

Bi-goal evolution for many-objective optimization problems



Miqing Li^a, Shengxiang Yang^{b,*}, Xiaohui Liu^a

^a Department of Computer Science, Brunel University, London UB8 3PH, UK

^b Centre for Computational Intelligence (CCI), School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, UK

ARTICLE INFO

Article history:

Received 22 August 2014

Received in revised form 14 June 2015

Accepted 20 June 2015

Available online 3 July 2015

Keywords:

Evolutionary multi-objective optimization

Many-objective optimization

Proximity

Diversity

Bi-goal evolution

ABSTRACT

This paper presents a meta-objective optimization approach, called Bi-Goal Evolution (BiGE), to deal with multi-objective optimization problems with many objectives. In multi-objective optimization, it is generally observed that 1) the conflict between the proximity and diversity requirements is aggravated with the increase of the number of objectives and 2) the Pareto dominance loses its effectiveness for a high-dimensional space but works well on a low-dimensional space. Inspired by these two observations, BiGE converts a given multi-objective optimization problem into a bi-goal (objective) optimization problem regarding proximity and diversity, and then handles it using the Pareto dominance relation in this bi-goal domain. Implemented with estimation methods of individuals' performance and the classic Pareto nondominated sorting procedure, BiGE divides individuals into different nondominated layers and attempts to put well-converged and well-distributed individuals into the first few layers. From a series of extensive experiments on four groups of well-defined continuous and combinatorial optimization problems with 5, 10 and 15 objectives, BiGE has been found to be very competitive against five state-of-the-art algorithms in balancing proximity and diversity. The proposed approach is the first step towards a new way of addressing many-objective problems as well as indicating several important issues for future development of this type of algorithms.

© 2015 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Real-world problems commonly involve multiple objectives/criteria which are required to be optimized simultaneously. For example, an individual would like to maximize the chance of being healthy and wealthy while still having fun and time for family and friends. A software engineer would be interested in finding the cheapest test suite while achieving full coverage (e.g., statement coverage, branch coverage and decision coverage). When prescribing radiotherapy to a cancer patient, a doctor would have to balance the attack on tumor, potential impact on healthy organs, and the overall condition of the patient. These multi-objective optimization problems (MOPs) can be seen in many fields, including engineering, science, medicine and logistics. They share the same issue of pursuing several objectives at the same time, and have long been regarded as a substantial challenge in artificial intelligence (AI) [73,25].

There have been a variety of approaches for MOPs, including traditional mathematical programming methods, local search techniques, and evolutionary algorithms (EAs). Inspired by biological evolution mechanisms, EAs have been demonstrated to be successful in diverse AI applications [73,10]. For example, an EA-based AI planner, Divide and Evolutionary

* Corresponding author.

E-mail address: syang@dmu.ac.uk (S. Yang).

(DaE) [8], won the Deterministic Temporal Satisficing track during the International Planning Competition (IPC7) at the 21st International Conference on Automated Planning and Scheduling (ICAPS 2011).¹ Recently, DaE has been successfully applied to multi-objective AI planning (called MO-DaE) [58]. MO-DaE, working with a well-known multi-objective EA, i.e., the indicator-based EA (IBEA) [99], has shown clear advantage over the metric-based approach using LPG metric sensitive planner [58].

A key strength of EAs for MOPs is their population-based feature which allows individuals to simultaneously approximate different parts of the Pareto front within a single execution [19,97]. Intuitively, the search process of an EA has two basic goals:

- minimizing the distance of the population to the Pareto front (i.e., proximity) and
- maximizing the distribution of the population along the Pareto front (i.e., diversity).

Since the optimal outcome of an MOP is a set of Pareto optimal solutions, the Pareto dominance relation naturally becomes a criterion to distinguish between solutions. Given two solutions p and q for an MOP, p is said to *Pareto dominate* q , if and only if p is better than q for at least one objective and is not worse for any of the others. The Pareto dominance reflects the weakest assumption about the preferred structure of the decision-maker.

As the primary selection criterion in the evolutionary multi-objective optimization (EMO) area, Pareto dominance is commonly used to evaluate the proximity of solutions. When Pareto dominance fails (e.g., the interested solutions are non-dominated to each other), EMO algorithms often introduce a density-based criterion to maintain diversity of the population. For example, the nondominated sorting genetic algorithm II (NSGA-II) [23] separates individuals in a population into different layers (ranks) by their Pareto dominance relation, and prefers 1) individuals in lower layers and 2) individuals with lower crowding degrees (measured by the *crowding distance* [23]) when they are located in the same layer.

An MOP with more than three objectives is called a many-objective optimization problem. Many-objective optimization is an important but very challenging topic and there has been increasing interest in the use of EAs to tackle many-objective optimization problems [14,16,26,35]. Although Pareto-based algorithms are the most popular approaches, they scale up poorly with the number of objectives [18,48,75]. When dealing with an MOP with many objectives, Pareto dominance often loses its effectiveness to differentiate individuals [57], which makes most individuals in a population become incomparable in terms of proximity (e.g., in NSGA-II most individuals fall into the first layer). Consequently, the density-based selection criterion will play a decisive role in determining the survival of individuals during the evolutionary process, leading to the individuals in the final population distributed widely over the objective space but far from the desired Pareto front [85].

A straightforward way to handle this problem (i.e., the ineffectiveness of Pareto-based algorithms in many-objective optimization) is to modify the Pareto dominance relation. Some interesting attempts include loosening the dominance condition or controlling the dominance angle, such as ϵ -dominance [22,36,61,84], α -dominance [43], ϵ -box dominance [60], and dominance area control [78]. By relaxing the area of an individual dominating, these dominance relations are able to provide sufficient selection pressure towards the Pareto front. However, how to set a proper value of the parameter(s) to determine the relaxation degree is a crucial issue in these methods, needing further studies [62,69,79].

On the other hand, the way of comparing individuals according to their quantitative difference in objectives has been found to be effective in converging towards the Pareto front. Many recent EMO algorithms originate from this motivation, introducing a variety of new criteria to distinguish between individuals, e.g., average ranking [52,70], fuzzy Pareto optimality [37,39], subspace partition [2,51], preference-inspired rank [88,87], grid-based rank [70,92], distance-based rank [32,71,91], and density adjustment strategies [1,66]. These methods provide ample alternatives to deal with many-objective optimization problems, despite some having the risk of leading the population to concentrate in one or several sub-areas of the whole Pareto front [50,67,81,65].

Recently, there has been significant interest in the use of selection criteria that involve both proximity and diversity to solve MOPs. Some such criteria, like the decomposition-based [94] and indicator-based [99] criteria, have been shown to be very promising in many-objective optimization [15,20,41,44,85]. The former uses the idea of single-objective aggregated optimization, decomposing an MOP into a number of scalar subproblems and optimizing them simultaneously. The latter defines an optimization criterion with regard to a specified performance indicator and uses this criterion to guide the search of the population. The indicator *hypervolume* is one of the most popular indicator-based criteria due to its good theoretical and empirical properties [7,13,29,42,101]. Whereas super-polynomial time complexity is required in the calculation of the hypervolume indicator (unless $P = NP$) [11], lots of effort is being made to reduce its computational cost, in terms of both the exact computation [6,12,90] and the approximate estimation [4,14,49]. Nevertheless, balancing proximity and diversity using one single criterion is not an easy task [76,38,69,68], especially for a many-objective optimization problem in which the conflict between the objectives is generally more serious than that in an MOP with two or three objectives [75,1].

In fact, evolving a population towards the optimum as well as diversifying its individuals over the whole Pareto front in many-objective optimization is, by itself, a multi-objective problem. The advance at one aspect usually comes along with the degradation at the other [33,75].

¹ http://www.sigevo.org/wiki/tiki-read_article.php?articleId=1.

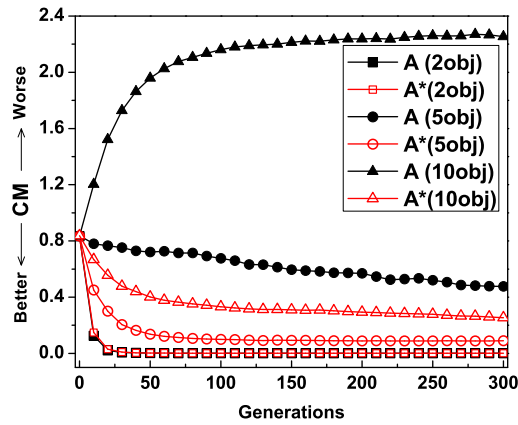


Fig. 1. Evolutionary trajectories of the average convergence metric (CM) for 30 runs of the original NSGA-II (denoted as **A**) and the modified NSGA-II without the diversity maintenance mechanism (denoted as **A***) on DTLZ2.

This paper presents a meta-objective optimization approach, called **Bi-Goal Evolution (BiGE)**, to deal with many-objective optimization problems. Inspired by two observations: 1) the conflict between proximity and diversity requirements is aggravated with the increase of the number of objectives and 2) the Pareto dominance loses its effectiveness for a high-dimensional space but works well on a low-dimensional space, BiGE converts a given many-objective optimization problem into a bi-goal (objective) optimization problem regarding individuals' proximity and crowding degree, and then handles it using Pareto dominance in this bi-goal domain.

The bi-goal evolution is implemented with two specific methods of estimating individuals' performance (i.e., proximity and crowding degree), and also with simple individual comparison strategies in the mating and environmental selection. In the implementation of the crowding degree estimation, a pragmatic approach is developed to prevent adjacent individuals from being assigned similar fitness in the bi-goal domain. In the environmental selection, BiGE, using Pareto nondominated sorting of proximity and crowding degree, attempts to put well-converged and well-distributed individuals into the first few layers such that they could then be chosen first.

It is worth mentioning that meta-objective optimization is not an uncommon approach in the (multi-objective) optimization field. For example, Jones and Jimenez introduced two meta-objectives, the number of unmet goals and closeness to the pairwise comparisons, into the extended goal programming framework [56]. Wang and Cai considered the constraint violation as a meta-objective to solve single-objective constrained optimization problems [89]. Toffolo and Benini viewed diversity as an additional objective and turned an m -objective MOP into an $(m + 1)$ -objective MOP [82]. Ishibuchi et al. considered the hypervolume maximization of a solution set with m reference points as m meta-objectives to optimize a number of solution sets [46]. An interesting difference between BiGE and these meta-objective approaches lies in that these meta-objective approaches typically introduce more objectives into the original optimization problem, while BiGE deals with less objectives via converting a many-objective problem into a bi-objective problem with two meta-objectives.

The rest of this paper is organized as follows. In Section 2, the motivation of BiGE is described. Section 3 is devoted to the presentation of BiGE's framework and its implementation. Section 4 introduces the experimental design. Empirical results of BiGE in comparison with five peer algorithms are shown in Section 5. Further investigation of the proposed algorithm and some discussions are given in Sections 6 and 7, respectively. Section 8 provides some concluding remarks along with pertinent observations.

2. Motivation

An EMO algorithm pursues two basic but often conflicting goals, proximity and diversity. Such conflict has a detrimental impact on the algorithm's optimization process and can be aggravated in many-objective optimization. Fig. 1 gives the comparison trajectories of the proximity results between the original NSGA-II (involving both proximity and diversity maintenance mechanisms) and its modified version in which the diversity maintenance mechanism is removed, on the 2-, 5- and 10-objective DTLZ2 [24]. These results are evaluated by a convergence metric (CM) [21], which calculates the average normalized Euclidean distance from the solution set to the Pareto front.

As can be seen in Fig. 1, the interval between the CM trajectories of the two algorithms becomes more visible with the increase of the number of objectives. This divergence behavior has been first reported in [75]. For the 2-objective problem, both algorithms perform well, with their CM trajectories being virtually overlapping. For the 5-objective problem, the NSGA-II without the diversity maintenance mechanism achieves better CM results than the original NSGA-II during the evolutionary process, which means that diversity maintenance has an unfavorable impact on the proximity of the algorithm. For the 10-objective problem, the diversity maintenance mechanism in NSGA-II even makes the evolving population gradually move away from the Pareto front; the great interval between the two trajectories in Fig. 1 indicates a serious conflict between proximity and diversity obtained.

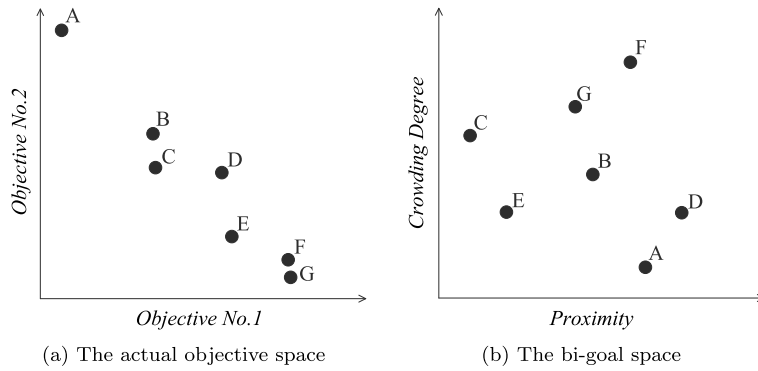


Fig. 2. An illustration of the conversion from the actual objective space to the bi-goal space of proximity and crowding degree on a bi-objective minimization problem.

Algorithm 1 Bi-goal evolution (BiGE).

