A decomposition-based multiobjective evolutionary algorithm with weights updated adaptively

Liu, Yuan; Hu, Yikun; Zhu, Ningbo; Li, Kenli; Zou, Juan; Li, Miqing

DOI:
10.1016/j.ins.2021.03.067

A Decomposition-based Multiobjective Evolutionary Algorithm with Weights Updated Adaptively

Yuan Liu[a,*], Yikun Hu[a], Ningbo Zhu[a,*], Kenli Li[a], Juan Zou[b], Miqing Li[c]

[a]*College of Computer Science and Electronic Engineering, Hunan University, Hunan, China*
[b]*School of Computer Science, Xiangtan University, Hunan, China*
[c]*CERCIA, School of Computer Science, University of Birmingham, Birmingham B15 2TT, U. K.*

Abstract

Recently, decomposition-based multiobjective evolutionary algorithms (DMEAs) have become more prevalent than other patterns (e.g., Pareto-based algorithms and indicator-based algorithms) for solving multiobjective optimization problems (MOPs). They utilize a scalarizing method to decompose an MOP into several subproblems based on the weights provided, resulting in the performances of the algorithms being highly dependent on the uniformity between the problem's optimal Pareto front and the distribution of the specified weights. However, weight generation is generally based on a simplex lattice design, which is suitable for "regular" Pareto fronts (i.e., simplex-like fronts) but not for other "irregular" Pareto fronts. To improve the efficiency of this type of algorithm, we develop a DMEA with weights updated adaptively (named DMEA-WUA) for the problems regarding various Pareto fronts. Specifically, the DMEA-WUA introduces a novel exploration versus exploitation model for environmental selection. The exploration process finds appropriate weights for a given problem in four steps: weight generation, weight deletion, weight addition and weight replacement. Exploitation means using these weights from the exploration step to guide the evolution of the population. Moreover, exploration is carried out when the exploitation

*Corresponding author

Email addresses: liu3yuan@hnu.edu.cn (Yuan Liu), yikunhu@hnu.edu.cn (Yikun Hu), quietwave@hnu.edu.cn (Ningbo Zhu), lkl@hnu.edu.cn (Kenli Li), zoujuan@xtu.edu.cn (Juan Zou), m.li.8@bham.ac.uk (Miqing Li)

[1]Since 1880.

process is stagnant; this is different from the existing method of periodically updating weights. Experimental results show that our algorithm is suitable for solving problems with various Pareto fronts, including those with "regular" and "irregular" shapes.

Keywords: The decomposition-based multiobjective evolutionary algorithm, multiobjective optimization problems, exploration, exploitation, weights updated adaptively.

1. Introduction

Multiobjective optimization problems (MOPs) generally have dual characteristics: 1) multiple objectives should be solved simultaneously, and 2) these objectives conflict with one another. Formally, an MOP can be defined as

$$\begin{aligned} &\min_{x} \quad \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x})) \\ &\textit{subject to} \quad \mathbf{x} \in \Omega \end{aligned} \tag{1}$$

where $\mathbf{f}(\mathbf{x})$ is an m-dimensional objective function and $\mathbf{x} = (x_1, \cdots, x_d)$ is a d-dimensional decision vector. Ω is the feasible set in the decision space $\mathbb{R}^k$, and $\mathbf{f} : \Omega \to \mathbb{R}^m$ is the corresponding attainable set in the objective space $\mathbb{R}^m$. Due to the characteristics of MOPs, there is no single optimal solution that is able to optimize all the objectives at the same time. Alternatively, a number of solutions, also known as the Pareto optimal set, can be obtained as trade-offs between different objectives, and decision-maker can choose from them. For approimating the Pareto optimal set, multiobjective evolutionary algorithms (MOEAs) have received significant attention over the last two decades [1]. MOEAs use a population of evolving solutions in their iterative steps by establishing an implicit parallel search. These population-based approaches allow a number of diverse nondominated solutions to be processed simultaneously, thereby enabling the population to converge near the Pareto-optimal front with a proper distribution.

The existing MOEAs can be roughly categorised into three patterns: 1) Pareto-based MOEAs and their variants [2]; 2) indicator-based MOEAs [3]; and 3) decomposition-based MOEAs [4]. The basic idea of Pareto-based MOEAs is to distinguish solutions based on their dominance relationships and density estimations. The former is considered a type of the primary selection and prefers nondominated solutions to dominated ones, while the latter is used to maintain the diversity of the nondominated solutions. Density estimation is performed when solutions cannot be compared using primary selection [5]. Indicator-based MOEAs use an indicator (i.e., inverted generational distance (IGD) [6] and hypervolume (HV) [7]) instead of (or together with) a dominance comparison procedure to guide the search during the evolutionary process [8]. Such indicator-based MOEAs provide stronger selection pressure toward the Pareto-optimal front than the Pareto-based MOEAs, but the computational costs of obtaining the solutions required by these methods increase exponentially with the number of objectives. Decomposition-based MOEAs decompose an MOP into several sub-problems based on the predefined weights via a scalarizing method (i.e., weighted sum (WS) [4], penalty-based boundary intersection (PBI) [4] and Tchebycheff (TCH) [4] methods), and use a heuristic search to optimize these subproblems simultaneously [4].

Different from the first two categories, decomposition-based algorithms are used as posteriori multiobjective algorithms, and their performances are limited by the distribution of the predefined weights [9]. Taking multiobjective evolutionary algorithm based on decomposition (MOEA/D) [4] as an example, it first specifies a set of weights, each of which corresponds to one subproblem, and then it uses the specified set of weights to maintain a diverse set of trade-off solutions. MOEA/D assumes during decomposition that optimizing multiple subproblems can produce a set of well-distributed Pareto-optimal solutions if the weight of each subproblem is appropriately chosen [10], and it ultimately expects these subproblems to be closely associated with one solution in the population. Ordinarily, decomposition-based algorithms used a simplex lattice method to construct uniformly distributed weights on a hyperplane [11]. The weights constructed can make the decomposition-based algorithms suitable for MOPs with “regular” (i.e., simplex-like) Pareto fronts. Once the distribution of specified weights is made consistent with the Pareto front shape of a given problem, decomposition-based algorithms have significant advantages over the other two types of algorithms in solving MOPs, especially many-objective optimization problems (MaOPs) with objective numbers greater than 3 [12]. Such advantages include having lower computational complexity at each generation [4], providing high selection pressure toward the Pareto front [4], owing to their high searchability for combinatorial optimization problems [13], and being easy to design a local search operator [14].

Unfortunately, the Pareto fronts of most actual MOPs are usually not known in advance [15]. When the shape of the Pareto front is far from that of the unit simplex, a set of weights from the simplex lattice method may not result in a uniform distribution of Pareto optimal solutions [9]. For example, when faced with convex/concave Pareto fronts, the Pareto optimal solutions distributed in the middle of these Pareto fronts will be more/less crowded than those on the border [16]; when an MOP is degenerated or disconnected, only a portion of the weights may be intersected by the Pareto front, resulting in several weights associated with multiple Pareto optimal solutions [17]. In such MOPs with "irregular" Pareto fronts, the performance of the decomposition-based MOEA is substantially degraded because of the difference between the Pareto front shape and the distribution of weights. Thus, to adapt the weights to MOPs with various Pareto fronts, studies have made efforts in the following two categories.

The first includes the model-based generation methods [18, 19, 20], where a pseudo Pareto front is obtained by fitting the obtained nondominated solutions using a machine learning method, and a uniform set of weights is generated directly from the estimated Pareto front. Experiments have shown that these methods are effective for MOPs with 2-3 objectives, but they are difficult to extend for the purpose of solving MaOPs. There may be many noise samples in the nondominated solutions due to the ineffectiveness of the Pareto dominance relationships in the MaOPs [21], and these will affect the quality of the fitted model. In addition, a complex modelincurs high computational costs during training [20]. The second group includes the set-based update methods [12, 17, 22], where an archive or the current population is used to update the weights during the optimization process so that their distribution gradually conforms to the Pareto front shape. These methods have achieved significantly better performances on MOPs with irregular Pareto fronts than other approaches, but their performances may be drastically reduced when solving MOPs with regular Pareto fronts. As an example in [8], a distribution of multiple solutions obtained by A-NSGA-III [17] was noted as being worse than that obtained by NSGA-III [11] on DTLZ1 [23] with a "regular" Pareto front. Two types of detailed algorithms will be introduced in the related work section of this paper (Section II).

Therefore, weight adaptation is not yet a mature approach because it does not handle a wide variety of MOPs as effectively as a predefined weight approach dealing with simplex-like Pareto fronts. To improve the efficiency of the decomposition-based algorithm, we propose a novel decomposition-based

MOEA with weights updated adaptively, denoted as the DMEA-WUA. The DMEA-WUA, as a set-based update approach, introduces the concept of an exploration versus exploitation (EvE) model for environmental selection. In the model, the exploration and exploitation process work together to guide the population to evolve quickly toward the optimal front while simultaneously maintaining the diversity of the population during the evolutionary process.

Exploration is the process of updating weights, and it allows our algorithm to explore entirely new regions of a search space. This exploration process is mainly composed of four parts: weight generation, weight deletion, weight addition and weight replacement. Weight generation uses a simple mapping operation. Weight deletion removes ineffective weights that are not associated with solutions or are associated with abnormal solutions. Weight addition provides new weights in sparse regions of the search space. Weight replacement improves the uniformity of the obtained solutions. Moreover, an archive with a size of $1.5N$ is introduced to guide the weight update procedure, where N is the size of the population.

Exploitation is the process of visiting the regions covered by the determined weights. This process involves not only the selection of superior solutions but also the calculation of the stability in each search region. The systematic association and niche preservation mechanisms of the selection operation are similar but not identical to those in NSGA-III [11]. The degree of stability reflects the evolutionary trend of the population. The end of an exploitation episode occurs when the solutions associated with each weight are fully evolved, and this is reflected in the stability value.

In short, the significant innovations and contributions of this paper are summarized as follows.

- Exploration is carried out when the exploitation process is stagnant. Different from periodically adapting the weights, it is not necessary to consider the impact of a frequent change on the performance of the algorithm;

- A novel approach is proposed for the adaptation of weights during the evolutionary search procedure. During exploration, we not only consider whether the weights have associated solutions, but also consider the distribution following by the solutions associated with each weight.

- The solutions in the archive set are strictly controlled within the valid search region determined by the extreme point of the current population.

- In the exploitation process, we provide different selection mechanisms for internal solutions and boundary solutions. For internal solutions, we prefer the nonominated solutions closest to the weights. For boundary solutions, we use a constraint to improve the convergence quality of the population.

The remainder of this paper is organized as follows. In Section II, some related works are briefly reviewed. The general architecture of the EvE model is described in Section III. The proposed algorithm, the DMEA-WUA, is elaborated in Section IV. Experimental results are presented in Section V. Finally, the conclusion is drawn in Section VI.

2. Related Work

In this section, we first present some notable works in terms of adaptive weighting methods. A more detailed summary of adaptive weight adjustment techniques can be found in [24]. Then, a brief review of exploration and exploitation in evolutionary algorithms (EAs) is given.

2.1. Adaptive Weighting Methods

In this paper, we generalize these methods into two categories based on their characteristics: model-based generation methods and set-based update methods.

2.1.1. Model-based Generation Methods

As described previously, model-based methods uniformly sample a set of points as weight vectors on a model constructed from a nondominated solution. In this line of research, various interpolation-based methods [19, 25] are applied between nondominated solutions to approximate the Pareto front. These methods can model low-dimensional problems but struggle to solve many-dimensional problems because of the impact of outliers. $pa\lambda$-MOEA/D [19] uses a paradigm $\sum_i^m f_i^p = 1$ to approximate the Pareto fronts in each generation, where p is used to control the curvature of the Pareto front estimated by calculating the area of the nondominated solutions in an archive. The symmetry of the paradigm makes $pa\lambda$-MOEA/D unable to deal with an asymmetric Pareto front. Furthermore, GFM [26] considers a more general paradigm $\sum_i^m a_i f_i^{p_i} = 1$ to fit each Pareto front, where a_i and p_i are used to control the scale and curvature of the Pareto front in terms of the i-th objective, respectively. In theory, GFM can substitute $pa\lambda$ to help the algorithm approximate various types of Pareto fronts. There are works that have used a self-organizing map (SOM) network [20] and a growing neural gas network [25] to confirm that the learning of the topological structure of a given Pareto front also contributes to the design of the utilized weights. However, both methods have expensive computational overhead since the network needs to be trained in each generation. In addition, Wu et al. [18] designed a learning-to-decompose (LTD) paradigm in which Gaussian process regression is employed to learn an analytical model for the Pareto front every 20 generations. Similarly, Ge et al. [27] proposed an incremental learning model to achieve the adaptive setting of weights. The additional computational costs associated with the use of these models also cannot be ignored.

2.1.2. Set-based Update Methods

The implementation process for the set-based update methods is more straightforward than that for model-based generation methods. The set-based update methods directly adjust the weight vectors periodically according to the distribution of the population or archive set. Following this line of research, VaEA [28], DDEA [29] and MOEA/D-AM2M [30] treat the current population as an approximation of the Pareto front and reset the feasible subregions based on the distributions of the population in the objective space. However, unlike VaEA and DDEA, MOEA/D-AM2M needs to define weights in advance and then resets the weights periodically. Delete-and-add patterns [31, 16, 32, 22, 33] are effective methods for updating the weights in such algorithms. Specifically, RVEAa [31] uses random weights to replace the invalid weights that are not associated with any solution in each generation. g-DBEA [22] deletes invalid weights and adds new weights close to the deleted weights in each generation. MOEA-AWA [16] and MOEA/D-URAW [33] periodically delete weights located in crowded areas and add new weights to the sparsest areas. MaOEA/D-2ADV [32] periodically replaces the poor weights by inserting new weights at the midpoint of two effective neighboring weights. Conversely, there are also some algorithms [17, 12] that use add-and-delete patterns to update the weights. Specifically, A-NSGA-III [17] first adds a simplex with m neighbor weights for each crowded weight and deletes invalid weights that are associated with no individuals. AdaW [12] periodically generates promising, undeveloped weights according to archive solutions and then deletes the poor-performing weights.

