Empirical Investigations of Reference Point Based Methods When Facing a Massively Large Number of Objectives: First Results

Ke Li[1,2(✉)], Kalyanmoy Deb[3], Tolga Altinoz[4], and Xin Yao[2]

[1] University of Exeter, North Park Road, Exeter EX4 4QF, UK
k.li@exeter.ac.uk
[2] University of Birmingham, Edgbaston, Birmingham B15 2TT, UK
x.yao@cs.bham.ac.uk
[3] Michigan State University, East Lansing, MI 48864, USA
kdeb@egr.msu.edu
[4] Ankara University, Ankara, Turkey
taltinoz@ankara.edu.tr

Abstract. Multi-objective optimization with more than three objectives has become one of the most active topics in evolutionary multi-objective optimization (EMO). However, most existing studies limit their experiments up to 15 or 20 objectives, although they claimed to be capable of handling as many objectives as possible. To broaden the insights in the behavior of EMO methods when facing a massively large number of objectives, this paper presents some preliminary empirical investigations on several established scalable benchmark problems with 25, 50, 75 and 100 objectives. In particular, this paper focuses on the behavior of the currently pervasive reference point based EMO methods, although other methods can also be used. The experimental results demonstrate that the reference point based EMO method can be viable for problems with a massively large number of objectives, given an appropriate choice of the distance measure. In addition, sufficient population diversity should be given on each weight vector or a local niche, in order to provide enough selection pressure. To the best of our knowledge, this is the first time an EMO methodology has been considered to solve a massively large number of conflicting objectives.

1 Introduction

During the past three decades, a large number of evolutionary multi-objective optimization (EMO) algorithms have been developed for solving multi-objective optimization problems (MOPs) with two and three objectives [8]. With the development of science and technology, real-life applications nowadays consider more complicated problems with four or more objectives, as known as many-objective optimization problems. Unfortunately, the curse of dimensionality has always been the Achilles's heel of optimization algorithms. It is widely accepted that the performance of Pareto dominance-based EMO algorithms such as NSGA-II [7] severely degrade with the increase of the number of objectives [19].

H. Trautmann et al. (Eds.): EMO 2017, LNCS 10173, pp. 390–405, 2017.
DOI: 10.1007/978-3-319-54157-0_27

Although some indicator-based EMO algorithms such as SMS-EMOA [4] claim to be scalable to any number of objectives in theory, their computational overheads in practice increase exponentially with the number of objectives [2]. In the past three years or so, many-objective optimization has become one of the most active topics within the EMO community and numerous studies have been conducted [13], e.g., remedy strategies for the canonical Pareto dominance to improve the convergence property [20], diversity management mechanisms to reimburse the loss of selection pressure [1], techniques for speeding up the calculation of some computationally expensive performance metrics in the indicator-based methods [3,15]. Recently, the reference point based method[1], e.g., MOEA/D [22] and NSGA-III [9], has shown very competitive performance for handling problems with many objectives. Generally speaking, its basic idea is using a set of pre-defined weight vectors to guide the search process. In particular, the weight vectors can be used to decompose the original MOP into a set of subproblems, either as scalarizing functions or simplified MOPs. As a consequence, the convergence is guaranteed by the optimization of each subproblem whereas the diversity is implicitly controlled by the uniform distribution of weight vectors.

Although the existing many-objective optimizers claim to be able to handle problems scalable to any number of objectives, most, if not all, existing studies limit their experiments to problems with the number of objectives up to 15 or 20 [9,14,21]. In view of the competitive performance reported in recent studies [12], this paper empirically investigates the performance of three selected reference point based EMO algorithms, i.e., MOEA/D, NSGA-III and R-NSGA-II [10], on MOPs with a massively large number of objectives. From the experimental results, we surprisingly find that R-NSGA-II, which was originally used to approximate the preferred Pareto-optimal solutions rather than the entire Pareto-optimal front (PF), shows the most competitive performance. In contrast, the performance of MOEA/D and NSGA-III deteriorate significantly with the massively growing number of objectives.

The remainder of this paper is started by a detailed description of the experimental settings of our comparative study in Sect. 2. Then, Sect. 3 presents the discussions of the experimental results. At last, Sect. 4 summarizes the main findings and provide an outlook of possible future directions.

2 Experimental Settings

2.1 Benchmark Problems

As a first study, here we use DTLZ1 to DTLZ4 from the widely used DTLZ suite [11] to form the benchmark set. For each benchmark problem, we consider the 25-, 50-, 75- and 100-objective cases separately. As for DTLZ1, the number of decision variables is set as $n = m + 4$; and for DTLZ2 to DTLZ4, it is set as $n = m + 9$, where m is the number of objectives.

[1] Also known as decomposition-based method, but here we use the terminology reference point based method without loss of generality.

2.2 Performance Metrics

To have a quantitative comparison of different algorithms, we use the following two indicators as the performance metrics.

- **Convergence Measure (CM):** As discussed in [11], the objective functions of a Pareto-optimal solution $\mathbf{x}^*$ satisfy: $\sum_{i=1}^{m} f_i(\mathbf{x}^*) = 0.5$ for DTLZ1 and $\sum_{i=1}^{m} f_i^2(\mathbf{x}^*) = 1.0$ for DTLZ2 to DTLZ4, respectively. Let S be the set of solutions obtained by an EMO algorithm, and the CM is calculated as:

$$CM(S) = \begin{cases} \frac{\sum_{\mathbf{x}\in S} | \sum_{i=1}^{m} f_i(\mathbf{x}) - 0.5|}{|S|}, & \text{DTLZ1} \\ \frac{\sum_{\mathbf{x}\in S} | \sum_{i=1}^{m} f_i^2(\mathbf{x}) - 1.0|}{|S|}, & \text{DTLZ2 to DTLZ4} \end{cases} \tag{1}$$

  where $|S|$ is the cardinality of S.
- **Inverted Generational Distance (IGD)** [5]**:** Let P^* be a set of points uniformly sampled along the PF, and the IGD value of S is calculated as:

$$\text{IGD}(S, P^*) = \frac{\sum_{\mathbf{x}^*\in P^*} dist(\mathbf{x}^*, S)}{|P^*|} \tag{2}$$

  where $dist(\mathbf{x}^*, S)$ is the Euclidean distance between the point $\mathbf{x}^* \in P^*$ and its nearest neighbor of S in the objective space. We use the method developed in [14] to get the Pareto-optimal samples that form P^*.

The CM metric only evaluates the convergence property while the IGD metric can evaluate both the convergence and diversity simultaneously. The lower are the CM and IGD metric values, the better is the quality of a solution set for approximating the PF. Each algorithm is independently run 31 times, and we use the Wilcoxon's rank sum test at a 5% significance level to validate the significance of a better result.

