# Niche-based and Angle-based Selection Strategies for Many-Objective Evolutionary Optimization

Jinlong Zhou[a,b,*], Juan Zou[a,b,*], Shengxiang Yang[a,d], Jinhua Zheng[a,c], Dunwei Gong[e], Tingrui Pei[a,b]

[a] *Key Laboratory of Hunan Province for Internet of Things and Information Security, Xiangtan University, Xiangtan, Hunan, 411105, China*
[b] *Key Laboratory of Intelligent Computing and Information Processing, Ministry of Education, School of Computer Science, Xiangtan University, Xiangtan, Hunan, 411105, China*
[c] *Hunan Provincial Key Laboratory of Intelligent Information Processing and Application, Hengyang, 421002, China*
[d] *School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH,U.K.*
[e] *School of Information and Electronic Engineering, China University of Mining and Technology, Xuzhou, 221116, China*

**Abstract**

It is well known that balancing population diversity and convergence plays a crucial role in evolutionary many-objective optimization. However, most existing multiobjective evolutionary algorithms encounter difficulties in solving many-objective optimization problems. Thus, this paper suggests niche-based and angle-based selection strategies for many-objective evolutionary optimization. In the proposed algorithm, two strategies are included: niche-based density estimation strategy and angle-based selection strategy. Both strategies are employed in the environmental selection to eliminate the worst individual from the population in an iterative way. To be specific, the former estimates the diversity of each individual and finds the most crowded area in the population. The latter removes individuals with weak convergence in the same niche. Experimental studies on several well-known benchmark problems show that the proposed algorithm is competitive compared with six state-of-the-art many-objective algorithms. Moreover, the proposed algorithm has

*Jinlong Zhou and Juan Zou contributed equally to this work.
*Email address:* zoujuan@xtu.edu.cn (Juan Zou )

also been verified to be scalable to deal with constrained many-objective optimization problems.

*Keywords:* Many-objective optimization, Pareto optimality, niche, angle-based selection.

## 1. INTRODUCTION

Many real-world optimization problems involve more than one conflicting objective. These problems are often referred to as multiobjective optimization problems (MOPs). MOPs with at least four conflicting objectives are known as many-objective optimization problems (MaOPs). Many-objective optimization research has become a hot topic due to its wide applications in the real world [1, 2].

Without loss of generality, an unconstrained MOP (or MaOP) can be mathematically defined as [3]:

$$\begin{aligned} &min\ \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})), \\ &subject\ to\ \mathbf{x}\ \in\ \Omega, \end{aligned} \tag{1}$$

where $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ represents an $n$-dimensional decision variable vector from the decision space $\Omega$, and $\mathbf{F}(\mathbf{x})$ defines $m$ objective functions. Due to the conflicts often existing in different objectives, there is no absolute or unique optimal solution, but a set of solutions representing tradeoffs among different objectives. For solution $\mathbf{x}_1$, $\mathbf{x}_2 \in \Omega$, $\mathbf{x}_1$ is said to Pareto dominate $\mathbf{x}_2$, if and only if $f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2)$ for each $1 \leq i \leq m$ and there exists at least one $j \in \{1, \ldots, m\}$ such that $f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2)$. If there is no other solution $\mathbf{x} \in \Omega$ Pareto-dominating $\mathbf{x}^*$, then $\mathbf{x}^*$ is nondominated or the Pareto optimal solution. The set of all Pareto optimal solutions is called the Pareto optimal set (PS), and when mapped into objective space is termed the Pareto optimal front (PF).

Being straightforward and free from derivatives, multiobjective evolutionary algorithms (MOEAs) have been demonstrated to be an efficient approach for solving MOPs [4]. Many MOEAs have been proposed to solve MOPs, for example, NSGA-II [5] and SPEA2 [6]. Among these MOEAs, Pareto dominance-based approaches are the most popular class of methods in the community. NSGA-II and SPEA2 are representative Pareto-based approaches, which have achieved gratifying results on the optimization of two-or three-objective MOPs. However, their performance dramatically deteriorates when dealing with MaOPs. The primary reason is that almost all the solutions in the population become nondominated when the number of objectives increases [7].

To enhance the scalability of MOEAs to handle MaOPs, a variety of approaches have been developed in recent years. These can be roughly summarized by the following categories [8].

The first category of approaches redefines the dominance relationship to increase the selection pressure towards the PF. This category includes: $L$-optimality [9] and preference order ranking [10]. Although these modified dominance relationships can improve the convergence of MOEAs for MaOPs, they may lead to a decrease in the diversity of the solutions. Grid-based algorithms, for example, $\varepsilon$-MOEA [11] and GrEA [12], relax the dominance relationship. This series of algorithms makes full use of the grid position to distinguish individuals, so as to maintain a good distribution of the population. However, adaptively setting the size of the grid for different optimization problems is a problem that needs to be solved.

The second category aims to improve the diversity maintenance mechanism in the algorithm. For instance, Li *et al.* [13] proposed a shift-based density estimation (SDE) strategy. The strategy covers both the diversity and convergence information of individuals, which can effectively discard solutions that lead to poor convergence. What's more, a knee point-driven evolutionary algorithm (KnEA) [14] was proposed, which is based on the idea of preferring the knee points among nondominated solutions.

The third category covers various decomposition-based methods. Existing methods belonging to this category contain two different types. The first type aggregates the objectives into several scalarizing functions, each of which generates a single scalar value. MOEA/D [15] decomposes a MOP into a number of scalar optimization subproblems and optimizes them simultaneously. The second type divides a MaOP into a group of sub-MaOPs. In NSGA-III [16] and MOEA/DD [17], the objective space is partitioned into a number of subspaces using the reference vectors to manage nondominated solutions. For both types, one key point is the setting of weight vectors, which have significant influences on the diversity of a population [18]. To achieve good performance in MaOPs with an irregular PF, a series of decomposition-based adaptive methods, such as ESOEA [19] and A-NSGA-III [20], has been produced.

The fourth category is known as the indicator-based approaches. Population evolution is usually guided by a single indicator, which accounts for both convergence and diversity performances of the solution set. Among others, hypervolume (HV) [21] and Inverted Generational Distance (IGD) [22] are the most commonly used indicators. The shortcoming of HV-based approaches is that the computational complexity for calculating hypervolume increases exponentially as the number of objectives increases. For IGD-based methods, how to obtain a set of uniformly distributed reference points for the calculation of the IGD indicator is a crucial issue.

The last category is the objective reduction methods. The natural idea is to reduce the number of optimization objectives and maintain as much information of all objectives as possible. Pal *et al.* [23] proposed differential evolution using clustering-based objective reduction. Bandyopadhyay and Mukherjee [24] periodically reorder the objectives based on their conflict status and select $\alpha$ conflicting objectives for further processing. Additionally, He *et al.* [25] proposed a strategy to make the whole population quickly approach a small number of target points near the true PF, to reduce the objective space.

Although these studies have greatly improved the ability of MOEAs to tackle MaOPs, these methods all have their own defects, and it is theoretically impossible to have an optimization method that is most suitable for all problems [26]. Among MaOPs, the proportion of nondominated solutions is too large to distinguish among different solutions, which is a great obstacle in the process of solving MaOPs. Many efforts have been made to modify the dominance relationship to increase the pressure on environmental selection [9, 10, 11, 12]. Moreover, the most straightforward way to increase selection pressure is to select individuals according to their convergence performance. However, if we choose individuals based on the convergence indicator, only a small part of the PF can be obtained [27]. In order to address this issue, mechanisms to enhance the distribution of the population are needed. Intuitively, when an individual is selected, its neighbor individual will be punished. However, how to implement the punishment operation (i.e., determine the neighbor individual and assign how much it will be punished) is not a trivial task. For example, the grid ranking adjustment process of GrEA introduces several penalty parameters.

This paper proposes a novel algorithm using niche-based and angle-based selection strategies for many-objective evolutionary optimization (NAEA). The task of environmental selection is to choose some promising individuals from the union population, which is composed of the parent and offspring populations. The environmental selection of NAEA is mainly composed of two components. First, the nondominated sort is used to eliminate a few worse candidate solutions and identify some excellent individuals. Then, the worst individual is found by the niching method, which takes an individual's convergence and diversity into consideration. Therefore, the proposed NAEA belongs to the second class of approaches for solving MaOPs discussed above.

The contributions of this paper can be summarized as follows:

- Different from mainstream decomposition-based and indicator-based methods, this paper devises the niche-based density estimation strategy and angle-based selection strategy to strike a balance of diversity and convergence. Specifically, individuals with poor diversity are identified by niche density estimation, and the ones with worse convergence are removed via angle-based selection.

- Systematic simulation experiments have been performed on the DTLZ and WFG test suites to demonstrate the effectiveness of NAEA. Compared with the other six state-of-the-art MaOEAs, our proposed algorithm achieved better performance in terms of widely used performance metrics: IGD [22], HV [21], SPREAD [28], etc. Additionally, NAEA can be easily extended to constrained optimization problems and has promising results.

The remainder of this paper is organized as follows: Section 2 introduces the background knowledge of this work. Then the proposed algorithm is elaborated in Section 3. To evaluate the performance of our algorithm in addressing MaOPs, a series of experiments is presented and analyzed in Section 4. Finally, conclusions are drawn in Section 5.

## 2. PRELIMINARIES

In this section, we present the motivation of our paper, and then several concepts related to our proposed algorithm are introduced.

### *2.1. Motivation*

Many existing methods use the Euclidean distance to estimate the distribution of the population. Unfortunately, the Euclidean distance has been shown to be unsuitable for determining the distribution of high-dimensional space [29], especially when the number of objectives increases. BCE-IBEA

[30] and NPGA [31] employed the Euclidean distance to estimate the distribution of solutions, which often selects the dominance resistant solutions (DRSs, i.e., the solutions with a quite poor value at least one objective but with near optimal values in some other objectives) [32]. For instance, consider the case in Fig. 1(a), A and G (both are DRSs) are considered to have good distribution because there are no other individuals in their niche. As a matter of fact, they are apart from Pareto optimal.

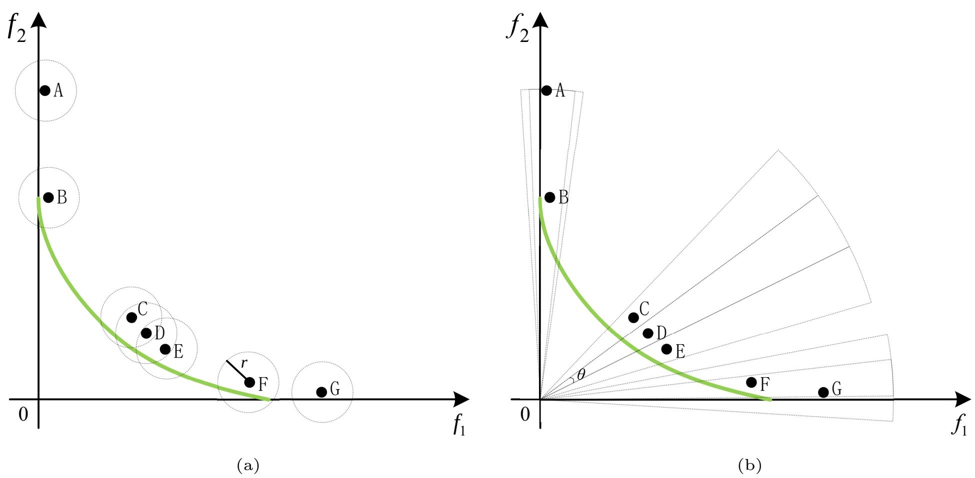



Figure 1: Illustrative examples of diversity estimates. (a) Estimation of diversity based on Euclidean distance niche, $r$ is the niche radius. (b) Estimation of diversity based on angle niche, $\theta$ is the niche radius.

This paper proposes an algorithm which removes the poorest individuals (in terms of diversity and convergence) from the population one by one. The proposed algorithm uses niche-based techniques to identify the most crowded regions. In traditional methods, the niche's radius is determined by the Euclidean distance [30, 31] of solutions. However, as discussed, the Euclidean distance is not well suited for measuring distribution in a high-dimensional space. The angle similarity has been widely used for measuring similarity in a high-dimensional space, for example, in document clustering [33] and VaEA [34]. Therefore, we use the angle information between individuals to determine the niche radius. Compared with the Euclidean distance based niche density estimation of Fig. 1(a), the angle-based niche density estimation of Fig. 1(b) is more objective in estimating the density of individuals. A and G are considered to be in crowded areas because there are other individuals in their angle-based niche. This distribution estimation method is very helpful in reducing the DRSs.

At the same time, the convergence of the population must also be taken into consideration. Nondominated sort can promote population convergence and prevent the final selected solution from being dominated by other solutions. What's more, the convergence of the algorithm is effectively guaranteed by searching for individuals with the most similar search directions and deleting individuals with the worst convergence indicators. In summary, the proposed algorithm removes individuals from the population by using an iterative method, which can adaptively estimate the convergence and diversity of the population as a whole in the evolutionary process. Thus, the worst individual can be deleted accurately, so both the diversity and convergence of the algorithm can be better balanced.

### *2.2. Concepts*

1) *Normalization:* Firstly, we find the ideal point $Z^{*}=(Z_1^{min}, Z_2^{min}, ..., Z_m^{min})^T$ and estimate the nadir point $Z^{nadir}=(Z_1^{max}, Z_2^{max}, ..., Z_m^{max})^T$, where the $Z_i^{min}$ and the $Z_i^{max}$ are the minimum and maximum values of the $i$th objective of all individuals in objective space, respectively. Then, for the $j$th individual $\mathbf{x}_j$, its objective vector $\mathbf{F}(\mathbf{x}_j)$ can be normalized as $\mathbf{F}'(\mathbf{x}_j) = (f_1'(\mathbf{x}_j), f_2'(\mathbf{x}_j), ..., f_m'(\mathbf{x}_j))$ according to the following equation:

$$f_i'(\mathbf{x}_j) = \frac{f_i(\mathbf{x}_j) - z_i^{min}}{z_i^{max} - z_i^{min}}, i = 1, 2, ..., m. \tag{2}$$

2) *Objective Vector Angle* [34]: After normalization of the objective vector, the definition of the angle between any two solutions $\mathbf{x}$ and $\mathbf{y}$ is given as follows:

$$Angle(\mathbf{x}, \mathbf{y}) = \arccos\left|\frac{\mathbf{F}'(\mathbf{x}) \bullet \mathbf{F}'(\mathbf{y})}{\|\mathbf{F}'(\mathbf{x})\| \times \|\mathbf{F}'(\mathbf{y})\|}\right|, \tag{3}$$

where $\mathbf{F}'(\mathbf{x}) \bullet \mathbf{F}'(\mathbf{y})$ denotes the inner product of $\mathbf{F}'(\mathbf{x})$ and $\mathbf{F}'(\mathbf{y})$, $\|\cdot\|$ is the two-norm of a vector, and $|\cdot|$ denotes the absolute value sign.

We only care about the acute angle between objective vectors of solutions. Given any two individuals, the angle between them in the objective space is assumed to be $\phi$. It is clear that the angle between any two solutions is less than or equal to $\pi/2$, which means that the range of $\phi$ between any two solutions belongs in $[0, \pi/2]$.

3) *Extreme Solutions* [34]: The extreme solution is defined as the solution with the smallest angle to the coordinate axis in the objective space. For the $j$th axis, the extreme solution $\mathbf{e}_j$ is defined as the solution with the smallest vector angle to the axis vector $\mathbf{w} = (w_1, w_2, \ldots, w_m)^T$.

$$\mathbf{e}_j = \arg\min_{\mathbf{x}\in\Omega}\{Angle(\mathbf{x}, \mathbf{w})\}, \tag{4}$$

$$w_i = \begin{cases} 1, & i = j \\ 1e-6, & i \neq j \end{cases} \quad i \in \{1, 2, \ldots, m\}, \tag{5}$$

where $i$ denotes the $i$th objective and $m$ is the objective number.

4) *Density Estimation* [13]: Density estimation strategies always estimate the similarity degree of individuals in the population. They consider the mutual positional relationship between an individual and other individuals to estimate its density. Specifically, the density of an individual $p$ in the population $P$ can be expressed as:

$$Density(p) = D(dist(p, q_1), dist(p, q_2), ..., dist(p, q_{n-1})), \tag{6}$$

where $q_i \in P$ and $q_i \neq p$; $n$ is the size of $P$. $dist(p, q)$ is the similarity degree of $p$ and $q$, usually measured by its angle or distance. $D()$ denotes the function to calculate the similarity degree between the interested individual and other individuals in the population.