2.2. Exploration and Exploitation

Exploration and exploitation are the two cornerstones of problem solving when using a search method. Exploration is the ability of a search algorithm to discover entirely new regions of the search space, while exploitation is its capability to access the regions of the search space that are within the neighborhood of promising previously searched solutions [34]. A search algorithm needs to establish an equilibrium between exploration and exploitation [35], and EAs are no exception. Excessive exploration often leads to globally optimal discoveries, but severely reduces the rate of convergence. In contrast, excessive exploitation leads to convergence at a local optimum.

In EAs, exploration and exploitation relationships are maintained throughout their execution by crossover, mutation and selection. With the crossover operator, two or more parents are combined to generate the possibility of improved offspring. From the perspective that good information exchange between good solutions produces better offspring [36], the crossover operator is much like an exploitation operator. However, an excellent crossover operator should also produce solutions in sparse search areas. The potential number of ways in which an operator can generate new solutions is called its exploratory ability [37]. A mutation operator is like an exploration operator when it randomly modifies solutions with a given probability and thus increases the diversity of the population. Conversely, a mutation operator can also be seen as an exploitation operator because most of the genetic characteristics are preserved [38]. Additionally, the levels of exploration and exploitation can be controlled by varying the selection pressure during the environment selection process. Higher selection pressure drives the search to more toward exploitation, and lower selection pressure urges the search to provide more exploration. Maintaining a suitable selection pressure to strike an equilibrium between exploration and exploitation is needed for many MOPs.

For decomposition-based MOEAs, the dynamic updating of weights and the search for solutions among these weights also require an equilibrium between exploration and exploitation. For example, if the update freuency is too low, insufficient exploration leads to considerable computational resources wasted on the exploitation of unpromising search directions. Conversely, if the update frequency is too high, excessive exploration may cause solutions to wander in the search space, and this seriously affects the convergence of the population [12]. As described in Section 2.1, most of the existing adaptive weighting methods use periodic update frequencies to balance exploration and exploitation. However, fixed-frequencies greatly hamper the universality of such algorithms.

3. Exploitation Versus Exploration Model

The exploitation module optimizes the population, stability matrix and archive set. The exploration module aims to find the appropriate weights for a given problem in four steps: weight generation, weight deletion, weight addition and weight replacement. The EvE model switches from the exploitation module to the exploration module when the stability factor δ becomes close to one ($\delta \approx 1$). In the following subsections, we describe each part of the model.

3.1. The Exploitation Module

3.1.1. Population Optimization

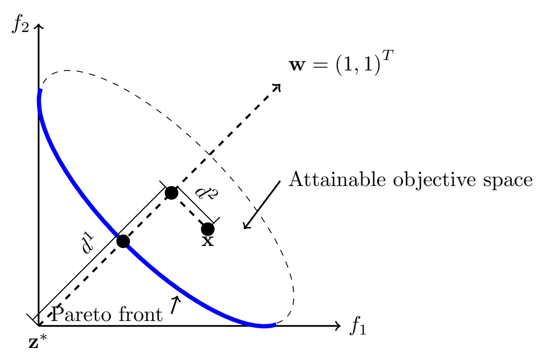


Figure 1: An illustration of the projection distance d_1 and perpendicular distance d_2 for a given solution and weight.

Population optimization technique in our algorithm uses the NSGA-III [11] framework but varies in the selection operator. In our selection operator, to obtain a suitable set of solutions, we first divide the current solutions into boundary sets and internal sets, which are associated with the boundary weights and internal weights, respectively, and then we use different methods (named the fitness-based method for the internal weights and the constraint fitness-based method for the boundary weights) to evaluate the fitness values of the solutions in both sets. Specifically, the systematic association is based on the perpendicular distance (e.g., d^2 in Fig. 1) of each solution from each of the weights. The solution that is closest to a weight in the objective space is considered to be associated with that weight. Obviously, a weight may be associated with one or more solutions, or it may not have any solution associated with it. We hope that each weight can eventually have a unique associated solution. In niche preservation, the fitness-based method chooses the superior solution(s) $\mathbf{x}$ of each subproblem (weight) by comparing the fitness values of all solutions; this is defined as

$$fit(\mathbf{x}) = DS(\mathbf{x}) + PD(\mathbf{x}), \tag{2}$$

where $DS(\mathbf{x})$ is the local dominance strength of $\mathbf{x}$, which is represented by the sorted non-dominance value of the solutions within the same subproblem. $PD(\mathbf{x})$ is calculated as

$$PD(\mathbf{x}) = \frac{d_2(\mathbf{x})}{\max_{\mathbf{x}_j \in \mathbf{C}} d_2(\mathbf{x}_j) + 1}, \tag{3}$$

where $\mathbf{C}$ denotes the set of solutions in subproblem $\mathbf{w}$ and $d_2(\mathbf{x})$ is the perpendicular distance from $\mathbf{x}$ to its associated weight $\mathbf{w}$. Unlike the fitness-based method for the internal weights, the constraint fitness-based method for the boundary weights introduces a convergence constraint ε_k to prevent degradation in this search direction, and it is defined as follows.

$$\varepsilon_k = \overline{d}_k \cdot (1 + cos(g/G \cdot \pi/2)), \tag{4}$$

where $\overline{d}_k$ is the minimum projection distance (e.g., d^1 in Fig. 1) among the historical solutions associated with $\mathbf{w}_k$. g and G are the current generation and the maximum number of generations, respectively. The solutions within the constraint have higher priority levels than the solutions outside the constraint. An example is shown in Fig. 2; there are four weights in the two-dimensional objective space, and each weight has two associated solutions. For the internal weights $\mathbf{w}_1$ and $\mathbf{w}_2$, the fitness of $\mathbf{x}_1$ is better than that of $\mathbf{x}_2$ in terms of $\mathbf{w}_1$ because the former dominates the latter. When two solutions $\mathbf{x}_3$ and $\mathbf{x}_4$ do not dominate each other, the one with the shortest perpendicular distance from weight $\mathbf{w}_2$ is selected. For the boundary weights $\mathbf{w}_3$ and $\mathbf{w}_4$, the convergence constraint raises the high priority levels of the solutions in the effective regions of the boundary weights. For example, $\mathbf{x}_5$ is superior to $\mathbf{x}_6$ in terms of weight $\mathbf{w}_3$, even though the perpendicular distance of the former to $\mathbf{w}_3$ is greater than that of the latter.

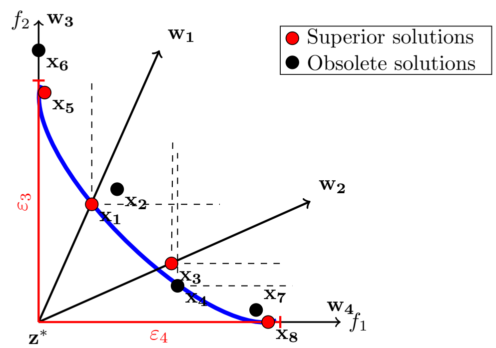


Figure 2: An example of the niche preservation mechanism, including the choice of internal solutions and boundary solutions.

3.1.2. Stability optimization

A stability matrix $\mathbf{S}$ is introduced to maintain an equilibrium between exploration and exploitation. In this matrix, the minimum convergence value among the historical solutions associated with each weight is recorded. As shown in Fig. 3, each row in $\mathbf{S}$ records the minimum convergence value $\widehat{d}_{k,t}$ owned by a weight $\mathbf{w}_k$ over nearly $2T$ generations. Formally, $\widehat{d}_{k,t}$ is described as follows:

$$\widehat{d}_{k,t} = \begin{cases} \min\ (\overline{d}_k, \widetilde{d}_k) & \text{if } \mathbf{C}_k \neq \emptyset \text{ and } \overline{d}_k \neq 0 \\ \widetilde{d}_k & \text{if } \mathbf{C}_k \neq \emptyset \text{ and } \overline{d}_k = 0 \\ 0 & \text{otherwise} \end{cases}, \tag{5}$$

where $k \in \{1, 2, \cdots, N\}$, $\mathbf{C}_k$ contains the solutions associated with $\mathbf{w}_k$ in the current generation and $\widetilde{d}_k$ is the minimum projection distance of the solutions in the set $\mathbf{C}_k$. Moreover, each row in $\mathbf{S}$ is equivalent to a first-in, first-out queue, where a new element $\widehat{d}_{k,t}$ is added and the oldest element $\widehat{d}_{k,1}$ is processed first.

$$
\begin{array}{c}
\\
\mathbf{w_1} \\ \mathbf{w_2} \\ \mathbf{w_3} \\ \leftarrow \vdots \\ \mathbf{w_{N-2}} \\ \mathbf{w_{N-1}} \\ \mathbf{w_N}
\end{array}
\begin{array}{c}
\begin{array}{cccccccc}
\text{t-2T} & \text{t-(2T-1)} & \cdots & \text{t-T} & \text{t-(T-1)} & \text{t-(T-2)} & \cdots & \text{t-1}
\end{array} \\
\left(\begin{array}{cccc:cccc}
\widehat{d}_{1,1} & \widehat{d}_{1,2} & \cdots & \widehat{d}_{1,T} & \widehat{d}_{1,T+1} & \widehat{d}_{1,T+2} & \cdots & \widehat{d}_{1,2T} \\
\widehat{d}_{2,1} & \widehat{d}_{2,2} & \cdots & \widehat{d}_{2,T} & \widehat{d}_{2,T+1} & \widehat{d}_{2,T+2} & \cdots & \widehat{d}_{2,2T} \\
\widehat{d}_{3,1} & \widehat{d}_{3,2} & \cdots & \widehat{d}_{3,T} & \widehat{d}_{3,T+1} & \widehat{d}_{3,T+2} & \cdots & \widehat{d}_{3,2T} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\widehat{d}_{N-2,1} & \widehat{d}_{N-2,2} & \cdots & \widehat{d}_{N-2,T} & \widehat{d}_{N-2,T+1} & \widehat{d}_{N-2,T+2} & \cdots & \widehat{d}_{N-2,2T} \\
\widehat{d}_{N-1,1} & \widehat{d}_{N-1,2} & \cdots & \widehat{d}_{N-1,T} & \widehat{d}_{N-1,T+1} & \widehat{d}_{N-1,T+2} & \cdots & \widehat{d}_{N-1,2T} \\
\widehat{d}_{N,1} & \widehat{d}_{N,2} & \cdots & \widehat{d}_{N,T} & \widehat{d}_{N,T+1} & \widehat{d}_{N,T+2} & \cdots & \widehat{d}_{N,2T}
\end{array}\right)
\end{array}
\begin{array}{c}
\text{t} \\
\widehat{d}_{1,t} \\ \widehat{d}_{2,t} \\ \widehat{d}_{3,t} \\ \vdots \; \leftarrow \\ \widehat{d}_{N-2,t} \\ \widehat{d}_{N-1,t} \\ \widehat{d}_{N,t}
\end{array}
$$

Figure 3: Stability matrix $\mathbf{S}$.

3.1.3. Archive Optimization

The purpose of archive optimization is to make a limited number of nondominated solutions in the archive evenly distributed in the valid objective space. Here, the valid area is determined by the extreme point $\mathbf{e}_i$ of each objective, and the calculation method for the extreme points is the same as that of NSGA-III [11].

First, we store the nondominated solutions located in the effective target space into the archive set. During the subsequent optimization process, we first perform a dominance check between the efficient nondominated solutions in the population and the efficient solutions in the archive. If one of the nondominated solutions is dominated by an archive member, this nondominated solution is not accepted. Otherwise, if any archive member is dominated by one of the nondominated solutions, those archive members are deleted, and the nondominated solution is accepted. If neither of the above two cases occurs, this means that the nondominated solutions are nondominated by the archive members and that all of these nondominated solutions are accepted. Once the number of solutions in the archive exceeds the preset capacity, a distribution maintenance mechanism is used to remove some solutions located in crowded regions. Here, we directly adopt the archive maintenance method in [12] due to its efficiency and effectiveness. This mechanism iteratively deletes the solution in the archive with the largest crowding degree until the number of archive members is equal to the preset capacity. Considering both the number and locations of neighbours, we define the crowding degree of a solution $\mathbf{x}_a$ in the archive $\mathbf{A}$ as follows:

$$D(\mathbf{x}_a) = 1 - \prod_{\mathbf{x}_b \in \mathbf{A}, \mathbf{x}_a \neq \mathbf{x}_b} R(\mathbf{x}_a, \mathbf{x}_b), \tag{6}$$

where

$$R(\mathbf{x}_a, \mathbf{x}_b) = \begin{cases} d(\mathbf{x}_a, \mathbf{x}_b)/r & \text{if} \quad d(\mathbf{x}_a, \mathbf{x}_b) \leq r \\ 1 & \text{otherwise} \end{cases}, \tag{7}$$

in which $d(\mathbf{x}_a, \mathbf{x}_b)$ is the Euclidean distance between solutions $\mathbf{x}_a$ and $\mathbf{x}_b$. The parameter r, as the largest neighbouring region, is set to the median of the distances from all the solutions to their m-th nearest solution in the archive and m is the number of objectives in the problem. The whole procedure of archive optimization is given in Fig. 4.

