



ARTICLE

A Reference Vector-Assisted Many-Objective Optimization Algorithm with Adaptive Niche Dominance Relation

Fangzhen Ge^{1,3}, Yating Wu^{1,*}, Debao Chen^{2,4} and Longfeng Shen^{1,5}

¹School of Computer Science and Technology, Huaibei Normal University, Huaibei, 340604, China

²School of Physic and Electronic Information, Huaibei Normal University, Huaibei, 340604, China

³Anhui Engineering Research Center for Intelligent Computing and Application on Cognitive Behavior (ICACB), Huaibei Normal University, Huaibei, 340604, China

⁴Anhui Province Key Laboratory of Intelligent Computing and Applications, Huaibei Normal University, Huaibei, 340604, China

⁵Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei, 230000, China

*Corresponding Author: Yating Wu. Email: wwwuyating@163.com

Received: 14 June 2023 Accepted: 31 January 2024 Published: 21 May 2024

ABSTRACT

It is still a huge challenge for traditional Pareto-dominated many-objective optimization algorithms to solve many-objective optimization problems because these algorithms hardly maintain the balance between convergence and diversity and can only find a group of solutions focused on a small area on the Pareto front, resulting in poor performance of those algorithms. For this reason, we propose a reference vector-assisted algorithm with an adaptive niche dominance relation, for short MaOEA-AR. The new dominance relation forms a niche based on the angle between candidate solutions. By comparing these solutions, the solution with the best convergence is found to be the non-dominated solution to improve the selection pressure. In reproduction, a mutation strategy of k -bit crossover and hybrid mutation is used to generate high-quality offspring. On 23 test problems with up to 15-objective, we compared the proposed algorithm with five state-of-the-art algorithms. The experimental results verified that the proposed algorithm is competitive.

KEYWORDS

Many-objective optimization; evolutionary algorithm; Pareto dominance; reference vector; adaptive niche

1 Introduction

Many problems in practical applications can be seen as multi-objective optimization problems (MOPs), such as cloud computing [1], industrial copper burdening system [2], and engineering design [3,4]. Many-objective optimization problem (MaOP) [5] is a task that needs to simultaneously optimize m ($m > 3$) conflicting objectives. The mathematical form of MaOP is represented by Eq. (1).

$$\begin{cases} F(x) = (f_1(x), f_2(x), \dots, f_m(x))^T \\ s.t. x \in \Omega \end{cases} \quad (1)$$



where $\Omega \in R^n$ is the feasible search region of the decision vector $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, $f(\mathbf{x}) \in R^m$ is the objective vector, and n is the number of decision variables. Generally, we assume that $f(\mathbf{x})$ is a minimization problem. Due to the duality principle, the maximization problem can be transformed into the minimization problem. Unlike single-objective optimization, due to the conflicting nature of multi-objective functions, multi-objective optimization problems cannot obtain a single optimal solution, but a group of compromise solutions. These compromise solutions form Pareto set (PS) in the decision space and map to the corresponding Pareto front (PF) in the objective space.

Owing to the strong population-based search properties of multi-objective evolutionary algorithms (MOEAs), they can be solved efficiently for MOPs. However, when encountering MaOPs, the performance of MOEA will deteriorate rapidly due to the curse of dimensional [6]. Researchers have developed new algorithms to solve MaOPs. It can be classified into four types: Convergence enhancement, Decomposition-based, Indicator-based and Secondary selection criteria.

The first category includes various convergence enhancement methods. The convergence pressure loss of most traditional MOEAs is the direct result of the inability to distinguish the traditional dominance relation. The most direct convergence enhancement method is to modify the dominance relation to improve the selection pressure. For example, ϵ -dominance [7] compares the dominance relation between solutions using relaxation factors; Yang et al. [8] proposed a grid-based method to choose solutions with a greater dominance priority and control the percentage of Pareto optimal solutions by changing the grid size.

The second category refers to Decomposition-based algorithms that decompose the MOP into multiple single-objective optimization problems. For example, the Multi-objective evolutionary algorithm based on decomposition (MOEA/D) [9] proposed by Zhang and Li decomposes m -objective MOP/MaOP into m single-objective problems for simultaneous optimization. Then, each sub-problem is then handled in a surrounding region that is bound by its associated reference vector. Another decomposition method involves decomposing the complicated MOP/MaOP into multiple simpler MOPs. For example, MOEA/D based on MOP to MOP (MOEA/D-M2M) [10] divides the entire PF into a set of segments, and each of which can be regarded as a separate issue. And many algorithms have been proposed based on the decomposition [11].

The third category is indicator-based methods [12], which proposes a number of MOEAs based on hypervolume (HV). Their main disadvantage is the high computational complexity of HV, especially when solving the MaOP. Sun et al. [13] introduced an Inverse Generation Distance (IGD) indicator in MOEA to solve the MaOP with up to 20-objective. An evolutionary algorithm (IGD+EMOA) based on inverse generation distance plus (IGD+) indicator was proposed [14] to solve the MaOP with up to 8-objective. The IGD+ indicator is a variant of the IGD.

The fourth category is characterized by a combination of Pareto dominance and other selection criteria. These methods choose high ranking non-dominated solutions based on Pareto dominance first, and then use the secondary selection criterion to distinguish non-comparable candidates. Existing methods in this genre mainly employ three typical ideas. The first is to develop new criteria to support methods with good convergence and diversity in non-comparable candidate solutions, such as Promising-region based EMO algorithm (PREA) [15]. The second idea is to use performance indicators to distinguish the quality of non-comparable candidate solutions by selecting those with great contributions, such as the Kriging-assisted Two_Arch2 (KTA-2) [16]. The typical algorithm of the third idea is Many-objective evolutionary algorithm based on dominance and decomposition (MOEA/DD) [17], which combines Pareto dominance with decomposition-based algorithms.

The existing four classes of methods have been able to solve many-objective optimization problems well, but there are still various shortcomings. Decomposition-based algorithms are seriously affected by the weight vector setting, and the computational complexity of indicator-based algorithms is high. Most MaOEAs mainly improve the performance of the algorithm from the perspective of environmental selection, and little attention has been paid to offspring generation. In addition, as far as most dominance relations are concerned, some methods can easily generate the solution set concentrated in a small region of the Pareto front. Considering these issues, in this paper, we propose a reference vector-assisted many-objective optimization algorithm with adaptive niche dominance relations (MaOEA-AR). First using k -bit crossover [18] and hybrid mutation in the mating strategy to generate high-quality offspring. Using a new dominance relation to select non-dominated solutions in environmental selection, and then using angle penalized distance (APD) [19] to further balance convergence and diversity. The contributions of this study are summarized as follows:

1. A new dominance relation, called the adaptive niche dominance relation (A-NDR), is proposed for many-objective optimization. By developing niche techniques based on the angle between candidate solutions and having just one candidate solution with the best convergence in each niche, A-NDR balances convergence and diversity, where convergence is calculated in terms of the PF shape adaptively by selecting the convergence indicator. The proposed A-NDR is parameter-free since the size of the niche is determined adaptively by the distribution of candidate solutions.
2. Mating selection also uses the niche mating method to select neighbor solutions in the objective space, the k -bit crossover to avoid the generation of the identical offspring to the parents and to improve the quality of the parents, and hybrid mutation to produce high-quality offspring.

The rest of this paper is structured as follows. The proposed algorithm is further explained in [Section 2](#). The experimental design and some of the data analysis are presented in [Section 3](#). Finally, [Section 4](#) presents the conclusions and outlook for the future. The related works, motivation, some of the experimental results and analyses of this paper are given in the Supplementary Material.

2 Proposed Algorithm

This section describes the proposed algorithm MaOEA-AR in detail. First, the general framework of the algorithm is given. The reproduction strategy was then discussed. Subsequently, we describe the environment selection, as well as the adaptive niche dominance relation and angle penalty distance used in the environment selection.

2.1 Framework of MaOEA-AR

The details of the general framework of the proposed MaOEA-AR are presented in Algorithm 1. First, initialize (lines 1–3) to generate a set of uniformly distributed reference vectors, initial populations, and train the initial improved growing neural gas (iGNG) net [20] using reference vectors with associated solutions. Copy the non-dominant solution from the initial population into archive A . Then, k -bit crossover and hybrid mutation are used to generate high-quality parent and offspring (lines 5–6, Algorithm 2 and Algorithm 3) individuals and update archive A . The solution in archive A and the solution in the current population are used as input data to train the iGNG network and set the weights of nodes. Finally, in the environment selection (Algorithm 4), N solutions are selected from the joint population of the current and its offspring using adaptive niche dominance relation and angle penalty distance.

Algorithm 1: The framework of MaOEA-AR

```

1: Initialize: reference vectors  $V_0$ , scale=1,  $f_r=0.1$     \\ frequency of reference vector updates
2: net←Initialize iGNG( $V_0, P, g$ )    \\  $g$  is parameters of iGNG
3:  $A$ ←empty set,  $A$ ←Update Archive( $P, A$ )
4: while  $G < G_{max}$  do
5:    $P' = \text{Matingselection}(P)$ 
6:    $Q = \text{Reproduction}(P')$     \\ Algorithm 2 and Algorithm 3
7:    $S = P' \cup Q$ ;
8:    $A = A$ ←Update Archive( $S, A$ );
9:   if  $G < 0.9 * G_{max}$  then
10:    net←TrainingiGNG( $S, A, g$ )
11:     $V$ ←the weights of nodes of net multiply scale
12:  end if
13:  if  $((G/G_{max}) \bmod f_r) == 0$  then
14:    scale = max( $A.objs$ )-min( $A.objs$ )
15:  end if
16:   $P = \text{EnvironmentalSelection}(S, A, V)$ 
17: end while
18: Return  $P$ 

```

2.2 Reproduction

As shown in Algorithm 2, in lines 1–12, the niche mating randomly selects 80% or 20% of the parent individuals from local neighbors or whole populations, which is similar to the traditional setting [21], to avoid reproducing the same offspring as the parents. Niche is a more stable and effect way to exchange evolutionary information between two solutions with similar objective vectors. In line 3, the solution is selected by normalizing the objective values to the same range to ensure fairness. In lines 13–20, k -bit crossover randomly modifies at least one but not more than $(|j| - 1)$ decision variables between one or two parent individuals. Then, we perform mutation operation (Algorithm 3).