## 3. PROPOSED ALGORITHM

### *3.1. Framework*

The framework of the NAEA is described in Algorithm 1. The basic procedure of the algorithm is similar to the traditional evolutionary many-objective optimization algorithms, such as NSGA-III and GrEA. First, the initial population $P$ with $N$ members is randomly generated in the decision space. Then, the evolutionary operators, including mating selection and variation, are executed to generate offspring individuals. Specifically, an offspring population $Q$ is generated from $P$ through mating selection and variation. Finally, the environmental selection procedure is implemented to select $N$ elitist solutions from the population $R$, which is a combination of $P$ and $Q$. The evolution procedure is repeated until some stopping criterion is satisfied. The key components of NAEA are described in detail as follows.

**Algorithm 1** The Framework of NAEA

**Input:** $N$ (population size)
**Output:** $P$ (the final population)

```
1: P = Initialize(N)
2: while the termination criterion is not satisfied do
3:     P' = Mating_Selection(P)
4:     Q = Variation(P')
5:     R = P ∪ Q
6:     P = Environmental_Selection(R)
7: end while
8: Return P
```

*3.2. Mating selection and variation*

Mating selection and variation aim to generate a number of offspring. They are usually implemented by selecting promising solutions from the current population to form a mating pool. NAEA does not use a special mechanism to select individuals that make up the mating pool, nor does it use a special crossover mutation operator to generate offspring. Instead, the binary tournament selection is used as the mating selection. The simulated binary crossover ($SBX$) and the polynomial mutation are utilized to generate $Q$. The reasons can be summarized as follows: the binary tournament selection, $SBX$, and polynomial mutation are widely used in many-objective optimization algorithms. Such a measure also better guarantees the fairness of our algorithm compared with other algorithms.

*3.3. Environmental selection*

The purpose of environmental selection is to select fitter individuals for the next generation. Similar to the NSGA-II, NAEA selects fitter individuals from a combination of the parent and offspring populations for the next generation. However, NAEA's niche-based and angle-based selection mechanism is quite different from other many-objective evolutionary algorithms. Algorithm 2 describes the environmental selection procedure.

In Algorithm 2, NAEA considers the dominant relationship in candidate solution set $R$. The candidate solutions are divided into several different fronts $(F_1, F_2, ..., F_l, ...)$ by the efficient non-dominant sorting (ENS) algo-

**Algorithm 2** Environmental selection ($R$)

**Input:** $R$ (the combination of $P$ and $Q$)
**Output:** $P$ (next population)

$P = \emptyset$ // Generate an empty set $P$ for archive
$\{F_1, F_2, \ldots, F_l, \ldots\}$ = Pareto_Nondominated_Sort($R$)
$P = F_1 \cup F_2 \cup ... \cup F_{l-1}$
**if** $|P| = N$ **then**
    return $P$
**else**
    $S = F_1 \cup F_2 \cup ... \cup F_{l-1} \cup F_l$
    Calculating the $Angle$ between any two individuals in $R$ according to equation (2) and (3)
    $\theta$ = niche_radius($Angle$)
    $Ncd$ = Niche crowding degree($S$, $\theta$, $Angle$)
    Add $m$ extreme solutions into $S$
    **while** $|S| > N$ **do**
        $x$ = Find_worst($F_l$, $Ncd$, $\theta$, $Angle$)
        $F_l = F_l \setminus \{x\}$
        Update_Ncd($S$, $Ncd$, $x$, $\theta$, $Angle$)
    **end while**
**end if**
$P = P \cup F_l$
return $P$

rithm [35]. Then, the critical front [1] is found, and correspondingly the first $(l-1)$ nondominated fronts ($F_1, F_2, ..., F_{l-1}$) are moved into the archive.

Since optimization problems may have different ranges for each objective, the population $R = P \cup Q$ is recommended to be normalized. In NAEA, we use the method in equation (2) to adaptively normalize $P \cup Q$. Hereafter, when the objective values of a solution are mentioned, we always refer to the normalized ones. Then, the vector angles between any two individuals in $R$ are calculated according to equation (3).

After calculating vector angles, the parameter $\theta$ that determines the size

[1]This paper calls $F_l$ the critical front, when ($|F_1 \cup F_2 \cup ... \cup F_{l-1}| \leq N$ *and* $|F_1 \cup F_2 \cup ... \cup F_l| > N$, where $N$ denotes the archive size)

of the niche is determined. To ensure that $\theta$ can better distinguish individuals of the population and can be adaptive in the evolutionary process of each generation, $\theta$ is set to be the median of the vector angle from all the solutions to their $m$th nearest solution in the set.

To maintain a good distribution of the population, this paper adopts a niche density estimation strategy to calculate the diversity information of each individual in the population (line 11 in Algorithm 2). Additionally, the extreme solution shares a similar concept of the extreme solution defined in [34], and $m$ extreme solutions are added into the $S$ (line 12). The inclusion of extreme solutions aims to promote diversity, especially the extensiveness of the population.

As part of the process of eliminating individuals one by one, function Find_worst (line 14) in Algorithm 2 is designed to find the worst individual $x$ in the $F_l$. Then, $x$ is eliminated from $F_l$. Then, because there is an individual removed from the population, we need to update the niche crowding degree ($Ncd$) of the individuals in the population.

**Algorithm 3** Niche crowding degree($S$, $\theta$, $Angle$)

**Input:** $S$, $\theta$, $Angle$
**Output:** $Ncd$
**for** each $p \in S$ **do**
    **for** each $q \in S$ & $p \neq q$ **do**
        **if** $Angle(p, q) < \theta$ **then**
            $sh(p, q) = Angle(p, q)/\theta$
        **else**
            $sh(p, q) = 1$
        **end if**
    **end for**
    $Ncd$ = calculating niche crowding degree according to equation (7)
**end for**
return $Ncd$

### *3.4. Niche crowding degree*

In evolutionary algorithms, the niche technique, which simulates the phenomenon that things of one kind come together, is often used to maintain population diversity in multimodal optimization [36]. The niche technique originated from the idea of resource sharing which derates each population solutions fitness by an amount related to the number of similar solutions in the population. In BCE-IBEA [30], a niching method based on Euclidean distance, which considers both the number and the location of the solutions in its niche, is proposed to maintain the diversity. Nevertheless, as discussed in Section 2.1, the Euclidean distance is not well suited to distinguishing individuals in high-dimensional space. In NAEA, the niche technology makes full use of angle information between individuals, which is shown in Algorithm 3. Formally, the density of a solution $p$ in the population $S$ is defined as:

$$Ncd(p) = 1 - \prod_{q \in S, q \neq p} sh(p, q), \tag{7}$$

$$sh(p, q) = \begin{cases} Angle(p,q)/\theta, & \text{if } Angle(p,q)<\theta, \\ 1, & \text{otherwise,} \end{cases} \tag{8}$$

where $Angle(p,q)$ denotes the vector angle between solutions $p$ and $q$. The parameter $\theta$ is the niche radius. The goal is to make most individuals have one or several neighbors in their niche, so the niche radius is set to the median of the angle from all the individuals to their $m$th nearest individual in the population. $m$ is set to the number of objectives for the problem. This results in fewer parameters for the algorithm and allows the niche radius to adaptively change with the number of objectives. The $Ncd$ value of an individual lies between [0,1], with a lower value being preferable. Clearly, the niche crowding degree depends on the number of neighbors (i.e., individuals in the niche) and the angle between the neighbors and it. If an individual has more neighbors or a smaller angle with its neighbors, it is likely to obtain a bigger $Ncd$ value, indicating that it is more crowded.

### *3.5. Find_worst*

First, the most crowded individual, $x_p$, is identified based on $Ncd$. The set $U$ is produced by the individuals from the critical front in the same niche as $x_p$. Namely, the vector angle between each individual in $U$ and $x_p$ is less than $\theta$. Then, the solution $x_q$ in the set $U$ with the smallest angle of $x_p$ is found. Finally, we identify the worst one between $x_p$ and $x_q$ (denoted by $x$) in terms of the length of the objective vectors (line 9 in Algorithm 4).

**Algorithm 4** Findout_worst($F_l$, $Ncd$, $\theta$, $Angle$)

**Input:** $S$, $F_l$, $Ncd$, $\theta$, $Angle$
**Output:** $x$ (the worst solution in $F_l$)

$x_p = \arg\max\{Ncd(F_l)\}$ // most crowded individual $x_p$
$U = \emptyset$
**for** all $p \in F_l$ **do**
  **if** $Angle(p,\ x_p) < \theta$ **then**
    $U = U \cup p$
  **end if**
**end for**
$x_q = \arg\min\{Angle(x_p,\ U)\}$
$x = \arg\max\{\|F(x_p)\|, \|F(x_q)\|\}$ // find the worst individual $x$ in terms of the length of the objective vectors
return $x$

**Algorithm 5** Update_Ncd($S$, $Ncd$, $x$, $\theta$, $Angle$)

**Input:** $Ncd$, $x$, $\theta$, $Angle$

**for** all $p \in S$ **do**
  **if** $Angle(p,\ x) < \theta$ **then**
    sh($p$, $x$)=1
    $Ncd(p) = 1 - \prod_{q \in S, q \neq p} sh(p, q)$
  **end if**
**end for**

*3.6. Update_Ncd*

If the solution $x$ is eliminated, then we need to update the $Ncd$ of individuals with $x$ in the same niche. That is, if the angles of the solutions $p$ and $x$ are less than $\theta$, we update the $Ncd$ of $p$.

To illustrate the environmental selection procedure, consider the case in Fig. 2, which illustrates the selection process of our proposed algorithm in a two-dimensional objective problem. Suppose that there are nine individuals in the population, and our task is to select five promising individuals for the next generation. First, it can be observed that solution D is Pareto dominated by E. Thus, D is the first eliminated individual. Next, we can determine the niche radius $\theta$ based on the minimum angle between any two of the individuals. In this case, the niche radius is determined by the angle between individuals G and I. The most crowded individual B is found with

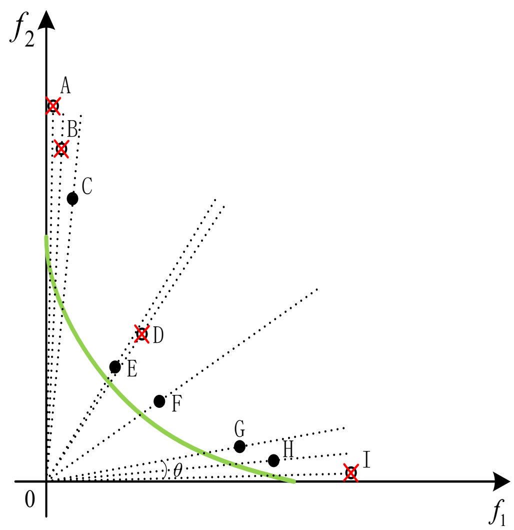



Figure 2: Illustration of the environmental selection of NAEA. There are nine individuals in the population, i.e., A, B, C, D, E, F, G, H and I and the task is to select five promising individuals for the next generation.

the angle-based niche. Individuals A and C are in the same niche territory as B. It is worth noting that A and B have the most similar search directions, so we decided to delete an individual from them. Because individual A performs worse on the length of the objective vector, A is eliminated from the population. In the same way, individuals I and B are eliminated from the population one by one.

From this analysis, we can observe that:

- The proposed algorithm uses the Pareto dominance relationship, which can effectively prevent our final solution as a solution dominated by other individuals.

- NAEA neatly combines two strategies: novel angle-based niche density estimation and angle-based individual selection. The former can guarantee the diversity of the population and eliminate the DRSs to a certain extent. The latter makes efficient use of computing resources and enhances the convergence of the algorithm. It can be seen from the selection results that NAEA is able to strike a good balance between convergence and diversity.

## 4. Experimental design and analysis

### *4.1. Benchmark test problems*

As pointed out in [37] and [38], the PF of the DTLZ and WFG test problems have various characteristics, for example, they are linear, convex, concave, mixed, multimodal, etc. They pose a momentous challenge for MOEAs to find a well-converged and well-distributed solution set.

For the purpose of evaluating the performance of NAEA, four test problems from the DTLZ [37] test suite (DTLZ1 to DTLZ4), and nine from the WFG [38] test suite (WFG1 to WFG9) were chosen for our empirical studies. All the test problems can be scaled to any number of objectives and decision variables. For each problem, the number of objectives was set to 5, 8, 10, and 15, respectively. Following the suggestion in [37], the number of decision variables $n$ was set to $n = m + k - 1$ for the DTLZ test suite, where $m$ denotes the number of objectives; $k = 5$ for DTLZ1, and $k = 10$ for DTLZ2-DTLZ4. According to suggestions in [38], the number of decision variables in the WFG test problems was set to $n = 2 \times (m - 1) + l'$, where $l'$ is the distance-related variable that was set to 20.

### *4.2. Performance metrics*

To evaluate the performance of the proposed algorithm, six widely used metrics were chosen as the performance metrics. They are the IGD [22], HV [21], SPREAD [28], Generational Distance (GD) [39], D_metric [40] and Purity [41].

1) *IGD:* This metric measures the average distance from the points in the true PF to their closest solution in the obtained set. $PF^*$ is a set of nondominated solutions uniformly sampled on the true PF using the two-layer method of Tian *et al.* [42] and $|PF^*|$ is about 10,000. Suppose that $P$ is an approximation set, the IGD metric is defined as follows:

$$IGD(P) = \frac{1}{|PF^*|} \sum_{z^* \in PF^*} d(z^*, P), \tag{9}$$

where $d(z^*, P)$ denotes the minimum Euclidean distance between $z^*$ and all members in $P$, and $|PF^*|$ is the cardinality of $PF^*$. A low IGD value is preferable, which means that the obtained solution set is close to the true PF and has a good distribution.

2) *HV:* HV measures the volume of the hypercube dominated by an approximation set. Similarly, suppose that $P$ is the approximation to PF obtained by a MaOEA. It can be expressed as the following formula:

$$HV(P) = \bigcup_{p_i \in P} S(p_i), \tag{10}$$

where $S(p_i)$ is the hypercube bounded by a solution $p_i$ and a reference point $y^*$. $y^* = (y_1^*, ..., y_M^*)$ is a reference point in the objective space that is dominated by all Pareto-optimal solutions. The HV value of $P$ (with respect to $y^*$ ) is the volume of the region, which dominates $y^*$ and is dominated by $P$. For MaOEAs, the larger the HV value, the better the result.

In this paper, $y^* = (1, 1, ..., 1)$, and the objective values are normalized by the point $1.1 \times z^{nadir}$ before the calculation, where $z^{nadir}$ is the nadir point of the true PF.

3) $SPREAD$: To measure the extent of spread achieved among the final nondominated solutions. SPREAD (denoted as $\Delta$) is defined as follows:

$$\Delta(P) = \frac{\sum_{i=1}^{m} d(e_i, P) + \sum_{z \in P} \left| d(z, P) - \overline{d} \right|}{\sum_{i=1}^{m} d(e_i, P) + |P| \, \overline{d}}, \tag{11}$$

where $P$ is a set of solutions obtained by a MaOEA; $e_1, ..., e_m$ are $m$ extreme solutions in $PF^*$ (roughly 10,000 points are uniformly sampled on the true PF using the method in [42]) and

$$d(z, P) = \min_{y \in P, y \neq z} \|F(y) - F(z)\|^2. \tag{12}$$

$$\overline{d} = \frac{1}{|PF^*|} \sum_{x \in PF^*} d(x, P). \tag{13}$$

If $P$ is well distributed and the extreme solutions in PF are included in $P$, its $\Delta$ value will be zero. A smaller $\Delta$ indicates a better population diversity.