2.3 Examined Reference Point Based EMO Algorithms

This paper considers the following three algorithms as the representatives.

- **MOEA/D** [22]**:** Its basic idea is to decompose a MOP into several scalarizing subproblems and optimizes them in a collaborative manner. To leverage the similarity information among neighboring subproblems, the mating selection and population update have a large chance to take place within the neighborhood. The offspring solution can replace its parent only when it has a better scalarizing function value for the corresponding subproblem. In particular, we consider the penalty-based boundary intersection (PBI) as the scalarizing function [22] in view of its reported superior performance for many-objective optimization [9]. Formally, the PBI function is defined as:

$$\begin{aligned} &\text{minimize } g^{PBI}(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = d_1 + \theta d_2, \\ &\text{subject to } \mathbf{x} \in \Omega, \end{aligned} \tag{3}$$

where Ω is the feasible region, $\mathbf{w}$ is a priori defined weight vector, $\mathbf{z}^*$ is the ideal objective vector, $\theta > 0$ is a user-defined penalty parameter (here we set $\theta = 5.0$), $d_1 = \frac{\|(\mathbf{F}(\mathbf{x})-\mathbf{z}^*)^T\mathbf{w}\|}{\|\mathbf{w}\|}$, $d_2 = \|\mathbf{F}(\mathbf{x})-(\mathbf{z}^*+d_1\mathbf{w})\|$ and $\|\cdot\|$ is the ℓ^2-norm.

- **NSGA-III** [9]**:** Its subproblem is defined to achieve the local optimum, in terms of convergence and diversity, of a subregion specified by the corresponding weight vector. More specifically, each solution is associated with its closest weight vector according to the perpendicular distance. Afterwards, the solutions associated with a particular weight vector form a niche. The survival of a solution is determined by the Pareto dominance relationship and the crowdedness of a niche. In particular, the ones associated with a less crowded niche have a larger chance to survive to the next generation.
- **R-NSGA-II** [10]**:** It was originally proposed to consider the decision maker's (DM's) preference information into the search process and to approximate the region of interest (ROI) rather than the entire PF. Specifically, the DM elicits his/her preference information as one or multiple aspiration level vectors which can be regarded as a discrete approximation of the PF. During the selection procedure, solutions closer to the given aspiration level vector have a larger chance to survive to the next generation. To maintain the population diversity, R-NSGA-II employs a ϵ-clearing idea to control the spread of those selected solutions within the ROI. If we set the aspiration level vectors the same as the weight vectors used in MOEA/D and NSGA-III, we can expect R-NSGA-II to approximate the entire PF as well.

2.4 Parameter Settings

All EMO algorithms use the simulated binary crossover (SBX) and polynomial mutation [8] for offspring reproduction. As suggested in [9], the distribution indices for SBX and polynomial mutation are set to $\mu_c = 30$ and $\mu_m = 30$, respectively. The crossover probability is set to $p_c = 0.9$ and the mutation probability is set to $p_m = \frac{1}{n}$. As suggested in [22], the neighborhood size is set as 20 in MOEA/D, and the probability to select within the neighborhood is set as 0.9. In addition, the maximum number of replacement of an offspring is set as 2. As for R-NSGA-II, the ϵ-clearing parameter is set as $\epsilon = 0.01$ as suggested in [10]. The stopping criterion for each algorithm is the predefined number of function evaluations (FEs). The settings of population size and the maximum number of FEs for different benchmark problems are given in Table 1. Note that the population size of R-NSGA-II is set twice as that of MOEA/D and NSGA-III. This is because R-NSGA-II was originally proposed to approximate the ROI where each aspiration level vector is supposed to hold more than one solution.

Note that the selection mechanism of a reference point based method mainly relies on a predefined set of weight vectors. They not only guide the search direction, and their distribution also determines the population diversity to a certain extent. The Das and Dennis's method [6], which samples $N = \binom{H+m-1}{m-1}$ uniformly distributed weight vectors with a uniform space $\Delta = \frac{1}{H}$ ($H > 0$ is the number of divisions considered along each objective coordinate) from a canonical simplex, is the most popular one for weight vector generation. Although

Table 1. Settings of population size and maximum number of FEs

m	MOEA/D	NSGA-III	R-NSGA-II	Problem	♯ of FEs	Problem	♯ of FEs
25	125	128	256	DTLZ1,3,4	250,000	DTLZ2	200,000
50	250	252	500		750,000		600,000
75	375	376	752		1,500,000		1,125,000
100	500	500	1,000		2,500,000		2,000,000

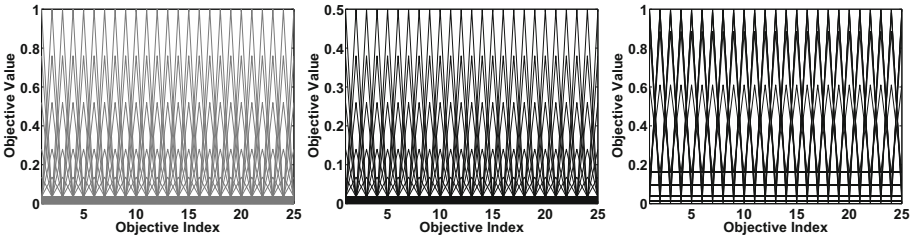
this method works well in the two- or three-objective case, it becomes impractical when the number of objectives becomes large. As discussed in [14], in order to have intermediate weight vectors within each edge of the simplex, we should set $H \geq m$. However, even for a 7-objective case, $H = 7$ will results in $\binom{7+7-1}{7-1} = 1,716$ weight vectors. Such large amount of weight vectors obviously aggregates the computational burden of an EMO algorithm. On the other hand, if we simply set $H < m$, all weight vectors will sparsely distribute along the boundary of the simplex. This is apparently harmful to the population diversity. To attack this issue, especially when encountering a massively large number of objectives, the two-layer weight vector generation method suggested in [14] is a viable resolution. Its basic idea is to make the boundary weight vectors contract inward the simplex by a coordinate transformation. Though it is originally suggested for generating two layers of weight vectors, it can be generalized to a multiple-layer case. Suppose that we need to generate σ ($\sigma \geq 2$) layers of weight vectors. First of all, σ layers of weight vectors (denoted as $W^k = \{\mathbf{w}_1^k, \cdots, \mathbf{w}_{\ell_k}^k\}$, where $k \in \{1, \cdots, \sigma\}$ and ℓ_k is the number of weight vectors in the k-th layer) are initialized according to the Das and Dennis's method, with appropriate H settings. Afterwards, the coordinates of weight vectors in the inside layer (i.e., except the first layer) are shrunk by a coordinate transformation. Specifically, as for a weight vector in the k-th layer, denoted as $\mathbf{w}^k = (w_1^k, \cdots, w_m^k)$, where $k \in \{2, \cdots, t\}$, its j-th component is re-evaluated as:

$$w_j^k = \frac{1-\tau}{m} + \tau_k \times w_j^k \tag{4}$$

where $j \in \{1, \cdots, m\}$ and $\tau_k \in [0, 1]$ is the shrinkage factor for the k-th layer. All σ layers of weight vectors combine to form the final weight vector set W.