Require: P (population), N (population size)

```

1:  $P \leftarrow \text{initialize}(P)$ 
2: while termination criterion not fulfilled do
3:    $\text{proximityEstimation}(P)$ 
4:    $\text{crowdingDegreeEstimation}(P)$ 
5:    $P' \leftarrow \text{matingSelection}(P)$ 
6:    $P'' \leftarrow \text{variation}(P')$ 
7:    $P \leftarrow \text{environmentalSelection}(P \cup P'')$ 
8: end while
9: return  $P$ 

```

On the other hand, Pareto dominance, which is popular and effective to distinguish between individuals in 2- or 3-objective MOPs, fails in many-objective optimization. In fact, the portion of any two individuals being comparable in an m -dimensional objective space is $\eta = 1/2^{m-1}$. For a 2- or 3-dimensional space, η is equal to 0.5 or 0.25, respectively, but when m reaches 6, η is already as low as 0.03125. Such exponential decrease of the portion leads to the dramatic decline of Pareto dominance's effectiveness with the number of objectives.

Given the above, it could then be viable to use Pareto dominance to only optimize the two goals (objectives) of proximity and diversity rather than to cope with all the objectives of an MOP. This way, sufficient selection pressure can be provided even in a very high-dimensional space. Bearing this in mind, we propose a bi-goal evolution approach, BiGE, to tackle many-objective optimization problems.

3. Bi-goal evolution (BiGE)

BiGE treats an MOP with many objectives as a bi-goal optimization problem regarding minimizing the proximity of individuals towards the optimal direction and minimizing the crowding degree of individuals in the population. Fig. 2 gives a bi-objective scenario to illustrate the conversion from the actual objective space to the bi-goal space.

As can be seen from Fig. 2, by conversion, some of the nondominated individuals A–G in the objective space become comparable. In the bi-goal space, only three individuals (C, A, and E) are Pareto nondominated (i.e., the best individuals in the population), given that C and A perform best in terms of proximity and crowding degree, respectively, and the performance of E can be regarded as the tradeoff between that of C and A. In contrast, individual F, which performs poorly in both proximity and crowding degree, is dominated by most of the individuals in the population.

Below, we introduce the main procedure of BiGE and its specific implementations.

3.1. Basic procedure of bi-goal evolution

The aim of BiGE is to deal with the ineffectiveness of the Pareto dominance relation in the high-dimensional objective space. BiGE only considers the individuals when they are incomparable on the basis of Pareto dominance in the selection process. Algorithm 1 gives the basic procedure of BiGE. Firstly, N individuals are randomly generated to form an initial population P . Then, the proximity and crowding degree of individuals in the current population are estimated. Next, mating selection is performed to select promising solutions in the bi-goal space for variation. Finally, the environmental selection procedure is implemented to keep a record of the N best solutions with respect to the two goals for survival.

3.2. Proximity estimation

Conversion from an MOP with a number of objectives to a bi-goal problem involves an integration of the objectives. In order to make the integration feasible (i.e., to be able to deal with an MOP with non-commensurable objective functions), in BiGE each objective of individuals is normalized (with respect to its minimum and maximum values in the current population) before estimating their proximity and crowding degree. For convenience, in the description of the proposed algorithm, the objective value of individuals refers to their normalized objective value in the range [0, 1].

BiGE estimates the proximity (denoted as f_{pr}) of an individual p in the population by summing its value in each objective:

$$f_{pr}(p) = \sum_{k=1}^m f^k(p) \quad (1)$$

where $f^k(p)$ denotes the objective value of individual p in the k th objective, and m is the number of objectives. This estimation function is determined by two factors: the number of objectives and the performance in each objective. An individual with good performance in the majority of objectives is likely to obtain a lower (better) f_{pr} value.

It is worth pointing out that the proximity information of an individual with m objectives (i.e., an m -dimensional vector) cannot be completely reflected and represented by the scalar value f_{pr} . The accuracy of the estimation can be influenced by the shape of an MOP's Pareto front. For example, individuals around the knee of the Pareto front often have better estimation result than those far away from the knee even if they are non-dominated to each other. To solve this issue, we introduce the goal of minimizing the crowding degree of individuals in the population. We consider the Pareto dominance relation of the two goals, preferring individuals with a good tradeoff between them.

3.3. Crowding degree estimation

Niching techniques are a kind of popular density estimation methods in the EA field. Bearing the idea of sharing resource in mind, niching techniques can effectively measure the crowding degree of an individual in the population. Here, we consider the following sharing function between two individuals p and q :

$$sh(p, q) = \begin{cases} (1 - \frac{d(p, q)}{r})^2, & \text{if } d(p, q) < r \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where $d(p, q)$ denotes the Euclidean distance between individuals p and q in the objective space, and r is the radius of a niche, determined by the population size N and the number of objectives m of a given MOP:

$$r = \frac{1}{\sqrt[m]{N}} \quad (3)$$

Note that the considered individuals are already normalized according to the range of the current population. Thus, the niche radius here is actually adaptive, varying with the evolutionary population. Using the sharing function in Eq. (2), the crowding degree (denoted as f_{cd}) of an individual p in a population P is defined as follows:

$$f_{cd}(p) = (\sum_{q \in P, q \neq p} sh(p, q))^{1/2} \quad (4)$$

Up to now, the performance of an individual in the population has been reflected by f_{pr} and f_{cd} . However, a problem may arise when applying these two estimation functions in the conversion from the actual objective space into the bi-goal space. Since the performance estimation of an individual depends on its position in comparison with other individuals in the population, the individuals located closely in the objective space may have similar behaviors regarding both proximity and crowding degree, thus also being situated closely in the bi-goal space. For example, similar nondominated individuals **A** and **B** in Fig. 3(a), after conversion, are still located closely and nondominated to each other (shown in Fig. 3(b)). In this case, it is likely that such individuals are preserved or eliminated simultaneously, which may result in congestion in some regions yet vacancy in some other regions.

To overcome this problem, we make a modification to the sharing function in Eq. (2) in order to distinguish between similar individuals. Two individuals will be assigned different sharing function values according to their performance comparison in terms of proximity. Specifically, we introduce a weight parameter (called the sharing discriminator) in the sharing function:

$$sh(p, q) = \begin{cases} (0.5(1 - \frac{d(p, q)}{r}))^2, & \text{if } d(p, q) < r, f_{pr}(p) < f_{pr}(q) \\ (1.5(1 - \frac{d(p, q)}{r}))^2, & \text{if } d(p, q) < r, f_{pr}(p) > f_{pr}(q) \\ rand(), & \text{if } d(p, q) < r, f_{pr}(p) = f_{pr}(q) \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

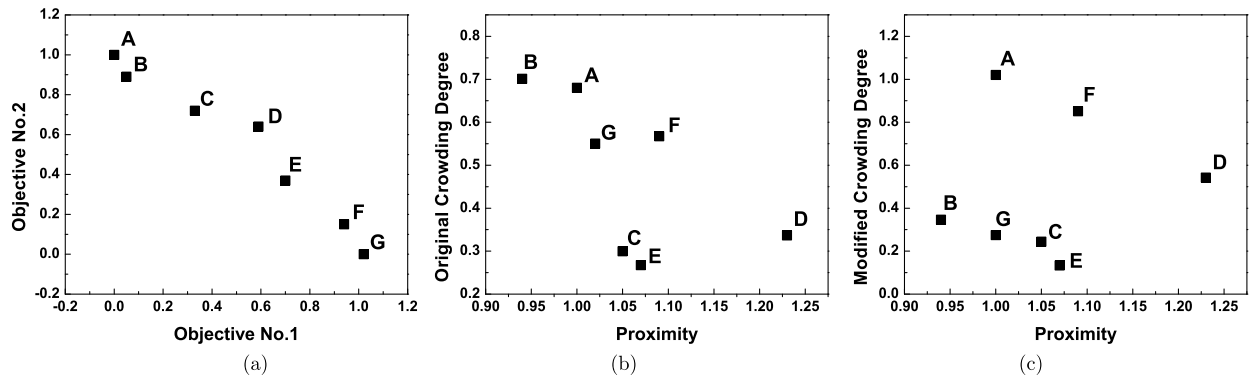


Fig. 3. An illustration of the case that similar individuals in the objective space may be located closely and nondominated to each other in the bi-goal space, and its remedy. (a) The actual objective space; (b) The bi-goal space with respect to the proximity and the original crowding degree; (c) The bi-goal space with respect to the proximity and the modified crowding degree. The numerical values of the individuals in these three spaces are given in Table 1.

Table 1

Individual values in the three spaces for the example of Fig. 3.

	(Objective No. 1, objective No. 2)	(Proximity, original crowding degree)	(Proximity, modified crowding degree)
A	(0.00, 1.00)	(1.00, 0.68031)	(1.00, 1.02047)
B	(0.05, 0.89)	(0.94, 0.70114)	(0.94, 0.34663)
C	(0.33, 0.72)	(1.05, 0.29965)	(1.05, 0.24422)
D	(0.59, 0.64)	(1.23, 0.33658)	(1.23, 0.54256)
E	(0.70, 0.37)	(1.07, 0.26737)	(1.07, 0.13369)
F	(0.94, 0.15)	(1.09, 0.56741)	(1.09, 0.85112)
G	(1.02, 0.00)	(1.02, 0.55022)	(1.02, 0.27511)

where the function $rand()$ means to assign either $sh(p, q) = (0.5(1 - \frac{d(p, q)}{r}))^2$ and $sh(q, p) = (1.5(1 - \frac{d(p, q)}{r}))^2$ or $sh(p, q) = (1.5(1 - \frac{d(p, q)}{r}))^2$ and $sh(q, p) = (0.5(1 - \frac{d(p, q)}{r}))^2$ randomly.

The sharing function now contributes differently to the crowding degree of individuals in the niche. An individual with better proximity than its neighbors will obtain a lower crowding degree. For two individuals which are the sole neighbor to each other in a population, they had the same crowding degree before, but now the better individual (in terms of proximity) will only have half of the original crowding degree and the worse one will have one and a half of the original crowding degree.

In general, this modification enables adjacent individuals to be located distantly. More importantly, it could lead to similar individuals comparable on the basis of the Pareto dominance criterion of the proximity and diversity goals, which is well suited to BiGE. Fig. 3(c) gives an illustration to explain the effect of this modification. As shown, individual A will become dominated by B when evaluated by the modified crowding degree. Table 1 shows the values of individuals in the three spaces for the example of Fig. 3.

3.4. Mating selection

Mating selection, which aims to make a good preparation for exchanging the information of individuals, picks out promising solutions from the current population to form a mating pool. BiGE uses a type of binary tournament selection strategy based on Pareto dominance in the bi-goal domain, as given in Algorithm 2. For two candidates, if they are Pareto-comparable in the two goal functions (e.g., $f_{pr}(p) < f_{pr}(q) \wedge f_{cd}(p) < f_{cd}(q)$), then the better one will be selected; otherwise, the tie will be split randomly. Note that the variation operations (e.g., crossover and mutation) are not fixed in BiGE and can be freely chosen by users. Here, we use the simulated binary crossover (SBX) and polynomial mutation for continuous MOPs and the uniform crossover and bit-flip mutation for combinatorial MOPs.

3.5. Environmental selection

Environmental selection, which aims to obtain a well-approximated and well-distributed new population, chooses the “best” solutions from the previous population and newly created individuals. BiGE implements environmental selection according to individuals’ Pareto dominance relation in the bi-goal domain. Here, we adopt a popular Pareto-based rank strategy in the area: nondominated sorting [34]. Nondominated sorting is an effective method to rank individuals in a low-dimensional space. First, the nondominated individuals in the population are identified as the first layer. Then, the remaining individuals are regarded as the current population, from which nondominated individuals are selected to form the second layer. This process is continued until the entire population is classified into different layers.

Algorithm 2 Tournament selection.**Require:** individuals p, q

```

1: if  $p < q$  in the bi-goal domain then
2:   return  $p$ 
3: else if  $q < p$  in the bi-goal domain then
4:   return  $q$ 
5: else if  $\text{random}(0, 1) < 0.5$  then
6:   return  $p$ 
7: else
8:   return  $q$ 
9: end if

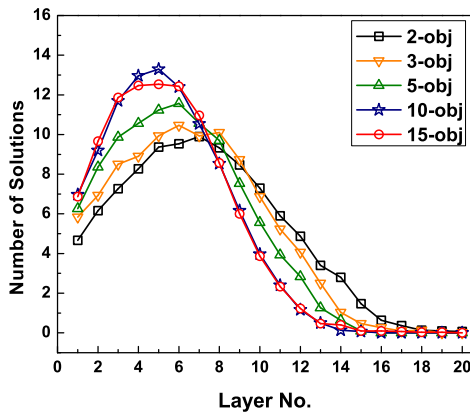
```

Algorithm 3 *environmentalSelection(Q)*.**Require:** N (population size)

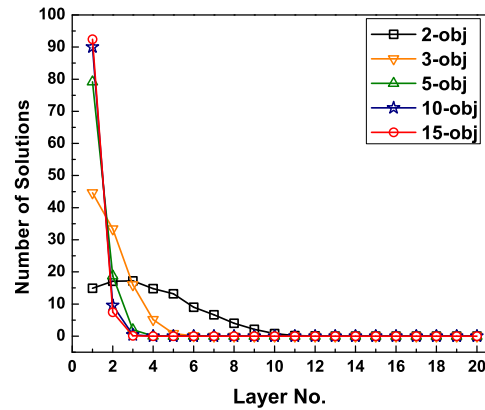
```

1: Generate an empty population  $P$ 
2: proximityEstimation(Q)
   /* Compute proximity of each individual in  $Q$  by Eq. (1) */
3: crowdingDegreeEstimation(Q)
   /* Compute crowding degree of each individual in  $Q$  by Eqs. (4) and (5) */
4:  $\{L_1, L_2, \dots, L_i, \dots\} \leftarrow \text{nondominatedSorting}(Q)$ 
   /* Partition  $Q$  into different layers ( $L_1, L_2, \dots, L_i, \dots$ ) by using Pareto nondominated sorting regarding proximity and crowding degree, and find the critical layer  $L_i$  (i.e.,  $0 \leq N - |L_1 \cup L_2 \cup \dots \cup L_{i-1}| < |L_i|$ ) */
5:  $P \leftarrow L_1 \cup L_2 \cup \dots \cup L_{i-1}$ 
6: if  $|P| < N$  then
7:   randomSelection(P, L_i, N - |P|)
   /* Select  $N - |P|$  individuals from  $L_i$  into  $P$  at random */
8: end if
9: return  $P$ 

```



(a) Nondominated sorting of the two goals



(b) Nondominated sorting of the actual objectives

Fig. 4. The average number of solutions in all the nondominated layers under (a) the bi-goal Pareto nondominated sorting and (b) the original Pareto nondominated sorting, where the population size is 100, the number of runs is 30, and the test instance is DTLZ2.