4) *D_metric*: This metric evaluates the distribution uniformity of the solutions in the objective space. Each member of the obtained solutions set $P$ is associated with the nearest reference line, as done in [43] (the number of reference lines is $L$). To this end, a vector called *Spread* is obtained, and each

element of the vector represents the number of solutions associated with each reference line. Thereafter, another vector called $Ideal\_Spread$ is defined. Each element of this vector is $\frac{|P|}{L}$ and the length of the vector is $L$. D_metric calculates the Euclidean distance between $Spread$ and $Ideal\_Spread$ vector. The smaller the value of the distance, the better the diversity.

$$D\_metirc(P) = \frac{L}{|P|}\sqrt{\sum_{i=1}^{L}(Spread_i - Ideal\_Spread_i)^2}. \tag{14}$$

5) $GD$: Suppose that $P$ is an obtained solutions set, the GD metric is defined as:

$$GD(P) = \frac{\sqrt{\sum_{i=1}^{|P|} d_i^2}}{|P|}, \tag{15}$$

where $|P|$ is the number of obtained solutions and $d_i$ is the Euclidean distance between each solution in $P$ and the nearest member in the true PF. A value nearer to 0 indicates a better convergence performance.

6) $Purity$: Suppose there are $C$ multiobjective optimization strategies. $r_i = |R_1^i|; i = 1, 2, \ldots; C$ is the number of rank one solutions for each strategy. The union of rank one solutions of all strategies is referred to as $R^*$. The ranking procedure is also performed in $R^*$, and the solutions to obtain a new rank one are called $R_1^*$. Let $r_i^*$ be the number of solutions that belong to both $R_1^i$ and $R_1^*$. The Purity for the $i$-th strategy is defined as:

$$Purity(i) = \frac{r_i^*}{r_i}, i = 1, 2, \ldots, C. \tag{16}$$

It should be clear that the Purity value may in the range [0,1], where a higher value implies better performance.

To have statistically comprehensive conclusions, the Wilcoxon rank-sum test [44] at a 0.05 significance level is adopted to test the significant difference between the result obtained by peer algorithms, where $+$, $-$, and $\approx$ indicate that the result is significantly better, significantly worse, and statistically similar to that obtained by NAEA, respectively.

### *4.3. Comparison with state-of-the-art algorithms*

In order to verify the performance of the proposed method, the state-of-the-art algorithms VaEA [34], MOEA/DD [17], NSGA-III [16], MOEA/D [15], MOEA/D-M2M [45] and MaOEA/IGD [46] were selected to compare with NAEA.

1) VaEA [34]: VaEA is a vector angle-based evolutionary algorithm which presents a maximum-vector-angle-first and worse-elimination principle as the selection criterion.

2) MOEA/DD [17]: MOEA/DD adopts a unified paradigm of convergence and diversity, which combines dominance and decomposition-based approaches, for many-objective optimization.

3) NSGA-III [16]: Based on the dominance relationship, NSGA-III uses a set of reference points to generate reference vectors. Those candidate solutions close to the reference vectors are selected for the next generation.

4) MOEA/D [15]: MOEA/D decomposes $m$ objectives into a set of single-objective optimization problems. Here, the PBI function is selected since MOEA/D with PBI has been found to be more competitive when solving problems with a high-dimensional objective space.

5) MOEA/D-M2M [45]: MOEA/D-M2M decomposes MOPs into a set of simple multi-objective optimization subproblems and solves them in a collaborative way.

6) MaOEA/IGD [46]: the IGD indicator is employed in each generation to select the solutions with favorable convergence and diversity.

### *4.4. General parameters*

- **Population size:** Table 1 presents the population size of VaEA, NAEA, MOEA/D, and MOEA/DD. Regarding other algorithms, the population size was the same as MOEA/D.

- **Reproduction operators:** In all algorithms, the $SBX$ and polynomial mutation were used to generate offspring solutions. For the $SBX$, we set the crossover probability $p_c$ to 1.0 and the distribution index $\eta_c$

to 20. As for the polynomial mutation, the probability $p_m$ and distribution index $\eta_m$ were set to be $\frac{1}{n}$ ($n$ represents decision number) and 20, respectively [16], [17].

- **Number of independent runs and termination condition:** All algorithms were independently run 20 times on each test instance. The termination condition can be determined by predefined maximum function evaluations. In accordance with [47], the maximum number of evaluations in this paper was defined as 90,000.

- **Specific parameter settings in algorithms:** Following the practice in [17], three parameters $T$, $\theta$ and the probability $\delta$ in MOEA/DD were set as 20, 5 and 0.9, respectively. For MOEA/D [15], the neighborhood size was set to 20; the maximum replacement number was set to 2, and the penalty parameter was set to 5. For MOEA/D-M2M [45], the number of subproblems was set as $K$=10.

In this paper, all algorithms were implemented under the PlatEMO [2], proposed by Tian *et al.* [48].

Table 1: Population size ($N$) for four algorithms

| Objectives | No. of Vectors | VaEA<br>NAEA | MOEA/D<br>MOEA/DD |
|---|---|---|---|
| 5 | 210 | 212 | 210 |
| 8 | 156 | 156 | 156 |
| 10 | 275 | 276 | 275 |
| 15 | 135 | 136 | 135 |

### *4.5. Results on the DTLZ test suite*

The results on the four DTLZ test problems are given in Table 2, with both the mean and standard deviation of the IGD values averaged over 20 independent runs for the seven compared MOEAs, where the best mean among the seven algorithms is highlighted.

As can be seen from Table 2, both NAEA and MOEA/DD are effective algorithms in terms of the number of the best results. Our proposed NAEA

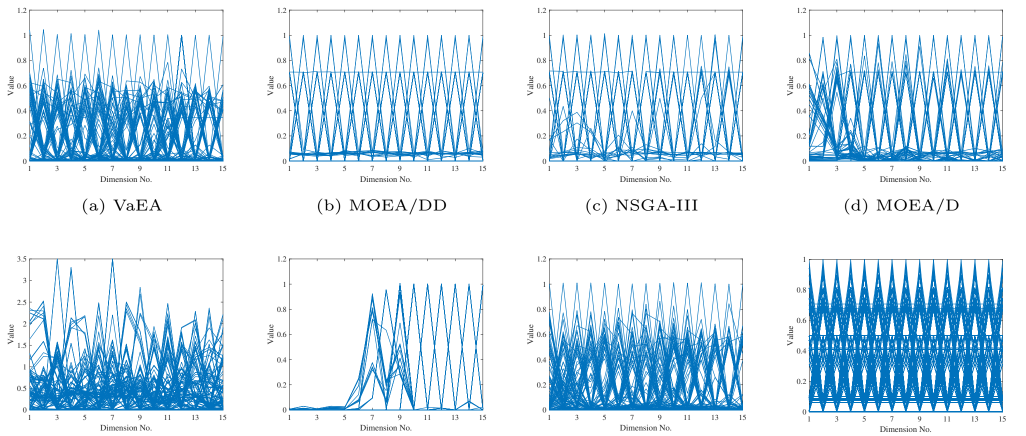
[2] The PlatEMO can be downloaded from: https://github.com/BIMK/PlatEMO/releases

Table 2: IGD results (mean and standard deviation) of the state-of-the-art algorithms on the DTLZ test suite.

| Problem | $m$ | $D$ | VaEA | MOEA/DD | NSGA-III | MOEA/D | MOEA/D-M2M | MaOEA/IGD | NAEA |
|---|---|---|---|---|---|---|---|---|---|
| DTLZ1 | 5 | 9 | 1.1517e-1 (2.55e-2) − | 5.2708e-2 (4.44e-5) + | 6.8093e-2 (4.60e-5) − | 5.2829e-2 (1.91e-4) + | 1.9415e+0 (1.47e+0) − | 2.0851e-1 (2.50e-1) ≈ | 5.3981e-2 (5.60e-4) |
| | 8 | 12 | 2.4784e-1 (1.81e-1) − | 9.6295e-2 (3.72e-4) + | 1.1479e-1 (4.10e-2) ≈ | 9.5151e-2 (5.50e-4) + | 2.5812e+0 (1.81e+0) − | 4.8501e-1 (3.62e-1) − | 1.0561e-1 (2.26e-3) |
| | 10 | 14 | 3.1285e-1 (1.52e-1) ≈ | 1.0617e-1 (4.20e-4) + | 1.5854e-1 (5.00e-2) + | 1.0590e-1 (4.88e-4) + | 2.6696e+0 (1.57e+0) − | 1.3773e-1 (9.75e-2) + | 2.1684e-1 (5.71e-2) |
| | 15 | 19 | 3.3725e-1 (2.25e-1) ≈ | 1.5354e-1 (9.94e-3) + | 2.1932e-1 (5.06e-2) + | 1.3844e-1 (2.84e-3) + | 1.5967e+1 (5.11e+0) − | 3.1592e-1 (1.79e-1) ≈ | 2.5582e-1 (3.38e-2) |
| DTLZ2 | 5 | 14 | 1.6803e-1 (9.70e-4) − | 1.6514e-1 (5.90e-6) − | 2.1222e-1 (2.06e-5) − | 1.6514e-1 (3.89e-6) − | 4.4212e-1 (8.72e-3) − | 1.6871e-1 (5.80e-4) − | 1.6287e-1 (9.45e-4) |
| | 8 | 17 | 3.6310e-1 (2.46e-3) − | 3.1501e-1 (3.66e-5) + | 3.5442e-1 (6.66e-2) ≈ | 3.1499e-1 (2.12e-5) + | 8.1472e-1 (9.07e-3) − | 3.4373e-1 (8.61e-3) + | 3.5015e-1 (2.37e-3) |
| | 10 | 19 | 4.2830e-1 (4.95e-3) − | 4.2147e-1 (3.04e-4) − | 4.6157e-1 (6.23e-2) − | 4.2128e-1 (3.61e-4) − | 8.1686e-1 (8.49e-3) − | 4.3574e-1 (5.35e-3) − | 3.9592e-1 (1.68e-3) |
| | 15 | 24 | 5.9565e-1 (7.05e-3) − | 6.2268e-1 (8.43e-4) − | 6.6845e-1 (4.84e-2) − | 6.2095e-1 (3.85e-3) − | 1.2731e+0 (1.85e-1) − | 8.4776e-1 (1.07e-1) − | 5.6474e-1 (1.85e-3) |
| DTLZ3 | 5 | 14 | 3.9018e-1 (1.47e-1) − | 1.6780e-1 (1.55e-3) + | 2.1364e-1 (1.92e-3) ≈ | 2.8970e-1 (5.42e-1) − | 5.0875e+1 (1.62e+1) − | 9.1581e+0 (5.15e+0) − | 2.2162e-1 (6.26e-2) |
| | 8 | 17 | 6.9269e+0 (4.61e+0) − | 3.4224e-1 (1.00e-1) + | 1.4379e+0 (1.57e+0) ≈ | 4.3768e-1 (2.54e-1) + | 4.6913e+1 (1.75e+1) − | 1.0664e+1 (1.09e+1) − | 7.5675e-1 (3.71e-1) |
| | 10 | 19 | 2.8254e+1 (8.77e+0) − | 5.2169e-1 (2.60e-1) + | 4.4066e+0 (4.53e+0) − | 6.7520e-1 (3.45e-1) + | 5.7708e+1 (1.31e+1) − | 5.4770e+0 (2.84e+0) − | 8.9835e-1 (4.26e-1) |
| | 15 | 24 | 1.5110e+1 (6.47e+0) − | 6.2927e-1 (2.20e-2) + | 3.7643e+0 (2.90e+0) ≈ | 9.5716e-1 (3.09e-1) + | 2.0342e+2 (2.45e+1) − | 7.5200e+0 (5.59e+0) − | 2.1051e+0 (7.93e-1) |
| DTLZ4 | 5 | 14 | 1.7021e-1 (1.13e-3) − | 1.6513e-1 (1.26e-5) ≈ | 2.9209e-1 (1.12e-1) − | 3.6938e-1 (1.94e-1) − | 4.4601e-1 (1.36e-2) − | 2.0253e-1 (8.28e-2) − | 1.6484e-1 (9.33e-4) |
| | 8 | 17 | 3.6438e-1 (2.00e-3) − | 3.2696e-1 (3.66e-2) + | 4.1177e-1 (1.11e-1) ≈ | 5.6701e-1 (1.39e-1) − | 7.7881e-1 (2.43e-2) − | 3.6862e-1 (5.15e-2) ≈ | 3.5182e-1 (1.45e-3) |
| | 10 | 19 | 4.3784e-1 (9.03e-3) − | 4.2066e-1 (5.39e-4) − | 4.5964e-1 (6.01e-2) − | 6.5887e-1 (7.27e-2) − | 8.5886e-1 (2.54e-2) − | 4.4638e-1 (1.74e-2) − | 4.0075e-1 (1.74e-3) |
| | 15 | 24 | 5.9422e-1 (2.97e-3) − | 6.2922e-1 (4.05e-3) − | 6.6649e-1 (4.93e-2) − | 7.5982e-1 (5.39e-2) − | 1.7630e+0 (3.08e-1) − | 6.4491e-1 (1.90e-2) − | 5.8290e-1 (3.65e-3) |
| +/-/≈ | | | 0/14/2 | 10/5/1 | 2/8/6 | 8/8/0 | 0/16/0 | 2/11/3 | |

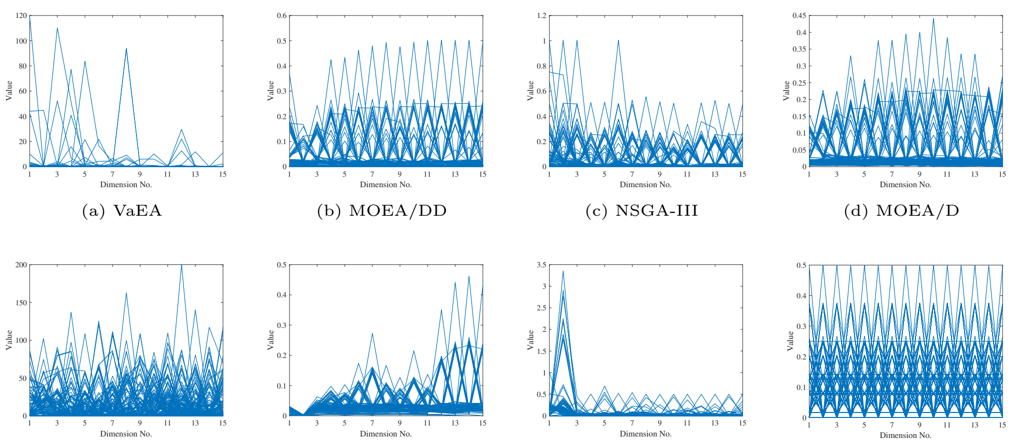
Table 3: Δ results (mean and standard deviation) of the state-of-the-art algorithms on the DTLZ test suite.