Note that one of the major assumptions here is that the uniform distribution of weight vectors can result in a uniform distribution of the obtained solutions. But unfortunately, this might not always be guaranteed in a high-dimensional space, especially when encountering a massively large number of objectives. Let us consider a 25-objective case, where weight vectors are generated by a five-layer weight vector generation method as shown in Fig. 1(a). In particular, the shrinkage factors are $\tau = \{1.0, 0.75, 0.5, 0.25, 0.1\}$ for different layers, respectively. And for each layer, we set $H = 1$. Correspondingly, we calculate the Pareto-optimal samples on the PFs of DTLZ1 and DTLZ2 by using the method introduced in [14]. From the results shown in Fig. 1(b) and (c) we find that,

although different layers of weight vectors have a uniform scale on the simplex, only the Pareto-optimal samples of DTLZ1, which has a linear PF shape, have a uniform spread over the PF; as for DTLZ2, which has a non-linear PF shape, the Pareto-optimal samples on the first three layers crowd around the boundary of the PF. This means that a linear setting of shrinkage factors, which results in a uniformly scaled weight vector layers, can lead an algorithm to search for a set of solutions having a biased distribution at the end. Obviously, this is harmful to the population diversity. Even worse, it aggravates with the increase of the number of objectives. To alleviate this side effect, here we suggest setting the shrinkage factors in a non-linear manner, according to Proposition 1.



(a) Distribution of weight vectors. (b) Pareto-optimal samples on DTLZ1. (c) Pareto-optimal samples on DTLZ2.

Fig. 1. Linear τ setting, i.e., $\tau = \{1.0, 0.75, 0.5, 0.25, 0.1\}$, and their corresponding Pareto-optimal samples on PFs of 25-objective DTLZ1 (linear PF) and DTLZ2 (non-linear PF), respectively.

Proposition 1. *For DTLZ2 to DTLZ4, suppose the expected objective value is f, the appropriate shrinkage factor should be set as the positive τ:*

$$\tau = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{5}$$

where $a = (f^2 - 1)m^2 - (f^2 - 2)m - 3$, $b = -2(m - 1)$ *and* $c = f^2 m - 1$.

Proof. Let us use a simple example to prove this proposition. Suppose an extreme weight vector $\mathbf{w} = (w_1, \cdots, w_m)^T$ is set as: $w_i = 1.0$ and $w_j = 0.0$ where $j \in \{1, \cdots, m\}$ and $j \neq i$. By using the weight vector transformation method introduced in Eq. (4), we can have the corresponding transformed weight vector as:

$$w_i = \frac{1 - \tau}{m} + \tau \times w_i, w_j = \frac{1 - \tau}{m}, \forall j \in \{1, \cdots, m\}, j \neq i \tag{6}$$

According to [14], the i-th objective value on the PF of DTLZ2 is calculated as:

$$\begin{aligned} f_i(\mathbf{x}) &= \frac{w_i}{\sqrt{\sum_{k=1}^{m} w_k^2}} \\ &= \frac{\frac{1-\tau}{m} + \tau \times w_i}{\sqrt{\sum_{k=1}^{m-1}(\frac{1-\tau}{m})^2 + (\frac{1-\tau}{m} + \tau \times w_i)^2}} \\ &= \frac{\frac{1-\tau}{m} + \tau \times w_i}{\sqrt{(m-1)(\frac{1-\tau}{m})^2 + (\frac{1-\tau}{m} + \tau \times w_i)^2}} \end{aligned} \tag{7}$$

Let $t = f_i(\mathbf{x})^2$, we can have:

$$\begin{aligned} \frac{(\frac{1-\tau}{m} + \tau \times w_i)^2}{\sqrt{(m-1)(\frac{1-\tau}{m})^2 + (\frac{1-\tau}{m} + \tau \times w_i)^2}} &= t \\ \implies (t-1)(\frac{1-\tau}{m} + \tau)^2 + t(m-1)(\frac{1-\tau}{m})^2 &= 0 \\ \implies [(t-1)m^2 - (t-2)m - 3]\tau^2 - 2(m-1)\tau + (tm-1) &= 0 \end{aligned} \tag{8}$$

Let $a = [(t-1)m^2 - (t-2)m - 3], b = 2(m-1), c = (tm-1)$, the above equation thus can be treated as a quadratic equation with τ as the unknown. Accordingly, this equation can be solved as:

$$\tau = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{9}$$

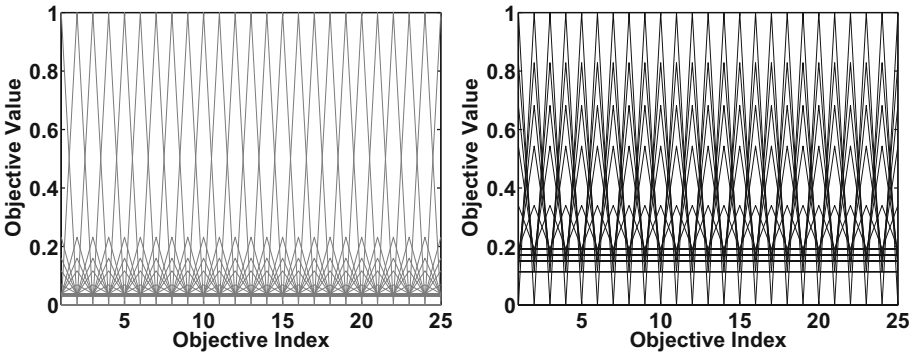
Obviously, the appropriate τ for the weight vector transformation purpose should be a positive number, i.e., the positive solution of Eq. (9).

Based on Proposition 1, we change the shrinkage factor setting of the example shown in Fig. 1 as $\tau = \{1.0, 0.2, 0.125, 0.08, 0.03\}$. As shown in Fig. 2, by using this τ setting, we can have a better spread of Pareto-optimal samples than using the linearly scaled τ setting. In our experiments, we use the 5-layer weight vector generation method suggested in this subsection to generate initial weight vectors. In particular, we set $\tau = \{1.0, 0.75, 0.5, 0.25, 0.1\}$ for the DTLZ1 problem and $\tau = \{1.0, 0.2, 0.125, 0.08, 0.03\}$ for the DTLZ2 to DTLZ4 problems.