Algorithm 3 gives the environmental selection procedure of BiGE. First, individuals' performance regarding proximity and crowding degree is estimated (Steps 2 and 3). Then, the candidate set Q is divided into different layers by the nondominated sorting procedure with respect to the two goals, and the first $(i - 1)$ layers are moved into the population P , where $|L_1 \cup L_2 \cup \dots \cup L_{i-1}| \leq N$ and $|L_1 \cup L_2 \cup \dots \cup L_{i-1} \cup L_i| > N$ (Steps 4 and 5). Finally, the slots in P are filled randomly by individuals in L_i (Steps 6–8). Note that BiGE employs a randomly-selected mode on the layer L_i , rather than a density-based selection mode. This is because the density of individuals in this bi-goal space does not reflect their own performance. An individual with high density in the bi-goal space does not mean that it is worse than individuals with low density but rather that there are some other individuals having similar proximity and crowding degree with it in the population (cf. individual **C** in the example of Fig. 3). Therefore, we randomly select individuals which are located in the same layer.

In order to investigate the effectiveness of the bi-goal nondominated sorting in providing the selection pressure, Fig. 4 demonstrates the average number of solutions in all the nondominated layers on the 2-, 3-, 5-, 10- and 15-objective DTLZ2, where, for contrast, the average number of solutions in all the nondominated layers obtained by nondominated sorting of the actual objectives is shown as well. As can be seen from Fig. 4(b), the number of individuals placed in the first layer (L_1) increases rapidly with the number of objectives, approximating 80% of the population size when the number of

Table 2

Properties of test problems in comparative studies.

Problem	Number of objectives (m)	Number of variables (n)	Properties
WFG1	5, 10, 15	$2 \times (m - 1) + 20$	Mixed, flat biased
WFG2	5, 10, 15	$2 \times (m - 1) + 20$	Convex, disconnected, nonseparable
WFG3	5, 10, 15	$2 \times (m - 1) + 20$	Linear, degenerate, nonseparable
WFG4	5, 10, 15	$2 \times (m - 1) + 20$	Concave, multimodal
WFG5	5, 10, 15	$2 \times (m - 1) + 20$	Concave, deceptive
WFG6	5, 10, 15	$2 \times (m - 1) + 20$	Concave, nonseparable
WFG7	5, 10, 15	$2 \times (m - 1) + 20$	Concave, parameter dependant biased
WFG8	5, 10, 15	$2 \times (m - 1) + 20$	Concave, nonseparable, parameter dependant biased
WFG9	5, 10, 15	$2 \times (m - 1) + 20$	Concave, nonseparable, deceptive, parameter dependant biased
Knapsack	5, 10, 15	500	Convex, constraint
TSP	5, 10, 15	30	Convex, zero correlation
Water	5	3	Convex, degenerate, constraint

objectives reaches 5. In contrast, the individuals in Fig. 4(a) are located in many different layers and distributed in a similar pattern. For example, L_1 is always small and has around 6 individuals. In all the instances, the number of individuals in L_i increases until the total number of individuals in L_1 to L_i reaches around half of the population size. This means that the bi-goal nondominated sorting can effectively distinguish between individuals, which is largely independent of the number of objectives.

4. Experimental design

BiGE focuses on the comparison among the individuals which are nondominated to each other in the objective space. For the individuals that can be differentiated by Pareto dominance, any existing comparison strategy in the EMO area, such as the nondominated sorting [34], nondominated ranking [28], and strength [100], can be used. Here, the nondominated sorting strategy is chosen to cooperate with BiGE due to its simplicity and popularity [23]. In this section, we introduce test problems, performance indicators, peer algorithms, and general parameter setting for the experimental studies.

4.1. Test problems

Three well-known continuous and combinatorial benchmark suites, the walking fish group (WFG) toolkit [40], the multi-objective 0–1 knapsack problem [100], and the multi-objective traveling salesman problem (TSP) [18], are included, with the objective number $m = 5, 10$, and 15 . Also, a real-world constraint problem, the water problem [72], is considered. Their characteristics are summarized in Table 2.

WFG is a continuous problem suite that can be scaled to any number of objectives and decision variables. Comprised of problems with various characteristics (such as having linear, convex, concave, multimodal, disconnected, biased, and degenerated Pareto fronts), the WFG suite is used to challenge varying capabilities of an EMO algorithm. According to the suggestion in [40], the parameters k and l in WFG are set to $2 \times (m - 1)$ and 20 , respectively, where m denotes the number of objectives.

The multi-objective 0–1 knapsack problem is one of the standard combinatorial problems in multi-objective optimization. Given a set of n items and a set of m knapsacks, the multi-objective knapsack problem can be defined as follows:

$$\begin{aligned}
 &\text{Maximize } f_i(x) = \sum_{j=1}^n p_{ij}x_j, \quad i = 1, \dots, m \\
 &\text{Subject to } \sum_{j=1}^n w_{ij}x_j \leq c_i, \quad i = 1, \dots, m \\
 &x = (x_1, \dots, x_n)^T \in \{0, 1\}^n
 \end{aligned} \tag{6}$$

where $p_{ij} \geq 0$ is the profit of item j in knapsack i , $w_{ij} \geq 0$ is the weight of item j in knapsack i , c_i is the capacity of knapsack i , and $x_j = 1$ means that item j is selected in the knapsacks. Following the study in [100], p_{ij} and w_{ij} are random integers in the interval $[10, 100]$, and the knapsack capacity is set to half of the total weight regarding the corresponding knapsack. Also, the greedy repair method for infeasible solutions presented in [100] (i.e., the order in which the items are removed from the knapsacks is determined by their maximum profit/weight ratio) is adopted in our experimental studies.

The multi-objective TSP is a typical combinatorial optimization problem and can be stated as follows [18]: given a network $L = (V, C)$, where $V = \{v_1, v_2, \dots, v_n\}$ is a set of n nodes and $C = \{c_k : k \in \{1, 2, \dots, m\}\}$ is a set of m cost matrices between nodes ($c_k : V \times V$), we need to determine the Pareto optimal set of Hamiltonian cycles that minimize each of the

m cost objectives. In our study, the m matrices are uncorrelated to each other, generated by assigning each distinct pair of nodes with a random number in the range $[0, 1]$. According to [18], the number of nodes is set to 30.

The water problem [72,77] is a three-variable, five-objective, seven-constraint optimization problem which relates to optimal planning for a storm drainage system in an urban area. It is frequently used in the area to challenge EMO algorithms in dealing with a problem with many objectives and constraints [19,81,80,53]. A detailed description of the problem can be found in [72].

4.2. Hypervolume indicator

Hypervolume (HV) [100] is a very popular quality indicator due to its good theoretical properties [13,29,101]. Calculating the volume of the objective space between the obtained solution set and a reference point, HV can give the set a comprehensive assessment in terms of proximity and diversity. For clarity, we provide a normalized HV value of each algorithm with respect to the proportion of the optimal HV result achieved. This normalization makes all of the obtained results reside in the range $[0, 1]$, with 1 representing the optimal value. For some of the test problems (i.e., WFG4–WFG9), the optimal HV value can be obtained by calculation; for the others, the optimal value is, as suggested in [36], approximately estimated by the HV result of the nondominated set with respect to the mixed population consisting of all the obtained solutions on a given problem.

In the calculation of HV, two crucial issues are the scaling of the search space [30] and the choice of the reference point [3,31]. Since the objectives in the WFG and water problems take different ranges of values, we standardize the objective value of the obtained solutions according to the range of the problem's Pareto front. Following the recommendation in [45], the reference point is set to 1.1 times the upper bound of the Pareto front (i.e., $r = 1.1^m$) to emphasize the balance between proximity and diversity of the obtained solution set. For the two combinatorial optimization problems, since the range of their Pareto front is unknown, we set the reference point slightly worse than the boundary values of the nondominated set with respect to the mixed population consisting of all the obtained solutions; that is, the points with 13,000 and 22 for each objective (i.e., $r = 13,000^m$ and $r = 22^m$) are fixed for the knapsack and TSP problems, respectively.

In addition, since the exact calculation of the HV indicator is generally infeasible for a solution set with 10 or more objectives, we approximately estimate the HV result of a solution set by the Monte Carlo sampling method used in [4]. Here, 10,000,000 sampling points are used to ensure accuracy [4].

4.3. State-of-the-art algorithms in comparison

We compare the proposed BiGE with the following algorithms.

- **Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D)**² [94]. Decomposing an MOP into a set of scalar optimization subproblems and optimizing them in a collaborative manner, MOEA/D is one of the most popular EMO algorithms developed recently. The high search ability of MOEA/D on both multi- and many-objective problems has already been demonstrated in the literature [47,63,69]. Here, the Tchebycheff scalarizing function³ is used in MOEA/D in the experiments.
- **Nondominated Sorting Genetic Algorithm III (NSGA-III)**⁴ [20]. NSGA-III is a very recent many-objective algorithm whose framework is based on NSGA-II but with significant changes in the selection mechanism. Instead of the crowding distance, NSGA-III uses a decomposition-based niching technique to maintain diversity. NSGA-III has been shown to outperform some popular decomposition-based algorithms as well as the classical generative method in many-objective optimization [20,53].
- **Hypervolume Estimation Algorithm (HypE)**⁵ [4]. HypE is an indicator-based algorithm for many-objective optimization. HypE adopts Monte Carlo simulation to approximate the hypervolume value, significantly reducing the algorithm's time cost and enabling hypervolume-based search to be easily applied to many-objective optimization, even when the number of objectives reaches 50 [4]. HypE has been demonstrated to be competitive in the WFG problem suite with many objectives [69,88].
- **Fuzzy Dominance-based NSGA-II (FD-NSGA-II)**⁶ [39]. To deal with the failure of Pareto dominance in many-objective optimization, a fuzzy dominance-based fitness evaluation mechanism has been developed in [39] to continuously differentiate individuals into different degrees of optimality. The concept of fuzzy logic is adopted to define a fuzzy Pareto dominance relation. Specifically, a fuzzy set based on a Gaussian function is applied to quantify the degrees of dominance, from dominating to being dominated as well as in various degrees of dominance in each objective. Incorporated into NSGA-II, the proposed fuzzy concept has been found to be promising in many-objective optimization [38,39].

² The code of MOEA/D is from <http://dces.essex.ac.uk/staff/zhang/webofmoead.htm>.

³ In order to obtain more uniform solutions, in the Tchebycheff scalarizing function, "multiplying the weight vector w_i " in the original MOEA/D [94] is replaced by "dividing w_i ", as suggested and practiced in recent studies [20,64].

⁴ The code of NSGA-III is from <http://web.ntnu.edu.tw/~tchiang/publications/nsga3cpp/nsga3cpp.htm>.

⁵ The code of HypE is from <http://www.tik.ee.ethz.ch/pisa>.

⁶ The code of FD-NSGA-II was provided by its authors.

- **Approximation-Guided Evolutionary Algorithm II (AGE-II)**⁷ [84]. Recently, an approximation-guided EA (AGE) has been proposed [15], which allows to incorporate a formal notion of approximation into an EA. Using the best knowledge obtained so far during the evolutionary process, AGE improves the approximation quality of the current population. AGE has been shown to outperform state-of-the-art EMO algorithms particularly in dealing with many-objective problems [15,83]. Despite its good performance, AGE could suffer from heavy computational cost as new incomparable solutions can unconditionally insert into AGE's archive. To tackle this issue, a fast, effective AGE (called AGE-II) has been developed [84]. AGE-II introduces an adaptive ϵ -dominance approach to balance the convergence speed and runtime. Also, the mating selection strategy is elaborately designed to emphasize the diversity of its solution set.

4.4. General experimental setting

All the results presented in this paper were obtained by executing 30 independent runs of each algorithm on each problem. Following the practice in [33,85], the population size was set to 100 and the termination criterion of a run was 30,000 evaluations (i.e., 300 generations) for the WFG, TSP and water problems. For the knapsack problem, more evaluations are required for one generation of an algorithm due to the repair method that deals with infeasible solutions, where we set 100,000 evaluations as the termination criterion. Note that the size of the population in MOEA/D and NSGA-III is often the same as the number of weight vectors and it is impossible for the algorithms to generate uniformly distributed weight vectors at an arbitrary number. Here, we uniformly generate a set of around 5000 weight vectors and then select 100 well-distributed weight vectors from the set using the method in [95].

Parameters need to be set in some peer algorithms. According to the study in MOEA/D [94], the neighborhood size was specified as 10% of the population size. For HypE, the number of sampling points in HypE was set to 10,000. Following the practice in [88], the reference point for calculating the hypervolume contribution in HypE was set to $2i + 1$ for all WFG problems, where i is the number of objectives; for other problems, the reference point was set to be the same as in the HV indicator. In FD-NSGA-II, parameter σ , which determines the spread of the Gaussian function, was set to 0.5, as suggested in [39]. In AGE-II, parameter ϵ_{grid} , which determines the size of the archive, was set to 0.1 since it can provide a good tradeoff between performance and runtime in many-objective problems [84].