| Problem | $m$ | $D$ | VaEA | MOEA/DD | NSGA-III | MOEA/D | MOEA/D-M2M | MaOEA/IGD | NAEA |
|---|---|---|---|---|---|---|---|---|---|
| DTLZ1 | 5 | 9 | 1.8151e+0 (2.45e-1) − | 5.3716e-3 (2.52e-3) + | 3.4074e-2 (4.51e-4) + | 1.2392e-2 (7.54e-3) + | 9.5756e-1 (4.79e-2) − | 5.2238e-1 (6.35e-1) ≈ | 2.0248e-1 (1.53e-2) |
| | 8 | 12 | 1.8416e+0 (1.38e-1) − | 7.3317e-2 (2.76e-3) + | 3.7009e-1 (5.06e-1) − | 8.5950e-2 (6.89e-3) + | 9.3523e-1 (7.06e-2) − | 4.1243e-2 (3.32e+0) + | 3.2201e-1 (3.49e-1) |
| | 10 | 14 | 1.6050e+0 (1.40e-1) ≈ | 9.5599e-2 (8.18e-3) + | 7.3761e-1 (7.68e-1) + | 1.2542e-1 (1.54e-2) + | 8.0576e-1 (6.18e-2) + | 4.6346e-1 (3.19e-1) + | 1.4870e+0 (6.03e-1) |
| | 15 | 19 | 1.5937e+0 (1.23e-1) + | 2.0271e-1 (7.04e-2) + | 1.4101e+0 (1.21e+0) ≈ | 3.5080e-1 (4.25e-2) + | 1.1329e+0 (6.34e-2) + | 8.6084e-1 (5.95e+0) + | 1.8483e+0 (2.53e-1) |
| DTLZ2 | 5 | 14 | 1.0012e-1 (8.39e-3) − | 1.5098e-1 (1.06e-4) − | 1.7512e-1 (9.50e-4) − | 1.5100e-1 (1.27e-4) − | 1.0529e+0 (4.83e-2) − | 1.5061e-1 (7.61e-3) − | 8.6808e-2 (5.22e-3) |
| | 8 | 17 | 1.2610e-1 (9.81e-3) − | 1.3904e-1 (3.57e-4) − | 1.8143e-1 (2.17e-1) − | 1.3892e-1 (3.25e-4) − | 1.2833e+0 (3.97e-2) − | 3.2498e-1 (5.70e-2) − | 8.4608e-2 (7.59e-3) |
| | 10 | 19 | 1.0553e-1 (7.37e-3) − | 1.5102e-1 (1.27e-3) − | 3.5830e-1 (3.21e-1) − | 1.5869e-1 (2.64e-3) − | 1.1955e+0 (2.72e-2) − | 2.6717e-1 (4.16e-2) − | 7.5032e-2 (4.44e-3) |
| | 15 | 24 | 2.5463e-1 (1.13e-2) − | 3.2107e-1 (3.75e-3) − | 8.8780e-1 (3.29e-1) − | 3.4393e-1 (3.44e-2) − | 1.1230e+0 (3.96e-2) − | 1.2821e+0 (1.94e-1) − | 8.8917e-2 (8.36e-3) |
| DTLZ3 | 5 | 14 | 1.6631e+0 (3.77e-1) − | 1.5759e-1 (6.10e-3) + | 1.7267e-1 (5.12e-3) + | 1.6782e-1 (2.91e-2) + | 9.5194e-1 (5.71e-2) ≈ | 1.0855e+0 (1.14e-1) ≈ | 8.1934e-1 (7.09e-1) |
| | 8 | 17 | 1.3065e+0 (2.09e-1) + | 1.4866e-1 (2.51e-2) + | 6.9384e-1 (6.07e-1) + | 3.2126e-1 (3.50e-1) + | 1.0000e+0 (8.45e-2) + | 1.0696e+0 (5.16e-2) + | 1.7281e+0 (2.53e-1) |
| | 10 | 19 | 4.7025e-1 (7.94e-2) + | 2.8861e-1 (2.18e-1) + | 1.2048e+0 (4.26e-1) + | 5.2282e-1 (4.66e-1) + | 9.3382e-1 (7.31e-2) + | 1.0385e+0 (2.89e-2) + | 1.6499e+0 (1.00e-1) |
| | 15 | 24 | 5.1117e-1 (4.98e-2) + | 4.4328e-1 (3.74e-1) + | 1.7051e+0 (6.78e-1) − | 8.0271e-1 (3.06e-1) + | 1.2303e+0 (6.28e-2) + | 1.0605e+0 (3.58e-2) + | 1.3690e+0 (1.14e-1) |
| DTLZ4 | 5 | 14 | 1.0147e-1 (7.24e-3) − | 1.5082e-1 (1.45e-4) − | 3.9650e-1 (3.16e-1) − | 6.8223e-1 (4.87e-1) − | 7.9856e-1 (4.74e-2) − | 2.4330e-1 (2.47e-1) − | 8.6570e-2 (7.62e-3) |
| | 8 | 17 | 1.2125e-1 (9.36e-3) − | 2.3874e-1 (3.14e-1) − | 3.6519e-1 (3.29e-1) − | 9.9388e-1 (4.00e-1) − | 9.6639e-1 (6.61e-2) − | 2.7432e-1 (1.44e-1) − | 7.9396e-2 (7.91e-3) |
| | 10 | 19 | 1.1118e-1 (9.07e-3) − | 1.5149e-1 (2.02e-3) − | 2.8754e-1 (2.23e-1) − | 1.2789e+0 (2.57e-1) − | 8.8911e-1 (4.72e-2) − | 2.3575e-1 (7.75e-2) − | 7.8028e-2 (7.28e-3) |
| | 15 | 24 | 2.3927e-1 (1.46e-2) − | 3.6103e-1 (8.32e-2) − | 6.8235e-1 (3.27e-1) − | 1.2985e+0 (1.53e-1) − | 7.1640e-1 (5.93e-2) − | 6.1834e-1 (2.34e-1) − | 1.2871e-1 (1.67e-2) |
| +/-/≈ | | | 4/11/1 | 8/8/0 | 5/10/1 | 8/8/0 | 5/10/1 | 6/8/2 | |



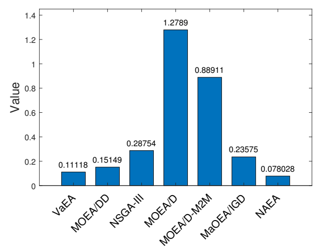
(a) VaEA (b) MOEA/DD (c) NSGA-III (d) MOEA/D (e) MOEA/D-M2M (f) MaOEA/IGD (g) NAEA (h) True PF

Figure 3: Final solution sets of seven compared algorithms and true PF on the 15-objective DTLZ2, shown by parallel coordinates.

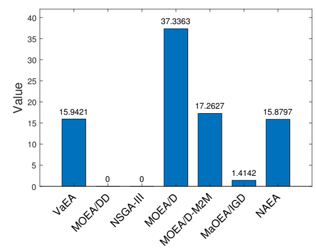


(a) VaEA (b) MOEA/DD (c) NSGA-III (d) MOEA/D (e) MOEA/D-M2M (f) MaOEA/IGD (g) NAEA (h) True PF

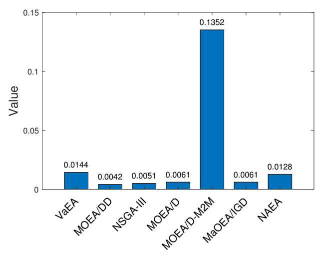
Figure 4: Final solution sets of seven compared algorithms and true PF on the 15-objective DTLZ1, shown by parallel coordinates.



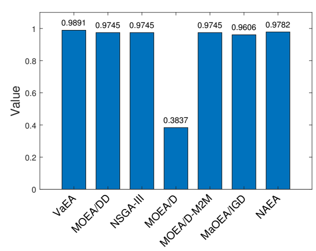

(a) Δ




(b) D_metric




(c) GD




(d) Purity

Figure 5: Plot of the bar of four metrics for seven algorithms on the 10-objective DTLZ4.

had significantly better performance than the other algorithms on 5- 10- and 15-objective problems of DTLZ2 and DTLZ4. DTLZ2 is mainly designed to test algorithms' diversity when the number of objectives increases [37]. In NAEA, the diversity of the population is guaranteed by the niche-based density estimation strategy. Therefore, NAEA achieves promising results on DTLZ2 as is shown in Fig. 3. Regarding DTLZ4, the density of points on its PF is non-uniform, which poses a challenge to the maintenance of population diversity. NAEA does not rely on weight vectors and has a novel diversity maintenance mechanism, so it can also work very well on DTLZ4.

At the same time, we can find that MOEA/D and MOEA/DD performed well on DTLZ1 and DTLZ3, respectively. DTLZ1 is a linear and multimodal problem. For solutions of the 15-objective DTLZ1 (see Fig. 4), NAEA has individual extreme values on a few objectives, which means that our proposed algorithm has relatively poor convergence performance on DTLZ1. The local optimal PFs of DTLZ3 increase exponentially with the number of objectives, which may weaken the convergence ability of the algorithm. NAEA obtained medium performance among all of the algorithms on DTLZ3. NAEA's performance on DTLZ1 and DTLZ3 was not particularly good, which can be attributed to the fact that DTLZ1 and DTLZ3 are multimodal test problems containing a large number of local Pareto optimal solutions and mainly selecting well-distributed individuals based on niche will lead to a local optimal solution.

From the 16 test instances of the DTLZ test suite presented in Table 2, we see that both NAEA and MOEA/DD achieved the best results of six instances in terms of IGD, while MOEA/D reached four. From the results, we can conclude that NAEA is a very competitive algorithm on the DTLZ suite test instances.

The main observation from Table 3 is that in terms of $\Delta$, NAEA and MOEA/DD are very competitive algorithms. NAEA obtained the best results for all test problems on DTLZ2 and DTLZ4. Additionally, MOEA/DD had the best results for almost all test problems on DTLZ1 and DTLZ3 except for 8-objective DTLZ1. This means that NAEA and MOEA/DD maintained good distribution in the DTLZ test suite.

Additionally, Fig. 5 plots the bar of four metrics for seven algorithms on the 10-objective DTLZ4. Fig. 5(a) and Fig. 5(b) show the diversity metric $\Delta$ and D_metric, respectively. $\Delta$ emphasizes the extent of spread among the obtained solution, and NAEA achieved the best performance among all peer algorithms. D_metric considers the diversity of solution distribution in the objective space. NAEA obtains medium performance in terms of D_metric. From Fig. 5(c)-(d), it can be seen that NAEA has obtained a smaller GD value and a larger Purity value, which shows that it is very competitive in terms of convergence.

### *4.6. Results on the WFG test suite*

IGD results on all WFG instances are shown in Table 4. The proposed NAEA demonstrated a clear advantage over its competitors on this test suite. More specifically, NAEA obtained the best results on 24 out of 36 test instances. The VaEA achieved 11 best results. NAEA performed best on WFG1, WFG4, WFG5, WFG6, WFG7, WFG8, and WFG9, while VaEA outperformed the other algorithms on WFG2 and WFG3. Additionally, MOEA/DD generated one best result on WFG1 with 10 objectives.

Table 5 gives the results of HV on all the WFG test instances. It can be seen from the table that NAEA and NSGA-III performed best, presenting a clear advantage over other algorithms on the majority of the test instances. As shown, NAEA performed best on WFG4, WFG6, WFG7, WFG8, and

Table 4: IGD results (mean and standard deviation) of the state-of-the-art algorithms on the WFG test suite.