3 Experimental Results

3.1 Comparison Results of MOEA/D, NSGA-III and R-NSGA-II

The CM and IGD metric values obtained by different algorithms are presented in Table 2. The best metric value is highlighted in bold face with a gray background. In Fig. 3, we plot the trajectories of the mean metric values obtained by different algorithms versus different number of objectives. To have a visual comparison,



(a) Distribution of weight vectors. (b) Pareto-optimal samples on DTLZ2.

Fig. 2. Non-linear τ setting, i.e., $\tau = \{1.0, 0.2, 0.125, 0.08, 0.03\}$, and their corresponding Pareto-optimal samples on the PF of 25-objective DTLZ2.

we show the parallel coordinate plot of the population holding the best IGD value for each algorithm[2].

From the experimental results, in terms of the IGD metric values and the parallel coordinate plots shown in the supplementary file, we find that R-NSGA-II is the best candidate on all benchmark problem instances. In particular, the solutions found by R-NSGA-II almost converge to the PF, and they well approximate the expected points on the PF. In contrast, the performance of NSGA-III is not satisfactory. Its performance deteriorates significantly with the growing number of objectives. For NSGA-III, the most direct effect from the curse of dimensionality is the convergence where NSGA-III can hardly drive the solutions fully converge to the PF on problems with the multi-modal property, e.g., DTLZ1 and DTLZ3. Although DTLZ4 does not have local PFs in its search space, its parametric mapping, which causes a biased density of solutions, impairs the selection pressure of NSGA-III and thus makes the solutions be drifted. As for MOEA/D, we observe a very decent performance on the CM metric. However, we also notice that the IGD metric values obtained by MOEA/D are not very competitive. From the parallel coordinate plots shown in the supplementary file, we find that although the solutions obtained by MOEA/D converge to the PF, they crowd on a narrow region. This means that MOEA/D fails to approximate the entire PF and thus explains its poor IGD metric values. It is interesting to note that the CM metric values obtained by MOEA/D are improved with the growth of the dimensionality, while their IGD metric values deteriorate accordingly. This might imply that diversity preservation becomes more different by increasing the number of objectives. In this case, most of the computational budgets in MOEA/D have been devoted to several selected weight vectors, thus results in a biased distribution in a high-dimensional space.

[2] Due to the page limit, the parallel coordinate plots are put in the supplementary file, which can be found in https://coda-group.github.io/publications/suppEMO.pdf.

Table 2. Comparison results on the CM and IGD metric

		CM				
Problem	m	MOEA/D	NSGA-III	R-NSGA-II	Generative Method	s
DTLZ1	25	**4.634E-3(2.01E-3)**	8.775E-2(1.52E-1)	2.768E-2(9.88E-3)	3.807E+1(1.70E+0)	†
	50	**3.877E-3(1.82E-3)**	2.594E+0(5.01E+0)	3.781E-2(4.36E-3)	3.871E+1(1.03E+0)	†
	75	**1.848E-3(5.23E-4)**	2.529E+1(9.89E+0)	6.762E-2(1.86E-2)	3.891E+1(1.68E+0)	†
	100	**1.612E-3(8.15E-4)**	2.985E+1(8.50E+0)	1.259E-1(1.74E-2)	3.907E+1(1.12E+0)	†
DTLZ2	25	4.425E-3(9.51E-4)	5.719E-2(3.65E-2)	1.262E-3(3.26E-4)	**4.613E-5(4.63E-5)**	†
	50	3.046E-3(3.30E-4)	5.497E-2(1.38E-2)	7.431E-4(1.34E-4)	**1.599E-3(1.19E-3)**	†
	75	2.722E-3(5.68E-4)	6.992E-2(2.71E-2)	5.566E-4(6.01E-5)	**2.058E-3(1.49E-3)**	†
	100	1.548E-3(6.60E-4)	1.191E-1(3.55E-2)	3.918E-4(6.01E-5)	**9.883E-4(3.56E-4)**	†
DTLZ3	25	2.478E-2(2.32E-2)	1.573E+3(1.79E+3)	**3.068E-3(2.96E-3)**	1.065E+4(5.06E+2)	†
	50	1.775E-2(1.89E-2)	7.793E+3(1.09E+3)	**4.667E-4(3.71E-4)**	1.038E+4(2.53E+2)	†
	75	4.263E-3(8.83E-3)	2.162E+4(1.40E+4)	**1.577E-4(5.21E-5)**	1.051E+4(1.29E+2)	†
	100	**6.794E-5(3.67E-5)**	3.292E+4(1.43E+4)	8.730E-5(2.78E-5)	1.059E+4(2.08E+2)	†
DTLZ4	25	2.974E-3(4.26E-3)	7.520E+1(1.18E+2)	1.068E-3(1.77E-4)	**1.074E-5(2.28E-5)**	†
	50	**9.323E-4(8.35E-4)**	9.869E+2(6.54E+2)	6.911E-4(7.58E-5)	1.011E-3(1.07E-3)	†
	75	**2.909E-4(8.20E-5)**	3.919E+3(2.52E+3)	5.144E-4(7.93E-5)	6.689E-4(4.05E-4)	†
	100	**2.617E-4(1.36E-4)**	2.798E+3(1.23E+3)	3.317E-4(7.02E-5)	6.086E-4(2.41E-4)	†
		IGD				
DTLZ1	25	1.830E-1(9.51E-4)	1.514E-1(2.33E-2)	**1.059E-1(4.41E-3)**	2.460E-1(8.35E-3)	†
	50	1.848E-1(1.72E-3)	2.335E-1(6.07E-2)	**8.388E-2(1.00E-3)**	2.588E-1(3.26E-3)	†
	75	1.848E-1(8.79E-4)	2.839E-1(9.27E-3)	**7.981E-2(3.22E-3)**	2.708E-1(1.50E-3)	†
	100	1.846E-1(4.72E-4)	2.643E-1(1.20E-2)	**7.585E-2(7.28E-3)**	2.731E-1(2.76E-3)	†
DTLZ2	25	2.110E-1(7.58E-3)	2.643E-1(1.20E-2)	1.391E-1(3.04E-3)	**2.368E-2(5.22E-3)**	†
	50	2.557E-1(4.47E-3)	2.207E-1(4.94E-2)	**1.769E-1(1.61E-3)**	4.934E-1(1.55E-2)	†
	75	2.654E-1(2.67E-2)	2.247E-1(2.57E-2)	**1.986E-1(2.24E-3)**	8.294E-1(1.80E-2)	†
	100	3.993E-1(2.95E-1)	2.374E-1(2.97E-2)	**2.108E-1(1.21E-3)**	9.565E-1(1.29E-2)	†
DTLZ3	25	6.798E-1(5.27E-1)	2.974E+0(1.30E+0)	**1.405E-1(2.74E-3)**	6.612E-1(9.61E-2)	†
	50	7.958E-1(5.60E-1)	1.695E+0(1.01E+0)	**1.795E-1(5.74E-3)**	8.573E-1(4.81E-2)	†
	75	1.136E+0(4.65E-1)	1.159E+0(6.99E-2)	**1.978E-1(3.57E-3)**	1.048E+0(2.54E-2)	†
	100	1.349E+0(3.74E-2)	1.134E+0(5.06E-2)	**2.118E-1(5.77E-3)**	1.139E+0(1.04E-2)	†
DTLZ4	25	5.966E-1(8.10E-2)	7.935E-1(3.13E-2)	**1.358E-1(2.37E-3)**	2.013E-1(1.20E-2)	†
	50	6.047E-1(7.11E-2)	8.596E-1(2.58E-2)	**1.731E-1(2.13E-3)**	3.911E-1(7.80E-3)	†
	75	6.058E-1(7.41E-2)	8.775E-1(1.91E-2)	**1.926E-1(3.70E-3)**	7.414E-1(1.30E-2)	†
	100	6.081E-1(6.99E-2)	8.744E-1(1.24E-2)	**2.061E-1(3.54E-3)**	8.871E-1(1.42E-2)	†