A crossover probability $p_c = 1.0$ and a mutation probability $p_m = 1/n$ (where n denotes the number of decision variables) were used. For continuous problems, operators for crossover and mutation are SBX crossover and polynomial mutation with both distribution indexes set to 20 [4,94]. As to the combinatorial problems, following the studies in [18,44], the uniform crossover and bit-flip mutation were used for the knapsack instance, and the order crossover and inversion mutation were used for the TSP instance.

5. Experimental results

In this section, we verify the performance of BiGE according to the experimental design described in the previous section. The HV results in the tables are the mean and standard deviation (SD) over 30 independent runs, and the best and second best mean values among the algorithms for each problem instance are shown with dark and light gray background, respectively. Moreover, in order to have statistically sound conclusions, we adopt the Wilcoxon's rank sum test [98] at a 0.05 significance level to examine the significance of the difference between the results obtained by BiGE and its competitors. The Wilcoxon test is a nonparametric alternative to the two-sample t-test with two advantages: 1) valid for data with a non-normal distribution and 2) much less sensitive to the outliers.

5.1. WFG problems

Table 3 gives the comparative results of the six algorithms on the WFG problems with 5, 10, and 15 objectives. As shown, BiGE and HypE perform the best, having a clear advantage over the other 4 algorithms on most of the test instances. Specifically, BiGE obtains the best and second best HV results on 14 and 10 out of the 27 instances respectively, and HypE on 10 and 15 respectively. NSGA-III performs the best on the 5-objective WFG3 and WFG9, and also generally outperforms the other three algorithms. AGE-II and MOEA/D typically work fairly well on the 5-objective WFG, but struggle on the 10- and 15-objective instances. FD-NSGA-II, which fails to maintain the diversity of individuals in the population, has the worst HV results on the WFG problem suite.

Concerning the statistical results, it can be observed that the difference between BiGE and the peer algorithms is significant on most of the test instances. Specifically, the proportion of the test instances where BiGE outperforms MOEA/D, NSGA-III, HypE, FD-NSGA-II and AGE-II with statistical significance is 26/27, 21/27, 11/27, 27/27, and 25/27, respectively. Conversely, the proportion of the instances where BiGE performs worse than MOEA/D, NSGA-III, HypE, FD-NSGA-II and AGE-II with statistical significance is 1/27, 2/27, 9/27, 0/27, and 0/27, respectively.

For a visual understanding of the solutions' distribution, Fig. 5 plots the final solutions of one run with respect to the 10-objective WFG9 by parallel coordinates. This particular run is associated with the result that is the closest to the mean

⁷ The code of AGE-II was provided by its authors.

Table 3

Normalized HV results (mean and SD) of the six algorithms on the WFG problem. The best and the second mean among the algorithms for each problem instance is shown with dark and light gray background, respectively.

Problem	Obj.	MOEA/D	NSGA-III	HypE	FD-NSGA-II	AGE-II	BiGE
WFG1	5	6.892E-1 (3.6E-2) [†]	6.773E-1 (6.3E-2) [†]	7.827E-1 (1.6E-2) [†]	2.194E-1 (7.2E-2) [†]	5.219E-1 (2.4E-2) [†]	6.151E-1 (3.9E-2)
	10	6.421E-1 (3.2E-2) [†]	7.710E-1 (5.4E-2)	8.324E-1 (2.3E-2) [†]	4.294E-1 (6.9E-2) [†]	6.092E-1 (2.8E-2) [†]	7.786E-1 (4.7E-2)
	15	5.913E-1 (2.2E-2) [†]	7.208E-1 (3.1E-2) [†]	8.138E-1 (3.1E-2) [†]	4.775E-1 (7.8E-2) [†]	5.735E-1 (2.2E-2) [†]	7.612E-1 (3.4E-2)
WFG2	5	7.999E-1 (9.0E-2) [†]	8.881E-1 (1.5E-1)	8.819E-1 (8.6E-2)	2.977E-1 (7.4E-2) [†]	9.180E-1 (7.2E-2)	9.014E-1 (8.2E-2)
	10	7.576E-1 (8.4E-2) [†]	8.647E-1 (7.5E-2) [†]	8.864E-1 (8.0E-2) [†]	2.698E-1 (1.4E-1) [†]	9.168E-1 (7.1E-2)	9.475E-1 (5.3E-2)
	15	6.454E-1 (7.7E-2) [†]	8.834E-1 (8.1E-2) [†]	9.247E-1 (8.3E-2)	2.841E-1 (1.2E-1) [†]	8.535E-1 (7.1E-2) [†]	9.402E-1 (6.1E-2)
WFG3	5	6.450E-1 (2.0E-2) [†]	9.139E-1 (1.7E-2)	9.137E-1 (7.4E-3)	1.330E-1 (1.9E-3) [†]	8.139E-1 (2.5E-2) [†]	9.092E-1 (1.1E-2)
	10	4.553E-1 (3.2E-2) [†]	7.949E-1 (6.6E-2) [†]	9.149E-1 (1.2E-2) [†]	1.242E-1 (1.8E-3) [†]	6.673E-1 (3.4E-2) [†]	8.705E-1 (1.8E-2)
	15	2.431E-1 (2.3E-2) [†]	8.136E-1 (7.7E-2) [†]	8.964E-1 (1.8E-2) [†]	1.213E-1 (2.4E-3) [†]	4.713E-1 (2.6E-2) [†]	8.545E-1 (2.6E-2)
WFG4	5	5.899E-1 (2.9E-2) [†]	7.269E-1 (6.5E-2) [†]	8.171E-1 (7.6E-3) [†]	2.770E-1 (8.2E-2) [†]	6.381E-1 (1.9E-2) [†]	8.117E-1 (8.4E-3)
	10	4.359E-1 (4.4E-2) [†]	4.223E-1 (7.6E-2) [†]	8.049E-1 (2.9E-2) [†]	2.207E-1 (9.8E-2) [†]	3.248E-1 (2.3E-2) [†]	8.313E-1 (1.1E-2)
	15	3.000E-1 (3.8E-2) [†]	5.332E-1 (4.8E-2) [†]	7.831E-1 (2.5E-2) [†]	1.223E-1 (5.2E-2) [†]	2.336E-1 (2.3E-2) [†]	8.073E-1 (1.9E-2)
WFG5	5	6.418E-1 (1.8E-2) [†]	7.607E-1 (5.8E-3) [†]	7.701E-1 (8.5E-3)	8.776E-2 (2.0E-4) [†]	6.301E-1 (1.5E-2) [†]	7.709E-1 (6.2E-3)
	10	4.911E-1 (3.4E-2) [†]	5.235E-1 (6.5E-2) [†]	8.142E-1 (1.7E-2) [†]	7.930E-2 (2.6E-4) [†]	4.040E-1 (3.5E-2) [†]	7.990E-1 (1.9E-2)
	15	3.170E-1 (6.4E-2) [†]	6.250E-1 (5.4E-2) [†]	7.628E-1 (1.9E-2) [†]	7.892E-2 (2.8E-4) [†]	2.735E-1 (6.5E-2) [†]	7.715E-1 (1.4E-2)
WFG6	5	5.753E-1 (2.6E-2) [†]	7.718E-1 (3.9E-2)	7.750E-1 (1.2E-2)	1.040E-1 (3.5E-2) [†]	6.577E-1 (1.8E-2) [†]	7.728E-1 (8.9E-3)
	10	5.158E-1 (4.3E-2) [†]	5.389E-1 (5.1E-2) [†]	8.352E-1 (1.1E-2) [†]	8.434E-2 (1.3E-3) [†]	3.897E-1 (3.0E-2) [†]	8.270E-1 (1.3E-2)
	15	2.845E-1 (4.5E-2) [†]	6.528E-1 (4.5E-2) [†]	8.139E-1 (2.3E-2) [†]	8.367E-2 (1.2E-3) [†]	3.402E-1 (7.1E-2) [†]	8.339E-1 (1.4E-2)
WFG7	5	6.413E-1 (4.3E-2) [†]	7.878E-1 (3.7E-2) [†]	8.262E-1 (8.7E-3) [†]	1.011E-1 (3.7E-5) [†]	6.770E-1 (2.0E-2) [†]	8.356E-1 (5.5E-3)
	10	5.852E-1 (4.3E-2) [†]	5.637E-1 (6.7E-2) [†]	8.808E-1 (1.2E-2)	1.140E-1 (5.3E-2) [†]	3.869E-1 (5.6E-2) [†]	8.827E-1 (1.2E-2)
	15	2.057E-1 (5.8E-2) [†]	6.919E-1 (5.2E-2) [†]	8.305E-1 (2.4E-2) [†]	9.381E-2 (1.7E-2) [†]	1.160E-1 (2.5E-2) [†]	8.787E-1 (1.3E-2)
WFG8	5	3.536E-1 (2.7E-2) [†]	5.609E-1 (9.7E-2) [†]	6.893E-1 (7.7E-3) [†]	1.529E-1 (8.6E-2) [†]	5.587E-1 (2.8E-2) [†]	6.822E-1 (9.1E-3)
	10	3.756E-1 (3.1E-2) [†]	4.944E-1 (7.0E-2) [†]	7.628E-1 (1.2E-2) [†]	1.279E-1 (5.9E-2) [†]	2.976E-1 (7.1E-2) [†]	7.722E-1 (6.1E-3)
	15	2.479E-1 (6.3E-2) [†]	6.112E-1 (8.8E-1) [†]	7.872E-1 (2.4E-2) [†]	1.116E-1 (5.2E-2) [†]	1.215E-1 (4.5E-2) [†]	8.179E-1 (1.0E-2)
WFG9	5	4.727E-1 (3.5E-2) [†]	7.253E-1 (1.6E-2) [†]	6.971E-1 (3.2E-2)	8.059E-2 (4.6E-5) [†]	5.963E-1 (2.9E-2) [†]	6.903E-1 (1.4E-2)
	10	3.675E-1 (5.3E-2) [†]	6.127E-1 (3.6E-2) [†]	6.727E-1 (2.4E-2) [†]	7.417E-2 (2.3E-3) [†]	4.273E-1 (5.1E-2) [†]	6.824E-1 (1.3E-2)
	15	2.044E-1 (5.3E-2) [†]	5.570E-1 (4.2E-2) [†]	6.677E-1 (1.2E-2) [†]	7.435E-2 (3.3E-3) [†]	3.408E-1 (5.2E-2) [†]	6.893E-1 (3.1E-2)

“†” indicates that the result of the peer algorithm is significantly different from that of BiGE at a 0.05 level by the Wilcoxon's rank sum test.

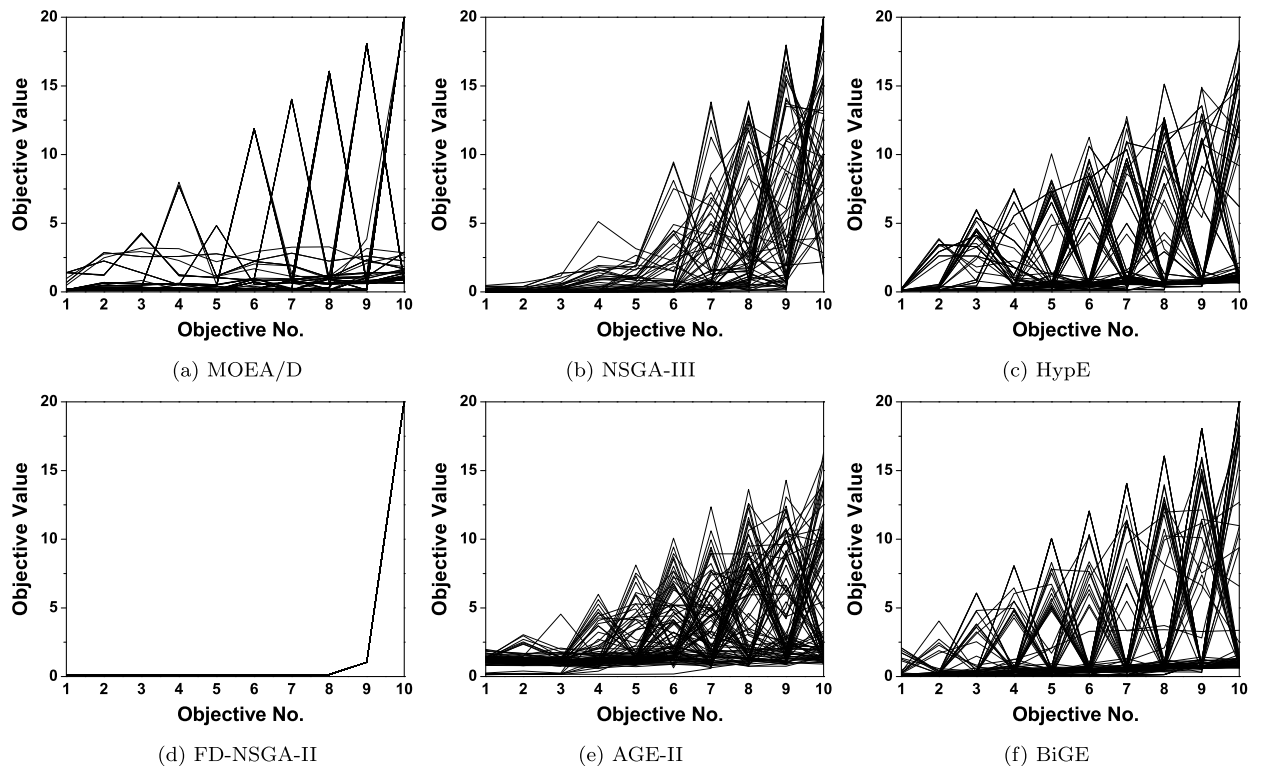


Fig. 5. The final solution set of the six algorithms on the ten-objective WFG9, shown by parallel coordinates.