| Problem | $m$ | $D$ | VaEA | MOEA/DD | NSGA-III | MOEA/D | MOEA/D-M2M | MaOEA/IGD | NAEA |
|---|---|---|---|---|---|---|---|---|---|
| WFG1 | 5 | 14 | 8.1009e-1 (9.32e-2) − | 7.0008e-1 (8.49e-2) − | 4.7338e-1 (5.35e-3) − | 8.2068e-1 (4.55e-2) − | 1.9786e+0 (3.56e-2) − | 4.4691e+0 (9.43e-1) − | 4.4603e-1 (3.53e-2) |
| | 8 | 17 | 1.3724e+0 (1.14e-1) − | 1.2382e+0 (1.27e-1) − | 1.0280e+0 (5.55e-2) − | 1.5497e+0 (9.81e-2) − | 2.5975e+0 (2.48e-2) − | 7.3383e+0 (1.82e+0) − | 9.2645e-1 (3.20e-2) |
| | 10 | 19 | 2.3822e+0 (1.16e-1) − | 1.3892e+0 (8.61e-2) + | 1.5209e+0 (1.06e-1) ≈ | 1.8765e+0 (8.61e-2) − | 2.9525e+0 (2.14e-2) − | 9.1861e+0 (3.07e+0) − | 1.5223e+0 (9.24e-2) |
| | 15 | 24 | 2.2919e+0 (1.82e-1) − | 1.9306e+0 (8.30e-2) − | 1.9030e+0 (1.87e-1) − | 2.9761e+0 (2.43e-1) − | 4.0277e+0 (3.36e-2) − | 1.0133e+1 (5.47e+0) − | 1.7034e+0 (5.02e-2) |
| WFG2 | 5 | 14 | 3.9164e-1 (4.63e-3) + | 4.9811e-1 (9.57e-3) − | 5.0410e-1 (1.88e-3) − | 7.6596e-1 (3.25e-2) − | 7.0743e-1 (3.22e-2) − | 1.5865e+0 (2.02e-1) − | 4.0461e-1 (7.63e-3) |
| | 8 | 17 | 9.3685e-1 (1.30e-2) + | 1.3379e+0 (2.28e-2) − | 1.1092e+0 (1.93e-1) ≈ | 1.7695e+0 (3.03e-2) − | 1.3445e+0 (5.28e-2) − | 2.2389e+0 (4.01e-1) − | 1.0469e+0 (3.11e-2) |
| | 10 | 19 | 1.0165e+0 (1.39e-2) + | 1.3699e+0 (3.47e-2) − | 1.2579e+0 (1.82e-1) − | 1.9359e+0 (2.76e-2) − | 1.3204e+0 (4.55e-2) − | 2.3698e+0 (5.84e-1) − | 1.1368e+0 (3.16e-2) |
| | 15 | 24 | 1.7537e+0 (2.70e-2) + | 2.1821e+0 (1.60e-2) − | 1.9540e+0 (5.29e-1) − | 2.5619e+0 (3.32e-2) − | 2.1175e+0 (5.71e-2) − | 4.0522e+0 (1.72e+0) − | 1.8887e+0 (6.88e-2) |
| WFG3 | 5 | 14 | 5.5104e-1 (3.78e-2) ≈ | 5.5903e-1 (4.11e-2) ≈ | 5.8995e-1 (5.19e-2) − | 7.5849e-1 (1.45e-1) − | 6.4090e-1 (5.13e-2) − | 5.1755e+0 (1.16e+0) − | 5.5296e-1 (9.12e-2) |
| | 8 | 17 | 1.5027e+0 (1.81e-1) ≈ | 1.9588e+0 (5.29e-2) − | 1.8725e+0 (3.86e-1) − | 3.6979e+0 (1.37e-1) − | 1.9890e+0 (2.77e-1) − | 8.4910e+0 (1.71e+0) − | 1.5426e+0 (3.05e-1) |
| | 10 | 19 | 1.8841e+0 (2.49e-1) + | 2.7149e+0 (6.47e-2) − | 2.0254e+0 (5.38e-1) ≈ | 5.4924e+0 (3.64e-1) − | 2.1274e+0 (4.21e-1) ≈ | 5.0554e+0 (4.60e+0) ≈ | 2.0857e+0 (2.89e-1) |
| | 15 | 24 | 3.7484e+0 (4.34e-1) ≈ | 7.0393e+0 (7.72e-2) − | 3.9051e+0 (1.75e+0) ≈ | 9.3099e+0 (2.44e-1) − | 8.1021e+0 (2.83e+0) − | 5.4534e+0 (2.04e-1) − | 3.7720e+0 (6.48e-1) |
| WFG4 | 5 | 14 | 9.4992e-1 (6.56e-3) − | 1.0603e+0 (4.18e-3) − | 1.2258e+0 (6.82e-4) − | 1.5255e+0 (9.71e-2) − | 1.3869e+0 (2.88e-2) − | 6.1362e+0 (7.26e-1) − | 9.3892e-1 (5.89e-3) |
| | 8 | 17 | 3.0058e+0 (2.98e-2) − | 4.1254e+0 (1.27e-1) − | 3.1701e+0 (3.38e-1) − | 6.8102e+0 (2.77e-1) − | 3.7580e+0 (6.91e-2) − | 9.9095e+0 (9.81e-1) − | 2.9319e+0 (1.28e-2) |
| | 10 | 19 | 4.0222e+0 (1.83e-2) − | 6.1263e+0 (1.57e-1) − | 4.7946e+0 (5.56e-1) − | 9.3054e+0 (2.17e-1) − | 5.0263e+0 (4.45e-2) − | 1.2013e+1 (2.45e-1) − | 3.9732e+0 (2.56e-2) |
| | 15 | 24 | 8.2499e+0 (8.90e-2) ≈ | 1.3310e+1 (1.06e+0) − | 9.7826e+0 (1.07e+0) − | 1.5860e+1 (3.07e-1) − | 1.0998e+1 (1.89e-1) − | 2.2469e+1 (3.50e+0) − | 8.2459e+0 (6.86e-2) |
| WFG5 | 5 | 14 | 9.4489e-1 (6.88e-3) − | 1.0391e+0 (6.23e-3) − | 1.2153e+0 (1.03e-4) − | 1.4413e+0 (7.14e-2) − | 1.3443e+0 (3.32e-2) − | 1.8950e+0 (1.49e+0) − | 9.3611e-1 (6.68e-3) |
| | 8 | 17 | 3.0291e+0 (3.61e-2) − | 4.0063e+0 (1.44e-1) − | 3.1184e+0 (2.75e-1) ≈ | 6.3237e+0 (2.33e-1) − | 3.6977e+0 (4.38e-2) − | 1.2520e+1 (4.05e+0) − | 2.9514e+0 (1.91e-2) |
| | 10 | 19 | 4.0056e+0 (2.12e-2) − | 6.4174e+0 (1.67e-1) − | 4.7816e+0 (5.93e-1) − | 8.8876e+0 (2.09e-1) − | 4.9581e+0 (5.49e-2) − | 8.9902e+0 (6.60e+0) − | 3.9304e+0 (2.12e-2) |
| | 15 | 24 | 7.9958e+0 (6.71e-2) + | 1.2709e+1 (1.80e-1) − | 9.5266e+0 (8.04e-1) − | 1.5238e+1 (1.32e-1) − | 9.8055e+0 (2.37e-1) − | 2.8189e+1 (3.01e+0) − | 8.0887e+0 (6.16e-2) |
| WFG6 | 5 | 14 | 9.7559e-1 (9.74e-3) − | 1.0494e+0 (6.07e-3) − | 1.2162e+0 (2.45e-3) − | 1.6394e+0 (3.18e-2) − | 1.5928e+0 (2.97e-2) − | 5.2928e+0 (1.26e+0) − | 9.3764e-1 (7.34e-3) |
| | 8 | 17 | 3.1333e+0 (5.49e-2) − | 4.0837e+0 (1.76e-1) − | 3.1346e+0 (2.75e-1) − | 6.9574e+0 (6.28e-2) − | 3.9623e+0 (4.13e-2) − | 1.0421e+1 (3.87e+0) − | 3.0298e+0 (2.15e-2) |
| | 10 | 19 | 4.0833e+0 (4.13e-2) − | 6.2112e+0 (1.66e-1) − | 4.7736e+0 (4.89e-1) − | 9.4365e+0 (1.66e-1) − | 5.2594e+0 (4.81e-2) − | 8.8833e+0 (4.63e+0) − | 3.9823e+0 (3.47e-2) |
| | 15 | 24 | 7.9255e+0 (6.68e-2) + | 1.2516e+1 (4.86e-1) − | 9.7079e+0 (9.41e-1) − | 1.6143e+1 (1.96e-1) − | 1.1660e+1 (1.97e-1) − | 1.7389e+1 (9.04e+0) ≈ | 8.1202e+0 (1.03e-1) |
| WFG7 | 5 | 14 | 9.5267e-1 (4.96e-3) − | 1.0628e+0 (5.41e-3) − | 1.2279e+0 (1.45e-3) − | 1.7208e+0 (2.45e-2) − | 1.5065e+0 (3.55e-2) − | 5.1582e+0 (1.09e+0) − | 9.3110e-1 (6.83e-3) |
| | 8 | 17 | 3.0573e+0 (6.06e-2) − | 3.6996e+0 (1.40e-1) − | 3.1504e+0 (2.81e-1) − | 7.0673e+0 (4.27e-2) − | 3.9197e+0 (5.92e-2) − | 1.0741e+1 (2.14e+0) − | 2.9489e+0 (1.95e-2) |
| | 10 | 19 | 4.0221e+0 (2.34e-2) − | 5.6024e+0 (2.66e-1) − | 4.7775e+0 (5.31e-1) − | 9.3620e+0 (1.59e-1) − | 5.1417e+0 (6.66e-2) − | 1.0790e+1 (5.06e+0) − | 3.9370e+0 (2.03e-2) |
| | 15 | 24 | 8.0856e+0 (7.53e-2) ≈ | 1.1698e+1 (1.55e+0) − | 9.5988e+0 (8.46e-1) − | 1.6392e+1 (1.71e-1) − | 1.0322e+1 (3.02e-1) − | 2.3693e+1 (2.45e+0) − | 8.0740e+0 (9.44e-2) |
| WFG8 | 5 | 14 | 1.0752e+0 (1.23e-2) − | 1.0626e+0 (6.80e-3) − | 1.2531e+0 (1.19e-2) − | 1.4035e+0 (1.91e-1) − | 1.6471e+0 (5.37e-2) − | 4.8295e+0 (1.04e+0) − | 9.7970e-1 (9.78e-3) |
| | 8 | 17 | 3.2603e+0 (2.52e-2) − | 3.6396e+0 (2.84e-1) − | 3.3626e+0 (2.37e-1) − | 6.3807e+0 (3.24e-1) − | 3.9563e+0 (6.65e-2) − | 1.1087e+1 (1.76e+0) − | 3.0689e+0 (4.23e-2) |
| | 10 | 19 | 4.3354e+0 (2.91e-2) − | 5.7541e+0 (4.76e-1) − | 4.8001e+0 (5.69e-1) − | 8.5831e+0 (2.75e-1) − | 5.3137e+0 (4.41e-2) − | 1.2362e+1 (1.82e+0) − | 4.0105e+0 (5.63e-2) |
| | 15 | 24 | 8.4942e+0 (1.81e-1) − | 9.6788e+0 (1.18e+0) − | 9.5386e+0 (1.25e+0) − | 1.4186e+1 (1.89e+0) − | 1.1646e+1 (2.24e-1) − | 2.3824e+1 (1.98e+0) − | 8.3268e+0 (1.22e-1) |
| WFG9 | 5 | 14 | 9.3294e-1 (6.83e-3) − | 1.0359e+0 (4.86e-3) − | 1.2049e+0 (6.83e-3) − | 1.5016e+0 (4.77e-2) − | 1.3082e+0 (4.11e-2) − | 4.4457e+0 (1.33e+0) − | 9.2373e-1 (6.96e-3) |
| | 8 | 17 | 3.0190e+0 (3.27e-2) − | 4.0566e+0 (1.19e-1) − | 3.1184e+0 (2.74e-1) ≈ | 6.4952e+0 (1.96e-1) − | 3.7200e+0 (6.97e-2) − | 1.0691e+1 (3.61e+0) − | 2.9584e+0 (2.64e-2) |
| | 10 | 19 | 3.9960e+0 (2.60e-2) − | 5.8627e+0 (2.36e-1) − | 4.6059e+0 (5.54e-1) − | 8.8681e+0 (6.13e-1) − | 5.0099e+0 (6.20e-2) − | 5.0853e+0 (1.93e+0) − | 3.9521e+0 (3.65e-2) |
| | 15 | 24 | 7.7587e+0 (6.70e-2) + | 1.0550e+1 (1.19e+0) − | 9.0475e+0 (9.06e-1) − | 1.5241e+1 (1.75e+0) − | 9.6672e+0 (2.05e-1) − | 1.8885e+1 (7.95e+0) − | 7.8518e+0 (1.17e-1) |
| +/-/≈ | | | 8/23/5 | 1/34/1 | 0/30/6 | 0/36/0 | 0/35/1 | 0/34/2 | |

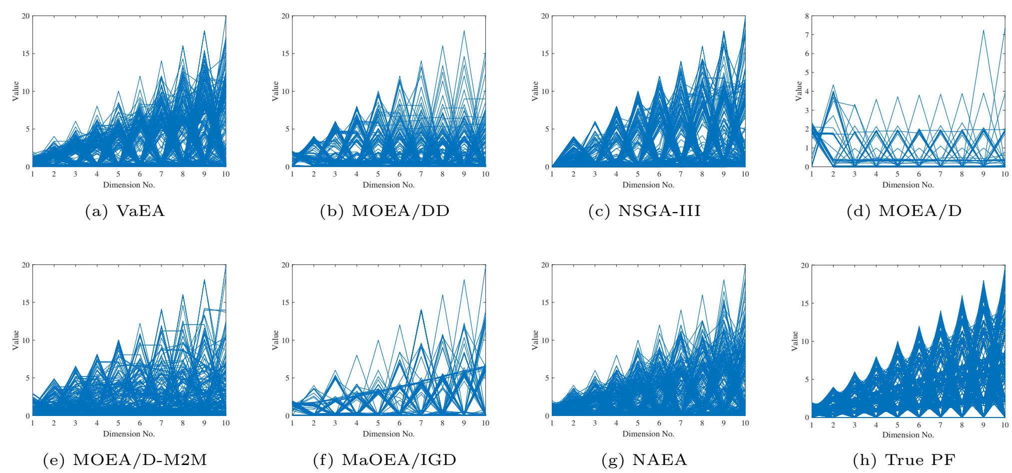
WFG9, while NSGA-III performed best on WFG1 and WFG5. For the WFG2, NSGA-III is a competitive algorithm to NAEA. MOEA/DD and VaEA achieved a good performance on WFG1 and WFG3 with 10 objectives and 8 objectives, respectively.

Table 5: HV results (mean and standard deviation) of the state-of-the-art algorithms on the WFG test suite.

| Problem | $m$ | $D$ | VaEA | MOEA/DD | NSGA-III | MOEA/D | MOEA/D-M2M | MaOEA/IGD | NAEA |
|---|---|---|---|---|---|---|---|---|---|
| WFG1 | 5 | 14 | 6.8420e-1 (4.52e-2) − | 7.7403e-1 (4.41e-2) − | 9.9738e-1 (2.39e-4) + | 9.2375e-1 (2.75e-2) + | 3.0867e-1 (6.08e-3) − | 2.2160e-1 (3.68e-2) − | 8.9446e-1 (3.27e-2) |
| | 8 | 17 | 5.9677e-1 (5.12e-2) − | 7.6985e-1 (1.02e-1) − | 9.2541e-1 (6.72e-2) + | 7.5487e-1 (1.20e-1) − | 2.6227e-1 (8.43e-3) − | 2.5870e-1 (8.19e-2) − | 8.8333e-1 (4.92e-2) |
| | 10 | 19 | 3.4132e-1 (2.66e-2) − | 8.3793e-1 (6.93e-2) + | 7.1112e-1 (1.51e-1) + | 5.7781e-1 (5.52e-2) ≈ | 2.3309e-1 (4.98e-3) − | 4.2715e-1 (1.68e-1) − | 6.0372e-1 (3.81e-2) |
| | 15 | 24 | 5.6174e-1 (6.11e-2) − | 9.6916e-1 (5.62e-2) − | 9.9844e-1 (1.87e-3) + | 4.2565e-1 (9.01e-2) − | 1.7647e-1 (1.07e-3) − | 3.4794e-1 (1.25e-1) − | 9.8331e-1 (3.41e-2) |
| WFG2 | 5 | 14 | 9.8733e-1 (1.42e-3) − | 9.7311e-1 (4.97e-3) − | 9.9558e-1 (4.48e-4) + | 9.5460e-1 (5.80e-3) − | 9.8623e-1 (2.35e-3) − | 9.3538e-1 (2.63e-2) − | 9.9007e-1 (1.26e-3) |
| | 8 | 17 | 9.9154e-1 (1.86e-3) − | 9.5583e-1 (8.79e-3) − | 9.9509e-1 (2.06e-3) ≈ | 9.2574e-1 (9.12e-3) − | 9.9385e-1 (2.06e-3) − | 9.4742e-1 (7.31e-2) − | 9.9589e-1 (1.34e-3) |
| | 10 | 19 | 9.8643e-1 (2.96e-3) − | 9.6432e-1 (6.65e-3) − | 9.9513e-1 (3.22e-3) + | 9.2499e-1 (5.02e-3) − | 9.8943e-1 (3.21e-3) − | 9.6186e-1 (5.49e-2) − | 9.9358e-1 (1.53e-3) |
| | 15 | 24 | 9.9134e-1 (2.78e-3) − | 9.4837e-1 (7.53e-3) − | 9.9044e-1 (1.22e-2) − | 9.1478e-1 (2.01e-2) − | 9.5738e-1 (5.51e-3) − | 9.0030e-1 (5.49e-2) − | 9.9622e-1 (1.92e-3) |
| WFG3 | 5 | 14 | 1.1273e-1 (2.13e-2) ≈ | 1.0217e-1 (2.11e-2) ≈ | 1.8467e-1 (1.83e-2) + | 2.5419e-2 (1.98e-2) − | 1.7822e-1 (2.08e-2) + | 9.2122e-2 (3.37e-2) − | 1.0944e-1 (1.81e-2) |
| | 8 | 17 | 5.5620e-2 (1.09e-2) + | 5.0087e-4 (1.66e-3) − | 2.9707e-2 (2.67e-2) + | 0.0000e+0 (0.00e+0) − | 1.8234e-3 (5.27e-3) − | 1.7016e-2 (1.79e-2) ≈ | 9.3102e-3 (1.03e-2) |
| | 10 | 19 | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) ≈ | 7.7811e-3 (2.02e-2) ≈ | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) ≈ | 1.1158e-3 (4.58e-3) ≈ | 0.0000e+0 (0.00e+0) |
| | 15 | 24 | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) ≈ | 0.0000e+0 (0.00e+0) |
| WFG4 | 5 | 14 | 7.5722e-1 (5.13e-3) − | 7.7463e-1 (2.46e-3) − | 7.7173e-1 (9.43e-4) − | 7.0595e-1 (2.14e-2) − | 6.9739e-1 (7.98e-3) − | 1.2707e-1 (4.83e-2) − | 7.9323e-1 (3.20e-3) |
| | 8 | 17 | 8.7073e-1 (5.24e-3) − | 8.0030e-1 (1.94e-2) − | 8.9076e-1 (2.82e-2) − | 4.3702e-1 (6.79e-2) − | 7.3089e-1 (1.67e-2) − | 1.0701e-1 (3.31e-2) − | 8.9359e-1 (3.69e-3) |
| | 10 | 19 | 8.8252e-1 (8.22e-3) − | 8.0849e-1 (1.50e-2) − | 9.3088e-1 (6.67e-3) − | 4.0130e-1 (5.19e-2) − | 7.6198e-1 (1.72e-2) − | 9.8942e-2 (2.53e-2) − | 9.4281e-1 (2.02e-3) |
| | 15 | 24 | 9.1518e-1 (6.08e-3) − | 6.4140e-1 (8.83e-2) − | 9.4643e-1 (7.67e-2) − | 3.1980e-1 (4.64e-2) − | 4.6771e-1 (3.27e-2) − | 1.3479e-1 (4.83e-2) − | 9.6494e-1 (2.84e-3) |
| WFG5 | 5 | 14 | 7.2869e-1 (4.47e-3) ≈ | 7.2934e-1 (3.02e-3) ≈ | 7.2370e-1 (4.14e-4) − | 6.7519e-1 (1.21e-2) − | 6.7630e-1 (1.07e-2) − | 5.9091e-1 (1.20e-1) − | 7.3156e-1 (5.03e-3) |
| | 8 | 17 | 8.2910e-1 (4.77e-3) + | 7.5574e-1 (1.54e-2) − | 8.4917e-1 (1.63e-2) + | 5.0898e-1 (1.69e-2) − | 6.9771e-1 (1.32e-2) − | 1.7430e-1 (2.16e-1) − | 8.2507e-1 (4.33e-3) |
| | 10 | 19 | 8.4590e-1 (5.41e-3) − | 7.3489e-1 (1.36e-2) − | 8.8213e-1 (2.55e-2) + | 4.8459e-1 (2.37e-2) − | 7.3104e-1 (1.50e-2) − | 5.2592e-1 (2.98e-1) − | 8.6466e-1 (4.00e-3) |
| | 15 | 24 | 8.5132e-1 (4.75e-3) − | 5.9963e-1 (2.74e-2) − | 8.8889e-1 (7.36e-2) + | 3.1942e-1 (3.42e-2) − | 3.8303e-1 (5.03e-2) − | 1.1990e-1 (1.53e-1) − | 8.8599e-1 (2.56e-3) |
| WFG6 | 5 | 14 | 7.0308e-1 (1.08e-2) − | 7.1511e-1 (1.89e-2) − | 7.0528e-1 (1.50e-2) − | 5.9924e-1 (3.15e-2) − | 5.3946e-1 (5.87e-3) − | 1.7326e-1 (7.76e-2) − | 7.3935e-1 (1.67e-2) |
| | 8 | 17 | 8.1299e-1 (1.45e-2) − | 7.2037e-1 (3.25e-2) − | 8.2306e-1 (2.02e-2) ≈ | 2.8885e-1 (3.43e-2) − | 6.1678e-1 (3.31e-2) − | 2.1535e-1 (1.50e-1) − | 8.2837e-1 (1.71e-2) |
| | 10 | 19 | 8.3426e-1 (1.99e-2) − | 7.2534e-1 (3.31e-2) − | 8.7041e-1 (2.08e-2) ≈ | 2.3678e-1 (4.67e-2) − | 6.6546e-1 (1.54e-2) − | 4.1976e-1 (1.98e-1) − | 8.7871e-1 (1.44e-2) |
| | 15 | 24 | 8.5771e-1 (2.67e-2) − | 6.0780e-1 (3.13e-2) − | 8.6294e-1 (7.51e-2) ≈ | 1.1166e-1 (4.12e-2) − | 3.7552e-1 (4.36e-2) − | 3.4096e-1 (2.58e-1) − | 8.8582e-1 (2.19e-2) |
| WFG7 | 5 | 14 | 7.7699e-1 (4.06e-3) − | 7.7558e-1 (3.36e-3) − | 7.7134e-1 (8.53e-4) − | 6.5274e-1 (9.91e-3) − | 6.9846e-1 (9.97e-3) − | 2.6829e-1 (1.10e-1) − | 7.9622e-1 (3.25e-3) |
| | 8 | 17 | 8.9814e-1 (2.96e-3) ≈ | 8.4468e-1 (1.42e-2) − | 9.0162e-1 (1.48e-2) + | 3.5297e-1 (9.15e-3) − | 7.2631e-1 (1.58e-2) − | 1.8559e-1 (7.26e-2) − | 8.9620e-1 (4.95e-3) |
| | 10 | 19 | 9.1810e-1 (4.68e-3) − | 8.5662e-1 (1.61e-2) − | 9.3985e-1 (6.81e-3) − | 3.3749e-1 (1.22e-2) − | 7.6920e-1 (2.92e-2) − | 4.2638e-1 (2.88e-1) − | 9.4475e-1 (3.34e-3) |
| | 15 | 24 | 9.4536e-1 (5.54e-3) − | 8.0610e-1 (9.20e-2) − | 9.6011e-1 (6.50e-2) − | 1.6415e-1 (3.43e-2) − | 3.9707e-1 (3.01e-2) − | 1.7414e-1 (6.46e-2) − | 9.6975e-1 (2.37e-3) |
| WFG8 | 5 | 14 | 6.4361e-1 (9.64e-3) − | 6.7654e-1 (1.62e-2) − | 6.5081e-1 (5.41e-3) − | 5.3030e-1 (9.80e-2) − | 5.5886e-1 (3.67e-2) − | 6.7352e-2 (5.68e-2) − | 6.9045e-1 (4.40e-3) |
| | 8 | 17 | 7.1729e-1 (1.27e-2) − | 7.4528e-1 (2.98e-2) − | 7.6476e-1 (2.93e-2) − | 7.6956e-2 (6.47e-2) − | 6.3136e-1 (2.64e-2) − | 1.6848e-1 (6.96e-2) − | 8.0945e-1 (2.97e-2) |
| | 10 | 19 | 7.6192e-1 (1.38e-2) − | 7.3271e-1 (4.43e-2) − | 8.4147e-1 (2.20e-2) − | 3.9106e-2 (2.70e-2) − | 6.2607e-1 (2.70e-2) − | 2.6176e-1 (1.13e-1) − | 8.8875e-1 (1.73e-2) |
| | 15 | 24 | 8.3226e-1 (1.44e-2) − | 8.3479e-1 (1.27e-1) ≈ | 8.8215e-1 (9.73e-2) − | 1.9920e-1 (3.17e-1) − | 2.4099e-1 (2.70e-2) − | 1.8840e-1 (6.47e-2) − | 8.9127e-1 (1.56e-2) |
| WFG9 | 5 | 14 | 7.2132e-1 (9.49e-3) − | 7.1545e-1 (1.74e-2) − | 7.1173e-1 (2.81e-2) − | 6.2842e-1 (4.85e-2) − | 6.9859e-1 (1.52e-2) − | 2.7072e-1 (1.40e-1) − | 7.5216e-1 (5.10e-3) |
| | 8 | 17 | 7.6834e-1 (5.87e-2) − | 6.9381e-1 (1.98e-2) − | 7.6437e-1 (7.06e-2) − | 3.7348e-1 (1.12e-1) − | 7.0385e-1 (3.31e-2) − | 2.1905e-1 (1.77e-1) − | 8.2233e-1 (9.17e-3) |
| | 10 | 19 | 7.9797e-1 (4.63e-2) − | 6.2493e-1 (3.25e-2) − | 8.3891e-1 (4.80e-2) ≈ | 3.2459e-1 (9.29e-2) − | 7.3808e-1 (1.88e-2) − | 6.9767e-1 (1.15e-1) − | 8.5906e-1 (5.95e-3) |
| | 15 | 24 | 7.5947e-1 (5.78e-2) − | 5.8760e-1 (1.12e-1) − | 8.4972e-1 (1.01e-1) − | 1.8275e-1 (1.57e-1) − | 5.5950e-1 (4.61e-2) − | 3.3564e-1 (2.26e-1) − | 8.6647e-1 (1.54e-2) |
| +/-/≈ | | | 2/29/5 | 1/30/5 | 12/17/7 | 1/32/3 | 1/32/2 | 0/32/3 | |

