, Constraint-handling in nature-inspired numerical optimization: Past, present and future, *SWEVO* 1 (4) (2011) 173–194.
- [35] T.T. Nguyen, S. Yang, J. Branke, Evolutionary dynamic optimization: A survey of the state of the art, *SWEVO* 6 (2012) 1–24.



- [36] R. Tanabe, H. Ishibuchi, A review of evolutionary multi-modal multi-objective optimization, *IEEE TEVC* (2019) in press.
- [37] F. Hutter, H.H. Hoos, K. Leyton-Brown, T. Stützle, ParamILS: An automatic algorithm configuration framework, *J. Artif. Intell. Res. (JAIR)* 36 (2009) 267–306.
- [38] M.J. Alves, J.C.N. Clímaco, A review of interactive methods for multi-objective integer and mixed-integer programming, *EJOR* 180 (1) (2007) 99–115.
- [39] A.E.I. Brownlee, J.A. Wright, Constrained, mixed-integer and multi-objective optimisation of building designs by NSGA-II with fitness approximation, *Appl. Soft Comput.* 33 (2015) 114–126.
- [40] E.J. Hughes, Radar waveform optimisation as a many-objective application benchmark, in: *EMO*, 2006, pp. 700–714.
- [41] S.J. Daniels, A.A.M. Rahat, R.M. Everson, G.R. Tabor, J.E. Fieldsend, A suite of computationally expensive shape optimisation problems using computational fluid dynamics, in: *PPSN*, 2018, pp. 296–307.
- [42] T. Kohira, H. Kemmotsu, A. Oyama, T. Tatsukawa, Proposal of benchmark problem based on real-world car structure design optimization, in: *GECCO (Companion)*, 2018, pp. 183–184.
- [43] V. Volz, B. Naujoks, P. Kerschke, T. Tusar, Single- and multi-objective game-benchmark for evolutionary algorithms, in: *GECCO*, 2019, pp. 647–655.
- [44] R. Tanabe, H. Ishibuchi, A. Oyama, Benchmarking multi- and many-objective evolutionary algorithms under two optimization scenarios, *IEEE Access* 5 (2017) 19597–19619.
- [45] T. Tusar, B. Filipic, Visualization of pareto front approximations in evolutionary multiobjective optimization: A critical review and the prosecution method, *IEEE TEVC* 19 (2) (2015) 225–245.
- [46] M. Li, L. Zhen, X. Yao, How to read many-objective solution sets in parallel coordinates [educational forum], *IEEE CIM* 12 (4) (2017) 88–100.
- [47] S. Jiang, S. Yang, An improved multiobjective optimization evolutionary algorithm based on decomposition for complex pareto fronts, *IEEE Trans. Cybern.* 46 (2) (2016) 421–437.
- [48] S. Yang, S. Jiang, Y. Jiang, Improving the multiobjective evolutionary algorithm based on decomposition with new penalty schemes, *Soft Comput.* 21 (16) (2017) 4677–4691.
- [49] A clustering-based adaptive evolutionary algorithm for multiobjective optimization with irregular pareto fronts, *IEEE Trans. Cybern.* 49 (7) (2019) 2758–2770.
- [50] E. Zitzler, L. Thiele, Multiobjective optimization using evolutionary algorithms - A comparative case study, in: *PPSN*, 1998, pp. 292–304.
- [51] E. Zitzler, D. Brockhoff, L. Thiele, The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration, in: *EMO*, 2006, pp. 862–876.
- [52] C.A.C. Coello, M.R. Sierra, A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm, in: *MICAI*, 2004, pp. 688–697.
- [53] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, *IEEE TEVC* 7 (2) (2003) 117–132.
- [54] Y. Yuan, H. Xu, B. Wang, X. Yao, A new dominance relation-based evolutionary algorithm for many-objective optimization, *IEEE TEVC* 20 (1) (2016) 16–37.
- [55] R. Tanabe, A.S. Fukunaga, Improving the search performance of SHADE using linear population size reduction, in: *IEEE CEC*, 2014, pp. 1658–1665.
- [56] J. Bader, E. Zitzler, Hype: An algorithm for fast hypervolume-based many-objective optimization, *Evol. Comput.* 19 (1) (2011) 45–76.
- [57] J.J. Durillo, A.J. Nebro, jMetal: A java framework for multi-objective optimization, *Adv. Eng. Softw.* 42 (10) (2011) 760–771.
- [58] H. Ishibuchi, R. Imada, Y. Setoguchi, Y. Nojima, Performance comparison of NSGA-II and NSGA-III on various many-objective test problems, in: *IEEE CEC*, 2016, pp. 3045–3052.
- [59] A.J. Nebro, J.J. Durillo, M. Vergne, Redesigning the jmetal multi-objective optimization framework, in: *GECCO (Companion)*, 2015, pp. 1093–1100.
- [60] K. Deb, R.B. Agrawal, Simulated binary crossover for continuous search space, *Complex Syst.* 9 (2) (1995).
- [61] I. Das, J.E. Dennis, Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems, *SIAM J. Optim.* 8 (3) (1998) 631–657.
- [62] A. Trivedi, D. Srinivasan, K. Sanyal, A. Ghosh, A survey of multiobjective evolutionary algorithms based on decomposition, *IEEE TEVC* 21 (3) (2017) 440–462.
- [63] K. Li, K. Deb, Q. Zhang, S. Kwong, An evolutionary many-objective optimization algorithm based on dominance and decomposition, *IEEE TEVC* 19 (5) (2015) 694–716.
- [64] H. Ishibuchi, K. Doi, Y. Nojima, Use of piecewise linear and nonlinear scalarizing functions in MOEA/D, in: *PPSN*, 2016, pp. 503–513.
- [65] R. Tanabe, H. Ishibuchi, An analysis of control parameters of MOEA/D under two different optimization scenarios, *Appl. Soft Comput.* 70 (2018) 22–40.
- [66] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, X. Yao, Benchmark Functions for the CEC'2017 Competition on Many-Objective Optimization, Tech. rep., Univ. of Birmingham, 2017.
- [67] H. Li, K. Deb, Q. Zhang, P.N. Suganthan, L. Chen, Comparison between MOEA/D and NSGA-III on a set of novel many and multi-objective benchmark problems with challenging difficulties, *SWEVO* 46 (2019) 104–117.