Table 4

Normalized HV results (mean and SD) of the six algorithms on the Knapsack problem. The best and the second mean among the algorithms for each problem instance is shown with dark and light gray background, respectively.

Obj.	MOEA/D	NSGA-III	HypE	FD-NSGA-II	AGE-II	BiGE
5	5.412E-1 (3.2E-2) [†]	4.657E-1 (1.6E-2) [†]	5.467E-1 (2.1E-2) [†]	5.319E-1 (2.7E-2) [†]	5.436E-1 (2.4E-2) [†]	5.738E-1 (2.1E-2)
10	1.385E-1 (3.6E-2) [†]	1.017E-1 (2.6E-2) [†]	2.529E-2 (3.2E-2) [†]	3.224E-1 (5.4E-2) [†]	3.171E-1 (4.0E-2) [†]	3.507E-1 (5.3E-2)
15	1.396E-1 (2.9E-2) [†]	4.793E-2 (1.4E-2) [†]	2.100E-1 (3.0E-2)	2.299E-1 (3.3E-2)	2.183E-1 (2.8E-2)	2.268E-1 (3.7E-2)

"†" indicates that the result of the peer algorithm is significantly different from that of BiGE at a 0.05 level by the Wilcoxon's rank sum test.

Table 5

Normalized HV results (mean and SD) of the six algorithms on the TSP problem. The best and the second mean among the algorithms for each problem instance is shown with dark and light gray background, respectively.

Obj.	MOEA/D	NSGA-III	HypE	FD-NSGA-II	AGE-II	BiGE
5	6.004E-1 (3.1E-2) [†]	4.173E-1 (3.9E-2) [†]	5.037E-1 (2.1E-2) [†]	5.636E-1 (3.8E-2) [†]	6.345E-1 (2.5E-2) [†]	6.186E-1 (2.1E-2)
10	2.246E-1 (3.9E-2) [†]	2.295E-2 (1.1E-2) [†]	3.000E-1 (1.8E-2) [†]	4.125E-1 (8.8E-2) [†]	3.477E-1 (2.8E-2) [†]	4.523E-1 (3.0E-2)
15	4.201E-2 (1.7E-2) [†]	8.876E-3 (6.4E-3) [†]	2.144E-1 (4.1E-2) [†]	2.446E-1 (6.3E-2) [†]	1.467E-1 (2.7E-2) [†]	2.860E-1 (4.7E-2)

"†" indicates that the result of the peer algorithm is significantly different from that of BiGE at a 0.05 level by the Wilcoxon's rank sum test.

HV value. Although all considered solution sets appear to converge into the optimal front (the upper and lower bounds of objective i in WFG's Pareto front are 0 and $2 \times i$, respectively), the six algorithms perform differently in terms of diversity maintenance. The solutions obtained by FD-NSGA-II converge into one point of the Pareto front, while the solutions of MOEA/D concentrate in the boundaries of the optimal front. The solutions of AGE-II and NSGA-III seem to have a good uniformity, but fail to reach some regions of the Pareto front. HypE and BiGE perform similarly. The only difference between them is that the solutions of HypE struggle to cover the problem's boundary on some objectives, while the solutions of BiGE appear to have a good coverage over the whole Pareto front.

5.2. The knapsack problem

Table 4 gives the results of the six algorithms on the 0–1 knapsack problem. As can be seen from the table, BiGE generally outperforms the five peer algorithms. Specifically, for the 5- and 10-objective instances, BiGE has the best HV value, and also the difference between BiGE and its competitors is statistically significant. For the 15-objective instance, BiGE ranks the second, only outperformed by FD-NSGA-II. In addition, it is interesting to note that FD-NSGA-II, which performs the worst in the WFG problems, works quite well in the knapsack problem (also in the TSP problem, as shown in Table 5 later). This indicates the different characteristics between continuous and combinatorial optimization problems. Some EMO algorithms may show better behavior on combinatorial optimization problems if their fitness assignment strategy is particularly suitable for the structure of the integral code in the problems.

5.3. The TSP problem

The normalized HV results of the six algorithms on the three TSP test instances are shown in Table 5. It is observed that BiGE performs better on the problem with a larger number of objectives. For the 5-objective TSP, AGE-II has the highest HV value, and BiGE outperforms the other four algorithms with statistical significance. For the 10- and 15-objective instances, BiGE and FD-NSGA-II, like on the knapsack problem, perform better than the other four algorithms. A difference from the results on the knapsack problem is that here BiGE always obtains a higher HV value than FD-NSGA-II on the instances. It is worth mentioning that HypE and NSGA-III, which are competitive in the WFG problems, perform constantly worse than BiGE on all the 6 knapsack and TSP instances.

To facilitate visual comparison, Fig. 6 plots the final solutions of a single run of the six algorithms regarding the two-dimensional objective space f_1 and f_2 of the 15-objective TSP. Similar plots can be obtained for other objectives of the problem. As shown, the solutions of BiGE have a good balance between proximity and diversity. In contrast, the five peer algorithms struggle in terms of proximity, with their solutions being generally distributed in the top-right region of the figures.

5.4. The water problem

The water problem is a three-variable, five-objective, seven-constraint real-world problem [72,77], which was designed to optimize the planning for a storm drainage system in an urban area. Table 6 gives the HV results of the six algorithms on this problem. As shown, BiGE outperforms the five peer algorithms with statistical significance. This indicates the effectiveness of the proposed bi-goal evolution in dealing with a problem with many objectives and constraints.

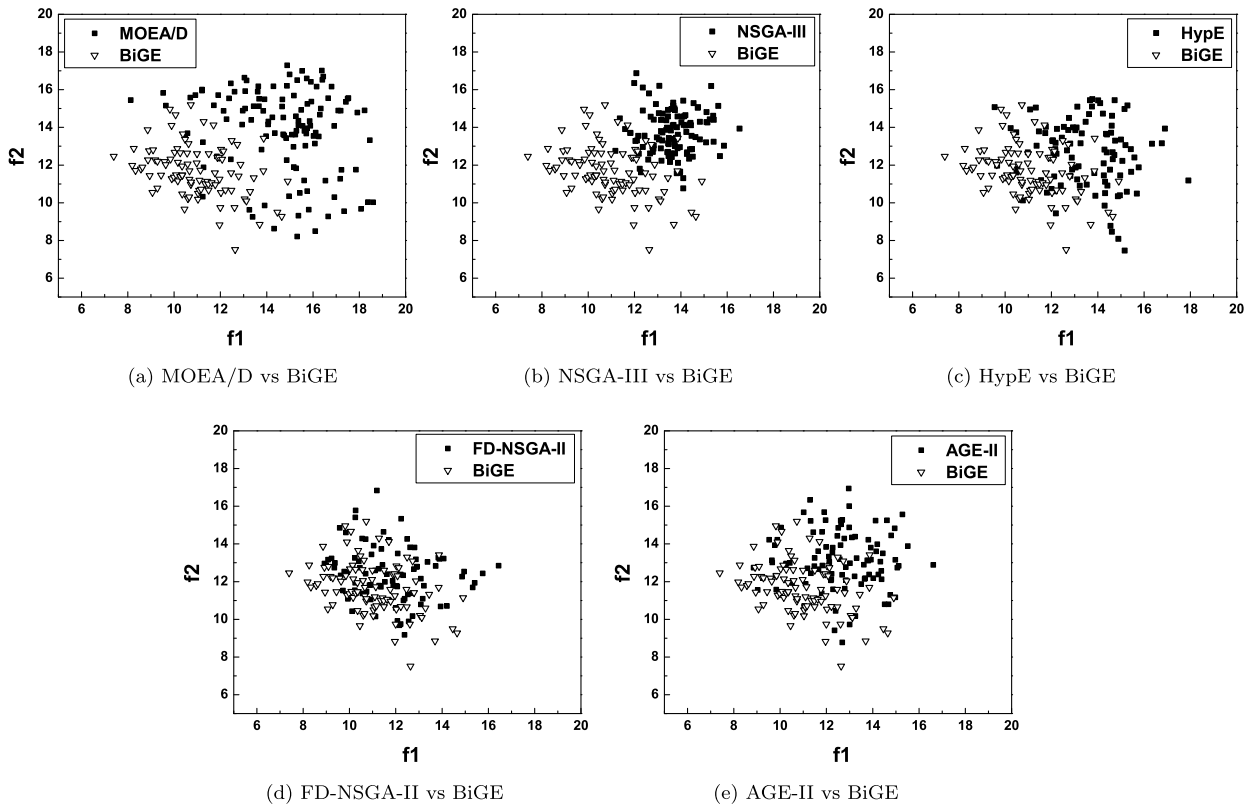


Fig. 6. Result comparison between BiGE and each of the other five algorithms on the 15-objective TSP. The final solutions of the algorithms are shown regarding the two-dimensional objective space f_1 and f_2 .

Table 6

Normalized HV results (mean and SD) of the six algorithms on the water problem. The best and the second mean among the algorithms for each problem instance is shown with dark and light gray background, respectively.

MOEA/D	NSGA-III	HypE	FD-NSGA-II	AGE-II	BiGE
8.589E-1 (9.1E-3) [†]	9.176E-1 (2.9E-3) [†]	9.133E-1 (3.8E-3) [†]	1.982E-1 (1.8E-3) [†]	8.960E-1 (1.5E-3) [†]	9.273E-1 (4.1E-3)

[†] indicates that the result of the peer algorithm is significantly different from that of BiGE at a 0.05 level by the Wilcoxon's rank sum test.

5.5. Result summary

To sum up, BiGE generally outperforms the five state-of-the-art algorithms, with the best and second best HV results in 19 and 12 out of all the 34 test instances, respectively. The five peer algorithms perform differently on problems with distinct properties. HypE and NSGA-III perform well on continuous MOPs, while FD-NSGA-II is competitive for combinatorial ones. AGE-II and MOEA/D work fairly well on 5-objective instances, but perform poorly in a higher-dimensional objective space. Similar observations have been reported in some recent studies [88,69,96,86].

In addition, the behavior difference between BiGE and the peer algorithms can also be seen from the simple artificial example in Table 1. In this example, BiGE can effectively distinguish between seven Pareto nondominated solutions, with **B**, **G**, **C** and **E** clearly outperforming the remaining ones (cf. Fig. 3(c)). In contrast, the five peer algorithms may fail to pick out the same solutions. The individual selection in MOEA/D, NSGA-III, HypE and AGE-II largely depends on some references associated with the algorithm, i.e., MOEA/D and NSGA-III on the predefined reference directions (on the basis of distribution of the weight vectors), HypE on the reference point in the hypervolume calculation, and AGE-II on the distribution of solutions in the archive. For FD-NSGA-II, since the fuzzy-based dominance relation prefers well-converged individuals, **B**, **A**, **G** and **C** would be assigned better fitness values; this may lead to the loss of population diversity.

6. Further investigations of BiGE

The experimental results in the previous section have shown the effectiveness of BiGE on diverse problems. Next, we will further examine BiGE by investigating the effect of parameter setting on the algorithm performance and comparing it with some algorithms that have similar components to the proposed algorithm.

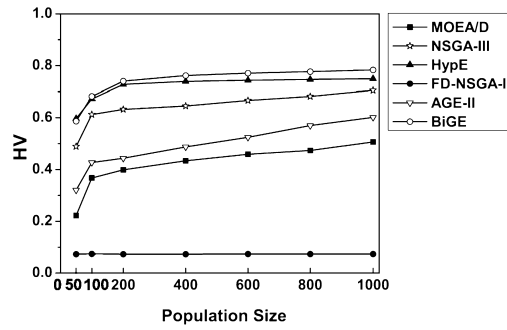


Fig. 7. Normalized HV of the six algorithms with different settings of the population size on the 10-objective WFG9.

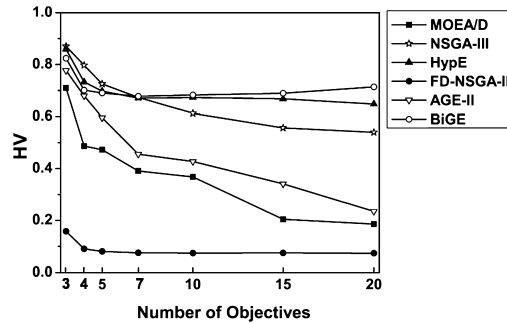


Fig. 8. Normalized HV of the six algorithms with different settings of the number of objectives on WFG9.

6.1. Effect of the population size and objective dimensionality

In BiGE, two parameters, i.e., the population size and the number of objectives, play an important role. They determine the niche radius in the crowding degree estimation of the algorithm. In this section, we investigate the effect of these two parameters on the algorithm's performance. Here, we show experimental results on WFG9, one of the most challenging test problems (this can be inferred from the HV values in Table 3). Similar results can also be observed on other problems.

First, we consider the effect of the population size on the performance of the six algorithms. The population size in the previous studies was fixed to 100. In this study, we give a wide range of the population size (from 50 to 1000) to test how the performance of the algorithm varies with it. Other parameters are kept unchanged in this study, except the function evaluations which are changed accordingly in order to keep the number of generations (300) fixed. Fig. 7 shows the HV results on the 10-objective WFG9. Clearly, except FD-NSGA-II, the HV result of all the algorithms increases with the population size, which means that a larger population size generally leads to a better performance. This is shown more evidently in AGE-II, NSGA-III, and MOEA/D. On the other hand, HypE and BiGE always outperform other four algorithms under all the seven settings of the population size. More specifically, HypE has the best HV when the population size is 50, while BiGE performs the best for the remaining cases. Overall, the above results indicate the insensitiveness of the proposed algorithm to the population size – BiGE can work well under various sizes of the evolutionary population.