Algorithm 2: Reproduce(P)

```

Input: Population  $P$ , Offspring  $Q$ 
1: Set  $|Q|=N$ 
2: set the Neighborhood  $|T|= \max(4, \lceil N * 0.2 \rceil)$ 
3: Normalize( $P$ )
4: Generate a matrix  $M$  of  $N*T$  size that stores neighbor individuals,  $M$  is composed of
 $T$  normalized Euclidean distance nearest solutions in  $P$ 
5: set  $P'=\emptyset$     //  $P'$  is the parent population
6: for  $i=1, \dots, N$  do
7:   if rand < 0.8 then
8:     Randomly select parent individual from  $M$  to join  $P'$ 
9:   else
10:    Randomly select parent individual from  $P$  to join  $P'$ 
11:   end if
12: end for

```

(Continued)

Algorithm 2 (continued)

```

13: Initialize Offspring  $Q = P$ 
14: for  $i=1, \dots, N$  do
15: Find index  $j$  such that  $P(i, j) \neq P'(i, j)$ 
16: if  $|j| > 1$  then
17:    $k = \text{randi}(j, \text{randi}(\{1, \dots, |j|-1\}, 1))$ 
18:   do crossover :  $Q(i, k) = P'(i, k)$ 
19: end if
20: end for
21: do mutation      \\ Algorithm 3

```

Algorithm 3: Mutate(Q)

```

Input: Offspring  $Q$ 
1: for  $i=1, \dots, |Q|$  do
2: if  $\text{rand} < 0.2$  then
3:   find index  $c^1$  such that  $Q(i, c^1) == 1$ 
4:   for  $k=1, \dots, |c^1|$  do
5:     if  $\text{rand} < 1/(1+|c^1|)$  then
6:       do mutation:  $Q(i, c^1(k)) == \tilde{Q}(i, c^1(k))$ 
7:     end if
8:   end for
9:   Find index  $c^0$  such that  $Q(i, c^0) == 0$ 
10:  Repeat the operation from line 4 to line 8 but replace  $c^1$  with  $c^0$ 
11: else
12:   get the decision space dimension  $D = |Q(1)|$ 
13:   for  $i=1, \dots, D$  do
14:     if  $\text{rand} < 1/D$  then
15:       do mutation:  $Q(i, t) == \tilde{Q}(i, t)$ 
16:     end if
17:   end for
18: end if
19: end for

```

Algorithm 3 describes the hybrid mutation method in detail, in which the mutation and the traditional mutation are randomly applied at a set ratio of 20% and 80%, respectively. We have no attempt to find the best ratio, but the probability of using the traditional bit-wise mutation method should be set relatively larger, because it provides a more general and stable variation. It is worth noting that c^1 and c^0 in line 3 and line 9 are not indicator of candidate solutions but indicator of genes with values of 1 or 0 in the decision vector of a solution. Furthermore, the specific mutation in Algorithm 3 is very simple. The index only needs to be flipped, and the specific gene represented by $c^1(k)$ in line 6 or t in line 15 changes from 1 to 0 or 0 to 1.

2.3 Environmental Selection

The environmental selection process is presented in Algorithm 4. The environmental selection consisted of three main parts. The first part combined the populations and archives for selection.

The second is to perform an adaptive niche dominance relation ranking to locate key layers. The third is the APD selection, which is the same as in [19]. Here, we use the adaptive niche dominance relation to identify non-dominated solutions in the population and remove extreme solutions with a large Euclidean distances from the ideal point. The number of solutions in the population after APD selection may be greater than N . Therefore, we will remove the crowded solutions to improve the diversity of the population. Finally, the solutions with the smallest Euclidean distance to their neighbors are removed individually until the number of solutions equals N .

Algorithm 4: Environmental selection(S)

Input: Population P , Archive A , reference vectors V

```

1:  $P = S \cup A$ 
2:  $P = \text{NDSort-A-NDR}(P)$ 
3:  $P = \text{DeleteExtremeSolution}(P)$ 
4:  $P = \text{APD-selection}(P, V)$ 
5: while  $|P| > N$  do
6:    $P = \text{DeleteCrowdedSolutions}(P)$ 
7: end while
8: return  $P$ 

```

2.3.1 Adaptive Niche Dominance Relation

Inspired by the literature [22], we propose an adaptive niche dominance relation (A-NDR) to balance the convergence and diversity of non-dominated solutions using the niche technique in this study. Specifically, in the A-NDR, if and only if

$$\begin{cases} \text{Con}(\mathbf{x}) < \text{Con}(\mathbf{y}), & \alpha_{xy} < \bar{\alpha} \\ \text{Con}(\mathbf{x}) \cdot \frac{\alpha_{xy}}{\bar{\alpha}} < \text{Con}(\mathbf{y}), & \alpha_{xy} > \bar{\alpha} \end{cases} \quad (2)$$

where α_{xy} denotes the acute angle between the objective values of the two candidate solutions, namely,

$$\alpha_{xy} = \arccos(f(\mathbf{x}), f(\mathbf{y})) \quad (3)$$

and $\bar{\alpha}$ is the size of the niche in which each candidate solution is located, which can be adjusted to the size of the niche adaptive, so that the candidate solutions concentrate on non-dominated solutions at a rate of approximately 50%, $\bar{\alpha}$ is set to the $\left\lfloor \frac{|P|}{2} \right\rfloor$ -th smallest element of $\{\min_{i \in P \setminus \{j\}} \alpha_{ij} | i \in P\}$, α_{ij} denotes the acute angle between any pair of candidate solutions. Eq. (2) shows that, each candidate solution \mathbf{x} in the population has a niche with size $\bar{\alpha}$, and the dominance relationship related to the candidate solution \mathbf{x} is determined by mainly considering the candidate solutions in its niche. According to the first formula in (2), if the angle between any \mathbf{x} and a candidate solution \mathbf{y} is smaller than $\bar{\alpha}$, \mathbf{x} is said to dominate \mathbf{y} once the convergence degree of \mathbf{x} is smaller than that of \mathbf{y} . It means that, inside each niche, since there do not exist two non-dominated solutions having a smaller angle than $\bar{\alpha}$, the diversity of the non-dominated solution set is naturally preserved. Because the PF is unknown prior, and the PF shape will seriously affect the performance of the algorithm, the ratio $q = d/\bar{d}$ of the distance from the ideal point to the solution and the distance from the ideal point to the hyperplane is used to determine

divide population P into N sub-populations $P_1, P_2, P_3, \dots, P_N$ associating each individual with its nearest reference vector. As shown in Fig. 3, the angle between an individual and vector v_2 is smaller, associating the individual with v_2 .

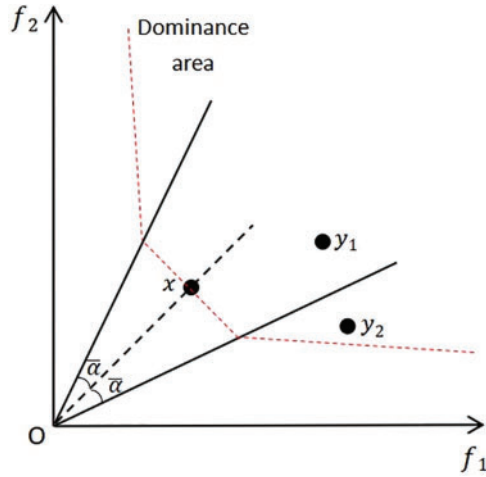


Figure 2: Dominant region of the solution obtained by A-NDR in bi-objective spaces

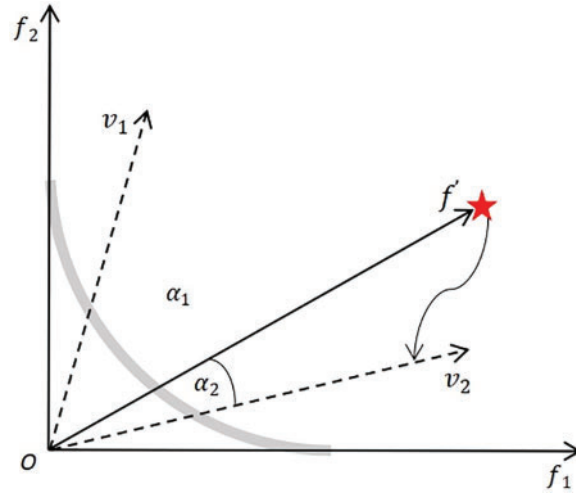


Figure 3: Individual and reference vector association diagram

The relation between the two sub-spaces is measured by the cosine value of the acute angle.

$$\cos \alpha_{i,j} = \frac{f'_i(\mathbf{x}) \cdot v_j}{\|f'_i(\mathbf{x})\| \|v_j\|} \tag{7}$$

where $\cos \alpha_{i,j}$ is the angle between $f'_i(\mathbf{x})$ and v_j . If and only if the angle between them is the smallest among all the reference vectors, that is, when the cosine value is the largest, the individual is assigned to the sub-population where vector v_j is located. After dividing the population, an elite individual from each sub-population is selected to continue participating in the next generation. We used the APD [19] to select elite individuals.

$$d_{i,j} = (1 + P(\alpha_{i,j})) \cdot \|f'_i(\mathbf{x})\| \quad (8)$$

where $P(\alpha_{i,j})$ is the $\alpha_{i,j}$ correlated penalty function.

$$P(\alpha_{i,j}) = m \cdot \left(\frac{t}{t_{max}} \right)^\theta \cdot \frac{\alpha_{i,j}}{\varphi_{v_j}} \quad (9)$$

and

$$\varphi_{v_j} = \min_{i=\{1,\dots,N\}, i \neq j} < v_i, v_j > \quad (10)$$

where t_{max} denotes the predefined maximum number of iterations. φ_{v_j} is the minimum angle value between the reference vector v_i and other vectors in the current generation. θ is a user-defined control $P(\alpha_{i,j})$ parameter. Consistent with the setting in [19], $\theta = 2$. In the early stage of evolution, $P(\alpha_{i,j})$ is close to 0, and convergence is emphasized, whereas in the late stage of evolution, the effect of $P(\alpha_{i,j})$ on $d_{i,j}$ gradually increases, and the distribution of the solution is emphasized.

2.4 Computational Complexity

MaOEA-AR consists of three main components: (1) Reproduction (k -bit crossover), (2) Hybrid mutation, and (3) Environmental selection. Details of the complexity analysis of these three components are given below. The complexity of reproduction (Algorithm 3) is concentrated in lines 6–20 of Algorithm 2, which has a complexity of $O(N)$, since each execution requires the selection of an excellent parent individual from a population of N , where N is the size of the population. Hybrid mutation requires the selection of a proportion of parents from local neighbour or from the whole population to be mutated, and therefore requires $O(|Q| * |c^1| * D)$, where $|Q|$ is the size of the offspring, $|c^1|$ is the gene that has been mutated, and D is the size of the offspring that has been mutated. The complexity of the environment selection depends on both A-NDR (line 2 of Algorithm 4) and APD (lines 3–7 of Algorithm 4). The complexity of A-NDR is determined by the PF estimation, which has a computational complexity of $O(mN^2)$, where m is the number of objectives. APD hold a computational complexity of $O(mN^2)$ in the worst case. Therefore, the complexity of environment selection is $O(mN^2)$. In summary, the overall computational complexity of MaOEA-AR is $O(mN^2)$.