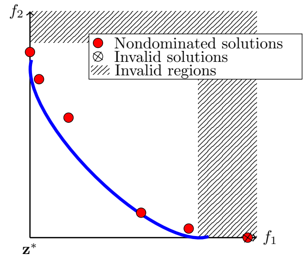
3.2. Balancing Exploration and Exploitation

The duality between exploitation and exploration has long been an important issue in decomposition-based MOEAs. This paper presents a new method for controlling the equilibria between exploitation and exploration for various problems. In the method, we extract the stability matrix $\mathbf{S}$ from the exploitation module to evaluate the evolutionary trend (reflected in terms of convergence) of the population under the guidance of the current weights. Specifically, we first divide $\mathbf{S}$ into two parts with the same dimensionality:

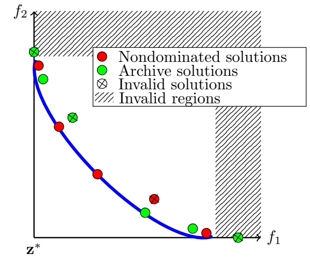
$$\mathbf{V} = \mathbf{S}(:, 1:T); \quad \mathbf{U} = \mathbf{S}(:, T+1:2T), \tag{8}$$

then evaluate the similarity of the both parts as the stability factor δ. Formally,

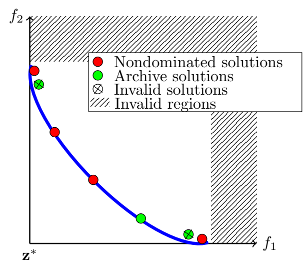
$$\delta = \frac{count\_zero(\Delta)}{T \cdot N}, \tag{9}$$



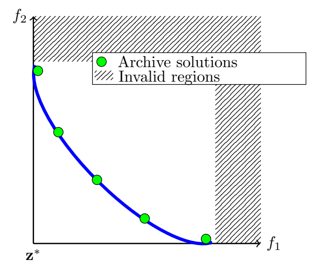
(a) The nondominated solutions in the valid regions are assigned to the archive when the archive is empty.



(b) The valid regions are updated, new nondominated solutions are added to the archive and a dominance check is made.



(c) A maintenance mechanism is used to remove some solutions with poor distribution.



(d) The archive solutions are evenly distributed over the valid regions of the objective space.

Figure 4: Illustration to the whole procedure of an optimization of archive. Here, we assume that the archive size is five.

where

$$\Delta_{k,j} = \begin{cases} 0 & \text{if} \quad | \mathbf{V}_{k,j} - \mathbf{U}_{k,j} | \leq 10^{-3} \\ 1 & \text{otherwise} \end{cases}, \tag{10}$$

in which $i \in \{1, 2, \cdots, N\}$ and $j \in \{1, 2, \cdots, T\}$, respectively. This design is mainly constructed in this way for two reasons: One is to provide sufficient cache space for leaving as little as possible to chance. For example, we not only retain the historical information of the last $2T$ generations in matrix $\mathbf{S}$ but also consider the similarity of the historical information between the intervals of length T. The other reason is to strictly control the stability factor in [0,1] for analysis. The higher the similarity between matrices $\mathbf{V}$ and $\mathbf{U}$ is, the greater the stability factor of the population. When $\delta \approx 1$, it is considered that the current weight-directed population is stable in terms of convergence and that an exploitation episode seems to be over. For simplicity, we say that the EvE model changes from the exploitation module to the exploration module when δ exceeds 0.99.

3.3. The Exploration Module

To obtain a set of well-distributed weights, we only allow the solution with the best fitness for each subproblem (weight) to enter the next generation when δ exceeds 0.99. Moreover, the adjustment of weights is often accompanied by the secondary selection of the population during the exploration phase.

3.3.1. Weight Deletion

When updating the weights, the weight deletion process aims to remove ineffective weights that are not associated with any solutions or that are associated with abnormal solutions. It is worth noting that deletion includes not only the weight itself but also its related information, such as the associated solution and the corresponding row in $\mathbf{S}$. For the first case, Eq. (5) provides reliable information about the exploration module for determining whether a weight has associated solutions. For example, if $\widehat{d}_{k,t}$ of $\mathbf{w}_k$ is equal to 0, this indicates that there are no solutions that belong to weight $\mathbf{w}_k$ in the current generation. Thus, we can use the stability matrix $\mathbf{S}$ to delete all of the weight $\mathbf{w}_k$ with a $\widehat{d}_{k,t}$ of 0.

After solving the first case, we need to further delete the weights associated with abnormal solutions. These abnormal solutions are somewhat misleading with respect to the generation of weights and may be selected throughout the optimization episode because they have the best perpendicular distances from certain weights. In this paper, the median absolute deviation (MAD) is introduced to detect and delete the weights associated with abnormal solutions. The MAD is a robust measure of central tendency that uses the absolute deviation from the median. Specifically, we first normalize all objectives of the selected solutions into the same range based on the maximum and minimum values of each objective. Then, a new virtual solution χ is created, where each objective consists of the median of the corresponding objectives of the selected solutions. Therefore, the MAD is defined as:

$$\text{MAD} = \text{median}\left(\sum_{i}^{m} |f_i(\mathbf{x}_j) - f_i(\chi)|\right), \tag{11}$$

where i and j represent the indexes of the objectives and solutions, respectively. According to Eq. (11), we consider that the greater the sum of absolute variances of all objectives of a solution is, the more likely the solution is to be an abnormal solution. Therefore, solution $\mathbf{x}_j$ can be identified as an abnormal solution if and only if the following conditions are met:

$$\frac{\sum_{i}^{m} |f_i(\mathbf{x}_j) - f_i(\chi)|}{b \cdot \text{MAD}} \geq c, \tag{12}$$

In Eq. (12), if we want to use the MAD as a consistent estimator for the standard deviation, we must use a constant b that is dependent on the distribution. Here, constant b is set to 1.4826 if the underlying distribution is assumed to be normal. As described in [39], if vector A is normally distributed, it can be shown that

$$\lim_{n\to\infty} \mathbb{E}(\text{MAD}(A)) = \sigma\Phi^{-1}(0.75), \tag{13}$$

where $\Phi^{-1}(0.75) \approx 0.6745$ is the 0.75^{th} quantile of the standard normal distribution and is used for consistency. In other words, it is chosen so that $\text{MAD}(A)/0.6745$ (or $1.4826 \cdot \text{MAD}(A)$) is a consistent estimator for the standard deviation σ. In addition, the constant c is the criterion for detecting abnormal solutions, and it is set to 3.

3.3.2. Weight Addition

After the weight deletion operation, the exploration module needs to add some weights whose search areas have not been developed to keep the number of weights unchanged. Weight addition is performed by comparing the distribution density of the solution in the population with the solution in the archive set using the Euclidean distance. More concretely, we first update the valid areas of the current population and keep the members of the archive set in those areas. Then, the distances from the solutions in the archive to their closest solution in the population are calculated. Finally, we only pick the archive member with the largest distance from the population each time, and its search direction is used as the new weight. After each selection step, the distance information of the archive members is updated again.

3.3.3. Weight Generation

Given a new solution from the weight addition operation, the optimal weight of this solution can be easily generated by using a linear mapping method. The main idea of our method is illustrated in Fig. 5(a). First, we assume that a linear hyperplane can be represented as $g_m = \sum_i^{m-1} \alpha_i g_i + \alpha_m$ for an m-dimensional objective space.

Then, considering that these extreme points are on the defined hyperplane, we have the following equation:

$$\mathbf{E}\mathbf{a} = \mathbf{b}, \tag{14}$$

where $\mathbf{E} = (\mathbf{e}_1^{1:m-1}, \mathbf{e}_2^{1:m-1}, \cdots, \mathbf{e}_{m-1}^{1:m-1}, \mathbf{i})^T$ in which $\mathbf{i}$ is an all-ones vector, $\mathbf{a} = (\alpha_1, \alpha_2, \cdots, \alpha_m)^T$ and $\mathbf{b} = (\mathbf{e}_1^m, \mathbf{e}_2^m, \cdots, \mathbf{e}_m^m)^T$. Accordingly, the parameter vector $\mathbf{a}$ can be computed by

$$\mathbf{a} = (\mathbf{E}^T\mathbf{E})^{-1}\mathbf{E}^T\mathbf{b}. \tag{15}$$

Finally, the optimal weight $\mathbf{w}$ of a given $\mathbf{f}(\mathbf{x})$ is defined as

$$\mathbf{w} = \eta\mathbf{f}(\mathbf{x}), \tag{16}$$

where

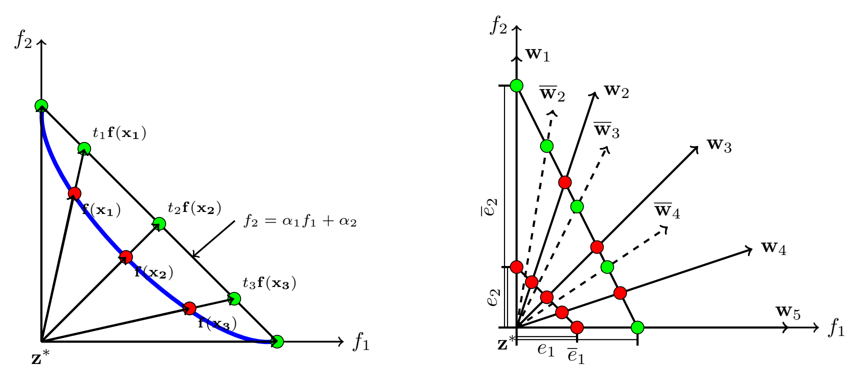
$$\eta = \frac{\alpha_m}{g_m - (\sum_i^{m-1} \alpha_i g_i)}. \tag{17}$$

Moreover, the preserved weights also need to be translated to the above hyperplane. Here, we introduce a new method for translating weights according to the ranges of the objective values of the current population. As shown in Fig. 5(b), assuming that there are two different hyperplanes, the weights evenly distributed over one of them may not be applicable to the other because the ranges of the objective values of both Pareto fronts are different.

However, the uniformly distributed weights on the two Pareto fronts satisfy the following relationship:

$$\begin{bmatrix}\beta_1\\ \beta_2\\ \cdots\\ \beta_m\end{bmatrix} \circ \begin{bmatrix}\mathbf{w}_1^k\\ \mathbf{w}_2^k\\ \cdots\\ \mathbf{w}_m^k\end{bmatrix} = \begin{bmatrix}\overline{\mathbf{w}}_1^k\\ \overline{\mathbf{w}}_2^k\\ \cdots\\ \overline{\mathbf{w}}_m^k\end{bmatrix} \tag{18}$$

where the symbol $\circ$ denotes the elementwise product between two vectors of the same dimension, and β_i is the ratio of the i-th objective ranges of the two Pareto fronts, such as $\beta_1 = \overline{e}_1/e_1$ and $\beta_2 = \overline{e}_2/e_2$ in Fig. 5(b). Through the translation method of Eq. (18), the uniform weights of the two Pareto fronts can be matched one by one.



(a) A linear mapping method (b) A weight translation method

Figure 5: Examples of weight generation.

3.3.4. Weights Replacement

After the above steps are performed, we are able to solve MOPs with "irregular" Pareto fronts, but the proposed method still struggles with MaOPs that have "regular" Pareto fronts. For MaOPs, the simplex-lattice design provides uniform weights [11], yet it suffers from the curse of dimensionality. Subsequently, the multilayer simplex-lattice design [40] controls the number of weights to some extent at the expense of uniformity. These weights may be highly consistent with the Pareto front but not totally uniform. Therefore, to alleviate this problem, a simple weight replacement operation is used to adjust the weights when all of these weights are located on the hyperplane constructed by the current extreme points and are effective for a given problem. First, the distance information of a solution in the population is determined by identifying the minimum distance between the solution and the others in the population. Then, a threshold γ is set to the median of those distances. Furthermore, when the distance information of an archive member is greater than γ, we use the member to replace the solution whose distance information is the smallest in the population, along with the replacement of the corresponding weights. Finally, with each replacement operation completed, the distances of the archive members and the solutions in the population need to be updated separately.

4. Algorithmic Framework

Algorithm 1 gives the main procedure of the DMEA-WUA. First, a set of weights $\mathbf{W}$ is generated by using the two-layer simplex-lattice design [11] in Step 1. Then, the initial population $\mathbf{P}$ is randomly sampled from Ω in Step 2. Step 3 randomly initializes the stability matrix $\mathbf{S}$ where all of the elements are assigned different but relatively large values. Step 4 initializes an empty archive set. During each iteration of the main while loop, we first produce the offspring population $\mathbf{Q}$ by using the recombination operator and merge it with the current population to form a new population $\mathbf{R}$ (Steps 6 and 7). The solutions of $\mathbf{R}$ are assigned to each subproblem according to the perpendicular distances of those solutions from each of the weights (Step 8). After that, we calculate the fitness of each solution for the corresponding subproblem in Step 9. Additionally, the archive set $\mathbf{A}$ is updated by using the current nondominated solutions (Step 10), and the stability factor δ is calculated to balance the EvE model (Step 11). If $\delta > 0.99$ and $g < 0.9 \cdot G$, a single solution is selected for each weight of $\mathbf{W}^*$ to the next population $\mathbf{P}$, and the exploration module is activated (Steps 12-19). When a solution is retained in a subproblem $\mathbf{w}_k$ (Step 14), the corresponding row $\mathbf{S}_{k,:}$ of $\mathbf{S}$ is updated (Step 15). During the exploration operation, Step 17 first deletes the ineffective weights that are not associated with any solutions or are associated with abnormal solutions. Then, some new weights are added to explore the undeveloped search areas in Step 18. After that, the weight replacement procedure is used in Step 19. Otherwise, if $\delta < 0.99$ or $g > 0.9 \cdot G$, we select solutions for the population one by one until the size of $\mathbf{P}$ is equal to N (Steps 20-34). Here, only the weights $\mathbf{W}^*$ associated with solutions are considered (Step 21). In Step 22, a parameter ρ_k is used to count the number of selected solutions in subproblem $\mathbf{w}_k$. When the population size does not reach N, we randomly pick a weight $\mathbf{w}_k$ from those with smallest value of ρ_k from $\mathbf{W}^*$, and select a superior solution $\mathbf{x}$ from $\mathbf{C}_k$ (Steps 24-26). If $\rho_k == 1$, the corresponding row $\mathbf{S}_{k,:}$ of $\mathbf{S}$ is updated (Steps 27-29). After that, $\mathbf{C}_k$, ρ_k and $\mathbf{W}^*$ are updated in sequence (Steps 30-32). The algorithm continues until the number of generations Gen reaches to the maximum number G.