Next, we consider the effect of the objective dimensionality on the performance of the six algorithms. In the previous studies, the algorithms have already been tested under 5, 10, and 15 objectives. Here, we extend the range of the number of objectives and investigate how the algorithms work in a lower- or higher-dimensional space. Fig. 8 shows the HV results of the six algorithms on the 3-, 4-, 5-, 7-, 10-, 15-, and 20-objective WFG9. As shown, NSGA-III, HypE, and BiGE outperform the other algorithms under all the seven settings of the number of objectives. Taking a closer comparison among these three algorithms, NSGA-III and HypE perform the best for the problem with from 3 to 5 objectives, while BiGE shows its advantage when the number of objectives reaches 10. In addition, an interesting difference of BiGE from the other algorithms is that its HV value remains quite steady (rather than degrades) with the increase of the number of objectives. This occurrence could be attributed to the fact that the bi-goal evolution can provide a good balance between proximity and diversity, which is largely independent of a problem's objective dimensionality.

6.2. Effect of the sharing discriminator in the sharing function

A feature in BiGE is that a sharing discriminator is introduced to differentiate individuals in a niche. When calculating the sharing function of two neighboring individuals, one with better proximity is encouraged by multiplying 0.5, while the other is discouraged by multiplying 1.5 (here we denote this sharing discriminator as (sd_e, sd_d)). This adjustment can lead the individual with better proximity to have a lower crowding degree and the individual with worse proximity to have a

Table 7

Normalized HV of BiGE with different settings of the sharing discriminator on the 10-objective WFG9.

(sd_e, sd_d)	Normalized HV
(0.00, 2.00)	6.211E-1 (1.5E-2)
(0.25, 1.75)	6.822E-1 (1.4E-2)
(0.50, 1.50)	6.824E-1 (1.3E-2)
(0.75, 1.75)	6.806E-1 (1.3E-2)
(1.00, 1.00)	5.650E-1 (1.5E-2)

Table 8

Normalized HV of BiGE with the settings of the sharing discriminator that only discourage the individual with worse proximity on the 10-objective WFG9.

(sd_1, sd_2)	Normalized HV
(1.00, 1.25)	6.720E-1 (1.3E-2)
(1.00, 1.50)	6.773E-1 (1.2E-2)
(1.00, 1.75)	6.758E-1 (1.3E-2)
(1.00, 2.00)	6.732E-1 (1.2E-2)
(0.50, 1.50)	6.824E-1 (1.3E-2)

higher one. Now a straightforward question is how much (sd_e, sd_d) affects the performance of the algorithm. In addition, one may also ask if we can only discourage the individual with worse proximity while remaining the other unchanged, such as (sd_e, sd_d) being set to (1.0, 1.5). In this case, two neighboring individuals can also be well differentiated.

In this section, we investigate the effect of the sharing discriminator and attempt to answer the above two questions. Due to space limitation, we only show the results on the 10-objective WFG9. Similar results can be obtained for other problems. Here, we consider four representative settings of the discriminator: (0.0, 2.0), (0.25, 1.75), (0.75, 1.25), and (1.0, 1.0). The setting (0.0, 2.0) is an extreme where the individual with better proximity is assigned zero sharing function value, while (1.0, 1.0) is the other extreme where neither of the individuals' sharing function value is changed. The settings (0.25, 1.75) and (0.75, 1.25) are two middle values between the extremes and the setting (0.5, 1.5) used in the paper. Table 7 gives the HV results of BiGE with the above four settings, along with (0.5, 1.5), on the 10-objective WFG9. As shown, the algorithm with the three settings (0.25, 1.75), (0.5, 1.5), and (0.75, 1.25) performs very similarly, and all significantly outperform the algorithm with the two extreme settings (0.0, 2.0) and (1.0, 1.0). This indicates the insensitiveness of the algorithm to the discriminator parameter within a certain range – BiGE can work well with different discriminator values, provided that they are away from the two extremes.

Next, we consider the case that only the individual with worse proximity is discouraged in the sharing function. That is, sd_e is set to 1.0 and sd_d to larger than 1.0. Here, we consider four settings of the discriminator: (1.0, 1.25), (1.0, 1.5), (1.0, 1.75), and (1.0, 2.0). The HV results of BiGE with them are given in Table 8, where the result of the algorithm with (0.5, 1.5) is repeated for comparison. As can be seen from the table, the algorithm where the individual with better proximity is not encouraged performs slightly worse than the original algorithm. This occurrence might be attributed to the following reason. In general, for a group of individuals in a niche, it is ideal to select a representative individual (i.e., with the best proximity) from them into the next population. However, with the discriminator setting that only discourages individuals with worse proximity, all the individuals in the niche could have a high crowding degree (in comparison with those having no neighbor in their own niche). This may lead to none of the individuals in this niche surviving in the next population. Thus, an encouragement for the individual with better proximity in the niche is beneficial to the diversity of the population – it further differentiates similar individuals and enables a representative one to be preserved in the evolutionary process.

6.3. Comparison with the average ranking (AR) methods

In BiGE, the proximity of an individual is estimated by the sum of its normalized values across the objectives. This estimation could be viewed as a slightly more fine-grained version of the well-known AR method [5]. AR estimates the proximity of an individual by summing its ranks (in the population) across the objectives. The difference between these two estimations is that AR considers individuals' rank in the population on each objective, while the proposed proximity estimation considers quantitative difference of individuals on each objective.

As an individual comparison criterion, AR is popular in many-objective optimization. Corne and Knowles have demonstrated that AR can provide sufficient selection pressure towards the optimal front in a high-dimensional objective space [18]. However, due to the lack of a diversity maintenance scheme, AR may lead the evolutionary population to converge into a sub-area of the Pareto front [52]. Recently, some methods have been proposed to enhance diversity for AR. For

Table 9

Normalized HV results (mean and SD) of the four algorithms on all the 34 test instances. The best and the second mean among the algorithms for each problem instance is shown with dark and light gray background, respectively.

Problem	Obj.	AR	AR+sharing	BiGE(AR)	BiGE
WFG1	5	1.553E-2 (6.9E-2) [†]	5.001E-1 (1.4E-2) [†]	5.530E-1 (2.8E-2) [†]	6.151E-1 (3.9E-2)
	10	1.577E-1 (1.4E-1) [†]	4.842E-1 (2.9E-2) [†]	5.998E-1 (3.9E-2) [†]	7.786E-1 (4.7E-2)
	15	4.961E-1 (2.9E-1) [†]	4.716E-1 (3.1E-2) [†]	6.001E-1 (3.7E-2) [†]	7.612E-1 (3.4E-2)
WFG2	5	1.714E-1 (7.6E-2) [†]	9.663E-1 (8.2E-3)	9.255E-1 (8.4E-2)	9.014E-1 (8.2E-2)
	10	2.276E-1 (9.0E-2) [†]	9.717E-1 (1.4E-2) [†]	9.715E-1 (5.2E-2)	9.475E-1 (5.3E-2)
	15	2.442E-1 (9.9E-2) [†]	9.587E-1 (1.9E-2)	9.541E-1 (6.3E-2)	9.402E-1 (6.1E-2)
WFG3	5	2.274E-1 (2.9E-2) [†]	7.470E-1 (3.7E-2) [†]	8.975E-1 (1.4E-2) [†]	9.092E-1 (1.1E-2)
	10	2.001E-1 (2.8E-2) [†]	7.486E-1 (5.5E-2) [†]	8.609E-1 (2.1E-2) [†]	8.705E-1 (1.8E-2)
	15	1.868E-1 (3.9E-2) [†]	7.559E-1 (4.7E-2) [†]	8.349E-1 (2.3E-2) [†]	8.545E-1 (2.6E-2)
WFG4	5	1.351E-1 (1.7E-2) [†]	6.745E-1 (1.5E-2) [†]	8.340E-1 (6.5E-3)	8.117E-1 (8.4E-3)
	10	1.114E-1 (1.3E-2) [†]	6.455E-1 (2.3E-2) [†]	8.304E-1 (1.1E-2)	8.313E-1 (1.1E-2)
	15	1.053E-1 (1.1E-2) [†]	5.448E-1 (3.6E-2) [†]	7.829E-1 (2.7E-2) [†]	8.073E-1 (1.9E-2)
WFG5	5	1.210E-1 (2.1E-2) [†]	6.634E-1 (1.4E-2) [†]	7.778E-1 (7.4E-3) [†]	7.709E-1 (6.2E-3)
	10	9.355E-2 (1.1E-2) [†]	6.754E-1 (1.8E-2) [†]	7.909E-1 (1.1E-2) [†]	7.990E-1 (1.9E-2)
	15	9.344E-2 (1.0E-2) [†]	6.224E-1 (2.4E-2) [†]	7.870E-1 (1.3E-2) [†]	7.715E-1 (1.4E-2)
WFG6	5	1.196E-1 (2.3E-2) [†]	6.283E-1 (1.9E-2) [†]	7.891E-1 (9.3E-3) [†]	7.728E-1 (8.9E-3)
	10	9.997E-2 (2.4E-2) [†]	6.607E-1 (2.4E-2) [†]	8.190E-1 (1.0E-2) [†]	8.270E-1 (1.3E-2)
	15	1.019E-1 (2.6E-2) [†]	6.673E-1 (3.0E-2) [†]	8.120E-1 (1.8E-2) [†]	8.339E-1 (1.4E-2)
WFG7	5	2.239E-1 (7.4E-2) [†]	6.876E-1 (1.5E-2) [†]	8.426E-1 (6.8E-3) [†]	8.356E-1 (5.5E-3)
	10	2.166E-1 (6.0E-2) [†]	7.303E-1 (2.1E-2) [†]	8.489E-1 (1.2E-2) [†]	8.827E-1 (1.2E-2)
	15	1.734E-1 (5.1E-2) [†]	6.858E-1 (3.5E-2) [†]	8.182E-1 (1.9E-2) [†]	8.787E-1 (1.3E-2)
WFG8	5	1.832E-1 (3.0E-2) [†]	5.064E-1 (1.4E-2) [†]	6.833E-1 (8.9E-3)	6.822E-1 (9.1E-3)
	10	1.581E-1 (2.5E-2) [†]	5.769E-1 (2.7E-2) [†]	7.478E-1 (8.6E-3) [†]	7.722E-1 (6.1E-3)
	15	1.449E-1 (3.0E-2) [†]	6.427E-1 (2.8E-2) [†]	7.843E-1 (1.4E-2) [†]	8.179E-1 (1.0E-2)
WFG9	5	1.218E-1 (4.4E-2) [†]	6.078E-1 (1.4E-2) [†]	6.910E-1 (8.7E-3)	6.903E-1 (1.4E-2)
	10	9.933E-2 (3.7E-2) [†]	6.314E-1 (1.9E-2) [†]	6.742E-1 (2.1E-2) [†]	6.824E-1 (1.3E-2)
	15	1.131E-1 (5.7E-2) [†]	6.122E-1 (2.6E-2) [†]	6.616E-1 (3.5E-2) [†]	6.893E-1 (3.1E-2)
Knapsack	5	4.199E-1 (3.2E-2) [†]	4.538E-1 (2.9E-2) [†]	5.652E-1 (2.1E-2)	5.738E-1 (2.1E-2)
	10	1.409E-1 (3.6E-2) [†]	2.879E-1 (1.3E-2) [†]	3.126E-1 (3.9E-2) [†]	3.507E-1 (5.3E-2)
	15	7.471E-2 (1.9E-2) [†]	1.403E-1 (3.9E-2) [†]	2.096E-1 (3.3E-2) [†]	2.268E-1 (3.7E-2)
TSP	5	2.850E-1 (3.0E-2) [†]	3.295E-1 (3.2E-2) [†]	5.788E-1 (2.9E-2) [†]	6.186E-1 (2.1E-2)
	10	1.251E-1 (2.4E-2) [†]	1.837E-1 (2.4E-2) [†]	4.094E-1 (3.3E-2) [†]	4.523E-1 (3.0E-2)
	15	5.187E-2 (2.7E-2) [†]	1.013E-1 (5.5E-2) [†]	2.367E-1 (4.8E-2) [†]	2.860E-1 (4.7E-2)
Water	5	7.904E-1 (7.7E-3) [†]	8.668E-1 (5.6E-3) [†]	9.213E-1 (8.0E-3) [†]	9.273E-1 (4.1E-3)

"†" indicates that the result of the peer algorithm is significantly different from that of BiGE at a 0.05 level by the Wilcoxon's rank sum test.

example, Purshouse et al. made a modification of the AR-based fitness by combining it with a sharing scheme based on the Epanechnikov kernel [76]. Li et al. imposed a punishment on individuals who are neighbors of the best-AR individual to prohibit or postpone their entry in the next population [70]. Kong et al. repeatedly initialized the population by a chaotic method after some generations, in order to enhance diversity of individuals in the decision space [59]. Instead of considering the objectives in the original AR, Yuan et al. summed up the aggregation function values based on uniformly-distributed weight vectors [93].

A clear difference of BiGE from these AR-based algorithms is that BiGE uses the idea of Pareto dominance to deal with proximity and diversity. This could be well suited to many-objective optimization where the conflict between proximity and diversity goals is more serious than that in bi- or tri-objective optimization. Considering the dominance relation of these two goals can provide a good balance between them and lead the algorithm to be less affected by the increase of the objective dimensionality.