3 Experimental Studies

In this section, we experiments on 23 benchmark test problems from three widely used test suites (i.e., Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler (DTLZ) [25], MaF [26] and Benchmark MOP proposed by Walking Fish Group (WFG) [27]) to compare MaOEA-AR with five state-of-the-art algorithms (i.e., Many-objective evolutionary algorithms based on an independent two-stage (MaOEA-IT) [28], IGD based many-objective evolutionary algorithm (MaOEA-IGD) [13], Many-objective evolutionary algorithm based on objective space reduction (MaOEA-RD) [29], MOEA/D with update when required (MOEA/D-UR) [30] and MOEA/D [9]). We chose the commonly used comprehensive evaluation indicators HV [31] and IGD [32]. The experiments were conducted on Evolutionary multi-objective optimization platform (PlatEMO) [33], and the comparison algorithm was obtained from the platform.

3.1 Benchmark Test Problem

On the 23 benchmark problems of DTLZ, MaF and WFG test suite, we compared the proposed algorithm with the five state-of-the-art algorithms. All problems in the DTLZ and WFG test suites can have any set of decision variable dimensions and objective numbers. MaF was proposed by Cheng et al. [26] in 2017 and includes various representative problems in the real-world. Its test suite contains 15 multi-objective optimization problems chosen by MaF 1–7 to evaluate the algorithm's performance in solving complex problems. Some properties of these benchmark function suites are listed in Table 1.

Table 1: The properties of test instances of DTLZ, MaF and WFG problem suite

Problem	No. of decision variables (D)	Properties
DTLZ 1	$m - 1 + 5$	Linear
DTLZ 2–4	$m - 1 + 10$	Concave
DTLZ 5–6		Degenerated
DTLZ 7	$m - 1 + 20$	Discontinuous
MaF1	$m - 1 + 10$	Linear
MaF2		Concave
MaF3–4		Convex, multi-modal
MaF5		Convex, biased
MaF6		Concave, degenerate
MaF7	$m - 1 + 20$	Mixed, disconnected, multi-modal
WFG 1	$m - 1 + 10$	Mixed
WFG 2		Convex, disconnected, non-separable
WFG 3		Linear, partially degenerate
WFG 4–9		Concave

3.2 Performance Indicator

With the continuous proposal of MaOEAs, evaluating the performance of MaOEA has also become an important research direction. When the algorithms solve MaOPs, it is necessary to quantify the performance of the algorithm through performance indicators, and to quantify the performance of one algorithm in comparison with the performance of other algorithms. The comprehensive indicators HV and IGD are the most often utilized indicators. Based on the magnitude of the range covered by the PS computed by the algorithm in the objective space, HV is used to evaluate convergence and diversity. By evaluating the similarity between the true PF and the PF generated by the algorithm, the IGD evaluates convergence and diversity. In general, the algorithm performs better the lower the IGD value, whereas the opposite is true for HV. In this study, we use these two indicators to evaluate the performance of the proposed MaOEA-AR, as shown in Eqs. (11) and (12).

$$HV = volume(\cup_{i=1}^{|PF|} v_i) \quad (11)$$

$$IGD(PF^*, PF) = \frac{\sum_{i \in PF^*} d(i, PF)}{|PF^*|} \quad (12)$$

where PF is obtained by algorithm, PF^* is the standard Pareto front, v_i is the hypervolume formed by the reference points and the individuals, $d(i, PF)$ is the Euclidean distance between the individual i on the PF and the nearest individual i , and $|PF^*|$ denotes the total number of points on the true Pareto front.

3.3 Experiment Setting

Population size N : In algorithms for reference vector or point generation, a two-layer strategy is usually used to generate uniformly distributed reference points N to avoid generating too many reference points, which is determined by $N = C_{h_1+m-1}^{m-1} + C_{h_2+m-1}^{m-1}$, where h_1 and h_2 are the partition parameters of the inner and outer layers, respectively. As in [33], we set the population size equal to the number N of reference points. In this study, we set the objectives as $m = 5, 8, 10$ and 15 . Table 2 provides the details of the corresponding population size and partition parameters.

Table 2: Population size setting

m	Parameter (h_1, h_2)	N
5	(6, 0)	210
8	(3, 2)	156
10	(3, 2)	275
15	(2, 2)	135

Numbers of run and stop conditions: To obtain reliable experimental results, we ran each algorithm 20 times independently on each test instance. This end when the maximum number of function evaluations (maxFEs) is reached. To ensure that each algorithm converges, we uniformly set the maxFEs at 100,000 for each test instance.

Operator parameter settings: The crossover probability p_c and mutation probability p_m are set to 1.0 and $1/n$, respectively, and n is the number of decision variables. The distribution indicator of crossover operator η_c and mutation operator η_m is set to 20. To achieve the best performance of the comparison algorithm, we set the crossover probability p_c of MaOEA-IGD to 0.9.

Two parameters in MaOEA-AR require to be predefined, i.e., the user-defined parameter θ that controls the rate of change of the penalty function in Eq. (9), and the parameters f_r controlling the frequency of reference vector updates in Algorithm 1. In this paper, $\theta = 2$ makes APD work well on a variety of problems with different numbers of objectives without changing the settings. $f_r = 0.1$ is used to deal with different target scales, and we adjust the reference vector after $f_r * G_{max}$ each generation. Detailed parametric sensitivity analyses for θ and f_r are provided in Supplementary Materials III of [20].

3.4 Experimental Results Analysis

In this section, we analyze and evaluate the MaOEA-AR based on the statistical results of 20 independent runs of each algorithm on 16 benchmark problems (92 test instances). Wilcoxon rank sum test is used to compare the difference between the average value obtained by MaOEA-AR and the other five algorithms. The significance level was 0.05, and the '+', '-' and '=' in the table indicate that the performance of other algorithms is significantly better, significantly lower, and no different than MaOEA-AR. Table 3 exhibits the performance of MaOEA-AR on 23 benchmark problems with

five comparison algorithms, and the Wilcoxon rank-sum test of MaOEA-AR on the test instances is shown in Fig. 4. It can be seen that MaOEA-AR has strong competitiveness for both HV and IGD.

Table 3: The proportion of the 23 test problems (92 test instances) where MaOEA-AR is better than, worse than, or equal to each algorithm on HV and IGD indicators, respectively

MaOEA-AR vs.	MOEA/D-UR	MaOEA-IT	MaOEA/IGD	MaOEA-RD	MOEA/D	
HV	+	14/92	0/92	6/92	3/92	13/92
	−	63/92	88/92	79/92	84/92	75/92
	=	13/92	4/92	7/92	5/92	4/92
IGD	+	9/92	0/92	3/92	2/92	15/92
	−	79/92	91/92	84/92	88/92	71/92
	=	4/92	1/92	5/92	2/92	6/92

In this paper we conducted 5 groups of experiments to verify the performance of the algorithm and the effectiveness of each strategy. We conducted experiments on all three benchmark test suites, DTLZ, MaF and WFG. The results and analyses of the first group of DTLZ benchmark experiments and the second group of MaF benchmark experiments are given in the Supplementary Material, and the results and analyses of the third group of WFG benchmark experiments are given below. The 4th group of experiments are ablation experiments to validate the performance of the A-NDR and k -bit crossover strategies and the analysis of their results are also given in the Supplementary Material. The fifth group of experiments was designed to validate the algorithm’s ability to solve real-world problems, and in the following we give the results and analyses of the experiments that dealt with three real-world problems.

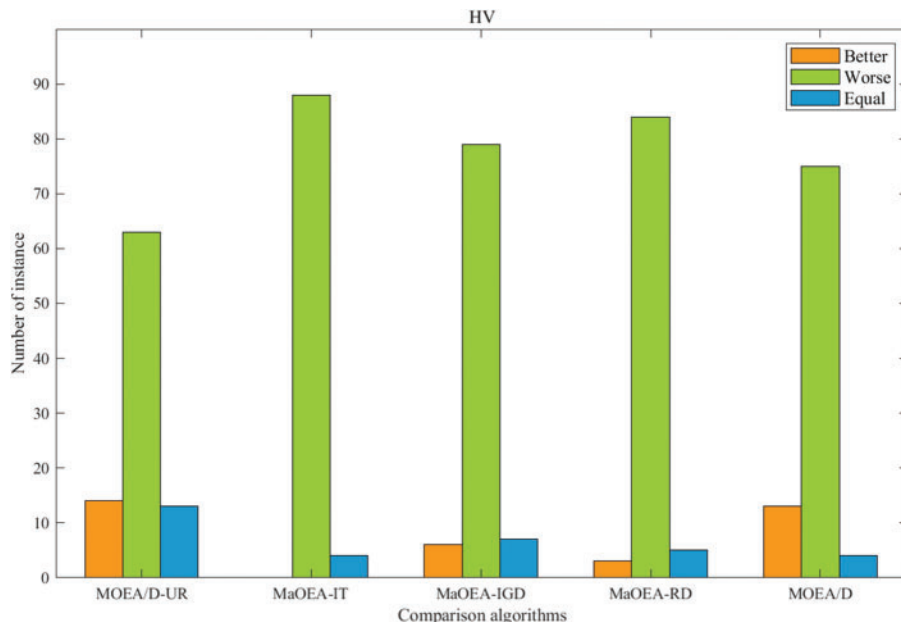


Figure 4: (Continued)