Wilcoxon's rank sum test at a 5% significance level is performed between the best metric value and others. † denotes the best mean metric value is significantly better than all other peers.

3.2 Further Discussions

In principle, the ultimate goal of these reference point based EMO algorithms is to find the appropriate solutions for each weight vector. Putting the algorithmic implementations aside, the key differences among them lie in their distance measures. To facilitate the illustration, let us consider the examples shown in Fig. 4. In particular, we connect a weight vector and the origin to form a reference line.

- MOEA/D uses an aggregated distance measure which adds the perpendicular distance between a solution and the reference line to the distance between the ideal point and the projection of a solution onto the reference line.
- NSGA-III only uses the perpendicular distance between a solution and the reference line as the criterion in its niching process.
- R-NSGA-II uses the direct Euclidean distance between a solution and the DM specified aspiration level vector, i.e., the weight vector, as the secondary selection criterion additional to the Pareto dominance.

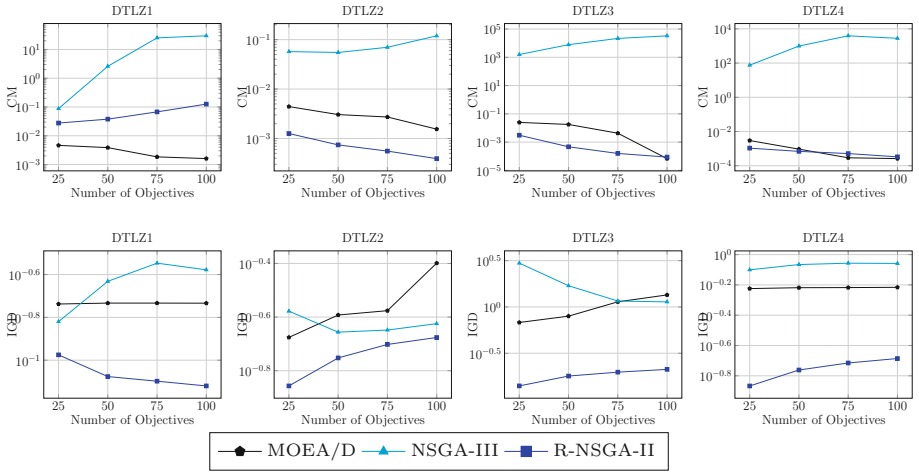
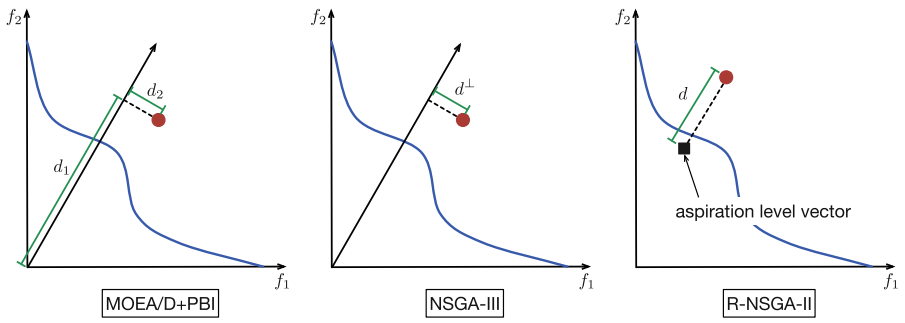


Fig. 3. Trajectories of CM and IGD values on different problems.

The experimental results discussed in Sect. 3.1 provides us an impression that the reference point based methods can be viable for handling MOPs with a massively large number of objectives given the use of a proper distance measure. The superior performance obtained by R-NSGA-II demonstrate that the direct Euclidean distance is the best choice, which might be unexpected at the first glance. The incorporation of the perpendicular distance between a solution and a reference line may help maintain the population diversity, but it may also impair the selection pressure towards the PF.

From another perspective, each weight vector used in R-NSGA-II plays as the DM supplied preference information relating to a particular region of the PF, i.e., the ROI. Accordingly, the use of a set of weight vectors not only implies the prior assumption of the geometrical characteristics of the PF, but also can be



(a) Aggregated distances. (b) Perpendicular distance. (c) Direct Euclidean distance.

Fig. 4. Comparisons of different distance measures.

regarded as its discrete approximation. To a certain extent, the approximation of each ROI plays as a reduction of the originally huge objective space; while R-NSGA-II uses a population-based technique to approximate a set of a priori defined ROIs in a parallel and collaborative manner.