Next, we empirically investigate the difference between BiGE and some AR-based algorithms. Specifically, we consider three peer algorithms: (1) the original AR [5], (2) AR combined with a fitness sharing scheme (called AR+sharing here) [76], and (3) a new version of BiGE where AR is used as the proximity estimation method, denoted as BiGE(AR). In [76], AR has been found to be competitive when combined with a sharing scheme based on the Epanechnikov kernel [27]. From some initial experiments, we found that replacing the Epanechnikov kernel with the proposed niching method, the algorithm can obtain very similar results. Therefore, here the proposed niching method is used in AR+sharing in order to investigate the difference of the algorithm framework. That is, AR+sharing and BiGE(AR) have the same proximity and crowding degree estimation methods and the only difference between them is the algorithm framework. In addition, it is worth noting that BiGE(AR) and BiGE have the same algorithm framework and the only difference between them is their proximity estimation.

Table 9 gives the HV results of the three algorithms on all the 34 test instances; for comparison, the results of BiGE are also included in the table. As shown, the diversity mechanism dramatically improves the HV results, with AR+sharing, BiGE(AR) and BiGE outperforming the original AR on all the 34 test instances. This suggests the importance of diversity maintenance in many-objective optimization.

Regarding the two algorithms having the same proximity and crowding degree estimators, BiGE(AR) performs better than AR+sharing in 31 out of the 34 instances. This clearly indicates the advantage of the bi-goal evolution framework for many-objective problems. In AR+sharing, the proximity and diversity information of an individual in the population is integrated into a scalar value. One may think that AR+sharing can work with a second negatively correlated “helper” objective (which is not explicitly related to diversity promotion). This type of structure has been presented in some “multiobjectivization” literature to solve single-objective problems [55]. However, AR+sharing under this structure may still struggle to balance proximity and diversity for many-objective problems, since there is no explicit diversity maintenance mechanism in the algorithm to guide the search towards different promising areas.

In addition, note that AR+sharing has the best HV result for the three WFG2 instances, which have a disconnected Pareto front. This is because AR+sharing can always find all the optimal regions of the Pareto front in all the 30 runs, while BiGE(AR) and BiGE can only do so in around half of the 30 runs.

Finally, considering the comparison between two versions of the bi-goal evolution algorithm. BiGE(AR) outperforms BiGE on 10 test instances (including seven 5-objective instances), while BiGE has a better HV result on the remaining 24 instances. An interesting observation is that the less fine-grained algorithm BiGE(AR) generally performs better on 5-objective instances. One possible explanation for this is that BiGE(AR) could prefer some boundary individuals in a population. These boundary solutions, which perform rather poorly on one objective but the best (or nearly the best) on other objectives, play an important role in extending the search range. Due to having no consideration of quantitative difference of individuals, BiGE(AR) would be in favor of these solutions. Nevertheless, it is worth pointing out that a fine-grained estimation of individual proximity can be more important in a high-dimensional space, where it is more needed to clearly differentiate between individuals.

7. Discussions

In BiGE, the diversity goal is estimated by a niche-based crowding degree. In this estimation, the radius of the niche depends on two factors: the population size and the number of objectives. A large population size (or low objective dimensionality) will lead to a small radius. The previous experiments (Section 6.1) have shown the effectiveness of BiGE under various settings of the population size and the number of objectives. This indicates that BiGE works well under this setting of the niche radius.

It is worth pointing out that the niche radius in the paper is a rough setting (estimate). A finely tuned setting based on the characteristics of a given MOP, such as varying with the Pareto front's shape, could lead to a better performance of the algorithm. Nevertheless, the algorithm under this radius setting has already shown high competitiveness against five state-of-the-art algorithms on diverse MOPs considered here. Also, this setting can benefit the applicability of the algorithm to real-world problems as it is hard (or even impossible) to know the problems' characteristics beforehand.

The major purpose of an EMO algorithm is to assist the decision-maker to select a single solution (or a few solutions) that fits his/her preferences [17,9,54,74]. However, since an EMO algorithm usually supply the decision-maker with an approximation of the whole Pareto front, it can be difficult for the decision-maker to choose his/her preferred one(s), especially in many-objective optimization. In spite of this, obtaining a Pareto front approximation with a set of well-distributed and well-converged solutions can be greatly useful to learn about the characteristics of the optimization problem. For example, the decision-maker can learn about the nature of the trade-offs among the objectives (e.g., discontinuousness, convexity, degeneration and knees) or discover inconsistencies of the model with regard to the real optimization problem [51]. This can help the decision-maker specify preferences that efficiently lead the search and eventually find a satisfied solution.

On the other hand, since the difficulties of representing the whole Pareto front as well as choosing a satisfied solution in a high-dimensional space, it could be appealing if an EMO algorithm can work collaboratively with the decision-maker preferences. This will lead to the search around the region of interest of the decision-maker.

Intuitively, there are three ways of implementing the incorporation of the decision-maker preferences into the proposed bi-goal evolution framework: 1) incorporating the preference information into the proximity goal, 2) incorporating the preference information into the diversity goal, and 3) incorporating the preference information into both the proximity and diversity goals. However, the first two ways may lead to the evolutionary population hard to get rid of some individuals that are far away from the region of interest of the decision-maker, since these individuals can perform very well on the other goal (i.e., the goal not including the preference information) and thus are nondominated in the population on the basis of the two goals. Therefore, it might be a better alternative to make both the proximity and diversity goals implicate the preference information (e.g., considering individuals' distance or relative position to the reference point supplied by the decision-maker). This could help the population evolve gradually towards the region of interest of the decision-maker while keep a relative balance between proximity and diversity in a high-dimensional objective space.

8. Conclusions

Many-objective optimization poses great challenges for EAs. The ineffectiveness of the Pareto dominance relation in a high-dimensional space suggests the need for new methodologies. This paper presents a meta-objective optimization approach, called BiGE, to deal with many-objective problems. Converting many objectives of a given problem into two

objectives of proximity and crowding degree, BiGE creates an optimization problem in which the objectives are the goals of the search process itself.

Systematic experiments have been carried out by providing extensive comparative studies between BiGE and five state-of-the-art algorithms on four groups of well-defined continuous and combinatorial benchmark suites with 5, 10, and 15 objectives. Unlike some peer algorithms, which work well on only a fraction of the test problems (e.g., AGE-II and MOEA/D on the 5-objective instances, HypE and NSGA-III on the continuous instances, and FD-NSGA-II on the combinatorial instances), BiGE can achieve a good balance between solutions' proximity and diversity on the test problems with different properties. In addition, the effect of several parameters on the algorithm was investigated. Experimental results have indicated the insensitiveness of BiGE to the population size and objective dimensionality and also the effectiveness of BiGE under different settings of the sharing discriminator within a certain range. Finally, a comparison with three AR-based algorithms has shown the advantage of the proposed framework and proximity estimation in dealing with many-objective problems.

Bi-goal evolution of proximity and diversity is a new concept in evolutionary multi-objective optimization. It performs the key decision process of mating and environmental selection in a two-layer process, with the lower level being a sort of recursive call to multi-objective optimization. This two-layer decision structure may open up many possibilities for hybridizations in the future, for example, instantiated by different comparison strategies in the selection process.

Despite high competitiveness of BiGE shown in our first attempt, more work is needed to further investigate its benefits and limitations in the future. In this regard, applying BiGE to more real-world problems and developing (or introducing) other proximity and crowding degree estimation methods are two focuses of our subsequent study.

Acknowledgements

The authors would like to thank Dr. Markus Wagner, School of Computer Science, University of Adelaide and Prof. Gary G. Yen, School of Electrical and Computer Engineering, Oklahoma State University for their help on carrying out the experiments. This work was supported in part by the Engineering and Physical Sciences Research Council (EPSRC) of UK under Grant EP/K001310/1, in part by the National Natural Science Foundation of China under Grant 71110107026, and in part by the EU FP7-Health under Grant 242193.

References

- [1] S.F. Adra, P.J. Fleming, Diversity management in evolutionary many-objective optimization, *IEEE Trans. Evol. Comput.* 15 (2011) 183–195.
- [2] H. Aguirre, K. Tanaka, Space partitioning with adaptive ϵ -ranking and substitute distance assignments: a comparative study on many-objective MNK-landscapes, in: *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, GECCO*, 2009, pp. 547–554.
- [3] A. Auger, J. Bader, D. Brockhoff, E. Zitzler, Theory of the hypervolume indicator: optimal μ -distributions and the choice of the reference point, in: *Proceedings of the 10th ACM SIGEVO Workshop on Foundations of Genetic Algorithms, FOGA*, 2009, pp. 87–102.
- [4] J. Bader, E. Zitzler, HypE: an algorithm for fast hypervolume-based many-objective optimization, *Evol. Comput.* 19 (2011) 45–76.
- [5] P.J. Bentley, J.P. Wakefield, Finding acceptable solutions in the Pareto-optimal range using multiobjective genetic algorithms, in: *Soft Computing in Engineering Design and Manufacturing*, 1997, pp. 231–240, chapter 5.
- [6] N. Beume, S-metric calculation by considering dominated hypervolume as Klee's measure problem, *Evol. Comput.* 17 (2009) 477–492.
- [7] N. Beume, B. Naujoks, M. Emmerich, SMS-EMOA: multiobjective selection based on dominated hypervolume, *Eur. J. Oper. Res.* 181 (2007) 1653–1669.
- [8] J. Bibai, P. Savéant, M. Schoenauer, V. Vidal, An evolutionary metaheuristic based on state decomposition for domain-independent satisficing planning, in: *Proc. 20th International Conference on Automated Planning and Scheduling, ICAPS*, 2010, pp. 18–25.
- [9] J. Branke, K. Deb, K. Miettinen, R. Slowinski (Eds.), *Multiobjective Optimization: Interactive and Evolutionary Approaches*, Springer-Verlag, Berlin, Heidelberg, 2008.
- [10] A.H. Brie, P. Morignot, Genetic planning using variable length chromosomes, in: *Proc. 20th International Conference on Automated Planning and Scheduling, ICAPS*, 2005, pp. 320–329.
- [11] K. Bringmann, T. Friedrich, Approximating the volume of unions and intersections of high-dimensional geometric objects, *Comput. Geom.* 43 (2010) 601–610.
- [12] K. Bringmann, T. Friedrich, An efficient algorithm for computing hypervolume contributions, *Evol. Comput.* 18 (2010) 383–402.
- [13] K. Bringmann, T. Friedrich, Approximation quality of the hypervolume indicator, *Artif. Intell.* 195 (2013) 265–290.
- [14] K. Bringmann, T. Friedrich, C. Igel, T. Voß, Speeding up many-objective optimization by Monte Carlo approximations, *Artif. Intell.* 204 (2013) 22–29.
- [15] K. Bringmann, T. Friedrich, F. Neumann, M. Wagner, Approximation-guided evolutionary multi-objective optimization, in: *Proceedings of the 22nd International Joint Conference on Artificial Intelligence, IJCAI*, 2011, pp. 1198–1203.
- [16] D. Brockhoff, T. Friedrich, N. Hebbinghaus, C. Klein, F. Neumann, E. Zitzler, On the effects of adding objectives to plateau functions, *IEEE Trans. Evol. Comput.* 13 (2009) 591–603.
- [17] C.A. Coello Coello, Handling preferences in evolutionary multiobjective optimization: a survey, in: *Proceedings of the Congress on Evolutionary Computation, CEC*, 2000, pp. 30–37.
- [18] D.W. Corne, J.D. Knowles, Techniques for highly multiobjective optimisation: some nondominated points are better than others, in: *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, GECCO*, 2007, pp. 773–780.
- [19] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley, New York, 2001.
- [20] 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 Trans. Evol. Comput.* 18 (2014) 577–601.
- [21] K. Deb, S. Jain, Running performance metrics for evolutionary multi-objective optimization, Technical Report 2002004, KanGAL, Indian Institute of Technology, 2002.