To analyze the computational complexity of the proposed DMEA-WUA, we consider the main steps of one generation process in the main while loop of the algorithm 1. The time complexity of offspring initialization in Step 6 is $\mathcal{O}(DN)$, where D is the number of decision variables and N is the size of the population. The association operation in Step 8 requires $\mathcal{O}(MN^2)$ computations, where M is the number of objectives. Assuming that $L_k = |\mathbf{C}_k|$ and $N = |\mathbf{W}|$, the time complexity of the fitness calculation in Step 9 is $\mathcal{O}(\sum_{k=1}^{N} L_k log^{M-2} L_k)$. In Step 10, the worst time complexity of the archive optimization operation is $\mathcal{O}(2MN^2)$. The calculation of the stability factor δ in Step 11 needs the $\mathcal{O}(NT)$ computations, where T is half of the row length of $\mathbf{S}$. In Steps 13-16, the selection of superior solutions and the update operation for the stability matrix require $\mathcal{O}(\sum_{k=1}^{N} L_k)$ and $\mathcal{O}(2NT)$ computations, respectively. For the weight deletion process in Step 17, $\mathcal{O}(MN)$ computations are needed. Assuming that H ineffective weights are deleted, we should find the same number of undeveloped solutions. The worst time complexity of this process requires $\mathcal{O}(2HMN^2)$ computations. In addition, the worst time complexity of weight replacement is $\mathcal{O}(2MN^2)$. At last, when $\delta < 0.99$ or $g > 0.9 \cdot G$, the computations in the solution selection method require a time complexity of $\mathcal{O}(N^2)$. Therefore, the total computational complexity for one generation is bounded by $\mathcal{O}(2HMN^2)$.

Algorithm 1: The Algorithm DMEA-WUA

Input : Population Size: N; Maximum Generation: G; Maximum Stagnation: $2T$; Generation Counter: $g \leftarrow 0$

Output: Final population: $\mathbf{P}$

$\mathbf{W} \leftarrow$ Generate N initial weights;
$\mathbf{P} \leftarrow$ Randomly generate N solutions;
$\mathbf{S} \leftarrow$ Randomly generate a $N \times 2T$ stablilty matrix;
$\mathbf{A} \leftarrow$ initialize an empty archive set $\emptyset$;
while $g < G$ **do**
    $\mathbf{Q} \leftarrow$ Create offspring population;
    $\mathbf{R} \leftarrow \mathbf{P} \cup \mathbf{Q}$, $\mathbf{P} \leftarrow \emptyset$;
    $(\mathbf{C}_1, \mathbf{C}_2, \cdots, \mathbf{C}_N) \leftarrow$ Associate the solutions of $\mathbf{R}$ to each weight (Section 3.1.1);
    Calculate the fitness value (fit) of each solution $\mathbf{x}$ on the corresponding weight (Section 3.1.1);
    $\mathbf{A} \leftarrow$ Update the archive set (Section 3.1.3);
    $\delta \leftarrow$ Evaluate the stability of the population under the guidance of current weights (Section 3.2);
    if $\delta > 0.99 \wedge g < 0.9 \cdot G$ **then**
        for $k = 0$; $k <= \mid \mathbf{W} \mid$; $k = k + 1$ **do**
            Select a superior solution $\mathbf{x}$ from $\mathbf{C}_k$ (Section 3.1.1);
            $\mathbf{S}_{k,:} \leftarrow$ Update the stability matrix (Section 3.1.2);
        end
        Delete the ineffective weights (Section 3.3.1);
        Generate new weights, and add them into $\mathbf{W}$ (Section 3.3.2, Section 3.3.3);
        Replace the weights (Section 3.3.3, Section 3.3.4);
    else
        $\mathbf{W}^* = \{\mathbf{w}_k : \mathbf{C}_k \neq \emptyset, \mathbf{w}_k \in \mathbf{W}\}$;
        $\rho_k \leftarrow$ count the number of selected solutions for weight $\mathbf{w}_k$;
        while $\mid \mathbf{P} \mid < N$ **do**
            $\overline{\mathbf{W}} = \{\mathbf{w}_k : \text{argmin}_{\mathbf{w}_k \in \mathbf{W}^*} \rho_k\}$;
            $\mathbf{w}_k = \text{random}(\overline{\mathbf{W}})$;
            Select a superior solution $\mathbf{x}$ from $\mathbf{C}_k$ (Section 3.1.1);
            if $\rho_k == 1$ **then**
                $\mathbf{S}_{k,:} \leftarrow$ Update the stability matrix (Section 3.1.2);
            end
            $\mathbf{C}_k \leftarrow \mathbf{C}_k \backslash \mathbf{x}$;
            $\rho_k \leftarrow \rho_k + 1$;
            $\mathbf{W}^* = \{\mathbf{w}_k : \mathbf{C}_k \neq \emptyset, \mathbf{w}_k \in \mathbf{W}^*\}$;
        end
    end
end

Table 1: Properties of test problems.

Problems	Properties	Problems	Properties	Problems	Properties	Problems	Properties
ZDT1	Convex	ZDT2	Concave	ZDT3	Discontinuous	ZDT4	Convex, Multimodal
ZDT5	Convex, Deceptive	ZDT6	Concave, Multimodal, Biased	UF1	Convex, Complex PS	UF2	Convex, Complex PS
UF3	Convex, Complex PS	UF4	Concave, Complex PS	UF5	Linear, Discrete, Complex PS	UF6	Linear, Discontinuous, Complex PS
UF7	Linear, Complex PS	UF8	Concave, Complex PS	UF9	Linear, Discontinuous, Complex PS	UF10	Concave, Complex PS
VNT1	Convex	VNT2	Mixed, Degenerate	VNT3	Mixed, Degenerate	VNT4	Mixed, Degenerate
DTLZ1	Linear, Multimodal	DTLZ2	Concave	DTLZ3	Concave, Multimodal	DTLZ4	Concave, Biased
DTLZ5	Concave, Degenerate	DTLZ6	Concave, Degenerate	DTLZ7	Mixed, Discontinuous, Multimodal	CDTLZ2	Convex
IDTLZ1	Inverted, Linear, Multimodal	IDTLZ2	Inverted, Concave	SDTLZ1	Badly-scaled, Linear, Multimodal	SDTLZ2	Badly-scaled, Concave
WFG1	Mixed, Biased	WFG2	Convex, Discontinuous, Nonseparable	WFG3	Linear, Degenerate, Nonseparable	WFG4	Concave, Multimodal
WFG5	Concave, Deceptive	WFG6	Concave, Nonseparable	WFG7	Concave, Biased	WFG8	Concave, Nonseparale, Biased
WFG9	Cocave, Nonseparable, Deceptive, Biased	Modified IDTLZ1	Inverted, Linear, Multimodal	DTLZ2BZ	Concave	ISDTLZ3	Convex, Multimodal
CSDTLZ4	Cocave, Multimodal	DPF1	Linear, Multimodal, Degenerate	DPF2	Mixed, Disconnected, Degenerate	DPF3	Concave, Biased, Degenerate
DPF4	Concave, Multimodal, Degenerate	DPF5	Linear, Partially Separable, Degenerate				

5. Experimental Results

In this section, we provide the simulation results of the DMEA-WUA on optimization problems with a variety of Pareto fronts [11, 23, 41, 42, 43, 44]. Table 1 summarizes the properties of these MOPs. Moreover, eight state-of-the-art decomposition-based approaches, AdaW [12], iRVEA [45], MOEA/D-AWA [16], MOEA/D-PaS [46], MOEA/D-URAW [33], RPEA [47], RVEAa [31], and DEA-GNG [25], are considered peer algorithms for evaluating the proposed DMEA-WUA. All optimization problems and test algorithms in the paper are implemented on the PlatEMO platform [48].

5.1. Experimental Settings

In the experiments, the population sizes of all the compared MOEAs are set to be the same for each MOP, namely, 100, 105, 220, 240 for 2-, 3-, 10- and 15-objective MOPs, respectively. All algorithms are independently executed 20 times on each problem. The number of generations is set to 1000 for 2-, 3-, 10-, and 15-objective problems. For the genetic operators of each algorithm, the simulated binary crossover (SBX) and the polynomial mutation operators are used for offspring reproduction [48]. For the SBX operator, the crossover probability p_c and the distribution index η_c are set to 1 and 20, respectively. For the polynomial mutation operator, we set the mutation probability $p_m = \frac{1}{d}$ and the distribution index $\eta_m = 20$. In addition, the specific parameter settings of the compared MOEAs are set as recommended in their original papers. For example, in RVEAa and iRVEA, the rate α of changing the penalty function and the frequency f_r of reference vector adaptation are set to 2 and 0.1, respectively. In MOEA/D-AWA, the maximal number of subproblem adjustments nus is set to $0.05N$, the computational resource utilization for the weight adaptation process is set to 20%, and the size of the archive is set as $1.5N$. Moreover, the size of neighborhood T is set to $\lceil 0.1N \rceil$, the probability δ of neighborhood selection is set to 0.9, and the maximum number of solutions replaced by each offspring n_r is set to $\lceil 0.01N \rceil$. Since MOEA/D-AWA and MOEA/D-PaS are the variants of MOEA/D, the general parameters of MOEA/D-PaS are consistent with those of MOEA/D-AWA. In AdaW, the time of updating the weight vectors and the time spent not allowing the update are every 5% of the total generations and the last 10% of the generations, respectively. The maximum capacity of the archive is set to 2N.

Furthermore, we use two performance indicators, the modified inverted generational distance (IGD+) [6] and the hypervolume (HV) [7], to evaluate the performances of these algorithms in this paper. The IGD+ indicator calculates the average modified distance from points in the Pareto front to their closet solution in a Pareto front approximation. Mathematically, suppose that $\mathbf{P}$ is an approximation set and that $\mathbf{P}^*$ is a reference set representing the Pareto front; then, the IGD+ indicator is defined as follows:

$$IGD+(\mathbf{P}) = \frac{1}{|\mathbf{P}^*|}\left(\sum_{\mathbf{x}^*\in\mathbf{P}^*,\mathbf{p}\in\mathbf{P}} d_+^2(\mathbf{x}^*,\mathbf{p})\right)^{1/2}, \tag{19}$$

where $|\mathbf{P}^*|$ denotes the size of $\mathbf{P}^*$ and $d_+(\mathbf{x}^*,\mathbf{p})$ is the minimum $max\{\mathbf{p}-\mathbf{x}^*\}$ representing the modified distance from $\mathbf{x}^*$ to the closest solution in $\mathbf{P}$ with the corresponding point $\mathbf{p}$. It is well-known that a low IGD+ value is desirable, as it indicates that the obtained solution set is close to the Pareto front and has good diversity.

The HV is calculated as the volume of the objective space between the obtained solution set and a specified reference point. In the calculation of the HV, all reference points used to calculate the HV are set to 1.1 times the worst value of each objective in the Pareto optimal solutions. The HV can be calculated in the following way:

$$HV(\mathbf{P},\mathbf{r}) = L\left(\cup_{\mathbf{x}\in\mathbf{P}}[f_1(\mathbf{x}), r_1]\times\cdots\times[f_M(\mathbf{x}), r_M]\right), \tag{20}$$

where $\mathbf{r}$ is a reference point and $\mathbf{x}$ is a solution in the population $\mathbf{P}$. The higher the HV value, the better performance obtained by the algorithm.

We also use two tests, the Wilcoxon rank-sum test [49] and the Friedman rank test [50], to determine whether there are significant differences among the comparative algorithms in terms of their IGD+ and HV results. For the

Table 2: Statistical results (means and standard deviations) of the obtained IGD+ values on the ZDT test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
ZDT1	2	30	2.6638e-3 (2.28e-5)	2.6558e-3 (2.36e-5) ≈	2.9132e-3 (6.49e-5) −	2.6491e-3 (7.27e-5) ≈	3.1829e+1 (3.39e+1) −	2.9254e-3 (7.25e-4) −	8.1830e-3 (6.77e-4) −	2.5166e-2 (1.00e-1) −	3.0152e-3 (7.99e-5) −	
Friedman Ranking			2.2500	1.9500	5.8500	2.1000	8.8500	4.5500	8.1000	4.7500	6.6000	9.0452e-27
ZDT2	2	30	2.5711e-3 (8.08e-6)	2.5927e-3 (2.35e-5) −	2.9208e-3 (8.06e-5) −	3.7237e-3 (3.12e-3) −	5.9113e+1 (3.08e+1) −	2.7771e-3 (2.46e-4) −	1.1392e-2 (1.79e-3) −	2.8374e-1 (1.21e-1) −	2.7500e-3 (6.10e-5) −	
Friedman Ranking			1.4500	2.1500	5.6000	3.9000	8.9500	4.1500	7.1000	7.4000	4.3000	1.4408e-24
ZDT3	2	30	2.8830e-3 (7.49e-5)	2.7364e-3 (5.16e-5) +	3.1846e-3 (1.02e-4) −	2.5409e-2 (5.10e-2) −	3.0809e+1 (3.44e+1) −	3.1050e-3 (9.55e-5) −	7.7599e-3 (7.81e-4) −	4.7739e-2 (6.56e-2) −	3.9086e-3 (4.86e-3) −	
Friedman Ranking			2.8000	1.2000	4.8500	6.2500	8.9500	4.1500	6.9500	7.5500	2.300	1.2789e-27
ZDT4	2	10	2.7851e-3 (1.12e-4)	2.7691e-3 (7.90e-5) ≈	3.0111e-3 (1.49e-4) −	3.6556e-3 (8.10e-4) −	2.0236e+1 (1.94e+1) −	3.3416e-3 (3.29e-4) −	8.7330e-3 (7.84e-4) −	3.4059e-3 (3.04e-4) −	3.2654e-3 (1.85e-4) −	
Friedman Ranking			1.6500	1.6500	3.6000	5.4000	8.9000	5.2500	8.1000	5.4500	5.0000	8.9940e-25
ZDT5	2	80	3.9202e-1 (1.10e-1)	5.0789e-1 (1.04e-1) −	5.0476e-1 (1.03e-1) −	6.8027e-1 (8.94e-2) −	1.1848e+0 (1.45e-1) −	5.3903e-1 (9.76e-2) −	5.8819e-1 (1.01e-1) −	5.4025e-1 (8.57e-2) −	5.1280e-1 (9.67e-2) −	
Friedman Ranking			1.8000	3.8000	4.0750	7.2250	9.0000	4.9000	5.5000	4.7000	4.0000	7.7248e-17
ZDT6	2	10	2.1432e-3 (1.05e-5)	2.1499e-3 (1.29e-5) ≈	2.3090e-3 (3.91e-5) −	2.1859e-3 (7.21e-5) −	5.4665e-1 (2.43e+0) −	2.2485e-3 (2.41e-5) −	6.7832e-3 (6.63e-4) −	2.1352e-3 (5.81e-5) ≈	2.2820e-3 (7.85e-5) −	
Friedman Ranking			2.3000	2.9000	7.4500	4.1500	3.9500	6.3000	8.9500	2.3000	6.7000	2.6189e-23
+/ − / ≈				1/2/3	0/6/0	0/5/1	0/6/0	0/6/0	0/6/0	0/5/1	0/6/0	
Average Ranking			2.0416	2.2750	5.2375	4.8375	8.1000	4.8833	7.4500	5.3580	4.8166	