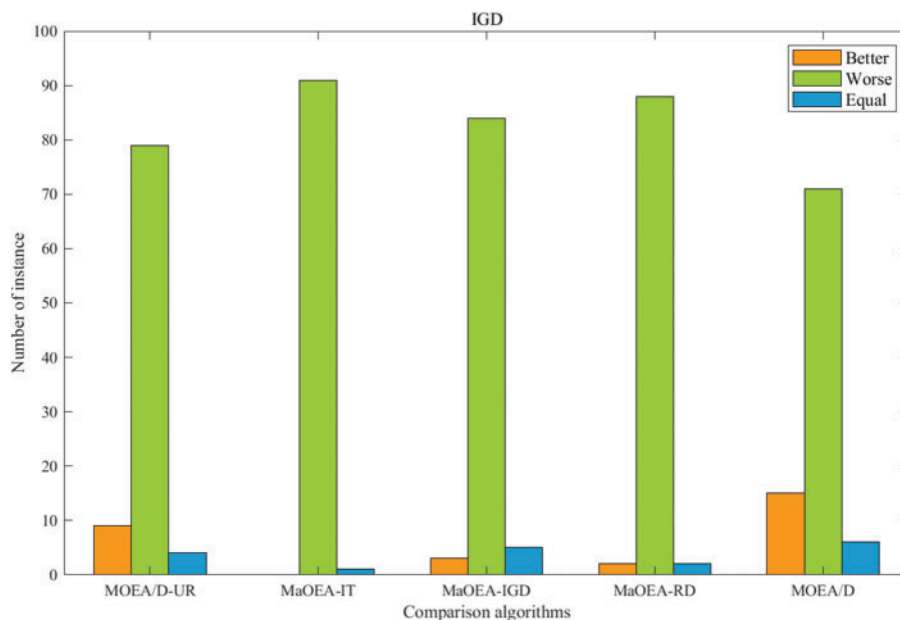


Figure 4: Wilcoxon rank sum test of MaOEA-AR on test instances in terms of HV and IGD

3.4.1 Performance on WFG

The experimental statistics of the six algorithms on the WFG test problem are given in [Tables 4](#) and [5](#). MaOEA-AR has performed very competitively on most problems. In [Table 4](#), we can see that WFG2 is a difficult problem, where PF is disconnected. Similar to other algorithms, MaOEA-AR does not handle this problem well, but MOEA/D-UR can solve the WFG2 problem well and obtain the best performance on its objectives in any dimension. This is because the decomposition-based algorithm predefines a set of well-distributed weight vectors that improve the diversity of the solutions obtained, in addition, the algorithm uses a specific mechanism to maintain a good balance between diversity and convergence. MaOEA-AR obtained 23 optimal values on 36 test problems, and optimal values on any dimension of the objective in WFG4, WFG5, WFG7 and WFG9. MOEA/D-UR followed closely behind, obtaining 11 optimal values. Only two optimal values were obtained for MaOEA-IGD. MaOEA-IT, MaOEA-RD and MOEA/D did not obtain optimal values. There are several reasons for this phenomenon: First, at mating selection, k -bit crossover was used to select excellent paternal individuals and a hybrid mutation strategy was used to generate high-quality offspring individuals; Secondly, the proposed A-NDR improves the selection pressure of the algorithm and ensures that the population can converge. [Table 5](#) lists the IGD values for the six algorithms on the WFG problem. MaOEA-AR can handle all WFG problems except WFG3 very well. WFG3 is a difficult problem in which the PF is degenerate and the decision variables are non-separable. Similar to other algorithms, MaOEA-AR does not handle this problem well, but MOEA/D-UR can solve the WFG3 problem well and obtain the best performance on its objectives in any dimension. To better observe the performance of the six algorithms on the WFG problem, we chose to plot the PF obtained by each algorithm on a WFG9 of 15-objective. The PF of WFG9 is biased, multi-modal, deceptive and in-separable. It can be seen that MaOEA-AR, MOEA/D-UR and MaOEA-IGD have similar distributions, but MaOEA-AR has better performance. It not only converges to the true PF, but covers the true PF more extensively, indicating that it has better diversity. The second best is MOEA/D-UR, which also converges well

to true PF, but does not cover 1-3-objective well and does not fully converge on 13-objective. This is followed by MaOEA-IGD, which also converges well to true PF but does not have good coverage on 5-15-objective. However, the PFs obtained by MaOEA-IT, MaOEA-RD and MOEA/D hardly cover the true PF. In addition, in Figs. 4 and 5, we can see that the visualization in the high-dimensional objective space is very poor, so we show in Fig. 6 the performance of each algorithm on the 3-objective problem, where the more complex WFG9 is still chosen for the test problem. Compared to other algorithms, the solutions obtained by MaOEA-AR in the figure cover the entire PF with a wide and uniform coverage. MOEA/D-UR, MaOEA-RD and MOEA/D can also cover the entire PF, but not uniformly. The solutions of MOEA/-UR and MOEA/D are concentrated around the second objective. The MaOEA-IT solution, on the other hand, is concentrated in the middle region and is less diverse. The solutions of MaOEA-IGD are distributed at the vertices of the three objectives.

Table 4: Comparison of HV among the 6 algorithms on benchmarks WFG with 5-, 8-, 10-and 15-objective

Problem	m	MOEA/D-UR	MaOEA-IT	MaOEA-IGD	MaOEA-RD	MOEA/D	MaOEA-AR
WFG1	5	9.9575e - 1 (6.21e - 4) +	2.2125e - 1 (3.76e - 2) -	3.9377e - 1 (1.79e - 1) -	7.2313e - 1 (3.21e - 1) -	9.5018e - 1 (1.64e - 2) -	9.8928e - 1 (5.00e - 3)
	8	9.9949e - 1 (1.62e - 4) +	1.7337e - 1 (4.35e - 2) -	3.0366e - 1 (6.51e - 2) -	8.0185e - 1 (1.52e - 1) -	9.2923e - 1 (3.71e - 3) -	9.8341e - 1 (2.81e - 2)
	10	9.5649e - 1 (4.72e - 2) -	6.2668e - 2 (4.24e - 2) -	6.9708e - 1 (2.20e - 1) -	8.0610e - 1 (1.27e - 1) -	9.3081e - 1 (1.38e - 2) -	9.7236e - 1 (4.06e - 2)
	15	9.9981e - 1 (1.46e - 4) +	5.1150e - 2 (2.91e - 2) -	5.7730e - 1 (2.61e - 1) -	5.4433e - 1 (2.40e - 1) -	8.5140e - 1 (7.83e - 2) -	9.9333e - 1 (1.31e - 3)
WFG2	5	9.9442e - 1 (7.10e - 4) +	6.0826e - 1 (4.39e - 2) -	9.1639e - 1 (3.80e - 2) -	9.7174e - 1 (3.29e - 3) -	9.5935e - 1 (5.26e - 3) -	9.7691e - 1 (3.49e - 3)
	8	9.9685e - 1 (9.18e - 4) +	5.9328e - 1 (2.94e - 2) -	9.5300e - 1 (6.49e - 2) =	9.4339e - 1 (3.02e - 2) -	9.2554e - 1 (8.70e - 3) -	9.8132e - 1 (3.67e - 3)
	10	9.9519e - 1 (1.04e - 3) +	6.0632e - 1 (2.38e - 2) -	9.6412e - 1 (5.38e - 2) =	9.6121e - 1 (2.37e - 2) -	9.2660e - 1 (4.74e - 3) -	9.8484e - 1 (3.07e - 3)
	15	9.9389e - 1 (2.52e - 3) +	5.4980e - 1 (3.56e - 2) -	9.0111e - 1 (9.59e - 2) -	9.6330e - 1 (1.42e - 2) -	9.1806e - 1 (8.76e - 3) -	9.8165e - 1 (2.99e - 3)
WFG3	5	2.7544e - 1 (1.48e - 3) +	1.0231e - 3 (2.26e - 3) -	8.6503e - 2 (2.71e - 3) -	1.0523e - 3 (4.71e - 3) -	3.9355e - 2 (3.38e - 2) -	1.0369e - 1 (1.66e - 2)
	8	1.4432e - 1 (2.39e - 2) +	0.0000e + 0 (0.00e + 0) =	1.8237e - 2 (2.17e - 2) +	0.0000e + 0 (0.00e + 0) =	0.0000e + 0 (0.00e + 0) =	0.0000e + 0 (0.00e + 0)
	10	0.0000e + 0 (0.00e + 0) =	0.0000e + 0 (0.00e + 0) =	2.6703e - 3 (6.76e - 3) +	0.0000e + 0 (0.00e + 0) =	0.0000e + 0 (0.00e + 0) =	0.0000e + 0 (0.00e + 0)
	15	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	7.8587e - 1 (2.22e - 3) -	3.1170e - 1 (9.79e - 2) -	1.1122e - 1 (3.62e - 2) -	6.3024e - 1 (9.76e - 3) -	7.1287e - 1 (1.52e - 2) -	7.9465e - 1 (5.97e - 3)
	8	8.8751e - 1 (3.42e - 3) -	2.9207e - 1 (9.84e - 2) -	9.9161e - 2 (2.54e - 2) -	5.9808e - 1 (8.66e - 2) -	4.3386e - 1 (6.06e - 2) -	8.7816e - 1 (4.02e - 2)
	10	9.0513e - 1 (5.52e - 3) -	3.6107e - 1 (7.60e - 2) -	1.0702e - 1 (3.32e - 2) -	6.4784e - 1 (9.34e - 2) -	4.0437e - 1 (4.84e - 2) -	8.9393e - 1 (4.49e - 2)
	15	9.2800e - 1 (1.41e - 2) -	3.3088e - 1 (2.43e - 2) -	1.1449e - 1 (3.75e - 2) -	6.2322e - 1 (7.55e - 2) -	3.1323e - 1 (4.93e - 2) -	9.1531e - 1 (4.76e - 2)

(Continued)

Table 4 (continued)