3.3 Comparison with the Classical Generative Method

From the experimental results discussed in Sect. 3.1, we find that the reference point based EMO methods, in particular R-NSGA-II, are capable of handling MOPs with a massively large number of objectives. As discussed in Sect. 3.2, the reference point based EMO methods use a set of predefined weight vectors, either supplied by the DM or systematically constructed, to guide the search process and to approximate a set of Pareto-optimal solutions in a single simulation run. However, the classical generative multi-criterion optimization methods [18] repeatedly solve a parameterized single-objective problem might achieve the same effect. Due to lack of parallelism, some studies [9] have reported that the classical generative methods might not be as efficient as the EMO methods. One may ponder whether a classical generative method can have a similar or even better performance than the EMO methods when tackling MOPs with a massively large number of objectives? For this purpose, we formulate a scalarized single-objective optimization problem for each weight vector in the following form:

$$\begin{aligned} &\text{minimize } \mathrm{ASF}(\mathbf{x}|\mathbf{z}^*, \mathbf{w}) = \max_{1\le j\le m} \left(\frac{f_i(\mathbf{x})-z_i^*}{w_i}\right), \\ &\text{subject to } \mathbf{x} \in \Omega, \end{aligned} \tag{10}$$

where $\mathbf{z}^*$ is an ideal objective vector. In this paper, we set $\mathbf{z}^*$ to be the origin for simplicity. As discussed in some recent studies [9,16,17], the optimal solution of the above optimization problem is the intersecting point of the reference direction of the corresponding weight vector and the PF. To make a fair comparison, for each weight vector, we allocate a maximum of $T = FE_{\max}/N$ (where $FE_{\max}$ is the total number of FEs suggested in Table 1 and N is the population size used by MOEA/D as suggested in Table 1, i.e., the number of weight vectors) FEs to its corresponding optimization procedure. In particular, each optimization procedure is run by the fmincon routine of MATLAB, where a random solution is used for initialization.

The classical generative method is run 31 independent times with different random seeds and the comparative results of the CM and IGD metric values are presented in Table 2 as well. In the supplementary file, we also plot the parallel coordinate plots of solutions which obtain the best IGD value in the corresponding benchmark problem instance. Due to the existence of many local PFs, the classical generative method cannot find any Pareto-optimal solution within the allocated number of FEs. Its poor convergence performance is also reflected by the large CM metric values shown in Table 2. It is also interesting to note that the differences of IGD values obtained by NSGA-III and the classical generative method become small with the growth of dimensionality. Comparing the parallel

coordinate plots of NSGA-III and the classical generative method, we find that both methods cannot find any meaningful solution on 50-, 75- and 100-objective DTLZ1 instances. As for the DTLZ2 problem, the classical generative method obtains the best CM metric value on all 25- to 100-objective scenarios. As shown in the parallel coordinate plots, the classical generative method almost obtains a perfect PF approximation on the 25-objective DTLZ2 problem instance. Accordingly, its IGD value is the best comparing to the other three reference point based EMO algorithms. However, the performance of the classical generative method deteriorates significantly with the growth of dimensionality. Although the solutions obtained by the classical generative method almost converge to the PF on the 50-, 75- and 100-objective scenarios, they all bias towards a particular region. This is reflected by its poor IGD metric values. Similar to the DTLZ1 problem, DTLZ3 is also featured by multi-modality. Solutions found by the classical generative method are obviously far away from the PF. However, they seem to have a better spread than those obtained by NSGA-III. Accordingly, the IGD values obtained by the classical generative method are better than NSGA-III. DTLZ4 is featured by its biased distribution. The classical generative method seems to have a relatively acceptable approximation to the PF on the 25- and 50-objective scenarios. This is reflected by its second best IGD metric values. However, with the growth of dimensionality, solutions found by the classical generative method have a obvious bias towards certain objectives.

From the experimental results discussed in this section, we find that the classical generative method is able to have a comparable or even better performance than the reference point based EMO methods when tackling MOPs with a massively large number of objectives without many local optima. However, with the growth of dimensionality and when the problem becomes complicated, the classical generative method can hardly find any meaningful solution as well.

3.4 Multiple-Solution Strategy

All these reference point based EMO algorithms give a high survival rate to the solutions close to the corresponding weight vector. As discussed in Sect. 3.2, one of their major differences is the method for measuring the distance between a solution and a weight vector. Besides, in R-NSGA-II, each weight vector can hold two or more solutions at once, whereas in MOEA/D and NSGA-III, each weight vector is only allowed to accommodate one solution. One may ponder whether the performance of MOEA/D and NSGA-III can be improved in case each weight vector is allowed to hold more than one solution? Bearing this consideration in mind, we make some modifications on the selection mechanisms of MOEA/D and NSGA-III. More specifically, for MOEA/D, each weight vector is assigned two solutions that hold the current best aggregation function values. In addition, the assigned solutions of a particular weight vector have an equal opportunity to participate the mating selection and the update procedure. For NSGA-III, the niche preservation operation is performed in case the niche count of a particular weight vector is greater than two.