Table 3: Statistical result (means and standard deviations) of the obtained HV values on the ZDT test instances., where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
ZDT1	2	30	7.2049e-1 (6.65e-5)	7.2052e-1 (6.93e-5) ≈	7.2005e-1 (1.00e-4) −	7.2051e-1 (1.09e-4) ≈	3.0598e-1 (3.48e-1) −	7.1960e-1 (1.55e-3) −	7.1158e-1 (1.38e-3) −	6.9771e-1 (9.86e-2) −	7.2021e-1 (7.79e-5) −	
Friedman Ranking			7.7500	8.3500	4.6500	7.8500	1.1500	4.2500	1.9000	3.5500	5.5500	2.0140e-27
ZDT2	2	30	4.4498e-1 (7.37e-5)	4.4498e-1 (2.22e-4) −	4.4455e-1 (1.25e-4) −	4.4207e-1 (7.32e-3) ≈	1.6330e-2 (7.30e-2) −	4.4363e-1 (6.86e-4) −	4.3433e-1 (1.74e-3) −	1.4402e-1 (1.30e-1) −	4.4492e-1 (6.66e-5) −	
Friedman Ranking			7.4000	8.2000	5.4000	6.0500	1.0500	4.1500	2.9500	2.8000	7.0000	4.1903e-23
ZDT3	2	30	6.0053e-1 (7.87e-4)	5.9965e-1 (2.23e-4) −	5.9902e-1 (2.82e-4) −	6.2164e-1 (6.57e-2) +	2.0587e-1 (2.88e-1) −	5.8588e-1 (1.50e-3) −	5.9578e-1 (5.82e-4) −	6.6739e-1 (7.78e-2) ≈	6.0424e-1 (1.98e-2) +	
Friedman Ranking			7.6500	6.3500	5.0500	4.1000	1.7500	2.9000	3.9500	6.3000	6.9500	1.0732e-14
ZDT4	2	10	7.2021e-1 (2.07e-4)	7.2035e-1 (1.63e-4) +	7.1977e-1 (3.01e-4) −	7.1881e-1 (1.33e-3) −	7.1568e-2 (2.20e-1) −	7.1892e-1 (6.88e-4) −	7.1168e-1 (9.23e-4) −	7.1868e-1 (5.63e-4) −	7.1984e-1 (2.65e-4) −	
Friedman Ranking			8.0000	8.7000	5.9500	4.7000	1.1000	4.3000	1.9000	3.8000	6.5500	1.5469e-26
ZDT5	2	80	8.2555e-1 (1.22e-2)	8.1221e-1 (1.31e-2) −	8.1139e-1 (1.16e-2) −	7.9192e-1 (9.59e-3) −	7.7310e-1 (1.03e-2) −	8.0800e-1 (1.03e-2) −	8.1968e-1 (1.18e-2) ≈	8.0727e-1 (9.61e-3) −	8.1739e-1 (1.28e-2) −	
Friedman Ranking			7.9000	5.5500	5.5750	2.5750	1.2500	4.5000	6.4000	5.0000	6.2500	1.0586e-15
ZDT6	2	10	3.8865e-1 (1.58e-4)	3.8883e-1 (8.95e-5) +	3.8862e-1 (5.84e-5) ≈	3.8885e-1 (1.05e-4) +	3.6913e-1 (8.69e-2) −	3.8809e-1 (3.23e-4) −	3.8243e-1 (9.64e-4) −	3.8889e-1 (8.20e-5) +	3.8865e-1 (1.37e-4) ≈	
Friedman Ranking			4.6000	7.2000	3.9000	7.1000	6.2500	2.4500	1.0500	8.2500	4.2000	1.7810e-22
+/ − / ≈				2/3/1	0/5/1	2/2/2	0/6/0	0/6/0	0/5/1	1/4/1	1/4/1	
Average Ranking			7.2166	7.3916	5.0875	5.3958	2.0916	3.7583	3.0250	4.9500	6.0833	

Wilcoxon rank-sum test, the null hypothesis is rejected at a significance level of 0.05, where "+", "≈" and "−" in the tables indicate that the performance of a compared MOEA is significantly better, statistically similar and significantly worse to that of the proposed DMEA-WUA, respectively. For the Friedman rank test, the null hypothesis is also rejected at a significance level of 0.05.

5.2. Comparisons on the ZDT Test Suite

In this section, we present the performances of the nine algorithms when solving the ZDT test suite in Tables 2 and 3. This test suite can verify the performance of each algorithm in dealing with two-objective problems with different properties, e.g., convex, concave and discontinuous problems.

From these two tables, we find that the DMEA-WUA and AdaW achieved outstanding performances in terms of uniformity and extensiveness. Specifically, the two indicators IGD+ and HV indicate that the DMEA-WUA performs the best on ZDT2 and ZDT5 instances, while AdaW performs best on the ZDT1 and ZDT4 instances. Although the performances of the DMEA-WUA and AdaW on other instances are not optimal, they are very competitive compared to other algorithms in terms of the values obtained from

Table 4: Statistical results (means and standard deviations) of the obtained IGD+ values on the VNT test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
VNT1	3	2	7.5942e-2 (1.53e-3)	7.5832e-2 (1.29e-3) ≈	8.0608e-2 (3.31e-3) −	9.1895e-2 (1.90e-3) −	1.1471e-1 (3.62e-3) −	8.1405e-2 (1.77e-3) −	8.7645e-2 (4.97e-3) −	7.7179e-2 (2.70e-3) ≈	7.9552e-2 (5.09e-3) −	
Friedman Ranking			2.1500	2.2000	4.8500	7.7500	9.0000	5.2000	6.8500	3.0000	4.0000	7.2532e-24
VNT2	3	2	6.8589e-3 (4.19e-4)	6.6321e-3 (3.61e-4) ≈	1.0495e-2 (2.19e-3) −	8.6699e-3 (6.15e-4) −	2.9125e-2 (1.35e-3) −	6.9511e-3 (4.18e-4) ≈	1.5574e-2 (1.82e-3) −	1.2355e-2 (3.65e-3) −	7.3471e-3 (9.55e-4) ≈	
Friedman Ranking			2.4500	1.8000	5.9500	5.1500	9.0000	2.7000	7.8000	6.7000	3.4500	4.0052e-26
VNT3	3	2	1.5395e-2 (1.61e-3)	1.5530e-2 (2.13e-3) ≈	4.7467e-2 (1.12e-2) −	3.4823e-2 (1.42e-2) −	6.5319e-2 (4.13e-3) −	1.5629e-2 (1.32e-3) ≈	4.3135e-2 (1.53e-2) −	3.1726e-2 (7.20e-3) −	2.4547e-2 (3.70e-3) −	
Friedman Ranking			1.8000	2.1500	7.2500	5.9500	8.8000	2.1500	6.6000	5.8500	4.4500	3.1432e-25
VNT4	3	2	1.8071e-1 (1.28e-2)	2.1634e-1 (3.25e-2) −	1.6391e-1 (1.87e-2) +	3.2306e-1 (5.56e-2) −	3.1763e-1 (2.79e-2) −	1.9078e-1 (1.47e-2) −	2.3445e-1 (2.90e-2) −	1.3536e-1 (5.64e-3) +	1.8861e-1 (2.64e-2) ≈	
Friedman Ranking			3.7500	5.4500	2.5000	8.4500	8.4000	4.4000	6.7000	1.0500	4.3000	2.0246e-25
+/ − / ≈				0/1/3	1/3/0	0/4/0	0/4/0	0/2/2	0/4/0	1/2/1	0/2/2	
Average Ranking			2.5375	2.9000	5.1375	6.8250	8.8000	3.6125	6.9875	4.1500	4.0500	

Table 5: Statistical results (means and standard deviations) of the obtained HV values on the VNT test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
VNT1	3	2	3.3823e-1 (2.08e-4)	3.3845e-1 (2.34e-4) +	3.3613e-1 (7.82e-4) −	3.3730e-1 (3.38e-4) −	3.2790e-1 (7.38e-4) −	3.3885e-1 (1.74e-4) +	3.3288e-1 (8.28e-4) −	3.3572e-1 (7.85e-4) −	3.3481e-1 (1.36e-3) −	
Friedman Ranking			7.2000	7.8500	4.5000	5.9500	1.0000	8.9500	2.1000	4.0500	3.4000	3.5217e-29
VNT2	3	2	3.3320e-1 (6.13e-5)	3.3348e-1 (4.57e-5) +	3.3283e-1 (8.11e-4) ≈	2.0295e+0 (2.37e-3) +	2.0140e+0 (2.91e-3) +	1.9928e+0 (2.71e-2) +	3.3185e-1 (3.09e-4) −	1.7635e+0 (2.34e-1) +	3.3318e-1 (1.75e-4) ≈	
Friedman Ranking			3.1500	4.7500	2.9000	8.8000	7.5500	7.1500	1.0000	6.5000	3.2000	7.2532e-24
VNT3	3	2	1.8428e-1 (5.25e-5)	1.8482e-1 (2.24e-5) +	1.8498e-1 (2.76e-5) +	1.8438e-1 (6.09e-5) +	1.8165e-1 (4.73e-4) −	1.8389e-1 (6.83e-5) −	1.8412e-1 (1.32e-4) −	1.8428e-1 (1.35e-4) ≈	1.8495e-1 (2.84e-5) +	
Friedman Ranking			4.1000	7.0000	8.8500	5.7000	1.0000	2.1000	3.4000	4.7000	8.1500	6.5573e-29
VNT4	3	2	2.4075e-1 (2.90e-3)	2.3458e-1 (5.83e-3) −	2.4157e-1 (3.00e-3) ≈	1.8649e-1 (6.71e-3) −	2.0641e-1 (1.85e-3) −	2.0646e-1 (1.91e-3) −	2.3609e-1 (3.62e-3) −	2.0632e-1 (1.90e-3) −	2.3674e-1 (4.86e-3) −	
Friedman Ranking			8.0000	6.2500	8.2000	1.0500	2.9000	2.9000	6.1500	3.1500	6.4000	4.9127e-26
+/ − / ≈				3/1/0	1/1/2	2/2/0	1/3/0	2/2/0	0/4/0	1/2/1	1/2/1	
Average Ranking			5.6125	6.4625	6.1125	5.3750	3.1125	5.2750	3.1625	4.6000	5.2875	

the IGD+ and HV indicators. MOEA/D-AWA has more of an advantage in extensiveness than in uniformity because it obtains three second-best values of the HV indicator, but only obtain one best value for the IGD+ indicator. RVEAa performs best on ZDT6, and has a better performance on ZDT3 in terms of the HV indicator. For the other algorithms, only the HV value of DEA-GNG on the ZDT3 instance exceeds that of our algorithm. Moreover, Friedman's test shows that the DMEA-WUA and AdaW achieve the best performances among all compared algorithms in terms of both the HV and IGD+.

5.3. Comparisons on the VNT Test Suite

This section provides the results of the compared algorithms on the VNT test suite. Compared the ZDT test suite, the problems in VNT have more complicated Pareto front shapes.

Tables 4 and 5 give the average performance of each algorithm in terms of the IGD+ and HV indicators, respectively, when solving the VNT test suite. The DMEA-WUA and AdaW achieve the best results for the IGD+ and HV indicators, respectively, according to Friedman's rank test. For the IGD+ indicator, our algorithm and AdaW obtain the most uniformly distributed populations in the VNT1-3 instances. In addition, it can be seen from the HV indicator that our algorithm also performs very well on the VNT4 instance.