Problem	m	MOEA/D-UR	MaOEA-IT	MaOEA-IGD	MaOEA-RD	MOEA/D	MaOEA-AR
WFG5	5	7.2209e - 1 (6.97e - 3) -	2.8073e - 1 (3.29e - 2) -	5.7233e - 1 (1.45e - 1) -	6.0235e - 1 (1.14e - 2) -	6.7984e - 1 (1.52e - 2) -	7.5305e - 1 (4.98e - 3)
	8	8.0994e - 1 (5.24e - 3) -	2.7669e - 1 (1.78e - 2) -	1.4958e - 1 (1.82e - 1) -	4.9520e - 1 (4.23e - 2) -	5.1488e - 1 (1.99e - 2) -	8.3679e - 1 (3.55e - 2)
	10	8.5740e - 1 (3.38e - 3) -	3.0322e - 1 (2.30e - 2) -	4.9755e - 1 (3.12e - 1) -	5.3218e - 1 (6.75e - 2) -	4.8934e - 1 (2.25e - 2) -	8.5338e - 1 (5.13e - 2)
	15	8.5768e - 1 (1.43e - 2) -	2.4389e - 1 (1.14e - 2) -	1.2827e - 1 (1.54e - 1) -	4.4021e - 1 (1.05e - 1) -	3.3363e - 1 (4.10e - 2) -	8.4463e - 1 (6.48e - 2)
WFG6	5	7.2182e - 1 (2.27e - 2) -	2.7278e - 1 (1.62e - 2) -	2.2582e - 1 (8.89e - 2) -	5.5737e - 1 (1.70e - 2) -	5.9400e - 1 (2.31e - 2) -	7.3501e - 1 (2.19e - 2)
	8	7.8989e - 1 (1.47e - 2) -	2.6375e - 1 (3.76e - 2) -	2.8991e - 1 (1.99e - 1) -	4.7584e - 1 (8.99e - 2) -	2.8591e - 1 (2.47e - 2) -	8.1388e - 1 (3.55e - 2)
	10	8.4320e - 1 (2.50e - 2) -	3.0503e - 1 (2.93e - 2) -	4.0493e - 1 (1.74e - 1) -	5.0365e - 1 (1.11e - 1) -	2.4939e - 1 (3.88e - 2) -	8.3615e - 1 (5.05e - 2)
	15	8.3841e - 1 (3.16e - 2) =	2.5900e - 1 (4.71e - 2) -	2.9965e - 1 (2.17e - 1) -	5.4092e - 1 (1.24e - 1) -	1.1966e - 1 (4.34e - 2) -	8.1695e - 1 (5.52e - 2)
WFG7	5	7.8663e - 1 (2.44e - 3) -	3.3479e - 1 (2.11e - 2) -	2.5433e - 1 (4.56e - 2) -	5.8840e - 1 (2.03e - 2) -	6.5768e - 1 (9.93e - 3) -	8.0535e - 1 (3.60e - 3)
	8	8.8624e - 1 (3.30e - 3) -	3.4025e - 1 (2.46e - 2) -	2.2213e - 1 (9.60e - 2) -	4.2196e - 1 (5.10e - 2) -	3.6718e - 1 (3.21e - 2) -	9.0299e - 1 (2.83e - 2)
	10	9.3840e - 1 (1.56e - 3) -	3.6967e - 1 (3.25e - 2) -	2.7484e - 1 (1.65e - 1) -	3.9729e - 1 (3.60e - 2) -	3.4512e - 1 (1.15e - 2) -	9.3274e - 1 (2.78e - 2)
	15	9.2500e - 1 (2.16e - 2) -	3.2437e - 1 (3.67e - 2) -	1.9348e - 1 (6.22e - 2) -	2.3127e - 1 (4.87e - 2) -	1.6982e - 1 (4.30e - 2) -	9.5148e - 1 (5.81e - 3)
WFG8	5	6.9200e - 1 (4.32e - 3) -	2.6905e - 1 (3.37e - 2) -	4.9468e - 2 (5.37e - 2) -	4.9634e - 1 (1.86e - 2) -	5.5004e - 1 (8.11e - 2) -	6.9491e - 1 (3.30e - 3)
	8	7.9123e - 1 (5.92e - 3) -	2.8066e - 1 (4.03e - 2) -	1.5336e - 1 (6.43e - 2) -	4.1211e - 1 (1.18e - 1) -	9.5124e - 2 (6.88e - 2) -	8.2399e - 1 (2.63e - 2)
	10	8.4484e - 1 (3.50e - 3) =	3.2251e - 1 (3.28e - 2) -	2.5333e - 1 (6.14e - 2) -	4.7094e - 1 (1.07e - 1) -	4.8696e - 2 (2.97e - 2) -	8.4460e - 1 (2.09e - 2)
	15	8.7969e - 1 (1.27e - 2) -	2.7452e - 1 (4.48e - 2) -	1.7006e - 1 (6.86e - 2) -	4.6699e - 1 (9.42e - 2) -	2.4026e - 1 (3.89e - 1) -	9.0415e - 1 (5.08e - 3)
WFG9	5	7.3609e - 1 (7.77e - 3) -	2.7612e - 1 (1.78e - 2) -	2.2369e - 1 (1.11e - 1) -	5.7629e - 1 (1.67e - 2) -	6.0040e - 1 (4.04e - 2) -	7.6473e - 1 (5.97e - 3)
	8	7.9231e - 1 (2.54e - 2) -	2.8491e - 1 (2.84e - 2) -	2.1298e - 1 (1.76e - 1) -	4.1666e - 1 (5.56e - 2) -	3.8567e - 1 (9.80e - 2) -	8.3234e - 1 (3.37e - 2)
	10	8.1327e - 1 (3.16e - 2) -	3.2807e - 1 (1.77e - 2) -	6.8156e - 1 (1.33e - 1) -	4.4150e - 1 (4.36e - 2) -	2.8962e - 1 (1.04e - 1) -	8.5383e - 1 (3.77e - 2)
	15	7.1961e - 1 (5.88e - 2) -	2.6957e - 1 (4.15e - 2) -	2.8284e - 1 (1.96e - 1) -	3.9297e - 1 (8.78e - 2) -	2.2646e - 1 (1.64e - 1) -	8.3285e - 1 (6.85e - 2)
+/-/=		9/23/4	0/33/3	2/31/3	0/33/3	0/33/3	

Table 5: Comparison of IGD among the 6 algorithms on benchmarks WFG with 5-, 8-, 10-and 15-objective

Problem	m	MOEA/D-UR	MaOEA-IT	MaOEA-IGD	MaOEA-RD	MOEA/D	MaOEA-AR
WFG1	5	5.4290e - 1 (3.17e - 2) -	3.0467e + 0 (1.15e + 0) -	3.9185e + 0 (1.12e + 0) -	1.7570e + 0 (1.97e + 0) -	7.8278e - 1 (3.71e - 2) -	3.7166e - 1 (6.79e - 3)
	8	1.3021e + 0 (8.16e - 2) -	2.5997e + 0 (4.41e - 1) -	7.7019e + 0 (2.11e + 0) -	1.5870e + 0 (1.27e + 0) -	1.5548e + 0 (2.35e - 2) -	8.4279e - 1 (2.02e - 2)
	10	1.5657e + 0 (1.03e - 1) -	3.1274e + 0 (2.61e - 1) -	6.2986e + 0 (3.04e + 0) -	1.8156e + 0 (2.42e + 0) -	1.7669e + 0 (7.17e - 2) -	9.9753e - 1 (4.24e - 2)
	15	2.2538e + 0 (7.15e - 2) -	4.3125e + 0 (2.06e - 1) -	1.1042e + 1 (7.38e + 0) -	1.3271e + 1 (8.46e + 0) -	2.3720e + 0 (8.18e - 2) -	1.8432e + 0 (9.75e - 2)
WFG2	5	4.0971e - 1 (8.17e - 3) -	1.7045e + 0 (3.29e - 1) -	1.6698e + 0 (2.33e - 1) -	5.0759e - 1 (2.48e - 2) -	7.5333e - 1 (4.34e - 2) -	3.9184e - 1 (8.43e - 3)
	8	1.0885e + 0 (4.18e - 2) -	2.9610e + 0 (5.77e - 1) -	2.1500e + 0 (4.22e - 1) -	1.2332e + 0 (1.50e - 1) -	1.7525e + 0 (3.12e - 2) -	9.6310e - 1 (2.67e - 2)
	10	1.1756e + 0 (4.57e - 2) -	3.7373e + 0 (8.13e - 1) -	2.3573e + 0 (5.70e - 1) -	1.4690e + 0 (2.52e - 1) -	1.9250e + 0 (2.43e - 2) -	1.0498e + 0 (6.29e - 2)
	15	1.8806e + 0 (9.14e - 2) =	6.5452e + 0 (1.50e + 0) -	4.3556e + 0 (1.61e + 0) -	3.1344e + 0 (1.00e + 0) -	2.5717e + 0 (2.88e - 2) -	1.0498e + 0 (6.29e - 2)
WFG3	5	1.0064e - 1 (1.56e - 2) +	1.2315e + 0 (8.98e - 1) -	5.4317e + 0 (1.70e - 2) -	1.3231e + 0 (2.43e - 1) -	7.9541e - 1 (2.08e - 1) -	3.2237e - 1 (2.03e - 2)
	8	3.3908e - 1 (4.20e - 2) +	2.1761e + 0 (1.70e + 0) -	8.8735e + 0 (4.21e - 2) -	2.7944e + 0 (6.20e - 1) -	3.6325e + 0 (1.69e - 1) -	7.0462e - 1 (7.20e - 2)
	10	4.0919e - 1 (5.75e - 2) +	1.7536e + 0 (1.40e + 0) -	8.8258e + 0 (4.20e + 0) -	3.0878e + 0 (6.07e - 1) -	5.4566e + 0 (4.16e - 1) -	8.7602e - 1 (7.88e - 2)
	15	1.0184e + 0 (1.95e - 1) +	2.9789e + 0 (2.55e + 0) -	9.5036e + 0 (5.60e + 0) -	5.2930e + 0 (1.92e + 0) -	9.3672e + 0 (3.30e - 1) -	1.4491e + 0 (1.49e - 1)
WFG4	5	9.6462e - 1 (9.36e - 3) -	3.0543e + 0 (1.87e + 0) -	6.4133e + 0 (7.33e - 1) -	1.0486e + 0 (2.50e - 2) -	1.5073e + 0 (7.49e - 2) -	9.4532e - 1 (7.16e - 3)
	8	3.1068e + 0 (2.17e - 2) -	7.9920e + 0 (2.47e + 0) -	9.8189e + 0 (2.58e - 1) -	3.5842e + 0 (3.39e - 1) -	6.7958e + 0 (2.20e - 1) -	3.0012e + 0 (4.72e - 2)
	10	4.4842e + 0 (3.68e - 2) -	1.0160e + 1 (2.29e + 0) -	1.1967e + 1 (2.68e - 1) -	5.0839e + 0 (3.66e - 1) -	9.3184e + 0 (2.07e - 1) -	4.0527e + 0 (3.51e - 2)
	15	9.8447e + 0 (4.32e - 1) -	1.9332e + 1 (1.38e + 0) -	2.4229e + 1 (4.32e + 0) -	1.2091e + 1 (9.35e - 1) -	1.5654e + 1 (2.46e - 1) -	8.6123e + 0 (4.17e - 1)
WFG5	5	9.9812e - 1 (2.80e - 2) -	2.0958e + 0 (4.77e - 1) -	1.9308e + 0 (1.20e + 0) -	1.0421e + 0 (1.69e - 2) -	1.4067e + 0 (7.76e - 2) -	9.4085e - 1 (1.01e - 2)
	8	3.2914e + 0 (4.57e - 2) -	5.3698e + 0 (3.37e - 1) -	1.2924e + 1 (3.45e + 0) -	3.4965e + 0 (1.79e - 1) -	6.3469e + 0 (1.61e - 1) -	3.0629e + 0 (4.18e - 2)
	10	4.3816e + 0 (4.58e - 2) -	7.3146e + 0 (3.87e - 1) -	9.6442e + 0 (6.95e + 0) -	4.6754e + 0 (2.08e - 1) -	8.9045e + 0 (2.08e - 1) -	4.0837e + 0 (3.20e - 2)
	15	9.4135e + 0 (2.10e - 1) -	1.5798e + 1 (5.69e - 1) -	2.7536e + 1 (5.04e + 0) -	1.2251e + 1 (1.30e + 0) -	1.5250e + 1 (1.48e - 1) -	8.1326e + 0 (9.77e - 2)
WFG6	5	9.8356e - 1 (3.19e - 2) -	2.0348e + 0 (2.38e - 1) -	4.5005e + 0 (1.13e + 0) -	1.1234e + 0 (2.84e - 2) -	1.6232e + 0 (3.60e - 2) -	9.5041e - 1 (4.26e - 3)
	8	3.3591e + 0 (4.74e - 2) -	6.0213e + 0 (8.98e - 1) -	8.9200e + 0 (4.50e + 0) -	3.4966e + 0 (1.41e - 1) -	6.9480e + 0 (5.22e - 2) -	3.1093e + 0 (5.07e - 2)