Overall, NAEA performed better than VaEA, MOEA/DD, NSGA-III, MOEA/D, MOEA/D-M2M and MaOEA/IGD on the WFG test suite in terms of hypervolume. NAEA achieved the best hypervolume on 23 test instances out of 36 WFG test instances considered in this work, while NSGA-III achieved the best hypervolume on 11 instances. VaEA and MOEA/DD produced one best result each. Note that NAEA performed very well in the WFG test suite, even on multimodal testing instances such as WFG4 and WFG9.

In order to compare the performance of each algorithm more intuitively, their results on WFG7 and WFG9 of the parallel coordinates system on the 10-dimensional problems are presented. As can be seen from Fig. 6, NAEA performed best in terms of both convergence and distribution, while VaEA performed second only to it. As for NSGA-III and MaOEA/IGD, the final solutions concentrate mainly on the boundary parts of the PF. The distribution of MOEA/D, MOEA/D-M2M and MOEA/DD is not very good, which can be attributed to the lack of a normalization procedure before the evaluation of individuals. At the same time, we can observe similar results from Fig. 7, and NAEA is still the best performing algorithm.

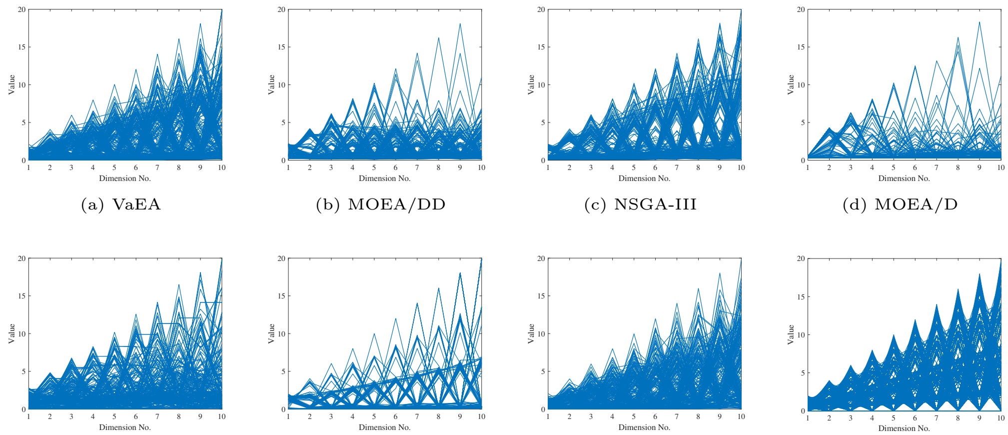



(a) VaEA (b) MOEA/DD (c) NSGA-III (d) MOEA/D

(e) MOEA/D-M2M (f) MaOEA/IGD (g) NAEA (h) True PF

Figure 6: WFG7 10-dimensional parallel coordinates system.

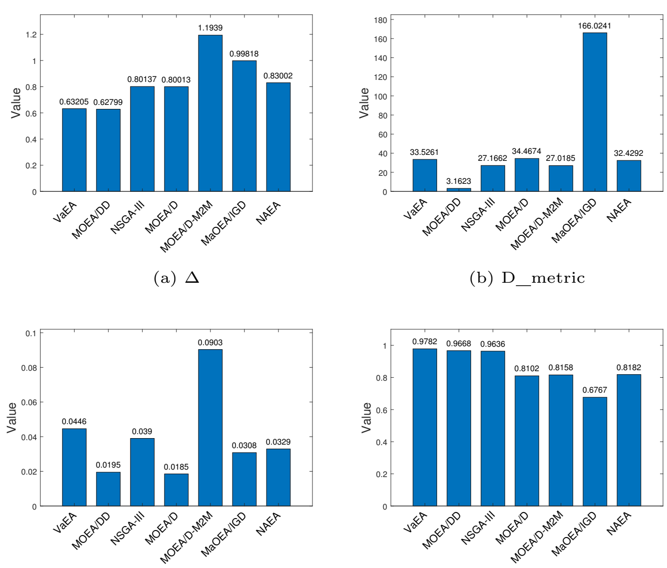
An interesting phenomenon that can be derived from the experimental results is that MOEA/D, MOEA/D-M2M and MOEA/DD obtained relatively poor IGD and HV indicators on this test suite compared with their peer algorithms. This may be attributed to the fact that the PFs of the WFG test suite are irregular, discontinued or mixed, and scaled with different ranges in each objective. Therefore, decomposition-based algorithms




(a) VaEA (b) MOEA/DD (c) NSGA-III (d) MOEA/D

(e) MOEA/D-M2M (e) MaOEA/IGD (f) NAEA (h) True PF

Figure 7: WFG9 10-dimensional parallel coordinates system.




(a) Δ (b) D_metric

(c) GD (d) Purity

Figure 8: Plot of the bar of four metrics for seven algorithms on the 10-objective WFG2.

with fixed weight vectors may suffer from performance degeneration in cases where the problems front shape does not match the shape of the distribution of weight vectors. The main reason for NAEA performing very well in most WFG test problems is that it does not rely on fixed weight vectors to guide population evolution. Niche-based density estimation ensures that the search area is wide enough and dynamically adjusts the search direction of the population.

Table 6 provides results for the $\Delta$ metric of the WFG test suites. NAEA produced significantly better results than all peer algorithms on almost all the test instances, except for WFG1, WFG2, 5-objective and 8-objective WFG3, and 15-objective WFG8. For WFG1 and WFG3 with five and eight objectives, VaEA was the most effective algorithm. MOEA/DD had an outstanding performance on WFG2 test instances. These results also indicate that NAEA can obtain a good distribution of solutions on most problems, and the distribution of NAEA is well reflected in figures 3−7.

Furthermore, Fig. 8 presents the bar of four metrics in all comparison algorithms on the 10-objective WFG2. NAEA has achieved better performance than most algorithms in $\Delta$ metric. Decomposition-based methods such as MOEA/DD and MOEA/D performed better in D_metric. The reason may be that the reference line of the D_metric is similar to the reference vector based on the decomposition algorithm. NAEA achieved moderate performance in both GD and Purity. The experimental results demonstrate that NAEA can better balance convergence and diversity.

### *4.7. Results on the imbalanced test suite*

To establish the extent of NAEA's explorative capability, this paper conducted a comparative experiment between NAEA and MOEA/D-M2M on imbalanced test problems [49]. The imbalance problems feature a serious imbalance of diversity preservation and achieving convergence. For the imbalanced problems IMB1-IMB10, the population size was set to 100 on problems with two objectives and 105 on problems with three objectives. It can be seen from Table 7 that the proposed algorithm showed a significant advantage over MOEA/D-M2M in almost all test cases. Specifically, NAEA obtained the best results on 29 out of 30 test instances. The reason might be that the imbalance problems require equal or more attention be given to diversity than to convergence. Compared with MOEA/D-M2M, the niche-based density estimation strategy of NAEA is very effective in maintaining population diversity.

Table 6: Δ results (mean and standard deviation) of the state-of-the-art algorithms on the WFG test suite.