Table 6: Statistical results (means and standard deviations) of the obtained IGD+ values on the UF test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
UF1	2	30	7.8120e-2 (1.78e-2)	8.4369e-2 (2.25e-2) ≈	1.1196e-1 (3.62e-2) −	1.1948e-1 (5.52e-2) −	5.4568e-2 (2.98e-2) +	9.7636e-2 (2.18e-2) −	9.2618e-2 (2.92e-2) ≈	9.9267e-2 (2.62e-2) −	7.3878e-2 (1.33e-2) ≈	
Friedman Ranking			4.0500	4.8000	6.4500	6.5000	1.9500	6.4000	5.0500	6.1500	3.6500	1.3197e-08
UF2	2	30	3.7364e-2 (9.97e-3)	3.9817e-2 (1.41e-2) ≈	4.0251e-2 (9.64e-3) ≈	3.4584e-2 (1.18e-2) ≈	2.8511e-2 (1.59e-2) +	4.3142e-2 (1.23e-2) ≈	5.0942e-2 (1.30e-2) −	5.0358e-2 (1.18e-2) −	3.5434e-2 (7.55e-3) ≈	
Friedman Ranking			4.4000	4.9500	5.1500	3.8500	2.0000	5.9000	7.2000	7.3000	4.2500	6.6083e-10
UF3	2	30	1.7692e-1 (2.52e-2)	1.7714e-1 (3.00e-2) ≈	2.0032e-1 (4.83e-2) −	2.1638e-1 (4.35e-2) −	1.2939e-1 (1.30e-1) +	1.9180e-1 (2.94e-2) ≈	1.9862e-1 (2.33e-2) −	1.9297e-1 (4.19e-2) ≈	1.8629e-1 (2.95e-2) ≈	
Friedman Ranking			4.2500	4.5000	5.8000	7.1500	1.9000	5.2500	5.8000	5.6000	4.7500	3.9691e-07
UF4	2	30	4.4625e-2 (1.05e-3)	4.3904e-2 (2.22e-3) ≈	4.3233e-2 (1.26e-3) +	5.4044e-2 (2.94e-3) −	8.5143e-2 (7.19e-3) −	4.9833e-2 (2.57e-3) −	6.5470e-2 (5.75e-3) −	5.0577e-2 (2.25e-3) −	4.3066e-2 (1.06e-3) +	
Friedman Ranking			3.5000	2.5500	2.0500	6.7500	9.0000	5.3500	7.9500	5.8000	2.0500	2.7550e-27
UF5	2	30	2.6155e-1 (7.07e-2)	2.9703e-1 (1.05e-1) ≈	2.8212e-1 (6.22e-2) ≈	4.6510e-1 (1.08e-1) −	7.7600e-1 (1.56e-1) −	3.8758e-1 (9.54e-2) −	3.2445e-1 (9.23e-2) −	3.7633e-1 (1.12e-1) −	2.7122e-1 (6.16e-2) ≈	
Friedman Ranking			2.6500	3.9500	3.4000	7.0000	9.0000	5.8500	4.3000	5.5000	3.3500	7.4666e-16
UF6	2	30	1.4428e-1 (6.65e-2)	1.7372e-1 (9.41e-2) ≈	2.4083e-1 (1.52e-1) ≈	4.0043e-1 (2.10e-1) −	3.9449e-1 (7.21e-1) −	1.9163e-1 (8.88e-2) −	1.7779e-1 (8.79e-2) ≈	2.7001e-1 (1.86e-1) −	1.5936e-1 (8.41e-2) ≈	
Friedman Ranking			3.4000	4.3000	5.2000	7.1000	5.9500	5.3500	4.3000	5.8500	3.5500	1.1104e-04
UF7	2	30	8.4118e-2 (9.20e-2)	9.6177e-2 (1.03e-1) ≈	2.0476e-1 (1.38e-1) −	1.9450e-1 (1.40e-1) −	2.2185e-2 (7.67e-3) +	2.1526e-1 (1.23e-1) −	1.7380e-1 (9.70e-2) −	1.8850e-1 (1.11e-1) −	9.0210e-2 (9.11e-2) ≈	
Friedman Ranking			3.8500	3.8000	6.8000	5.8000	1.4000	6.5000	6.1000	6.6000	4.1500	4.3755e-12
UF8	3	30	2.3618e-1 (9.58e-3)	2.2081e-1 (4.40e-2) ≈	2.2410e-1 (4.30e-2) +	1.9769e-1 (1.15e-1) ≈	2.3823e-1 (5.26e-2) −	2.1378e-1 (4.08e-2) ≈	2.3418e-1 (4.53e-3) +	2.3236e-1 (1.56e-2) ≈	2.3573e-1 (2.93e-2) +	
Friedman Ranking			4.3500	4.9500	5.7500	3.8000	7.9500	4.3000	3.5500	3.8500	6.5000	2.2553e-07
UF9	3	30	2.4460e-1 (7.45e-2)	2.0311e-1 (6.78e-2) ≈	2.4101e-1 (6.54e-2) ≈	2.4958e-1 (1.75e-2) ≈	2.8953e-1 (1.24e-1) ≈	2.4384e-1 (7.43e-2) ≈	4.1469e-1 (6.29e-2) −	3.1862e-1 (7.81e-2) −	2.4221e-1 (6.56e-2) ≈	
Friedman Ranking			4.5500	2.8500	4.3500	4.6000	4.7500	4.1500	8.6500	6.6000	4.5000	5.7543e-10
UF10	3	30	3.1574e-1 (9.79e-2)	3.6877e-1 (1.03e-1) −	3.4386e-1 (1.04e-1) ≈	4.2668e-1 (1.04e-1) −	8.1002e-1 (1.48e-1) −	3.8698e-1 (1.11e-1) −	4.2917e-1 (1.21e-1) −	4.2754e-1 (1.03e-1) −	3.5648e-1 (9.57e-2) ≈	
Friedman Ranking			3.1500	4.4000	3.8500	5.6500	9.0000	4.4500	5.3500	5.3500	3.8000	1.3687e-10
+/ − / ≈				0/1/9	2/3/5	0/7/3	4/5/1	0/6/4	1/7/2	0/8/2	2/0/8	
Average Ranking			3.8150	4.1050	4.8800	5.8200	5.2900	5.3500	5.8250	5.8600	4.0550	

iRVEA achieves good performance on the VNT test suite because it obtains two of the best results in terms of the HV.

5.4. Comparisons on the UF Test Suite

In this section, we compare the performances of the nine algorithms on the UF test suite. As shown in Tables 6 and 7, the DMEA-WUA, AdaW and DEA-GNG perform best on the UF test suite in terms of the IGD+ and HV indicators. The DMEA-WUA and AdaW are only significantly different for IGD+ on UF9 and for HV on UF10, while the DMEA-WUA and DEA-GNG are significantly different on the UF4 and UF8 instances. MOEA/D-AWA, MOEA/D-URAW, RPEA and RVEAa are obviously inferior to the DMEA-WUA because they perform significantly worse than our algorithm on most problems. Specifically, MOEA/D-AWA obtains the best results on the UF8 instance. When solving the UF8 instance, MOEA/D-URAW and RPEA surpass our algorithm in terms of the IGD+ and HV indicators, respectively. RVEAa does not significantly outperform our algorithm on the whole UF test suite. MOEA/D-PaS performs very well on the UF test suite because it achieved the best results on four of the problems, e.g., UF1, UF2, UF3 and UF6. The iRVEA algorithm performs better than our algorithm on the UF4 and UF8 instances but is significantly worse than our algorithm on UF1, UF3 and UF7. In short, the DMEA-UWA is also very competitive in the UF test suite compared to the other eight algorithms. We can see from these two tables that our algorithm has the highest ranking in terms of Friedman's rank test.

Table 7: Statistical results (means and standard deviations) of the obtained HV values on the UF test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
UF1	2	30	5.8789e-1 (2.61e-2)	5.8254e-1 (3.89e-2) ≈	5.3964e-1 (5.45e-2) −	5.4084e-1 (7.99e-2) ≈	6.3736e-1 (4.44e-2) +	5.6067e-1 (2.27e-2) −	5.6576e-1 (3.46e-2) ≈	5.5656e-1 (4.05e-2) −	5.9341e-1 (2.13e-2) ≈	
Friedman Ranking			6.0500	5.1500	3.2500	4.1000	8.2000	3.7000	4.5500	4.0000	6.0000	2.6086e-08
UF2	2	30	6.7467e-1 (1.05e-2)	6.7198e-1 (1.45e-2) ≈	6.7168e-1 (1.13e-2) ≈	6.7828e-1 (1.26e-2) ≈	6.8600e-1 (1.69e-2) +	6.6789e-1 (1.32e-2) ≈	6.5479e-1 (1.29e-2) −	6.5998e-1 (1.17e-2) −	6.7683e-1 (8.75e-3) ≈	
Friedman Ranking			5.6000	5.1000	5.0000	6.1500	8.1000	4.2000	2.1500	2.8000	5.9000	9.4457e-12
UF3	2	30	4.4166e-1 (3.77e-2)	4.3343e-1 (3.87e-2) ≈	4.0854e-1 (5.74e-2) −	4.1531e-1 (3.85e-2) −	5.4709e-1 (1.32e-1) +	4.1695e-1 (3.79e-2) ≈	4.0965e-1 (3.47e-2) −	4.1871e-1 (5.38e-2) ≈	4.2666e-1 (3.71e-2) ≈	
Friedman Ranking			5.3500	5.0000	4.0000	4.0500	8.2000	4.6000	4.0500	4.5500	5.2000	1.6231e-05
UF4	2	30	3.8569e-1 (1.55e-3)	3.8458e-1 (3.65e-3) ≈	3.8725e-1 (1.57e-3) +	3.6667e-1 (5.93e-3) −	3.2614e-1 (8.76e-3) −	3.7275e-1 (4.35e-3) −	3.4685e-1 (1.17e-2) −	3.7702e-1 (2.69e-3) −	3.8730e-1 (1.42e-3) +	
Friedman Ranking			7.0000	6.8500	8.0000	3.1500	1.0500	4.1000	2.0000	4.8000	8.0500	1.6204e-27
UF5	2	30	2.3677e-1 (5.52e-2)	2.2396e-1 (7.12e-2) ≈	2.1795e-1 (8.01e-2) ≈	9.7611e-2 (7.14e-2) −	5.9965e-3 (1.97e-2) −	1.4871e-1 (7.39e-2) −	2.0727e-1 (8.08e-2) ≈	1.4820e-1 (7.68e-2) −	2.3280e-1 (5.55e-2) ≈	
Friedman Ranking			7.0500	6.2250	6.2500	3.1250	1.1500	4.2750	6.0500	4.2750	6.6000	3.1012e-14
UF6	2	30	3.1461e-1 (6.90e-2)	2.9524e-1 (7.27e-2) ≈	2.6573e-1 (1.05e-1) ≈	1.8782e-1 (1.22e-1) −	2.6319e-1 (8.18e-2) −	2.8131e-1 (6.48e-2) −	3.0159e-1 (6.51e-2) ≈	2.5494e-1 (9.70e-2) −	3.0631e-1 (7.79e-2) ≈	
Friedman Ranking			6.4000	5.7500	4.7500	3.3500	3.9000	4.6500	5.6500	4.3500	6.2000	0.0025
UF7	2	30	4.7561e-1 (9.11e-2)	4.6563e-1 (1.03e-1) ≈	3.5058e-1 (1.27e-1) −	3.6708e-1 (1.36e-1) −	5.4955e-1 (1.24e-2) +	3.4604e-1 (1.20e-1) −	3.8302e-1 (9.34e-2) −	3.6630e-1 (1.09e-1) −	4.6936e-1 (9.19e-2) ≈	
Friedman Ranking			6.2000	6.1000	3.1500	4.1000	8.6000	3.5500	4.0500	3.3500	5.9000	4.1922e-12
UF8	3	30	3.3732e-1 (7.47e-3)	3.5341e-1 (3.60e-2) ≈	3.4371e-1 (2.08e-2) ≈	3.5599e-1 (8.56e-2) +	3.3058e-1 (3.21e-2) −	3.5353e-1 (2.35e-2) +	3.3734e-1 (1.95e-3) ≈	3.3626e-1 (7.63e-3) ≈	3.2761e-1 (1.29e-2) −	
Friedman Ranking			5.1000	5.3000	4.9000	6.8500	2.2500	7.0000	5.5000	5.7000	2.4000	4.1569e-10
UF9	3	30	5.1427e-1 (7.20e-2)	6.0671e-1 (5.72e-2) +	5.1747e-1 (6.24e-2) ≈	5.5202e-1 (2.76e-2) ≈	5.0821e-1 (1.42e-1) ≈	5.4590e-1 (7.56e-2) ≈	3.3186e-1 (5.64e-2) −	4.3507e-1 (7.90e-2) −	5.1810e-1 (6.31e-2) ≈	
Friedman Ranking			4.8500	7.7500	5.1500	6.1000	5.9000	6.1000	1.3500	2.8500	4.9500	2.1893e-13
UF10	3	30	2.2043e-1 (9.69e-2)	1.8015e-1 (8.15e-2) ≈	1.9677e-1 (9.44e-2) ≈	1.5173e-1 (7.99e-2) −	1.5159e-2 (1.48e-2) −	1.7336e-1 (8.75e-2) ≈	1.6722e-1 (1.01e-1) ≈	1.4402e-1 (7.13e-2) −	1.7387e-1 (5.74e-2) ≈	
Friedman Ranking			6.6500	5.4500	6.2000	4.6000	1.0000	5.4500	5.0000	4.7500	5.9000	1.3750e-09
+/ − / ≈				1/0/9	1/3/6	1/6/3	4/5/1	1/5/4	0/5/5	0/8/2	1/1/8	
Average Ranking			6.0250	5.8675	5.0650	4.5575	4.8350	4.7625	4.0350	4.1425	5.7100	