(Continued)

Table 5 (continued)

Problem	m	MOEA/D-UR	MaOEA-IT	MaOEA-IGD	MaOEA-RD	MOEA/D	MaOEA-AR
	10	4.4393e + 0 (2.78e - 2) -	7.7229e + 0 (8.01e - 1) -	8.7875e + 0 (3.96e + 0) -	4.7470e + 0 (2.48e - 1) -	9.3557e + 0 (1.66e - 1) -	4.1580e + 0 (4.11e - 2)
	15	9.5485e + 0 (2.67e - 1) -	1.6721e + 1 (1.64e + 0) -	1.9878e + 1 (8.50e + 0) -	1.1832e + 1 (1.65e + 0) -	1.6090e + 1 (2.42e - 1) -	8.1979e + 0 (1.04e - 1)
WFG7	5	9.6180e - 1 (8.50e - 3) -	2.1187e + 0 (2.97e - 1) -	5.2307e + 0 (5.92e - 1) -	1.1184e + 0 (3.45e - 2) -	1.6993e + 0 (2.93e - 2) -	9.4795e - 1 (7.05e - 3)
	8	3.1382e + 0 (2.38e - 2) -	6.0447e + 0 (5.53e - 1) -	1.0338e + 1 (2.49e + 0) -	3.7754e + 0 (1.88e - 1) -	7.0408e + 0 (5.32e - 2) -	3.0642e + 0 (5.43e - 2)
	10	4.5327e + 0 (4.23e - 2) -	8.2991e + 0 (7.62e - 1) -	1.2402e + 1 (2.96e + 0) -	4.9560e + 0 (1.98e - 1) -	9.3218e + 0 (1.62e - 1) -	4.0662e + 0 (6.29e - 2)
	15	9.5233e + 0 (4.63e - 1) -	1.7307e + 1 (1.21e + 0) -	2.3616e + 1 (2.73e + 0) -	1.3851e + 1 (8.08e - 1) -	1.6320e + 1 (2.26e - 1) -	8.4412e + 0 (2.47e - 1)
WFG8	5	1.0087e + 0 (7.89e - 3) -	2.3941e + 0 (5.98e - 1) -	4.6593e + 0 (3.11e - 1) -	1.2563e + 0 (2.32e - 2) -	1.4120e + 0 (1.85e - 1) -	9.7538e - 1 (7.34e - 3)
	8	3.1694e + 0 (5.34e - 2) -	6.0819e + 0 (7.25e - 1) -	1.1257e + 1 (1.41e + 0) -	4.0211e + 0 (1.58e - 1) -	6.3055e + 0 (4.07e - 1) -	3.0250e+0 (6.09e - 2)
	10	4.4226e + 0 (1.19e - 1) -	8.0757e + 0 (6.15e - 1) -	1.2553e + 1 (1.13e + 0) -	5.5141e + 0 (3.43e - 1) -	8.6403e + 0 (1.34e - 1) -	4.0086e + 0 (6.20e - 2)
	15	1.0290e + 1 (6.39e - 1) -	1.6817e + 1 (1.26e + 0) -	2.5278e + 1 (3.65e + 0) -	1.3569e + 1 (1.14e + 0) -	1.3566e + 1 (2.67e + 0) -	9.2683e + 0 (7.18e - 1)
WFG9	5	9.8008e - 1 (2.98e - 2) -	2.1194e + 0 (1.30e - 1) -	4.9295e + 0 (1.21e + 0) -	1.0897e + 0 (3.25e - 2) -	1.4739e + 0 (8.07e - 2) -	9.1971e - 1 (5.26e - 3)
	8	3.2283e + 0 (5.15e - 2) -	5.8094e + 0 (3.70e - 1) -	1.1085e + 1 (3.54e + 0) -	3.8121e + 0 (1.69e - 1) -	6.3405e + 0 (6.62e - 1) -	2.9387e + 0 (3.03e - 2)
	10	4.5496e + 0 (6.04e - 2) -	7.8930e + 0 (4.69e - 1) -	5.3182e + 0 (2.13e + 0) -	4.9184e + 0 (1.31e - 1) -	8.8993e + 0 (1.88e - 1) -	3.9293e + 0 (4.29e - 2)
	15	9.1411e + 0 (5.10e - 1) -	1.6316e + 1 (7.23e - 1) -	2.1079e + 1 (7.73e + 0) -	1.2450e + 1 (9.06e - 1) -	1.5052e + 1 (1.67e + 0) -	8.0613e + 0 (1.31e - 1)
+/-/=		4/31/1	0/36/0	0/36/0	0/36/0	0/36/0	

To observe the evolution of the algorithms, the evolutionary trajectory of the IGD values of the experimental results obtained using the five state-of-the-art algorithms and MaOEA-AR on the 10-objectives DTLZ4, WFG4 and WFG9 are shown in Fig. 7. In these plots, the X-axis denotes the number of functional assessments, and the Y-axis denotes the value of the IGD indicator, with the data used for plotting obtained from the data closest to the average IGD value. Throughout the evolutionary process, MaOEA-AR significantly outperformed other algorithms on the DTLZ4, WFG4 and WFG9 problems, which makes it competitive in terms of the balance between convergence and diversity. It is clear from Fig. 7 that MaOEA-AR all converge very quickly and are more robust than the other five algorithms. MaOEA-AR converges steadily and rapidly during the evolution. This is because the population itself dynamically guides its evolution towards PF, and A-NDR accelerates its convergence.