Table 3. Comparison results on the CM and IGD metrics

Problem	m	MOEA/D$^{\Delta}$	MOEA/D	NSGA-III$^{\Delta}$	NSGA-III	R-NSGA-II	s
				CM			
DTLZ1	25	**2.675E-3(1.18E-3)**	4.634E-3(2.01E-3)	3.568E-3(1.81E-3)	8.775E-2(1.52E-1)	2.768E-2(9.88E-3)	†
	50	**8.160E-4(4.99E-4)**	3.877E-3(1.82E-3)	1.699E-2(2.73E-2)	2.594E+0(5.01E+0)	3.781E-2(4.36E-3)	†
	75	**6.541E-4(4.07E-4)**	1.848E-3(5.23E-4)	6.985E-2(3.75E-2)	2.529E+1(9.89E+0)	6.762E-2(1.86E-2)	†
	100	**5.044E-4(4.44E-4)**	1.612E-3(8.15E-4)	6.985E-2(3.75E-2)	2.985E+1(8.50E+0)	1.259E-1(1.74E-2)	†
DTLZ2	25	2.952E-3(7.58E-4)	4.425E-3(9.51E-4)	2.509E-2(1.33E-3)	5.719E-2(3.65E-2)	**1.262E-3(3.26E-4)**	†
	50	2.133E-3(4.23E-4)	3.046E-3(3.30E-4)	1.557E-2(9.26E-3)	5.497E-2(1.38E-2)	**7.431E-4(1.34E-4)**	†
	75	1.932E-3(3.59E-4)	2.722E-3(5.68E-4)	1.154E-2(2.92E-3)	6.992E-2(2.71E-2)	**5.566E-4(6.01E-5)**	†
	100	1.527E-3(2.95E-4)	1.548E-3(6.60E-4)	3.858E-2(2.93E-3)	1.191E-1(3.55E-2)	**3.918E-4(6.01E-5)**	†
DTLZ3	25	2.335E-2(8.86E-3)	2.478E-2(2.32E-2)	4.796E-1(2.84E-1)	1.573E+3(1.79E+3)	**3.068E-3(2.96E-3)**	†
	50	1.511E-2(1.71E-2)	1.775E-2(1.89E-2)	3.412E-1(1.10E-1)	7.793E+3(1.09E+3)	**4.667E-4(3.71E-4)**	†
	75	1.790E-3(5.62E-3)	4.263E-3(8.83E-3)	3.831E-1(1.10E-1)	2.162E+4(1.40E+4)	**1.577E-4(5.21E-5)**	†
	100	**3.508E-6(4.20E-6)**	6.794E-5(3.67E-5)	3.998E-1(1.95E-1)	3.292E+4(1.43E+4)	8.730E-5(2.78E-5)	†
DTLZ4	25	**6.654E-6(1.97E-5)**	2.974E-3(4.26E-3)	1.46E-2(4.11E-3)	7.520E+1(1.18E+2)	1.068E-3(1.77E-4)	†
	50	**9.553E-7(6.82E-7)**	9.323E-4(8.35E-4)	1.776E-2(4.10E-3)	9.869E+2(6.54E+2)	6.911E-4(7.58E-5)	†
	75	**7.167E-7(4.48E-7)**	2.909E-4(8.20E-5)	1.802E-2(4.02E-3)	3.919E+3(2.52E+3)	5.144E-4(7.93E-5)	†
	100	**5.007E-7(4.50E-7)**	2.617E-4(1.36E-4)	1.867E-2(3.95E-3)	2.798E+3(1.23E+3)	3.317E-4(7.02E-5)	†
				IGD			
DTLZ1	25	1.166E-1(6.96E-4)	1.830E-1(9.51E-4)	1.204E-1(5.45E-3)	1.514E-1(2.33E-2)	**1.059E-1(4.41E-3)**	†
	50	1.208E-1(6.94E-5)	1.848E-1(1.72E-3)	1.502E-1(1.14E-2)	2.335E-1(6.07E-2)	**8.388E-2(1.00E-3)**	†
	75	1.308E-1(7.51E-4)	1.848E-1(8.79E-4)	2.056E-1(1.47E-2)	2.839E-1(9.27E-3)	**7.981E-2(3.22E-3)**	†
	100	1.414E-1(8.55E-5)	1.846E-1(4.72E-4)	3.587E-1(1.35E-1)	2.643E-1(1.20E-2)	**7.585E-2(7.28E-3)**	†
DTLZ2	25	3.442E-1(6.22E-4)	2.110E-1(7.58E-3)	1.026E+0(3.90E-2)	2.643E-1(1.20E-2)	**1.391E-1(3.04E-3)**	†
	50	4.313E-1(6.62E-4)	2.557E-1(4.47E-3)	1.178E+0(2.30E-2)	2.207E-1(4.94E-2)	**1.769E-1(1.61E-3)**	†
	75	4.611E-1(1.27E-3)	2.654E-1(2.67E-2)	1.225E+0(1.88E-2)	2.247E-1(2.57E-2)	**1.986E-1(2.24E-3)**	†
	100	4.926E-1(2.73E-3)	3.993E-1(2.95E-1)	1.276E+0(1.29E-2)	2.374E-1(2.97E-2)	**2.108E-1(1.21E-3)**	†
DTLZ3	25	3.492E-1(8.21E-4)	6.798E-1(5.27E-1)	8.748E-1(1.45E-1)	2.974E+0(1.30E+0)	**1.405E-1(2.74E-3)**	†
	50	8.674E-1(4.57E-1)	7.958E-1(5.60E-1)	8.528E-1(4.45E-2)	1.695E+0(1.01E+0)	**1.795E-1(5.74E-3)**	†
	75	1.070E+0(4.14E-1)	1.136E+0(4.65E-1)	8.610E-1(3.96E-2)	1.159E+0(6.99E-2)	**1.978E-1(3.57E-3)**	†
	100	1.255E+0(2.63E-1)	1.349E+0(3.74E-2)	8.844E-1(5.03E-2)	1.134E+0(5.06E-2)	**2.118E-1(5.77E-3)**	†
DTLZ4	25	6.909E-1(4.57E-2)	5.966E-1(8.10E-2)	3.327E-1(9.74E-4)	7.935E-1(3.13E-2)	**1.358E-1(2.37E-3)**	†
	50	6.917E-1(2.83E-2)	6.047E-1(7.11E-2)	4.221E-1(1.52E-3)	8.596E-1(2.58E-2)	**1.731E-1(2.13E-3)**	†
	75	6.703E-1(3.32E-2)	6.058E-1(7.41E-2)	4.678E-1(2.32E-3)	8.775E-1(1.91E-2)	**1.926E-1(3.70E-3)**	†
	100	6.850E-1(3.60E-2)	6.081E-1(6.99E-2)	5.398E-1(2.63E-3)	8.744E-1(1.24E-2)	**2.061E-1(3.54E-3)**	†

Wilcoxon's rank sum test at a 5% significance level is performed between the best metric value and others. † denotes the best mean metric value is significantly better than all other peers.

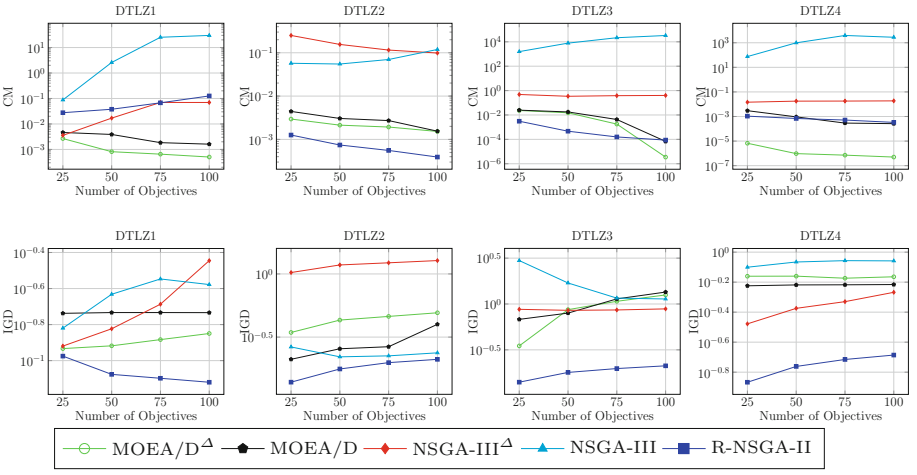


Fig. 5. Trajectories of CM and IGD values on different problems.