Table 8: Statistical results (means and standard deviations) of the obtained IGD+ values on the DPF test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
DPF1	3	11	1.6449e-3 (2.43e-4)	1.5445e-3 (1.55e-4) ≈	1.4239e-3 (4.29e-5) +	1.7018e-3 (2.24e-4) ≈	6.1631e-3 (4.31e-4) −	1.6398e-3 (2.55e-4) ≈	3.1749e-3 (2.94e-4) −	2.8359e-3 (9.70e-4) −	1.8283e-3 (4.29e-4) ≈	
	Friedman Ranking		3.7000	3.0000	1.6000	4.4000	8.9500	3.7500	7.8500	7.0500	4.7000	2.0572e-23
	10	11	1.0540e-3 (2.51e-4)	5.0710e-2 (2.22e-1) ≈	1.8562e-1 (8.26e-1) ≈	3.4881e-2 (4.92e-2) −	2.4185e+0 (4.63e+0) −	2.4763e-1 (1.10e+0) ≈	1.1173e-1 (4.92e-1) −	1.2144e-1 (3.60e-2) −	2.0330e-1 (9.04e-1) ≈	
	Friedman Ranking		2.8500	2.9000	3.7000	6.8000	9.0000	3.4500	5.8000	7.7000	2.8000	2.7937e-22
	15	11	9.3121e-4 (1.89e-4)	9.3320e-4 (8.70e-5) ≈	1.0071e-3 (1.05e-4) −	9.1015e-3 (1.83e-2) −	7.8929e-1 (1.50e+0) −	9.8498e-4 (1.34e-4) −	1.7561e-3 (1.53e-4) −	1.6725e-1 (4.68e-2) −	9.0785e-4 (5.53e-5) ≈	
	Friedman Ranking		2.2000	2.8500	4.2000	6.9500	9.0000	3.2000	6.0500	8.0000	2.5500	3.2652e-26
DPF2	3	21	1.1218e-2 (2.51e-3)	9.7503e-3 (2.65e-4) +	2.5455e-2 (9.61e-3) −	7.9585e-2 (2.99e-1) −	6.2499e+0 (1.51e+1) −	1.0411e-2 (3.40e-4) ≈	2.2584e-1 (4.85e-1) −	1.0608e-1 (2.94e-1) −	1.0372e-2 (5.77e-4) ≈	
	Friedman Ranking		3.4500	1.2500	6.3500	5.2500	8.7500	2.9500	7.1000	7.3000	2.6000	4.0826e-26
	10	21	1.0542e-2 (5.08e-4)	1.0054e-2 (3.43e-3) +	4.0783e-2 (1.06e-2) −	7.5607e-2 (1.91e-1) −	1.1870e+1 (2.74e+1) −	1.0570e-2 (1.89e-3) −	6.1754e-1 (1.05e+0) −	6.7564e-1 (7.82e-1) −	1.1760e-2 (3.63e-3) ≈	
	Friedman Ranking		3.3000	1.2500	6.1500	5.9500	8.6500	2.6000	6.2500	7.8000	3.0500	8.8927e-26
	15	21	1.2569e-2 (9.10e-4)	1.1130e-2 (1.10e-4) +	2.0594e-1 (6.87e-1) −	5.4144e-2 (2.82e-2) −	6.6565e+0 (1.89e+1) −	1.1923e-2 (1.96e-4) +	4.9685e-1 (1.13e+0) −	1.7615e+0 (1.48e+0) −	1.2674e-2 (1.02e-3) ≈	
	Friedman Ranking		3.2000	1.1000	6.2500	6.0500	8.5000	2.5500	5.8000	8.2500	3.3000	6.9606e-27
DPF3	3	7	2.5249e-3 (7.72e-5)	2.6068e-3 (4.92e-5) −	5.1357e-3 (7.22e-4) −	6.7094e-3 (2.69e-4) −	2.0542e-1 (2.43e-1) −	2.5455e-3 (1.99e-4) ≈	3.6777e-2 (1.14e-1) −	5.2268e-2 (1.28e-1) −	4.8998e-3 (1.50e-3) −	
	Friedman Ranking		1.5500	2.8000	5.0000	6.6500	8.6500	1.6500	8.1000	5.7500	4.8500	2.6855e-27
	10	14	2.1286e-3 (6.85e-4)	2.1971e-3 (8.62e-5) ≈	4.2722e-3 (5.93e-4) −	3.6763e-2 (5.89e-2) −	4.0849e-1 (1.33e-1) −	1.2701e-3 (1.69e-4) +	5.5840e-3 (9.68e-4) −	1.7962e-2 (3.09e-3) −	2.1821e-3 (1.16e-4) ≈	
	Friedman Ranking		2.8000	3.1500	5.1000	7.9500	9.0000	1.0500	5.8500	7.0500	3.0500	5.0747e-29
	15	19	1.7002e-3 (2.11e-4)	2.6076e-3 (1.11e-4) −	4.3832e-3 (4.34e-4) −	2.6684e-2 (3.39e-3) −	5.5183e-1 (1.94e-1) −	1.1520e-3 (1.60e-5) +	4.3411e-3 (9.26e-4) −	2.3753e-2 (4.32e-2) −	2.3594e-3 (1.43e-4) −	
	Friedman Ranking		1.9500	3.9000	5.6500	7.9500	9.0000	1.0500	5.3500	7.0500	3.1000	4.6117e-30
DPF4	3	11	2.7014e-3 (9.50e-5)	2.8035e-3 (6.88e-4) ≈	6.9935e-3 (2.78e-3) −	1.1864e-2 (1.50e-3) −	2.1320e+1 (3.53e+1) −	2.8000e-3 (1.13e-4) −	7.3163e-2 (1.95e-2) −	9.7749e-3 (2.11e-3) −	3.3023e-2 (6.18e-2) −	
	Friedman Ranking		1.8000	1.9500	4.0000	5.9000	8.3500	2.4000	8.5000	5.3500	6.7500	1.3037e-27
	10	11	1.2375e-3 (3.40e-5)	2.1913e-1 (9.70e-1) −	4.6154e-2 (1.36e-1) −	3.0570e-2 (5.60e-2) −	2.5001e+1 (2.18e+1) −	3.2817e-1 (1.46e+0) −	2.9874e+0 (1.33e+1) −	2.2394e-2 (3.19e-3) −	2.1623e-2 (4.62e-2) −	
	Friedman Ranking		1.0000	3.2000	5.6000	6.1000	9.0000	2.3000	5.7000	7.1000	5.0000	2.0580e-24
	15	11	1.1186e-3 (8.10e-6)	2.6887e-3 (7.50e-4) −	1.8253e-2 (3.04e-3) −	1.5234e-2 (3.47e-3) −	2.9105e+4 (2.25e+4) −	1.2621e-3 (3.78e-5) −	1.0372e-2 (6.91e-3) −	2.4016e-2 (4.13e-3) −	1.5692e-2 (1.18e-2) −	
	Friedman Ranking		1.0000	3.0500	6.3500	5.7500	9.0000	2.0000	4.8000	7.4000	5.6500	7.4199e-27
DPF5	3	12	2.2258e-2 (7.16e-4)	2.1564e-2 (4.91e-4) +	2.5069e-2 (2.04e-3) −	2.1445e-2 (4.81e-4) +	4.0748e-2 (7.05e-4) −	2.0568e-2 (6.32e-4) +	4.3347e-2 (3.71e-3) −	2.2122e-2 (1.02e-3) ≈	2.9262e-2 (4.86e-3) −	
	Friedman Ranking		4.3500	3.0000	6.2000	2.6500	8.2000	1.4000	8.8000	3.8000	6.6000	1.2135e-26
	10	19	2.5149e-1 (1.31e-2)	2.5807e-1 (8.42e-2) −	2.2445e-1 (1.66e-2) +	3.0723e-1 (1.02e-1) ≈	6.7110e-1 (1.29e-1) −	2.6765e-1 (1.13e-2) −	2.1881e-1 (6.38e-3) +	2.4940e-1 (1.54e-2) ≈	2.6422e-1 (1.84e-2) −	
	Friedman Ranking		5.2500	4.0000	2.1000	5.4000	9.0000	6.5000	1.6000	5.0500	6.1000	8.0649e-20
	15	24	3.8622e-1 (2.10e-2)	3.1942e-1 (1.95e-2) +	3.2916e-1 (5.62e-3) +	3.7002e-1 (4.67e-2) ≈	8.2712e-1 (2.91e-4) −	3.7793e-1 (2.31e-2) ≈	3.0753e-1 (4.36e-3) +	3.2448e-1 (6.20e-3) +	3.0719e-1 (7.79e-3) +	
	Friedman Ranking		7.2500	3.1500	4.5000	6.2500	9.0000	6.7500	1.9000	4.1500	2.0500	4.4446e-24
+/ − / ≈				5/5/5	3/11/1	1/11/3	0/15/0	4/6/5	2/13/0	1/12/2	1/7/7	
Average Ranking			3.0433	2.7033	4.8500	6.0000	8.8033	2.9066	5.9633	6.5866	4.1433	

Table 9: Statistical results (means and standard deviations) of the obtained HV values on the DPF test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
DPF1	3	11	2.7199e-1 (6.04e-4)	2.7233e-1 (3.73e-4) ≈	2.7278e-1 (1.19e-4) +	2.7208e-1 (5.38e-4) ≈	2.6401e-1 (5.83e-4) −	2.7200e-1 (6.06e-4) ≈	2.6887e-1 (6.84e-4) −	2.7010e-1 (1.40e-3) −	2.7174e-1 (1.04e-3) ≈	
	Friedman Ranking		5.8000	6.8000	8.8500	5.9500	1.0000	5.7500	2.1500	3.0500	5.6500	3.8152e-24
	10	11	6.0086e-5 (6.56e-7)	5.7107e-5 (1.35e-5) ≈	5.7113e-5 (1.35e-5) ≈	5.1572e-5 (1.22e-5) −	2.2431e-7 (1.90e-7) −	5.7169e-5 (1.35e-5) ≈	5.6204e-5 (1.32e-5) −	2.4201e-5 (9.19e-6) −	5.7307e-5 (1.35e-5) ≈	
	Friedman Ranking		7.0000	6.3750	6.3750	3.2500	1.1250	6.7750	4.3250	2.3000	7.4750	2.1726e-21
	15	11	1.0014e-7 (2.16e-9)	9.9646e-8 (2.40e-9) ≈	9.9362e-8 (1.92e-9) ≈	9.1555e-8 (1.27e-8) −	1.2406e-9 (3.22e-9) −	9.8949e-8 (2.37e-9) ≈	9.6763e-8 (2.06e-9) −	1.5622e-8 (1.45e-8) −	9.9659e-8 (1.41e-9) ≈	
	Friedman Ranking		7.2500	6.8000	6.6000	3.7000	1.0000	6.3000	4.5500	2.0000	6.8000	6.2148e-21
DPF2	3	21	9.1273e-2 (3.01e-4)	9.1470e-2 (8.87e-5) +	9.1038e-2 (3.35e-4) ≈	8.8727e-2 (1.16e-2) ≈	7.3515e-2 (3.36e-2) −	9.1354e-2 (7.30e-5) ≈	8.2434e-2 (1.86e-2) −	8.7124e-2 (1.14e-2) −	9.1460e-2 (5.94e-5) +	
	Friedman Ranking		5.6000	8.2000	5.1500	5.7500	1.4500	5.8000	2.4000	2.6000	8.0500	6.7889e-23
	10	21	1.1330e-4 (7.27e-7)	1.1345e-4 (9.28e-7) ≈	1.1371e-4 (1.32e-6) +	1.0935e-4 (3.44e-6) −	3.6618e-5 (2.42e-5) −	1.1318e-4 (8.41e-7) ≈	8.7490e-5 (4.56e-5) ≈	7.8723e-5 (3.01e-5) −	1.1369e-4 (1.42e-6) +	
	Friedman Ranking		6.0500	6.4500	7.4000	3.2500	1.3500	6.2500	4.8000	2.1500	7.3000	1.1625e-19
	15	21	2.0795e-6 (1.79e-8)	2.0932e-6 (2.22e-8) ≈	2.0001e-6 (4.67e-7) −	1.8557e-6 (1.65e-7) −	7.4960e-7 (4.41e-7) −	2.0887e-6 (1.73e-8) ≈	1.7780e-6 (7.60e-7) ≈	1.2961e-6 (7.63e-7) −	2.1011e-6 (1.90e-8) +	
	Friedman Ranking		5.4500	6.8000	7.2000	3.0500	1.3500	6.3500	5.6000	2.0500	7.1500	5.8102e-20
DPF3	3	7	4.0908e-1 (2.72e-4)	4.0955e-1 (7.94e-5) +	4.0775e-1 (6.69e-4) −	4.0621e-1 (2.56e-4) −	2.2986e-1 (1.97e-1) −	4.0911e-1 (1.87e-4) ≈	3.8594e-1 (8.89e-2) −	3.6542e-1 (1.09e-1) −	4.1010e-1 (4.10e-4) +	
	Friedman Ranking		6.6000	8.1500	4.4000	2.9500	1.1500	6.4500	2.3500	4.2000	8.7500	3.9908e-28
	10	14	3.9757e-6 (4.15e-8)	3.9827e-6 (4.92e-9) ≈	3.9529e-6 (1.27e-8) ≈	3.2790e-6 (8.63e-7) −	6.6552e-7 (7.78e-7) −	4.0191e-6 (4.48e-9) +	4.0006e-6 (8.59e-9) ≈	3.6090e-6 (1.09e-7) −	3.9910e-6 (5.00e-9) ≈	
	Friedman Ranking		6.3000	5.5500	4.5500	2.1500	1.0500	9.0000	7.2000	2.8000	6.4000	6.1255e-27
	15	19	9.2242e-11 (2.06e-13)	9.1196e-11 (1.15e-13) −	9.0801e-11 (2.40e-13) −	8.0006e-11 (1.35e-12) −	6.0247e-12 (1.32e-11) −	9.2654e-11 (8.92e-14) +	9.2246e-11 (9.93e-14) ≈	8.2017e-11 (1.93e-11) −	9.1668e-11 (1.89e-13) −	
	Friedman Ranking		7.6000	4.8500	4.1500	2.0500	1.0000	8.9000	7.5000	2.9500	6.0000	6.8207e-30
DPF4	3	11	7.7002e-1 (7.16e-4)	7.7255e-1 (7.20e-4) +	7.7256e-1 (3.56e-3) +	7.6201e-1 (6.14e-3) −	4.9486e-1 (3.73e-1) −	7.7126e-1 (6.50e-4) +	7.5499e-1 (4.28e-3) −	7.6609e-1 (3.56e-3) −	7.2413e-1 (8.20e-2) ≈	
	Friedman Ranking		5.7500	7.9000	8.3000	3.3000	2.5000	6.7500	2.0000	4.3000	4.2000	1.1545e-20
	10	11	1.1839e-4 (8.22e-8)	1.1214e-4 (2.64e-5) −	1.0302e-4 (2.44e-5) −	9.0786e-5 (2.66e-5) −	0.0000e+0 (0.00e+0) −	1.1245e-4 (2.65e-5) ≈	1.1132e-4 (2.62e-5) −	1.0133e-4 (3.96e-6) −	9.9791e-5 (2.67e-5) −	
	Friedman Ranking		8.6000	6.7250	4.4000	3.0000	1.0750	8.1250	5.6250	3.1000	4.3500	5.0775e-25
	15	11	9.3391e-9 (5.99e-12)	9.2723e-9 (5.39e-12) −	8.3320e-9 (1.65e-10) −	8.8372e-9 (1.08e-10) −	3.6491e-10 (1.63e-9) −	9.3363e-9 (4.97e-12) ≈	9.2287e-9 (4.58e-11) −	7.4380e-9 (3.76e-10) −	7.7246e-9 (1.81e-9) −	
	Friedman Ranking		8.6500	6.8500	3.4500	4.8000	1.0500	8.3500	6.1500	2.1500	3.5500	3.6598e-29
DPF5	3	12	5.1192e-1 (1.28e-3)	5.1275e-1 (1.39e-3) +	5.0500e-1 (2.66e-3) −	5.1491e-1 (4.90e-4) +	5.0221e-1 (8.97e-4) −	5.1426e-1 (5.61e-4) +	4.8165e-1 (5.64e-3) −	5.1190e-1 (9.96e-4) ≈	4.9309e-1 (5.13e-3) −	
	Friedman Ranking		5.9000	6.6500	3.8000	8.6500	3.2000	8.1000	1.0000	5.7000	2.0000	1.1975e-28
	10	19	9.3944e-1 (3.06e-3)	8.9349e-1 (1.72e-1) −	9.5757e-1 (1.79e-2) +	8.4207e-1 (1.62e-1) ≈	2.5898e-1 (1.92e-1) −	9.1856e-1 (1.32e-2) −	9.1610e-1 (8.79e-3) −	9.5404e-1 (2.88e-2) +	8.8352e-1 (1.55e-2) −	
	Friedman Ranking		5.9000	6.1000	8.3000	4.7500	1.0000	4.4500	4.1500	7.7000	2.6500	5.3063e-21
	15	24	9.3394e-1 (2.86e-2)	9.7401e-1 (6.77e-3) +	9.9124e-1 (1.02e-3) +	8.8572e-1 (4.98e-2) −	1.1631e-1 (8.03e-4) −	9.4182e-1 (1.88e-2) ≈	9.6533e-1 (1.93e-3) +	9.6596e-1 (1.53e-2) +	9.6626e-1 (5.48e-3) +	
	Friedman Ranking		3.4000	7.4000	9.0000	2.4500	1.0000	3.9000	5.7000	5.9000	6.2500	1.8399e-25
+/ − / ≈				5/4/6	5/6/4	1/11/3	0/15/0	4/1/10	1/10/4	2/12/1	5/5/5	
Average Ranking			6.3900	6.7733	6.1283	3.9366	1.3533	6.7500	4.3666	3.5300	5.7716	