- [22] K. Deb, M. Mohan, S. Mishra, Evaluating the ϵ -domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions, *Evol. Comput.* 13 (2005) 501–525.
- [23] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.* 6 (2002) 182–197.
- [24] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable test problems for evolutionary multiobjective optimization, in: A. Abraham, L. Jain, R. Goldberg (Eds.), *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, Springer, Berlin, Germany, 2005, pp. 105–145.
- [25] M. Ehrgott, Multiobjective optimization, *AI Mag.* 29 (2009) 47–57.
- [26] P. Fleming, R. Purshouse, R. Lygoe, Many-objective optimization: an engineering design perspective, in: *Proceedings of the 3rd International Conference on Evolutionary Multi-Criterion Optimization*, EMO, 2005, pp. 14–32.
- [27] C. Fonseca, P. Fleming, Multiobjective genetic algorithms made easy: selection, sharing and mating restriction, in: *Proceedings of the First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, 1995, pp. 42–52.
- [28] C.M. Fonseca, P.J. Fleming, An overview of evolutionary algorithms in multiobjective optimization, *Evol. Comput.* 3 (1995) 1–16.
- [29] T. Friedrich, K. Bringmann, T. Voß, C. Igel, The logarithmic hypervolume indicator, in: *Proceedings of the 11th ACM SIGEVO Workshop on Foundations of Genetic Algorithms*, FOGA, 2011, pp. 81–92.
- [30] T. Friedrich, C. Horoba, F. Neumann, Multiplicative approximations and the hypervolume indicator, in: *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, GECCO, 2009, pp. 571–578.
- [31] T. Friedrich, F. Neumann, C. Thyssen, Multiplicative approximations, optimal hypervolume distributions, and the choice of the reference point, *Evol. Comput.* (2014), in press.
- [32] M. Garza-Fabre, G. Toscano-Pulido, C.A.C. Coello, Two novel approaches for many-objective optimization, in: *Proceedings of the IEEE Congress on Evolutionary Computation*, CEC, 2010, pp. 1–8.
- [33] M. Garza-Fabre, G. Toscano-Pulido, C.A.C. Coello, E. Rodriguez-Tello, Effective ranking + speciation = many-objective optimization, in: *Proceedings of the IEEE Congress on Evolutionary Computation*, CEC, 2011, pp. 2115–2122.
- [34] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.
- [35] D. Hadka, P. Reed, Diagnostic assessment of search controls and failure modes in many-objective evolutionary optimization, *Evol. Comput.* 20 (2012) 423–452.
- [36] D. Hadka, P. Reed, Borg: an auto-adaptive many-objective evolutionary computing framework, *Evol. Comput.* 21 (2013) 231–259.
- [37] Z. He, G.G. Yen, A new fitness evaluation method based on fuzzy logic in multiobjective evolutionary algorithms, in: *Proceedings of the IEEE Congress on Evolutionary Computation*, CEC, 2012, pp. 1–8.
- [38] Z. He, G.G. Yen, Ranking many-objective evolutionary algorithms using performance metrics ensemble, in: *Proceedings of the IEEE Congress on Evolutionary Computation*, CEC, 2013, pp. 2480–2487.
- [39] Z. He, G.G. Yen, J. Zhang, Fuzzy-based Pareto optimality for many-objective evolutionary algorithms, *IEEE Trans. Evol. Comput.* 18 (2014) 269–285.
- [40] S. Huband, P. Hingston, L. Barone, L. While, A review of multiobjective test problems and a scalable test problem toolkit, *IEEE Trans. Evol. Comput.* 10 (2006) 477–506.
- [41] E.J. Hughes, Many-objective directed evolutionary line search, in: *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, GECCO, 2011, pp. 761–768.
- [42] C. Igel, N. Hansen, S. Roth, Covariance matrix adaptation for multi-objective optimization, *Evol. Comput.* 15 (2007) 1–28.
- [43] K. Ikeda, H. Kita, S. Kobayashi, Failure of Pareto-based MOEAs: does non-dominated really mean near to optimal?, in: *Proceedings of the IEEE Congress on Evolutionary Computation*, CEC, 2001, pp. 957–962.
- [44] H. Ishibuchi, N. Akedo, Y. Nojima, Behavior of multi-objective evolutionary algorithms on many-objective knapsack problems, *IEEE Trans. Evol. Comput.* (2014), in press.
- [45] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, Y. Nojima, Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space, in: *Proceedings of the International Conference on Parallel Problem Solving from Nature*, PPSN, 2010, pp. 91–100.
- [46] H. Ishibuchi, H. Masuda, Y. Nojima, Meta-level multi-objective formulations of set optimization for multi-objective optimization problems: multi-reference point approach to hypervolume maximization, in: *Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation*, GECCO, 2014, pp. 89–90.
- [47] H. Ishibuchi, Y. Sakane, N. Tsukamoto, Y. Nojima, Evolutionary many-objective optimization by NSGA-II and MOEA/D with large populations, in: *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, 2009, pp. 1758–1763.
- [48] H. Ishibuchi, N. Tsukamoto, Y. Nojima, Evolutionary many-objective optimization: a short review, in: *Proceedings of the IEEE Congress on Evolutionary Computation*, CEC, 2008, pp. 2419–2426.
- [49] H. Ishibuchi, N. Tsukamoto, Y. Sakane, Y. Nojima, Indicator-based evolutionary algorithm with hypervolume approximation by achievement scalarizing functions, in: *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation*, GECCO, ACM, 2010, pp. 527–534.
- [50] A.L. Jaimes, C.A. Coello Coello, Study of preference relations in many-objective optimization, in: *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, GECCO, 2009, pp. 611–618.
- [51] A.L. Jaimes, C.A. Coello Coello, H. Aguirre, K. Tanaka, Adaptive objective space partitioning using conflict information for many-objective optimization, in: *Proceedings of the 6th International Conference on Evolutionary Multi-Criterion Optimization*, EMO, 2011, pp. 151–165.
- [52] A.L. Jaimes, L.V.S. Quintero, C.A. Coello Coello, Ranking methods in many-objective evolutionary algorithms, in: R. Chiong (Ed.), *Nature-Inspired Algorithms for Optimisation*, Springer, Berlin, Germany, 2009, pp. 413–434.
- [53] 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 Trans. Evol. Comput.* 18 (2014) 602–622.
- [54] A. Jaszkiewicz, J. Branke, *Interactive multiobjective evolutionary algorithms*, in: *Multiobjective Optimization*, Springer, 2008, pp. 179–193.
- [55] M.T. Jensen, Helper-objectives: using multi-objective evolutionary algorithms for single-objective optimisation, *J. Math. Model. Algorithms* 3 (2004) 323–347.
- [56] D. Jones, M. Jimenez, Incorporating additional meta-objectives into the extended lexicographic goal programming framework, *Eur. J. Oper. Res.* 227 (2013) 343–349.
- [57] R. Joshi, B. Deshpande, Scalability of population-based search heuristics for many-objective optimization, in: *Proceedings of the 16th European Conference on Applications of Evolutionary Computation*, 2013, pp. 479–488.
- [58] M. Khoudjaia, M. Schoenauer, V. Vidal, J. Dréo, P. Savéant, Pareto-based multiobjective AI planning, in: *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI, 2013, pp. 2321–2327.

- [59] W. Kong, J. Ding, T. Chai, J. Sun, Large-dimensional multi-objective evolutionary algorithms based on improved average ranking, in: 49th IEEE Conference on Decision and Control, CDC, 2010, pp. 502–507.
- [60] N. Kowatari, A. Oyama, H.E. Aguirre, K. Tanaka, A study on large population MOEA using adaptive ϵ -box dominance and neighborhood recombination for many-objective optimization, in: Proceedings of the 6th Learning and Intelligent Optimization Conference, LION, 2012, pp. 86–100.
- [61] M. Laumanns, L. Thiele, K. Deb, E. Zitzler, Combining convergence and diversity in evolutionary multiobjective optimization, *Evol. Comput.* 10 (2002) 263–282.
- [62] M. Laumanns, R. Zenklusen, Stochastic convergence of random search methods to fixed size Pareto front approximations, *Eur. J. Oper. Res.* 213 (2011) 414–421.
- [63] H. Li, Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, *IEEE Trans. Evol. Comput.* 13 (2009) 284–302.
- [64] K. Li, Q. Zhang, S. Kwong, M. Li, R. Wang, Stable matching based selection in evolutionary multiobjective optimization, *IEEE Trans. Evol. Comput.* 18 (2014) 909–923.
- [65] M. Li, S. Yang, X. Liu, Diversity comparison of Pareto front approximations in many-objective optimization, *IEEE Trans. Cybern.* 44 (2014) 2568–2584.
- [66] M. Li, S. Yang, X. Liu, Shift-based density estimation for Pareto-based algorithms in many-objective optimization, *IEEE Trans. Evol. Comput.* 18 (2014) 348–365.
- [67] M. Li, S. Yang, X. Liu, A test problem for visual investigation of high-dimensional multi-objective search, in: Proceedings of the IEEE Congress on Evolutionary Computation, CEC, 2014, pp. 2140–2147.
- [68] M. Li, S. Yang, X. Liu, A performance comparison indicator for Pareto front approximations in many-objective optimization, in: Proceedings of the 17th Annual Conference on Genetic and Evolutionary Computation, GECCO, 2015, pp. 703–710.
- [69] M. Li, S. Yang, X. Liu, R. Shen, A comparative study on evolutionary algorithms for many-objective optimization, in: Proceedings of the 7th International Conference on Evolutionary Multi-Criterion Optimization, EMO, 2013, pp. 261–275.
- [70] M. Li, J. Zheng, K. Li, Q. Yuan, R. Shen, Enhancing diversity for average ranking method in evolutionary many-objective optimization, in: Proceedings of the International Conference on Parallel Problem Solving from Nature, PPSN, 2010, pp. 647–656.
- [71] S. Mostaghim, H. Schmeck, Distance based ranking in many-objective particle swarm optimization, in: Proceedings of the International Conference on Parallel Problem Solving from Nature, PPSN, 2008, pp. 753–762.
- [72] K. Musselman, J. Talavage, A trade-off cut approach to multiple objective optimization, *Oper. Res.* 28 (1980) 1424–1435.
- [73] M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems*, Pearson Education, 2005.
- [74] R.C. Purshouse, K. Deb, M.M. Mansor, S. Mostaghim, R. Wang, A review of hybrid evolutionary multiple criteria decision making methods, in: IEEE Congress on Evolutionary Computation, CEC, 2014, pp. 1147–1154.
- [75] R.C. Purshouse, P.J. Fleming, On the evolutionary optimization of many conflicting objectives, *IEEE Trans. Evol. Comput.* 11 (2007) 770–784.
- [76] R.C. Purshouse, C. Jalbá, P.J. Fleming, Preference-driven co-evolutionary algorithms show promise for many-objective optimisation, in: Proceedings of the 6th International Conference on Evolutionary Multi-Criterion Optimization, EMO, 2011, pp. 136–150.
- [77] T. Ray, K. Tai, K.C. Seow, Multiobjective design optimization by an evolutionary algorithm, *Eng. Optim.* 33 (2001) 399–424.
- [78] H. Sato, H. Aguirre, K. Tanaka, Controlling dominance area of solutions and its impact on the performance of MOEAs, in: Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization, EMO, 2007, pp. 5–20.
- [79] H. Sato, H.E. Aguirre, K. Tanaka, Self-controlling dominance area of solutions in evolutionary many-objective optimization, in: Proceedings of the 8th International Conference on Simulated Evolution and Learning, SEAL, 2010, pp. 455–465.
- [80] D.K. Saxena, J.A. Duro, A. Tiwari, K. Deb, Q. Zhang, Objective reduction in many-objective optimization: linear and nonlinear algorithms, *IEEE Trans. Evol. Comput.* 17 (2013) 77–99.
- [81] H.K. Singh, A. Isaacs, T. Ray, A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems, *IEEE Trans. Evol. Comput.* 15 (2011) 539–556.
- [82] A. Toffolo, E. Benini, Genetic diversity as an objective in multi-objective evolutionary algorithms, *Evol. Comput.* 11 (2003) 151–167.
- [83] M. Wagner, T. Friedrich, Efficient parent selection for approximation-guided evolutionary multi-objective optimization, in: Proceedings of the IEEE Congress on Evolutionary Computation, CEC, 2013, pp. 1846–1853.
- [84] M. Wagner, F. Neumann, A fast approximation-guided evolutionary multi-objective algorithm, in: Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation, GECCO, 2013, pp. 687–694.
- [85] T. Wagner, N. Beume, B. Naujoks, Pareto-, aggregation-, and indicator-based methods in many-objective optimization, in: Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization, EMO, 2007, pp. 742–756.
- [86] H. Wang, L. Jiao, X. Yao, An improved two-archive algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.* (2014), in press.
- [87] R. Wang, R.C. Purshouse, P.J. Fleming, On finding well-spread Pareto optimal solutions by preference-inspired co-evolutionary algorithm, in: Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation, GECCO, 2013, pp. 695–702.
- [88] R. Wang, R.C. Purshouse, P.J. Fleming, Preference-inspired co-evolutionary algorithms for many-objective optimisation, *IEEE Trans. Evol. Comput.* 17 (2013) 474–494.
- [89] Y. Wang, Z. Cai, Combining multiobjective optimization with differential evolution to solve constrained optimization problems, *IEEE Trans. Evol. Comput.* 16 (2012) 117–134.
- [90] L. While, L. Bradstreet, L. Barone, A fast way of calculating exact hypervolumes, *IEEE Trans. Evol. Comput.* 16 (2012) 86–95.
- [91] U.K. Wickramasinghe, X. Li, Using a distance metric to guide PSO algorithms for many-objective optimization, in: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, GECCO, 2009, pp. 667–674.
- [92] S. Yang, M. Li, X. Liu, J. Zheng, A grid-based evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.* 17 (2013) 721–736.
- [93] Y. Yuan, H. Xu, B. Wang, Evolutionary many-objective optimization using ensemble fitness ranking, in: Proceedings of the 16th Annual Conference on Genetic and Evolutionary Computation, GECCO, 2014, pp. 669–676.
- [94] Q. Zhang, H. Li, MOEA/D: a multiobjective evolutionary algorithm based on decomposition, *IEEE Trans. Evol. Comput.* 11 (2007) 712–731.
- [95] Q. Zhang, W. Liu, H. Li, The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances, in: Proceedings of the IEEE Congress on Evolutionary Computation, CEC, 2009, pp. 203–208.
- [96] X. Zhang, Y. Tian, Y. Jin, A knee point driven evolutionary algorithm for many-objective optimization, *IEEE Trans. Evol. Comput.* (2014), in press.
- [97] A. Zhou, B. Qu, H. Li, S. Zhao, P. Suganthan, Q. Zhang, Multiobjective evolutionary algorithms: a survey of the state of the art, *Swarm Evol. Comput.* 1 (2011) 32–49.

- [98] E. Zitzler, J. Knowles, L. Thiele, Quality assessment of Pareto set approximations, in: J. Branke, K. Deb, K. Miettinen, R. Slowinski (Eds.), *Multiobjective Optimization*, vol. 5252, Springer, Berlin, Heidelberg, 2008, pp. 373–404.
- [99] E. Zitzler, S. Künzli, Indicator-based selection in multiobjective search, in: *Proceedings of the International Conference on Parallel Problem Solving from Nature, PPSN, 2004*, pp. 832–842.
- [100] E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach, *IEEE Trans. Evol. Comput.* 3 (1999) 257–271.
- [101] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, *IEEE Trans. Evol. Comput.* 7 (2003) 117–132.