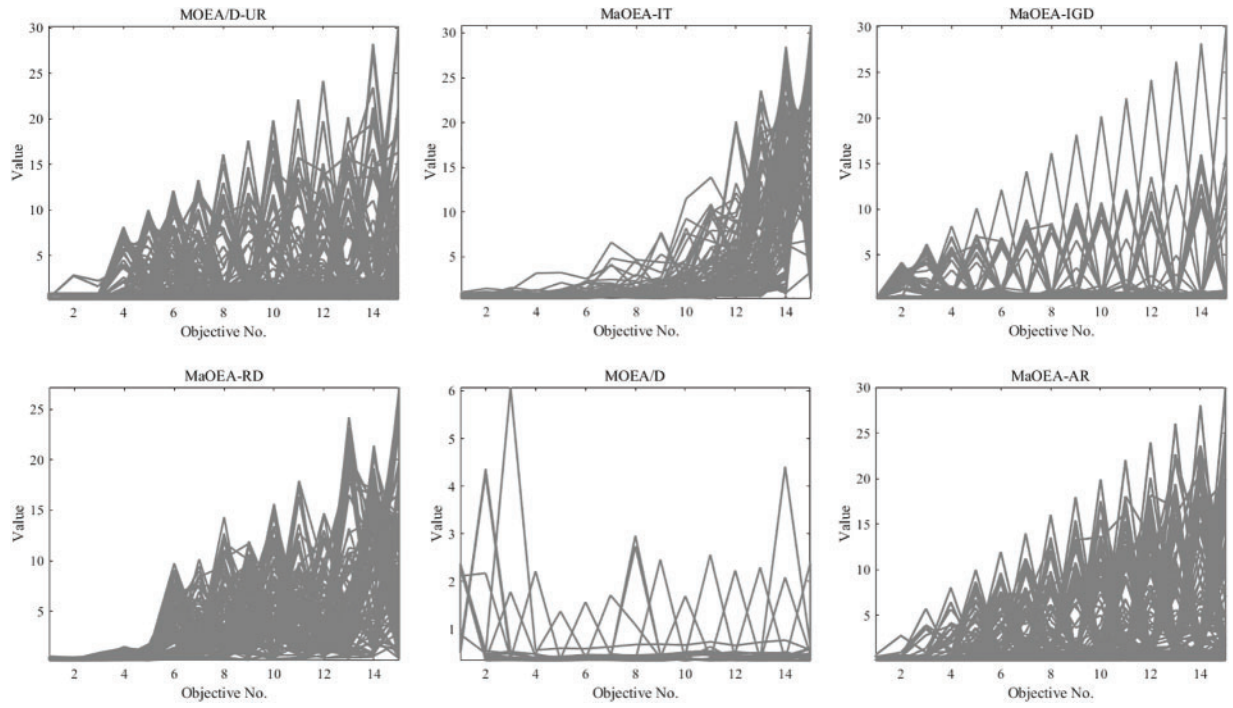


Figure 5: Distribution of solutions on WFG9 with 15-objective

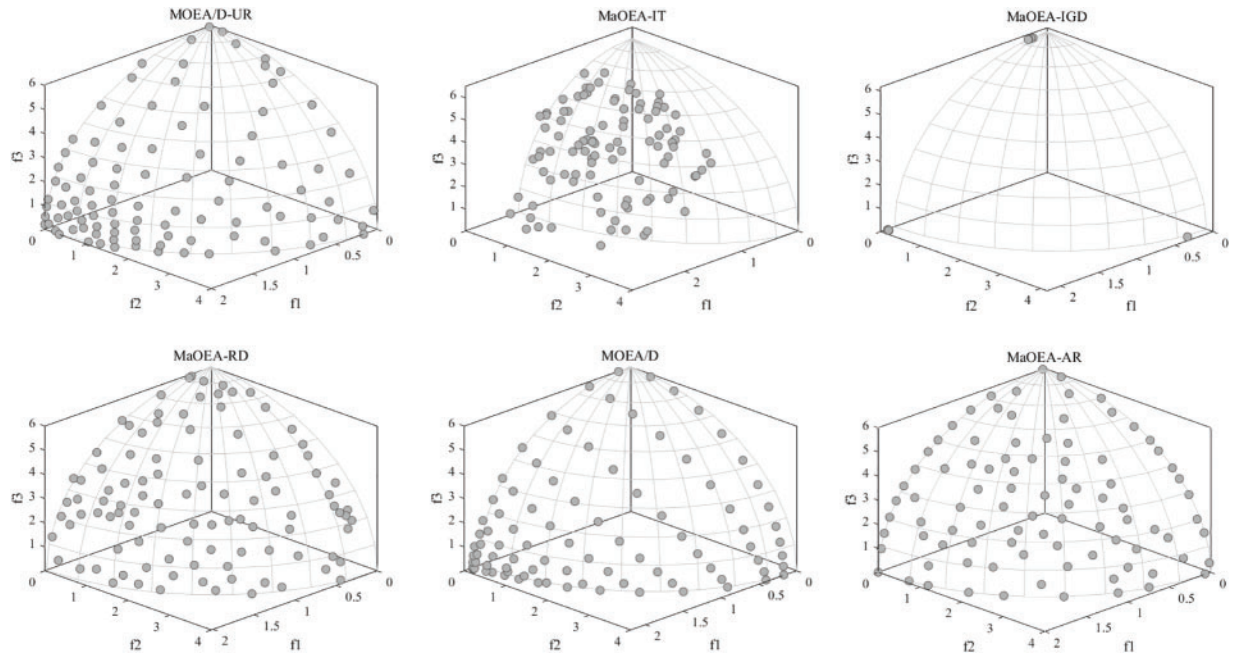


Figure 6: Distribution of solutions on WFG9 with 3-objective

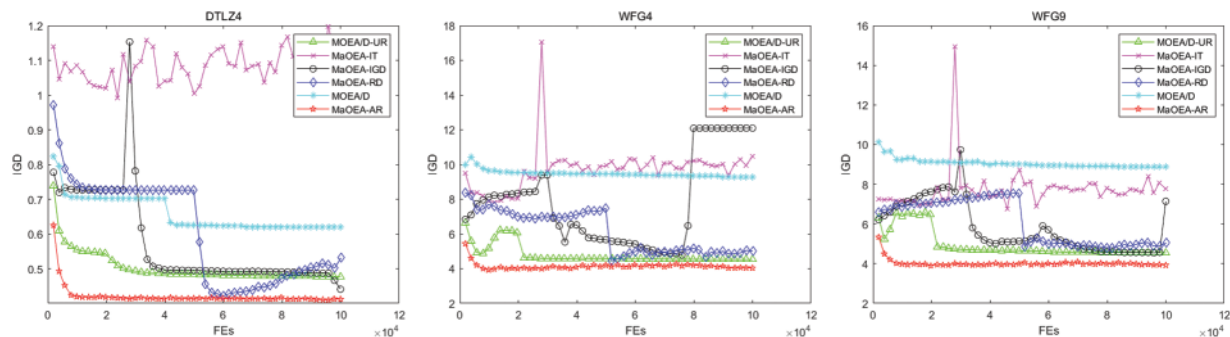


Figure 7: Convergence curves of the algorithms IGD on DTLZ4, WFG4 and WFG9 with 10-objective

3.4.2 Performance on Real-World Applications

To verify the algorithm’s ability to solve real-world optimization problems, three of the many-objective test problems from the RE test suite were selected to validate MaOEA-AR’s ability to solve real-world problems. The RE test suite consists of 16 real-world problems with bounded constraints, we have chosen among them RE4-7-1, RE4-6-2 and RE6-3-1 for experiments, whose corresponding real-world problems and their characteristics are given in Table 6. We refer to the settings in [34], where the population size is 120 when $m = 4$, and when $m = 6$, the population size is 182. The other parameters for each algorithm are set as shown in Section 3.3. In [35], it was noted that setting maxFEs to a small value estimates the performance of the algorithm on expensive problems. We referenced this study and also set the maxFEs to 10,000 in our experiments. Table 7 lists the HV averages obtained from 20 independent runs of the six algorithms on three real-world problems. The MaOEA-IGD solution on the RE4-6-2 problem deviates from the true PF and obtains a HV value of *NaN*. The experimental results show that MaOEA-AR significantly outperforms the other five algorithms.

Table 6: Real-world problems RE4-7-1, RE4-6-2, RE6-3-1 and their features

Problem	Real-world optimization	m	D	The type of design variables	PF
RE4-7-1	Car side impact design [36]	4	7	Continuous	Unknown
RE4-6-2	Conceptual marine design [37]	4	6	Continuous	Unknown
RE6-3-1	Water resource planning [38]	6	3	Continuous	Unknown

The above experimental results show that MaOEA-AR has strong competitiveness in these conventional problems and some real-world representative problems. This can be explained by the fact that our adaptive niche dominance relation is different from the traditional Pareto dominance relations and the existing improved dominance. Most dominance relations can only find a set of solutions concentrated in a small area on the Pareto front. In this study, we used a customized niche technique to balance the diversity of non-dominated convergence, which generates selection pressure even in the high-dimensional objective space, and comparability between non-dominated solutions, alleviating the dominance resistance phenomenon. In the process of generating offspring, we used k -bit crossover and hybrid mutation instead of traditional mutation operators to generate high-quality offspring, which further improves the selection pressure of the algorithm. In addition, we used the APD

to dynamically balance the convergence and diversity of the algorithm during environment selection. However, MaOEA-AR is less effective at dealing with degenerate and irregular problems.

Table 7: HV mean of 6 algorithms on RE4-7-1, RE4-6-2 and RE6-3-1 test problems

Problem	m	MOEA/D-UR	MaOEA-IT	MaOEA-IGD	MaOEA-RD	MOEA/D	MaOEA-AR
RE4-7-1	4	2.3038e - 1 (2.45e - 3) -	2.1082e - 1 (5.24e - 3) -	1.4608e - 1 (4.60e - 4) -	9.6164e - 2 (7.56e - 3) -	2.2941e - 1 (1.44e - 3) -	2.4239e - 1 (7.03e - 4)
RE4-6-2	4	1.9236e - 1 (2.30e - 3) =	1.8205e - 1 (2.45e - 3) -	1.7105e - 1 (1.77e - 3) -	2.0628e - 1 (3.33e - 3) -	1.8575e - 1 (1.80e - 3) -	2.1208e - 1 (2.33e - 3)
RE6-3-1	6	8.3711e - 2 (9.84e - 4) -	7.5515e - 2 (3.65e - 3) -	4.8933e - 2 (7.79e - 4) -	2.6702e - 2 (2.49e - 3) -	7.7563e - 2 (8.37e - 4) -	9.0993e - 2 (3.09e - 4)
+/-/=		0/2/1	0/3/0	0/3/0	0/3/0	0/3/0	

4 Conclusions

In this study, we proposed a reference vector-assisted algorithm MaOEA-AR with an adaptive niche dominance relation to solve MaOPs. We balance the convergence and diversity of the non-dominated solution set by a customized niche technique, where the convergence indicator is determined by the ratio of the distance from the ideal point to the solution and the distance from the ideal point to the hyperplane to select the appropriate convergence indicator by determining the PF shape to maximize the selection pressure of the algorithm. When reproducing offspring are generated, k -bit crossover and hybrid mutations are selected to generate high-quality offspring. Finally, in environment selection, the APD is used to sample solutions with good convergence and diversity. MaOEA-AR was tested with five state-of-the-art algorithms with up to 15-objective on 23 test problems in the DTLZ, MaF and WFG benchmark test suites, as well as on three test problems from the real-world test set RE. The experimental results show that the performance of MaOEA-AR is significantly better than the other five algorithms and has strong competitiveness. In the future, we will optimize the performance of the algorithm to solve degenerate and discontinuous problems, and apply the algorithm to more practical applications.

Acknowledgement: We thank PlatEMO, an evolutionary multi-objective optimization platform developed by BIMK group at Anhui University, for providing us with a good experimental environment.

Funding Statement: This research is supported by the National Natural Science Foundation of China (Grant No. 61976101), the University Natural Science Research Project of Anhui Province (Grant No. 2023AH040056), the Natural Science Research Project of Anhui Province (Graduate Research Project, Grant No. YJS20210463), the Funding Plan for Scientific Research Activities of Academic and Technical Leaders and Reserve Candidates in Anhui Province (Grant No. 2021H264), the Top Talent Project of Disciplines (Majors) in Colleges and Universities in Anhui Province (Grant No. gxbjZD2022021), the University Synergy Innovation Program of Anhui Province, China (GXXT-2022-033) and supported by the Innovation Fund for Postgraduates of Huaibei Normal University (Grant Nos. cx2022041, yx2021023, CX2023043). We would like to thank Editage (www.editage.cn) for English language editing.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization of this study, Methodology, Software, Writing: Yating Wu; Supervision, Writing-review & editing,

Funding acquisition: Fangzhen Ge; Visualization, Investigation: Debao Chen; Supervision: Longfeng Shen. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The experiment in this paper is a simulation experiment, which is completed on the evolutionary multi-objective optimization platform PlatEMO. The comparison algorithm and test function are all from this platform. The experimental data can be found in this paper and Supplementary Material.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

Supplementary Materials: The supplementary material is available online at <https://doi.org/10.32604/iasc.2024.042841>.