| Problem | $m$ | $D$ | VaEA | MOEA/DD | NSGA-III | MOEA/D | MOEA/D-M2M | MaOEA/IGD | NAEA |
|---|---|---|---|---|---|---|---|---|---|
| WFG1 | 5 | 14 | 5.0443e-1 (3.28e-2) ≈ | 7.5287e-1 (1.03e-1) − | 5.1079e-1 (1.95e-2) ≈ | 5.3647e-1 (5.58e-2) ≈ | 1.1046e+0 (4.39e-2) − | 5.5539e+0 (1.18e+1) − | 5.1577e-1 (2.37e-2) |
| | 8 | 17 | 6.3513e-1 (4.62e-2) + | 1.2030e+0 (5.94e-2) − | 7.2461e-1 (1.02e-1) ≈ | 8.0171e-1 (4.30e-2) − | 1.1112e+0 (3.28e-2) − | 2.2537e+0 (4.38e+0) − | 7.5682e-1 (4.99e-2) |
| | 10 | 19 | 6.0699e-1 (2.61e-2) ≈ | 1.1758e+0 (6.06e-2) − | 7.1595e-1 (1.04e-1) − | 8.4301e-1 (6.21e-2) − | 1.0213e+0 (2.43e-2) − | 9.2854e-1 (1.68e+0) − | 6.1270e-1 (5.61e-2) |
| | 15 | 24 | 8.6592e-1 (3.45e-2) ≈ | 1.1596e+0 (8.65e-2) − | 1.0243e+0 (2.51e-1) ≈ | 9.9741e-1 (4.39e-3) − | 9.4120e-1 (1.66e-2) ≈ | 1.5521e+0 (1.13e+0) − | 9.1250e-1 (8.48e-2) |
| WFG2 | 5 | 14 | 5.1029e-1 (2.21e-2) + | 3.5276e-1 (8.74e-3) + | 4.3999e-1 (8.25e-3) + | 4.4271e-1 (1.03e-2) + | 1.0221e+0 (5.85e-2) − | 1.0813e+0 (6.30e-2) − | 5.6485e-1 (3.20e-2) |
| | 8 | 17 | 6.9453e-1 (2.47e-2) + | 5.9404e-1 (7.52e-3) + | 7.8033e-1 (1.25e-1) ≈ | 7.4864e-1 (2.16e-2) + | 1.2716e+0 (4.17e-2) − | 1.0443e+0 (1.82e-2) − | 8.7207e-1 (2.90e-2) |
| | 10 | 19 | 6.3205e-1 (3.12e-2) + | 6.2799e-1 (1.44e-2) + | 8.0137e-1 (9.71e-2) + | 8.0013e-1 (3.91e-2) + | 1.1939e+0 (4.57e-2) − | 9.9818e-1 (1.39e-2) − | 8.3002e-1 (4.39e-2) |
| | 15 | 24 | 1.0337e+0 (3.07e-2) + | 9.6316e-1 (5.33e-3) + | 1.0523e+0 (1.70e-1) + | 9.9229e-1 (7.49e-3) + | 1.0384e+0 (2.88e-2) + | 1.0100e+0 (2.01e-2) + | 1.2893e+0 (6.91e-2) |
| WFG3 | 5 | 14 | 2.7369e-1 (1.42e-2) + | 1.2117e+0 (5.84e-2) − | 7.7390e-1 (6.43e-2) − | 4.7268e-1 (4.07e-2) − | 1.2985e+0 (5.02e-2) − | 1.0020e+0 (1.15e-3) − | 2.9332e-1 (2.13e-2) |
| | 8 | 17 | 2.8398e-1 (1.60e-2) + | 1.6660e+0 (3.40e-2) − | 8.0470e-1 (8.55e-2) − | 5.3185e-1 (1.55e-2) − | 1.4015e+0 (4.00e-2) − | 6.8461e-1 (0.00e+0) ≈ | 3.1145e-1 (2.85e-2) |
| | 10 | 19 | 2.7995e-1 (1.34e-2) ≈ | 1.5181e+0 (4.60e-2) − | 7.3415e-1 (1.27e-1) − | 6.7908e-1 (2.66e-2) − | 1.2340e+0 (4.25e-2) − | 7.1010e-1 (9.29e-2) − | 2.7433e-1 (1.48e-2) |
| | 15 | 24 | 4.7073e-1 (3.89e-2) − | 7.0441e-1 (3.34e-2) − | 9.8780e-1 (2.20e-1) − | 1.2493e+0 (1.62e-2) − | 1.0600e+0 (3.73e-2) − | 1.0972e+0 (1.54e-1) − | 3.9471e-1 (3.37e-2) |
| WFG4 | 5 | 14 | 1.9635e-1 (1.43e-2) − | 3.5272e-1 (9.81e-3) − | 2.3025e-1 (1.40e-3) − | 3.7555e-1 (3.45e-2) − | 8.2716e-1 (5.32e-2) − | 1.1298e+0 (2.62e-1) − | 1.8098e-1 (1.26e-2) |
| | 8 | 17 | 2.4607e-1 (2.20e-2) − | 4.1046e-1 (1.69e-2) − | 2.9079e-1 (1.41e-1) − | 8.3277e-1 (2.01e-1) − | 8.8845e-1 (5.99e-2) − | 1.0046e+0 (2.09e-2) − | 1.9556e-1 (1.52e-2) |
| | 10 | 19 | 2.0692e-1 (1.02e-2) − | 4.4439e-1 (1.42e-2) − | 3.5557e-1 (8.41e-2) − | 9.0105e-1 (2.96e-1) − | 7.8633e-1 (3.56e-2) − | 1.0208e+0 (9.28e-2) − | 1.7380e-1 (1.26e-2) |
| | 15 | 24 | 3.5318e-1 (2.52e-2) − | 6.5339e-1 (1.67e-1) − | 6.0760e-1 (1.82e-1) − | 1.0714e+0 (6.35e-2) − | 9.2326e-1 (3.73e-2) − | 1.0689e+0 (9.90e-2) − | 2.4912e-1 (2.90e-2) |
| WFG5 | 5 | 14 | 2.0264e-1 (1.10e-2) − | 3.4991e-1 (7.70e-3) − | 2.3245e-1 (5.32e-4) − | 4.0622e-1 (2.49e-2) − | 9.7093e-1 (5.69e-2) − | 5.3908e-1 (2.08e-1) − | 1.7685e-1 (1.50e-2) |
| | 8 | 17 | 2.6546e-1 (2.15e-2) − | 4.0461e-1 (2.52e-2) − | 2.4023e-1 (5.96e-2) − | 9.8507e-1 (1.00e-1) − | 1.0526e+0 (5.33e-2) − | 8.5420e-1 (2.77e-1) − | 1.8878e-1 (1.35e-2) |
| | 10 | 19 | 2.3526e-1 (9.45e-3) − | 3.8798e-1 (2.60e-2) − | 3.8959e-1 (1.76e-1) − | 1.2554e+0 (6.90e-2) − | 1.0029e+0 (3.51e-2) − | 5.9603e-1 (1.96e-1) − | 1.9070e-1 (1.28e-2) |
| | 15 | 24 | 3.8412e-1 (1.76e-2) − | 6.0058e-1 (1.92e-2) − | 6.3441e-1 (1.06e-1) − | 1.2021e+0 (1.13e-2) − | 1.1089e+0 (5.36e-2) − | 1.0149e+0 (5.56e-2) − | 2.4895e-1 (2.44e-2) |
| WFG6 | 5 | 14 | 1.9939e-1 (1.45e-2) − | 3.5773e-1 (8.44e-3) − | 2.3438e-1 (2.51e-3) − | 4.0390e-1 (4.16e-2) − | 1.1367e+0 (4.11e-2) − | 1.2154e+0 (3.26e-1) − | 1.8859e-1 (1.45e-2) |
| | 8 | 17 | 2.4777e-1 (2.72e-2) − | 3.9017e-1 (2.45e-2) − | 2.4099e-1 (5.53e-2) − | 8.7173e-1 (2.18e-1) − | 1.3460e+0 (4.58e-2) − | 3.9127e-1 (1.76e+0) − | 2.2583e-1 (1.38e-2) |
| | 10 | 19 | 2.4413e-1 (1.24e-2) − | 4.0499e-1 (2.30e-2) − | 3.4534e-1 (6.86e-2) − | 9.5743e-1 (2.08e-1) − | 1.2409e+0 (3.89e-2) − | 1.2083e+0 (6.38e-1) − | 2.2206e-1 (1.66e-2) |
| | 15 | 24 | 3.4433e-1 (2.77e-2) ≈ | 6.2209e-1 (2.04e-2) − | 6.6759e-1 (1.75e-1) − | 1.0630e+0 (2.94e-2) − | 1.0997e+0 (3.69e-2) − | 5.7816e-1 (1.33e+0) − | 3.3506e-1 (2.76e-2) |
| WFG7 | 5 | 14 | 1.9091e-1 (1.45e-2) − | 3.5040e-1 (1.06e-2) − | 2.3230e-1 (1.99e-3) − | 4.1074e-1 (5.53e-2) − | 1.0456e+0 (4.28e-2) − | 1.2100e+0 (3.61e-1) − | 1.8011e-1 (1.30e-2) |
| | 8 | 17 | 2.2718e-1 (2.19e-2) − | 4.3519e-1 (1.64e-2) − | 2.3725e-1 (5.63e-2) − | 9.3923e-1 (2.38e-1) − | 1.1917e+0 (5.40e-2) − | 1.4586e+0 (1.47e+0) − | 1.9581e-1 (1.70e-2) |
| | 10 | 19 | 2.1914e-1 (1.20e-2) − | 4.6047e-1 (2.71e-2) − | 3.9012e-1 (1.35e-1) − | 1.0256e+0 (2.31e-1) − | 1.0995e+0 (4.86e-2) − | 7.6370e-1 (2.20e+0) − | 1.7987e-1 (1.17e-2) |
| | 15 | 24 | 3.2428e-1 (2.36e-2) − | 7.3103e-1 (1.53e-1) − | 7.8691e-1 (2.18e-1) − | 1.0761e+0 (2.77e-2) − | 1.1158e+0 (3.78e-2) − | 7.2192e-1 (2.65e+0) − | 2.7996e-1 (2.43e-2) |
| WFG8 | 5 | 14 | 1.8915e-1 (1.23e-2) − | 4.2488e-1 (2.27e-2) − | 2.5449e-1 (2.86e-2) − | 4.1798e-1 (3.93e-2) − | 1.1317e+0 (5.86e-2) − | 1.0730e+0 (7.32e-2) − | 1.7006e-1 (8.49e-3) |
| | 8 | 17 | 2.2570e-1 (1.32e-2) ≈ | 4.7720e-1 (5.05e-2) − | 4.5746e-1 (9.29e-2) − | 1.0037e+0 (2.26e-1) − | 1.2447e+0 (5.12e-2) − | 3.2749e-1 (2.55e+0) ≈ | 2.2161e-1 (2.49e-2) |
| | 10 | 19 | 2.0462e-1 (1.91e-2) ≈ | 4.5796e-1 (5.01e-2) − | 5.3375e-1 (1.19e-1) − | 9.7366e-1 (2.14e-1) − | 1.1689e+0 (3.77e-2) − | 4.2619e-1 (3.83e+0) − | 2.0251e-1 (1.63e-2) |
| | 15 | 24 | 3.1992e-1 (2.10e-2) + | 9.8477e-1 (2.97e-1) − | 7.5190e-1 (1.49e-1) − | 1.1251e+0 (1.81e-1) − | 1.0882e+0 (4.48e-2) − | 1.2777e+0 (2.96e+0) − | 3.6203e-1 (3.99e-2) |
| WFG9 | 5 | 14 | 1.9657e-1 (1.33e-2) − | 3.7193e-1 (1.46e-2) − | 2.5460e-1 (9.43e-3) − | 4.3862e-1 (2.60e-2) − | 9.2951e-1 (8.12e-2) − | 1.2440e+0 (4.32e-1) − | 1.7111e-1 (1.06e-2) |
| | 8 | 17 | 2.4889e-1 (1.40e-2) − | 4.2083e-1 (2.24e-2) − | 2.7441e-1 (9.55e-2) − | 1.1994e+0 (3.86e-2) − | 9.8286e-1 (5.53e-2) − | 1.1281e+0 (3.72e-1) − | 1.7493e-1 (1.68e-2) |
| | 10 | 19 | 2.1989e-1 (1.28e-2) − | 4.2732e-1 (3.34e-2) − | 4.0047e-1 (1.40e-1) − | 1.3014e+0 (1.19e-1) − | 9.5708e-1 (4.05e-2) − | 5.9108e-1 (2.17e-1) − | 1.6522e-1 (1.24e-2) |
| | 15 | 24 | 3.7458e-1 (2.01e-2) − | 7.7511e-1 (1.56e-1) − | 6.5402e-1 (1.46e-1) − | 1.1357e+0 (6.84e-2) − | 1.0259e+0 (4.99e-2) − | 7.3364e-1 (1.47e+0) − | 2.4680e-1 (1.41e-2) |
| +/-/≈ | | | 8/21/7 | 4/32/0 | 3/29/4 | 4/31/1 | 1/34/1 | 1/33/2 | |

Table 7: IGD, HV and Δ results (mean and standard deviation) on the IMB test suite.

| IGD | $m$ | $D$ | MOEA/D-M2M | NAEA |
|---|---|---|---|---|
| IMB1 | 2 | 10 | 5.6446e-3 (2.73e-4) − | 5.2848e-3 (3.05e-4) |
| IMB2 | 2 | 10 | 5.5692e-3 (2.83e-4) − | 4.4157e-3 (2.51e-4) |
| IMB3 | 2 | 10 | 6.4503e-3 (3.43e-4) − | 4.5742e-3 (1.51e-4) |
| IMB4 | 3 | 10 | 3.1970e-1 (3.69e-3) − | 3.1015e-1 (1.48e-4) |
| IMB5 | 3 | 10 | 1.3126e-1 (1.93e-3) − | 9.8171e-2 (1.02e-3) |
| IMB6 | 3 | 10 | 3.0346e-1 (1.81e-3) − | 2.9587e-1 (6.71e-4) |
| IMB7 | 2 | 10 | 7.8096e-3 (4.26e-4) − | 5.2766e-3 (2.70e-4) |
| IMB8 | 2 | 10 | 8.1979e-3 (3.48e-4) − | 4.3109e-3 (1.09e-4) |
| IMB9 | 2 | 10 | 8.3362e-3 (3.85e-4) − | 4.5131e-3 (1.05e-4) |
| IMB10 | 3 | 10 | 3.2415e-1 (4.06e-3) ≈ | 3.2490e-1 (3.56e-3) |
| +/-/≈ | | | 0/9/1 | |
| HV | $m$ | $D$ | MOEA/D-M2M | NAEA |
| IMB1 | 2 | 10 | 7.1852e-1 (3.09e-4) ≈ | 7.1870e-1 (4.54e-4) |
| IMB2 | 2 | 10 | 5.8027e-1 (3.30e-4) − | 5.8162e-1 (2.93e-4) |
| IMB3 | 2 | 10 | 3.4514e-1 (3.93e-4) − | 3.4709e-1 (1.98e-4) |
| IMB4 | 3 | 10 | 1.1306e-1 (1.01e-2) − | 1.6471e-1 (1.27e-3) |
| IMB5 | 3 | 10 | 4.8893e-1 (2.89e-3) − | 5.2630e-1 (2.08e-3) |
| IMB6 | 3 | 10 | 1.3253e-1 (5.74e-3) − | 1.8334e-1 (2.00e-3) |
| IMB7 | 2 | 10 | 7.1337e-1 (6.95e-4) − | 7.1860e-1 (3.19e-4) |
| IMB8 | 2 | 10 | 5.7399e-1 (6.39e-4) − | 5.8160e-1 (1.57e-4) |
| IMB9 | 2 | 10 | 3.4074e-1 (6.11e-4) − | 3.4715e-1 (9.61e-5) |
| IMB10 | 3 | 10 | 9.5449e-2 (7.33e-3) − | 1.2551e-1 (6.30e-3) |
| +/-/≈ | | | 0/9/1 | |
| Δ | $m$ | $D$ | MOEA/D-M2M | NAEA |
| IMB1 | 2 | 10 | 7.6536e-1 (5.81e-2) − | 5.4725e-1 (3.81e-2) |
| IMB2 | 2 | 10 | 8.4601e-1 (2.24e-1) − | 3.1376e-1 (4.92e-2) |
| IMB3 | 2 | 10 | 7.4266e-1 (6.18e-2) − | 2.0917e-1 (3.07e-2) |
| IMB4 | 3 | 10 | 1.3899e+0 (1.22e-1) − | 5.8297e-1 (9.52e-2) |
| IMB5 | 3 | 10 | 1.3097e+0 (6.86e-2) − | 1.8768e-1 (1.58e-2) |
| IMB6 | 3 | 10 | 1.1406e+0 (4.84e-2) − | 3.7005e-1 (2.63e-2) |
| IMB7 | 2 | 10 | 1.3000e+0 (2.46e-1) − | 5.4145e-1 (3.27e-2) |
| IMB8 | 2 | 10 | 1.2360e+0 (2.17e-1) − | 3.0034e-1 (2.88e-2) |
| IMB9 | 2 | 10 | 1.3949e+0 (1.82e-1) − | 2.3818e-1 (1.97e-2) |
| IMB10 | 3 | 10 | 1.3255e+0 (9.55e-2) − | 4.4530e-1 (3.07e-2) |
| +/-/≈ | | | 0/10/0 | |

### *4.8. Discussion*

To balance convergence and diversity, the proposed algorithm integrates the concepts of niche and angle. The performance of the NAEA has been investigated on DTLZ and WFG benchmark problems with up to 15 objectives. It can be seen from the experimental results that the proposed algorithm shows its high performance in most test instances. It is worth noting that the distribution of NAEA shows a significant improvement over other peer algorithms. However, NAEA and VaEA do not perform very well in DTLZ1 and DTLZ3, which are representative of the hard-to-converge problems. The results indicate that for Pareto-based algorithms, more effective convergence strategies need to be developed on the basis of maintaining diversity.

### *4.9. Constrained NAEA*

NAEA does not need to use any weight vectors and maintains good distribution. This feature inspires us to speculate that it may be able to bypass the infeasible region of the constraint problem and obtain a true PF. We further extend NAEA to deal with constraints on many-objective optimization problems, and the resulting algorithm is called CNAEA.

To measure the degree of constraint violation, these constraints can be commonly summarized into a scalar value as follows:

$$CV(\mathbf{x}) = \sum_{i=1}^{p} |min(g_i(\mathbf{x}), 0)| + \sum_{j=1}^{q} |h_j(\mathbf{x})|, \tag{17}$$

where $g_i(\mathbf{x}) \geq 0$ is the $i$th inequality constraint, and $h_j(\mathbf{x}) = 0$ is the $j$th equality constraint. Based on this definition, if $CV(\mathbf{x}) = 0$, the solution $\mathbf{x}$ is feasible; otherwise, it is infeasible. The constrained dominance principle (CDP) [5] is the simplest and most commonly used constraint-handling technique in constrained multiobjective optimization. In CDP, any feasible solution is preferred to an infeasible solution and among two infeasible solutions, the one having the smaller constraint violation is preferred.

When dealing with the constraint problem, the environmental selection is slightly different. First, we divide the combined population $R$ of parent and offspring into two parts: the feasible solutions and infeasible solutions. If the number of feasible solutions is larger than $N$, we can temporarily regard this constraint problem as a problem with only box constraints, and use Algorithm 2 to select $N$ solutions from all feasible solutions. Otherwise, we will sort $R$ in an ascending order based on the $CV$ values. Then the first $N$ solutions with the least degree of constraint violation are selected for the next generation.

To verify the performance of CNAEA through the experimental study, we compared our proposed algorithm with CNSGA-III [20], which is the constrained version of NSGA-III, on CDTLZ benchmarks. The experimental settings were consistent with the unconstrained many-objective optimization algorithms unless otherwise specified.

The experimental results of IGD, HV and $\Delta$ are summarized as shown in Table 8. It can be seen that CNAEA performed best, presenting a clear advantage over CNSGA-III on the majority of the test problems. Accurately, the proportion of the test instances where CNAEA performed better than CNSGA-III on the IGD, HV, and Δ metric were 9/12, 5/12 and 8/12, respectively. Conversely, the proportion in which the peer algorithms defeated CNAEA was 2/12 for all three indicators. In general, our proposed CNAEA is easy to implement and can effectively deal with many-objective constrained optimization problems.