The parameters of MOEA/D and NSGA-III are set the same as described in Sect. 2.4, except the population size and the maximum number of FEs. In particular, the population size is doubled, while the maximum number of FEs is reduced by half accordingly. The CM and IGD metric values are presented in Table 3 and Fig. 5. In particular, MOEA/D and NSGA-III with a doubled population size are denoted as MOEA/D$^{\Delta}$ and NSGA-III$^{\Delta}$ respectively. From the experimental results, we clearly observe the performance improvement, in terms of convergence and diversity, after doubling the population size. Since each weight vector is able to hold at least two candidates simultaneously, we can expect an improvement on the population diversity. In the meanwhile, due to the existence of more than one candidate associated with a weight vector, we can also expect an enhanced selection pressure towards the PF within a local niche. From the parallel coordinate plots shown in the supplementary file, we find that the solutions obtained by MOEA/D$^{\Delta}$ and NSGA-III$^{\Delta}$ have a better convergence than those obtained by MOEA/D and NSGA-III. Nevertheless, we also observe that the IGD metric values obtained by R-NSGA-II are still better than MOEA/D$^{\Delta}$ and NSGA-III$^{\Delta}$. This further implies the importance of the use of an appropriate distance measure.

4 Conclusions and Future Directions

In this paper, we conduct a series of experiments that investigate the performance of three selected reference point based EMO methods on MOPs with a massively large number of objectives. From the experimental studies, we find that R-NSGA-II, originally proposed to search for the ROIs, shows very competitive and robust performance when handling problems with a massively large number of objectives. We attribute the success of R-NSGA-II to two aspects: (1) since each weight vector is able to hold more than one solution, a stronger selection pressure can be expected in the local niche; (2) the appropriate distance measure plays an important role in the selection criterion design of the reference point based EMO methods.

We admit that the results observed in this paper might not be conclusive, since we only focus on the reference point based EMO methods. Future research will deepen the insights in the behavior of more types of EMO methods, i.e., the Pareto- and indicator-based EMO methods. However, with a massively large number of objectives, the computational overheads for calculating the Hypervolume metric, which is the most popular metric used in the indicator-based EMO method, might become practically infeasible for the scalability studies. As for the Pareto-based method, we suspect that they might have some troubles to drive the population in a massively high dimensional search space without any direction information provided by the weight vectors. Furthermore, the benchmark problems considered in this paper are relatively less challenging. In our follow-up work, we will test the performance on a wider range of and more complicated benchmark problems. In addition, statistically guided parameter studies will be performed to figure out the suitable parameterizations for different EMO algorithms.

Acknowledgement. This work was partially supported by EPSRC (Grant No. EP/J017515/1).

References

1. Adra, S.F., Fleming, P.J.: Diversity management in evolutionary many-objective optimization. IEEE Trans. Evol. Comput. **15**(2), 183–195 (2011)
2. Auger, A., Bader, J., Brockhoff, D., Zitzler, E.: Hypervolume-based multiobjective optimization: theoretical foundations and practical implications. Theor. Comput. Sci. **425**, 75–103 (2012)
3. Bader, J., Zitzler, E.: HypE: an algorithm for fast hypervolume-based many-objective optimization. Evol. Comput. **19**(1), 45–76 (2011)
4. Beume, N., Naujoks, B., Emmerich, M.T.M.: SMS-EMOA: multiobjective selection based on dominated hypervolume. Eur. J. Oper. Res. **181**(3), 1653–1669 (2007)
5. Coello, C.A.C., Cortés, N.C.: Solving multiobjective optimization problems using an artificial immune system. Genet. Program. Evolvable Mach. **6**(2), 163–190 (2005)
6. Das, I., Dennis, J.E.: Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems. SIAM J. Optim. **8**, 631–657 (1998)
7. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
8. Deb, K.: Multi-objective Optimization Using Evolutionary Algorithms. John Wiley & Sons Inc., New York (2001)
9. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. IEEE Trans. Evol. Comput. **18**(4), 577–601 (2014)
10. Deb, K., Sundar, J., Bhaskara, U., Chaudhuri, S.: Reference point based multi-objective optimization using evolutionary algorithms. Int. J. Comput. Intell. Res. **2**(3), 273–286 (2006)
11. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable test problems for evolutionary multiobjective optimization. In: Abraham, A., Jain, L., Goldberg, R. (eds.) Evol. Multiobjective Optim. Advanced Information and Knowledge Processing, pp. 105–145. Springer, London (2005)
12. Ishibuchi, H., Setoguchi, Y., Masuda, H., Nojima, Y.: Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. IEEE Trans. Evol. Comput. **PP**(99), 1 (2016)
13. Li, B., Li, J., Tang, K., Yao, X.: Many-objective evolutionary algorithms: a survey. ACM Comput. Surv. **48**(1), 13 (2015)
14. Li, K., Deb, K., Zhang, Q., Kwong, S.: An evolutionary many-objective optimization algorithm based on dominance and decomposition. IEEE Trans. Evol. Comput. **19**(5), 694–716 (2015)
15. Li, K., Kwong, S., Cao, J., Li, M., Zheng, J., Shen, R.: Achieving balance between proximity and diversity in multi-objective evolutionary algorithm. Inf. Sci. **182**(1), 220–242 (2012)
16. Li, K., Kwong, S., Zhang, Q., Deb, K.: Interrelationship-based selection for decomposition multiobjective optimization. IEEE Trans. Cybern. **45**(10), 2076–2088 (2015)

17. Li, K., Zhang, Q., Kwong, S., Li, M., Wang, R.: Stable matching based selection in evolutionary multiobjective optimization. IEEE Trans. Evol. Comput. **18**(6), 909–923 (2014)
18. Miettinen, K.: Nonlinear Multiobjective Optimization, vol. 12. Kluwer Academic Publishers, Boston (1999)
19. Purshouse, R.C., Fleming, P.J.: On the evolutionary optimization of many conflicting objectives. IEEE Trans. Evol. Comput. **11**(6), 770–784 (2007)
20. Sato, H., Aguirre, H.E., Tanaka, K.: Self-controlling dominance area of solutions in evolutionary many-objective optimization. In: Deb, K., Bhattacharya, A., Chakraborti, N., Chakroborty, P., Das, S., Dutta, J., Gupta, S.K., Jain, A., Aggarwal, V., Branke, J., Louis, S.J., Tan, K.C. (eds.) SEAL 2010. LNCS, vol. 6457, pp. 455–465. Springer, Heidelberg (2010). doi:10.1007/978-3-642-17298-4_49
21. Wang, H., Jiao, L., Yao, X.: Two_arch2: An improved two-archive algorithm for many-objective optimization. IEEE Trans. Evolutionary Computation **19**(4), 524–541 (2015)
22. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evolutionary Computation **11**, 712–731 (2007)