References

- [1] J. L. Xu, Z. X. Zhang, Z. M. Hu, L. Du, and X. J. Cai, "A many-objective optimized task allocation scheduling model in cloud computing," *Appl. Intell.*, vol. 51, no. 6, pp. 3293–3310, 2021. doi: [10.1007/s10489-022-03963-w](https://doi.org/10.1007/s10489-022-03963-w).
- [2] L. B. Ma *et al.*, "Learning to optimize: Reference vector reinforcement learning adaption to constrained many-objective optimization of industrial copper burdening system," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 12698–12711, Dec. 2022. doi: [10.1109/TCYB.2021.3086501](https://doi.org/10.1109/TCYB.2021.3086501).
- [3] Z. Y. Hu, Z. H. Wei, H. Sun, J. M. Yang, and L. X. Wei, "Optimization of metal rolling control using soft computing approaches: A review," *Arch. Comput. Method Eng.*, vol. 28, no. 2, pp. 405–421, 2021. doi: [10.1007/s11831-018-9250-y](https://doi.org/10.1007/s11831-018-9250-y).
- [4] X. F. Liu, Z. H. Zhan, and J. Zhang, "Resource-aware distributed differential evolution for training expensive neural-network-based controller in power electronic circuit," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6286–6296, 2022. doi: [10.1109/TNNLS.2021.3075205](https://doi.org/10.1109/TNNLS.2021.3075205).
- [5] M. Farina and P. Amato, "On the optimal solution definition for many-criteria optimization problems," in *2002 Annu. Meet. North Am. Fuzzy Inf. Process. Soc. Proc. NAFIPS-FLINT 2002 (Cat. No. 02TH8622)*, New Orleans, LA, USA, 2002, pp. 233–238.
- [6] R. C. Purshouse and P. J. Fleming, "On the evolutionary optimization of many conflicting objectives," *IEEE Trans. Evolut. Comput.*, vol. 11, no. 6, pp. 770–784, 2007. doi: [10.1109/TEVC.2007.910138](https://doi.org/10.1109/TEVC.2007.910138).
- [7] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multi-objective optimization," *Evol. Comput.*, vol. 10, no. 3, pp. 263–282, 2002. doi: [10.1162/106365602760234108](https://doi.org/10.1162/106365602760234108).
- [8] S. X. Yang, M. Q. Li, X. H. Liu, and J. H. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evolut. Comput.*, vol. 17, no. 5, pp. 721–736, 2013. doi: [10.1109/TEVC.2012.2227145](https://doi.org/10.1109/TEVC.2012.2227145).
- [9] Q. F. Zhang and H. Li, "MOEA/D: A multi-objective evolutionary algorithm based on decomposition," *IEEE Trans. Evolut. Comput.*, vol. 11, no. 6, pp. 712–731, 2007. doi: [10.1109/TEVC.2007.892759](https://doi.org/10.1109/TEVC.2007.892759).
- [10] H. L. Liu, F. Q. Gu, and Q. F. Zhang, "Decomposition of a multi-objective optimization problem into a number of simple multi-objective sub-problems," *IEEE Trans. Evolut. Comput.*, vol. 18, no. 3, pp. 450–455, 2014. doi: [10.1109/TEVC.2013.2281533](https://doi.org/10.1109/TEVC.2013.2281533).
- [11] L. B. Ma, M. Huang, S. X. Yang, R. Wang, and X. W. Wang, "An adaptive localized decision variable analysis approach to large-scale multi-objective and many-objective optimization," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 6684–6696, Jul. 2022. doi: [10.1109/TCYB.2020.3041212](https://doi.org/10.1109/TCYB.2020.3041212).
- [12] D. Brockhoff and E. Zitzler, "Improving hypervolume-based multi-objective evolutionary algorithms by using objective reduction methods," in *2007 IEEE Congr. Evol. Comput.*, Singapore, 2007, pp. 2086–2093.

- [13] Y. N. Sun, G. G. Yen, and Y. Zhang, "IGD indicator-based evolutionary algorithm for many-objective optimization problems," *IEEE Trans. Evolut. Comput.*, vol. 23, no. 2, pp. 173–187, 2019. doi: [10.1109/TEVC.2018.2791283](https://doi.org/10.1109/TEVC.2018.2791283).
- [14] E. M. Lopez and C. A. C. Coello, "IGD+-EMOA: A multi-objective evolutionary algorithm based on IGD+," in *2016 IEEE Congr. Evol. Comput.*, Vancouver, BC, Canada, 2016, pp. 999–1006.
- [15] J. W. Yuan, H. L. Liu, F. Q. Gu, Q. F. Zhang, and Z. S. He, "Investigating the properties of indicators and an evolutionary many-objective algorithm using promising regions," *IEEE Trans. Evolut. Comput.*, vol. 25, no. 1, pp. 75–86, 2021. doi: [10.1109/TEVC.2020.2999100](https://doi.org/10.1109/TEVC.2020.2999100).
- [16] Z. S. Song, H. D. Wang, C. He, and Y. C. Jin, "A kriging-assisted two-archive evolutionary algorithm for expensive many-objective optimization," *IEEE Trans. Evolut. Comput.*, vol. 25, no. 6, pp. 1013–1027, 2021. doi: [10.1109/TEVC.2021.3073648](https://doi.org/10.1109/TEVC.2021.3073648).
- [17] K. Li, K. Deb, Q. F. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Trans. Evolut. Comput.*, vol. 19, no. 5, pp. 694–716, 2015. doi: [10.1016/j.knosys.2021.107392](https://doi.org/10.1016/j.knosys.2021.107392).
- [18] H. Xu, B. Xue, and M. J. Zhang, "A duplication analysis-based evolutionary algorithm for bi-objective feature selection," *IEEE Trans. Evolut. Comput.*, vol. 2, no. 2, pp. 205–218, 2021. doi: [10.1109/TEVC.2020.3016049](https://doi.org/10.1109/TEVC.2020.3016049).
- [19] R. Cheng, Y. C. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. Evolut. Comput.*, vol. 20, no. 5, pp. 773–791, 2016. doi: [10.1109/TEVC.2016.2519378](https://doi.org/10.1109/TEVC.2016.2519378).
- [20] Q. Q. Liu, Y. C. Jin, M. Heiderich, T. Rodemann, and Y. Guo, "An adaptive reference vector-guided evolutionary algorithm using growing neural gas for many-objective optimization of irregular problems," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 2698–2711, 2022. doi: [10.1109/TCYB.2020.3020630](https://doi.org/10.1109/TCYB.2020.3020630).
- [21] B. H. Nguyen, B. Xue, P. Andreae, H. Ishibuchi, and M. J. Zhang, "Multiple reference points-based decomposition for multi-objective feature selection in classification: Static and dynamic mechanisms," *IEEE Trans. Evolut. Comput.*, vol. 24, no. 1, pp. 170–184, 2020. doi: [10.1109/TEVC.2019.2913831](https://doi.org/10.1109/TEVC.2019.2913831).
- [22] Y. Tian, R. Cheng, X. Y. Zhang, Y. S. Su, and Y. C. Jin, "A strengthened dominance relation considering convergence and diversity for evolutionary many-objective optimization," *IEEE Trans. Evolut. Comput.*, vol. 23, no. 2, pp. 331–345, 2019. doi: [10.1109/TEVC.2018.2866854](https://doi.org/10.1109/TEVC.2018.2866854).
- [23] L. Li, G. G. Yen, A. Sahoo, L. Chang, and T. Gu, "On the estimation of pareto front and dimensional similarity in many-objective evolutionary algorithm," *Inform. Sci.*, vol. 563, pp. 375–400, 2021. doi: [10.1016/j.ins.2021.03.008](https://doi.org/10.1016/j.ins.2021.03.008).
- [24] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based non-dominated sorting approach, part I: Solving problems with box constraints," *IEEE Trans. Evolut. Comput.*, vol. 18, no. 4, pp. 577–601, 2014. doi: [10.1109/TEVC.2013.2281535](https://doi.org/10.1109/TEVC.2013.2281535).
- [25] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multi-objective optimization," in *Proc. 2002 Congr. Evol. Comput. (Cat. No.02TH8600)*, Singapore, 2002, pp. 105–145.
- [26] R. Cheng *et al.*, "A benchmark test suite for evolutionary many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 1, pp. 67–81, 2017.
- [27] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multi-objective test problems and a scalable test problem toolkit," *IEEE Trans. Evolut. Comput.*, vol. 10, no. 5, pp. 477–506, 2006. doi: [10.1109/TEVC.2005.861417](https://doi.org/10.1109/TEVC.2005.861417).
- [28] Y. N. Sun, B. Xue, M. J. Zhang, and G. G. Yen, "A new two-stage evolutionary algorithm for many-objective optimization," *IEEE Trans. Evolut. Comput.*, vol. 23, no. 5, pp. 748–761, 2019. doi: [10.1109/TEVC.2018.2882166](https://doi.org/10.1109/TEVC.2018.2882166).
- [29] Z. He and G. G. Yen, "Many-objective evolutionary algorithm: Objective space reduction and diversity improvement," *IEEE Trans. Evolut. Comput.*, vol. 20, no. 1, pp. 145–160, 2016. doi: [10.1109/TEVC.2015.2433266](https://doi.org/10.1109/TEVC.2015.2433266).
- [30] L. R. C. de Farias and A. F. R. Araujo, "A decomposition-based many-objective evolutionary algorithm updating weights when required," *Swarm and Evolutionary Computation*. vol. 68. pp. 100980, Feb. 2022.

- [31] L. While, P. Hingston, L. Barone, and S. Huband, "A faster algorithm for calculating hypervolume," *IEEE Trans. Evolut. Comput.*, vol. 10, no. 1, pp. 29–38, 2006. doi: [10.1109/TEVC.2005.851275](https://doi.org/10.1109/TEVC.2005.851275).
- [32] E. Zitzler, L. Thiele, and M. Laumanns, "Performance assessment of multi-objective optimizers: An analysis and review," *IEEE Trans. Evolut. Comput.*, vol. 7, no. 2, pp. 117–132, 2003. doi: [10.1109/TEVC.2003.810758](https://doi.org/10.1109/TEVC.2003.810758).
- [33] Y. Tian, R. Cheng, X. Y. Zhang, and Y. C. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, 2017. doi: [10.1109/MCI.2017.2742868](https://doi.org/10.1109/MCI.2017.2742868).
- [34] R. Tanabe and H. Ishibuchi, "An easy-to-use real-world multi-objective optimization problem suite," *Appl. Soft. Comput.*, vol. 89, pp. 106078, Apr. 2020. doi: [10.1016/j.asoc.2020.106078](https://doi.org/10.1016/j.asoc.2020.106078).
- [35] Q. L. Zhu *et al.*, "An elite gene guided reproduction operator for many-objective optimization," *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 765–778, 2021. doi: [10.1109/TCYB.2019.2932451](https://doi.org/10.1109/TCYB.2019.2932451).
- [36] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part II: Handling constraints and extending to an adaptive approach," *IEEE Trans. Evolut. Comput.*, vol. 18, no. 4, pp. 602–622, 2013. doi: [10.1109/TEVC.2013.2281534](https://doi.org/10.1109/TEVC.2013.2281534).
- [37] M. G. Parsons and R. L. Scott, "Formulation of multi-criterion design optimization problems for solution with scalar numerical optimization methods," *J. Ship. Res.*, vol. 48, no. 1, pp. 61–76, 2004. doi: [10.5957/jsr.2004.48.1.61](https://doi.org/10.5957/jsr.2004.48.1.61).
- [38] T. Ray, K. Tai, and K. C. Seow, "Multi-objective design optimization by an evolutionary algorithm," *Eng. Optimiz.*, vol. 33, no. 4, pp. 399–424, 2001.