Table 8: IGD, HV and Δ results (mean and standard deviation) on the C-DTLZ test suite.

| IGD | $m$ | $D$ | CNSGA-III | CNAEA |
|---|---|---|---|---|
| C1-DTLZ1 | 5 | 9 | 5.1988e-2 (9.33e-4) + | 5.4210e-2 (8.91e-4) |
| | 8 | 12 | 1.0073e-1 (8.78e-3) + | 1.0486e-1 (1.21e-3) |
| | 10 | 14 | 1.1033e-1 (8.08e-3) − | 1.0592e-1 (5.43e-4) |
| | 15 | 19 | 1.7275e-1 (9.72e-3) − | 1.3821e-1 (1.43e-3) |
| C2-DTLZ2 | 5 | 14 | 1.3509e-1 (1.21e-3) − | 1.3327e-1 (1.37e-3) |
| | 8 | 17 | 3.1225e-1 (1.61e-1) − | 2.4545e-1 (2.65e-3) |
| | 10 | 19 | 3.0958e-1 (9.93e-2) ≈ | 3.0463e-1 (1.39e-1) |
| | 15 | 24 | 4.1431e-1 (1.45e-1) − | 3.5731e-1 (2.01e-1) |
| C3-DTLZ4 | 5 | 14 | 2.7034e-1 (7.85e-2) − | 2.6029e-1 (1.87e-3) |
| | 8 | 17 | 6.8613e-1 (9.23e-2) − | 5.0570e-1 (3.67e-3) |
| | 10 | 19 | 7.5828e-1 (8.13e-2) − | 5.4866e-1 (2.06e-3) |
| | 15 | 24 | 8.8423e-1 (5.60e-2) − | 7.6160e-1 (4.55e-3) |
| +/-/≈ | | | 2/9/1 | |
| **HV** | $m$ | $D$ | CNSGA-III | CNAEA |
| C1-DTLZ1 | 5 | 9 | 9.5721e-1 (1.95e-2) ≈ | 9.5830e-1 (4.86e-3) |
| | 8 | 12 | 9.7463e-1 (2.39e-2) ≈ | 9.7665e-1 (1.68e-2) |
| | 10 | 14 | 9.5939e-1 (2.31e-2) − | 9.8194e-1 (6.23e-3) |
| | 15 | 19 | 9.8053e-1 (1.96e-2) ≈ | 9.6885e-1 (2.51e-2) |
| C2-DTLZ2 | 5 | 14 | 7.5008e-1 (3.00e-3) + | 7.3929e-1 (2.84e-3) |
| | 8 | 17 | 7.8445e-1 (1.45e-1) ≈ | 8.3685e-1 (4.00e-3) |
| | 10 | 19 | 8.7231e-1 (3.35e-2) + | 8.6358e-1 (1.55e-1) |
| | 15 | 24 | 8.7197e-1 (1.62e-1) − | 8.7719e-1 (2.30e-1) |
| C3-DTLZ4 | 5 | 14 | 9.5829e-1 (1.31e-2) − | 9.5832e-1 (9.97e-4) |
| | 8 | 17 | 9.8022e-1 (1.09e-2) − | 9.9403e-1 (4.40e-4) |
| | 10 | 19 | 9.9222e-1 (5.59e-3) − | 9.9867e-1 (1.01e-4) |
| | 15 | 24 | 9.9922e-1 (7.07e-4) ≈ | 9.9907e-1 (1.79e-4) |
| +/-/≈ | | | 2/5/5 | |
| **Δ** | $m$ | $D$ | CNSGA-III | CNAEA |
| C1-DTLZ1 | 5 | 9 | 1.3541e-1 (4.73e-2) + | 1.9814e-1 (1.15e-2) |
| | 8 | 12 | 2.7771e-1 (1.17e-1) ≈ | 2.3092e-1 (1.20e-2) |
| | 10 | 14 | 3.4431e-1 (6.54e-2) − | 2.0310e-1 (9.53e-3) |
| | 15 | 19 | 2.8260e-1 (2.99e-1) ≈ | 3.1392e-1 (1.71e-2) |
| C2-DTLZ2 | 5 | 14 | 1.6401e-1 (5.59e-2) − | 1.0481e-1 (7.80e-3) |
| | 8 | 17 | 6.1373e-1 (2.09e-1) − | 1.1447e-1 (1.82e-2) |
| | 10 | 19 | 7.8689e-1 (3.37e-2) − | 1.2359e-1 (2.97e-2) |
| | 15 | 24 | 8.2978e-1 (2.57e-1) − | 1.0854e-1 (9.60e-2) |
| C3-DTLZ4 | 5 | 14 | 1.1227e-1 (1.06e-1) + | 1.6332e-1 (9.77e-3) |
| | 8 | 17 | 5.0030e-1 (9.67e-2) − | 1.4672e-1 (1.16e-2) |
| | 10 | 19 | 4.8951e-1 (1.25e-1) − | 1.2637e-1 (7.65e-3) |
| | 15 | 24 | 5.7750e-1 (2.28e-1) − | 1.6253e-1 (1.58e-2) |
| +/-/≈ | | | 2/8/2 | |

### *4.10. Computational time complexity*

The computational cost of NAEA mainly comes from its environmental selection. In the environmental selection, the angle between each pair of individuals in the population is first calculated, which requires $O(mN^2)$ computations, where $m$ is the number of objectives and $N$ is the population size. Subsequently, determining the niche radius $\theta$ requires $O(N^2)$ operations in which finding the $m$th smallest angle for an individual needs $O(N)$ comparisons. Then, the niche-based $Ncd$ calculation is implemented, which requires $O(N^2)$ computations. Finally, the computational cost of eliminating individuals from critical front $F_l$ one by one is determined by two operations: finding the worst solutions and updating the $Ncd$. The time complexity of both operations is $O(N)$. It is worth mentioning that in the process of updating $Ncd$, we only need to update the $Ncd$ of the neighbors of the removed individual (i.e., the individuals that are in the same niche of the removed individual). In general, there are very few individuals in the niche due to $\theta$ being set to be the median of the vector angle from all the solutions to their $m$th nearest solution.

To summarize, the overall computational complexity of one generation of NAEA is bounded by $O(mN^2)$. Compared with most popular MOEAs for MaOPs, NAEA is computationally very efficient.

## 5. Conclusion

In this paper, we have proposed a novel many-objective optimization algorithm named NAEA for MaOPs. The main idea is to make use of two s-trategies (niche-based density estimation and angle-based selection) to delete poor individuals one by one during the environmental selection.

From the experimental results, we know the proposed NAEA outperforms other competitors on many test instances. The reasons can be summarized as follows: first, by taking advantage of Pareto dominance, NAEA eliminates a few of the poorest individuals from the population and identifies some excellent individuals. Additionally, a unique environmental selection incorporating the niche-based method is adopted, which can maintain the diversity of the population. Last but not least, in the same niche, finding the individual with the smallest angle of the most crowded individuals, then deleting the individuals with poor convergence can effectively guarantee the convergence of the algorithm.

This work demonstrates that the idea of using niche-based and angle-based selection strategies for MaOPs is very promising. In future research, it will be necessary to investigate how to apply this algorithm to practical MaOPs.

## 6. Acknowledgements

The authors wish to thank the support of the National Natural Science Foundation of China (Grant No. 61876164, 61772178), The MOEA Key Laboratory of Intelligent Computing and Information Processing, the Science and Technology Plan Project of Hunan Province (Grant No. 2016TP1020), the Provinces and Cities Joint Foundation Project (Grant No. 2017JJ4001), Science and Technology Planning Project of Guangdong Province of China (Grant No. 2017B010111005), the Hunan province science and technology project funds (Grant No. 2018TP1036).

[1] S. Tahmasebi, P. Moradi, S. Ghodsi, A. Abdollahpouri, An ideal point based many-objective optimization for community detection of complex networks, Information Sciences 502 (2019) 125–145.

[2] C. Hu, L. Dai, X. Yan, W. Gong, X. Liu, L. Wang, Modified NSGA-III for sensor placement in water distribution system, Information Sciences 509 (2020) 488–500.

[3] A. Zhou, Y. Wang, J. Zhang, Objective extraction via fuzzy clustering in evolutionary many-objective optimization, Information Sciences 509 (2020) 343–355.

[4] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, Q. Zhang, Multiobjective evolutionary algorithms: A survey of the state of the art, Swarm and Evolutionary Computation 1 (1) (2011) 32–49.

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

[6] E. Zitzler, M. Laumanns, L. Thiele, Improving the strength Pareto evolutionary algorithm, EUROGEN 2001, Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems (2000) 95–100.

[7] J. Z. X. C. W. Z. Zhihua Cui, Yu Chang, Improved NSGA-III with selection-and-elimination operator, Swarm and Evolutionary Computation 49 (2019) 23–33.

[8] K. Li, R. Wang, T. Zhang, H. Ishibuchi, Evolutionary many-objective optimization: A comparative study of the state-of-the-art, IEEE Access 6 (2018) 26194–26214.

[9] X. Zou, Y. Chen, M. Liu, L. Kang, A new evolutionary algorithm for solving many-objective optimization problems, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 38 (5) (2008) 1402–1412.

[10] F. Di Pierro, S.-T. Khu, D. A. Savic, An investigation on preference order ranking scheme for multiobjective evolutionary optimization, IEEE Transactions on Evolutionary Computation 11 (1) (2007) 17–45.

[11] K. Deb, M. Mohan, S. Mishra, Evaluating the $\varepsilon$-domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions, Evolutionary Computation 13 (4) (2005) 501–525.

[12] S. Yang, M. Li, X. Liu, J. Zheng, A grid-based evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 17 (5) (2013) 721–736.

[13] M. Li, S. Yang, X. Liu, Shift-based density estimation for Pareto-based algorithms in many-objective optimization, IEEE Transactions on Evolutionary Computation 18 (3) (2013) 348–365.

[14] X. Zhang, Y. Tian, Y. Jin, A knee point-driven evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 19 (6) (2014) 761–776.

[15] Q. Zhang, L. Hui, MOEA/D: a multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 712–731.

[16] 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 Transactions on Evolutionary Computation 18 (4) (2014) 577–601.

[17] K. Li, K. Deb, Q. Zhang, S. Kwong, An evolutionary many-objective optimization algorithm based on dominance and decomposition, IEEE Transactions on Evolutionary Computation 19 (5) (2015) 694–716.

[18] S. Jiang, S. Yang, A strength Pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization, IEEE Transactions on Evolutionary Computation 21 (3) (2017) 329–346.

[19] M. Pal, S. Bandyopadhyay, ESOEA: Ensemble of single objective evolutionary algorithms for many-objective optimization, Swarm and Evolutionary Computation 50 (2019) 100511.

[20] 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 Transactions on Evolutionary Computation 18 (4) (2014) 602–622.

[21] E. Zitzler, L. Thiele, Multiobjective optimization using evolutionary algorithmsa comparative case study, in: International Conference on Parallel Problem Solving from Nature, Springer, 1998, pp. 292–301.

[22] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, V. G. Da Fonseca, Performance assessment of multiobjective optimizers: An analysis and review, IEEE Transactions on Evolutionary Computation 7 (2) (2003) 117–132.

[23] M. Pal, S. Saha, S. Bandyopadhyay, DECOR: differential evolution using clustering based objective reduction for many-objective optimization, Information Sciences 423 (2018) 200–218.

[24] S. Bandyopadhyay, A. Mukherjee, An algorithm for many-objective optimization with reduced objective computations: A study in differential evolution, IEEE Transactions on Evolutionary Computation 19 (3) (2014) 400–413.

[25] Z. He, G. G. Yen, Many-objective evolutionary algorithm: Objective space reduction and diversity improvement, IEEE Transactions on Evolutionary Computation 20 (1) (2015) 145–160.

[26] D. H. Wolpert, W. G. Macready, et al., No free lunch theorems for optimization, IEEE Transactions on Evolutionary Computation 1 (1) (1997) 67–82.

[27] Y. Liu, D. Gong, J. Sun, Y. Jin, A many-objective evolutionary algorithm using a one-by-one selection strategy, IEEE Transactions on Cybernetics 47 (9) (2017) 2689–2702.

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

[29] R. Morgan, M. Gallagher, Sampling techniques and distance metrics in high dimensional continuous landscape analysis: Limitations and improvements, IEEE Transactions on Evolutionary Computation 18 (3) (2013) 456–461.

[30] M. Li, S. Yang, X. Liu, Pareto or non-Pareto: Bi-criterion evolution in multiobjective optimization, IEEE Transactions on Evolutionary Computation 20 (5) (2016) 645–665.

[31] J. rey Horn, N. Nafpliotis, D. E. Goldberg, A niched Pareto genetic algorithm for multiobjective optimization, in: Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, Vol. 1, Citeseer, 1994, pp. 82–87.

[32] K. Ikeda, H. Kita, S. Kobayashi, Failure of Pareto-based MOEAs: Does nondominated really mean near to optimal?, in: Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546), Vol. 2, IEEE, 2001, pp. 957–962.

[33] L. Muflikhah, B. Baharudin, Document clustering using concept space and cosine similarity measurement, in: 2009 International Conference on Computer Technology and Development, Vol. 1, IEEE, 2009, pp. 58–62.

[34] Y. Xiang, Y. Zhou, M. Li, Z. Chen, A vector angle based evolutionary algorithm for unconstrained many-objective optimization, IEEE Transactions on Evolutionary Computation 21 (1) (2017) 131–152.

[35] X. Zhang, Y. Tian, R. Cheng, Y. Jin, An efficient approach to non-dominated sorting for evolutionary multiobjective optimization, IEEE Transactions on Evolutionary Computation 19 (2) (2014) 201–213.

[36] B.-Y. Qu, J. J. Liang, P. N. Suganthan, Niching particle swarm optimization with local search for multimodal optimization, Information Sciences 197 (2012) 131–143.

[37] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable test problems for evolutionary multiobjective optimization, in: Evolutionary Multiobjective Optimization, Springer, 2005, pp. 105–145.

[38] S. Huband, P. Hingston, L. Barone, L. While, A review of multiobjective test problems and a scalable test problem toolkit, IEEE Transactions on Evolutionary Computation 10 (5) (2006) 477–506.

[39] C. A. C. Coello, M. R. Sierra, Multiobjective evolutionary algorithms: classifications, analyses, and new innovations, in: Evolutionary Computation, Citeseer, 1999.

[40] R. Sengupta, M. Pal, S. Saha, S. Bandyopadhyay, Population dynamics indicators for evolutionary many-objective optimization, in: Progress in Advanced Computing and Intelligent Engineering, Springer, 2019, pp. 261–271.

[41] S. Bandyopadhyay, S. K. Pal, B. Aruna, Multiobjective GAs, quantitative indices, and pattern classification, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 34 (5) (2004) 2088–2099.

[42] Y. Tian, X. Xiang, X. Zhang, R. Cheng, Y. Jin, Sampling reference points on the Pareto fronts of benchmark multi-objective optimization problems, in: 2018 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2018, pp. 1–6.

[43] Z. He, G. G. Yen, Visualization and performance metric in many-objective optimization, IEEE Transactions on Evolutionary Computation 20 (3) (2015) 386–402.

[44] E. Zitzler, J. Knowles, L. Thiele, Quality assessment of Pareto set approximations, in: Multiobjective Optimization, Springer, 2008, pp. 373–404.

[45] H. Liu, F. Gu, Q. Zhang, Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems, IEEE Transactions on Evolutionary Computation 18 (3) (2014) 450–455.

[46] Y. Sun, G. G. Yen, Z. Yi, IGD indicator-based evolutionary algorithm for many-objective optimization problems, IEEE Transactions on Evolutionary Computation 23 (2) (2018) 173–187.

[47] B. Li, J. Li, K. Tang, X. Yao, Many-objective evolutionary algorithms: A survey, ACM Computing Surveys (CSUR) 48 (1) (2015) 13.

[48] Y. Tian, R. Cheng, X. Zhang, Y. Jin, PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum], IEEE Computational Intelligence Magazine 12 (4) (2017) 73–87.

[49] H. Liu, L. Chen, K. Deb, E. D. Goodman, Investigating the effect of imbalance between convergence and diversity in evolutionary multiobjective algorithms, IEEE Transactions on Evolutionary Computation 21 (3) (2016) 408–425.