5.5. Comparisons on the DPF Test Suite

Here we provide the IGD+ and HV results of the compared algorithms for the DPF test suite. This test suite mainly verifies the performance of the algorithm on degenerate multiobjective problems. As shown in Tables 8 and 9, the results obtained by the DMEA-WUA for the IGD+ indicator are more advantageous than those obtained for the HV indicator. For the DPF1 instances, our algorithm and DEA-GNG achieve the ideal results for both the IGD+ and HV indicators under high-dimensional objectives, while iRVEA is outstanding with low-dimensional objectives. For the DPF2 instances, the best performing algorithm is AdaW, followed by the DEA-GNG, DMEA-WUA and MOEA/D-URAW algorithms. These four algorithms are significantly better than the others. AdaW and DEA-GNG obtain the best results in terms of the IGD+ and HV indicators, respectively. Our algorithm can also obtain good populations when optimizing these problems because the performance of our algorithm is not significantly different from that of DEA-GNG in terms of the IGD+ indicator. In addition, the results for the IGD+ and HV indicators show that the performances of our algorithm and MOEA/D-URAW are not much different on the DPF2 problems. For the DPF3 instances, our algorithm is only slightly inferior to MOEA/D-URAW. Especially in terms of the IGD+ indicator, the results obtained by our algorithm on DPF3 are ranked in the top two when compared to those of the other eight algorithms. For the DPF4 instances, our algorithm is superior to the other algorithms in terms of the IGD+ and comparable to MOEA/D-URAW in terms of the HV. Finally, for the DPF5 instances, except for MOEA/D-PaS, the numerical performance gaps between the compared algorithms regarding the IGD+ and HV results are not very large.

5.6. Comparisons on the DTLZ Test Suite

Tables 10 and 11 list the performances of the compared algorithms on the DTLZ test suite. The DTLZ test suite is composed of 17 test problems with various properties, e.g., the true Pareto fronts of IDTLZ1, ISDTLZ1, modified IDTLZ1 and IDTLZ2 are inverted; the true Pareto fronts of DTLZ5IM and DTLZ6 are degenerated; and the true Pareto fronts of DTLZ7 are disconnected. Overall, the AdaW and DMEA-WUA achieved the best and second=best performance on this test suite in terms of Friedman's rank test. AdaW obtains good IGD+ values in 25 instances and good HV values in 21 instances out of the 51 total test instances. The DMEA-WUA obtains good IGD+ values in 14 instances and good HV values in 17 instances out of the

Table 10: Statistical results (means and standard deviations) of the obtained IGD+ values on the DTLZ test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
DTLZ1	3	7	1.3537e-2 (1.46e-4)	1.3831e-2 (1.51e-4) −	1.4294e-2 (1.35e-4) −	1.3699e-2 (2.33e-4) −	1.1195e+0 (4.93e+0) −	1.4157e-2 (1.71e-4) −	1.1010e-1 (1.93e-2) −	1.4371e-2 (4.52e-4) −	3.5875e-2 (1.86e-2) −	
	Friedman Ranking		1.3000	2.8500	5.0500	2.2000	7.1000	4.6000	8.9000	5.0000	8.0000	1.0556e-27
	10	14	9.5877e-2 (4.59e-3)	9.3528e-2 (5.05e-3) ≈	8.3072e-2 (1.96e-2) +	9.8788e-2 (5.23e-3) ≈	2.1093e+1 (1.18e+1) −	7.5728e-2 (2.59e-3) +	1.8838e-1 (3.47e-2) −	8.1202e-2 (1.50e-3) +	1.7842e-1 (1.86e-2) −	
	Friedman Ranking		5.1000	4.5000	2.2500	5.2500	9.0000	1.2500	7.6000	2.6500	7.4000	2.7355e-28
	15	19	1.4512e-1 (2.97e-2)	1.1691e-1 (9.17e-3) +	1.3201e-1 (7.19e-2) +	9.2031e-2 (1.36e-2) +	3.0028e+1 (6.16e+0) −	1.1182e-1 (1.08e-2) +	1.8268e-1 (2.86e-2) −	9.0911e-2 (6.54e-3) +	1.6050e-1 (1.52e-2) ≈	
	Friedman Ranking		5.8000	4.4500	3.9500	1.8500	9.0000	3.9000	7.3500	2.0500	6.6500	7.9564e-23
DTLZ2	3	12	3.1348e-2 (1.33e-4)	3.2265e-2 (1.90e-4) −	3.4731e-2 (4.48e-4) −	3.1774e-2 (1.71e-4) −	3.9688e-2 (8.20e-4) −	3.2458e-2 (3.77e-4) −	5.7784e-2 (3.68e-3) −	3.1917e-2 (1.41e-4) −	3.2836e-2 (5.50e-4) −	
	Friedman Ranking		1.0500	4.2500	7.0000	2.2000	8.0000	4.9000	9.0000	3.0500	5.5500	6.4325e-29
	10	19	2.5611e-1 (2.08e-3)	2.5779e-1 (5.88e-3) ≈	2.2779e-1 (3.16e-3) +	2.4338e-1 (3.96e-2) +	7.8819e-1 (3.81e-2) −	2.7034e-1 (5.03e-3) −	2.4629e-1 (5.81e-3) +	2.4452e-1 (1.56e-3) +	2.6960e-1 (2.49e-2) −	
	Friedman Ranking		5.4000	5.9000	1.7000	2.6500	9.0000	7.4000	3.6000	3.0500	6.3000	4.3808e-23
	15	24	4.0095e-1 (1.30e-2)	3.5912e-1 (3.14e-2) +	3.1039e-1 (1.28e-2) +	4.0213e-1 (3.49e-2) ≈	8.5240e-1 (9.32e-3) −	4.1415e-1 (2.20e-2) −	3.1624e-1 (7.26e-3) +	3.2078e-1 (6.80e-3) +	3.0635e-1 (9.99e-3) +	
	Friedman Ranking		6.6500	5.2500	2.2000	6.7000	9.0000	7.3000	2.7000	3.4000	1.8000	1.4327e-26
DTLZ3	3	12	3.3192e-2 (1.11e-3)	3.4725e-2 (2.32e-3) −	3.5929e-2 (1.42e-3) −	3.5567e-2 (2.24e-3) −	1.3227e+0 (2.56e+0) −	3.7799e-2 (2.52e-3) −	7.9801e-2 (1.48e-2) −	4.3027e-2 (8.71e-3) −	5.6861e-2 (2.04e-2) −	
	Friedman Ranking		1.4500	2.5000	3.7000	3.3500	7.8500	4.6500	8.4000	5.7500	7.3500	1.6894e-24
	10	19	3.0883e-1 (2.58e-1)	3.3144e-1 (2.83e-2) −	6.8024e+1 (2.44e+1) −	2.8658e-1 (5.23e-2) +	4.9876e+1 (7.63e+1) −	3.2579e-1 (1.53e-2) −	4.3182e-1 (6.42e-2) −	2.5988e-1 (6.33e-3) +	4.8548e-1 (5.14e-2) −	
	Friedman Ranking		1.9500	4.3000	8.7000	3.1000	8.2500	4.0000	5.9000	2.1500	6.6500	3.8790e-25
	15	24	2.9470e-1 (2.41e-2)	4.5976e-1 (3.81e-2) −	1.5016e+2 (1.18e+2) −	4.5657e-1 (4.45e-2) −	1.3189e+2 (8.30e+1) −	4.7927e-1 (1.84e-2) −	5.4075e-1 (5.41e-2) −	4.0788e-1 (1.33e-1) −	4.6980e-1 (4.56e-2) −	
	Friedman Ranking		1.0000	4.2000	8.5500	4.0500	8.4500	5.0000	6.4000	2.6000	4.7500	7.1510e-25
DTLZ4	3	12	6.0511e-2 (7.13e-2)	4.2390e-2 (4.32e-2) +	3.4710e-2 (4.48e-4) +	5.1205e-2 (5.98e-2) +	1.3127e-1 (9.69e-2) −	8.5565e-2 (1.24e-1) −	1.5616e-1 (1.22e-1) −	1.1960e-1 (1.58e-1) −	3.2845e-2 (4.30e-4) +	
	Friedman Ranking		1.9500	4.2500	6.1500	2.7000	7.7500	4.3000	8.2000	4.8000	4.9000	1.2237e-16
	10	19	2.4735e-1 (8.81e-3)	2.4164e-1 (3.37e-3) +	2.2834e-1 (1.02e-2) +	2.6067e-1 (1.58e-2) −	2.2820e-1 (6.76e-3) +	2.5310e-1 (3.74e-3) −	2.4101e-1 (7.03e-3) +	2.4390e-1 (1.84e-3) ≈	2.4167e-1 (3.34e-3) +	
	Friedman Ranking		6.1000	4.4000	2.0500	8.1000	2.1000	8.1000	4.3500	5.2000	4.6000	1.1782e-18
	15	24	3.3203e-1 (3.14e-2)	2.9176e-1 (1.07e-2) +	3.7699e-1 (2.87e-2) −	3.1725e-1 (1.62e-2) ≈	2.8678e-1 (2.27e-2) +	3.4214e-1 (1.32e-2) −	3.0858e-1 (4.47e-3) +	3.3066e-1 (3.96e-3) +	2.8517e-1 (4.99e-3) +	
	Friedman Ranking		5.9500	2.5500	8.7500	4.9500	2.3500	7.6500	4.2500	6.6000	1.9500	1.4972e-23
DTLZ5IM	3	12	2.5713e-3 (1.07e-4)	2.4949e-3 (1.42e-5) +	2.8116e-3 (3.66e-4) −	4.5220e-3 (1.07e-4) −	1.2983e-2 (1.15e-3) −	2.5519e-3 (5.10e-5) ≈	1.6597e-2 (4.86e-3) −	4.5633e-3 (1.00e-3) −	2.7723e-3 (9.44e-5) −	
	Friedman Ranking		2.5000	1.2500	4.1500	6.5500	8.2000	2.5000	8.8000	6.4500	4.6000	1.2052e-28
	10	19	3.4866e-1 (5.50e-1)	1.3644e-3 (1.46e-5) +	1.3351e-3 (3.09e-5) +	6.5105e-2 (5.63e-2) ≈	7.2290e-1 (4.82e-1) −	1.2047e-3 (6.49e-6) +	1.0491e-2 (2.50e-3) +	3.8620e-2 (5.48e-2) +	4.0386e-1 (3.41e-1) −	
	Friedman Ranking		6.2000	2.8500	2.5500	5.8000	8.4500	1.0000	4.5000	5.7500	7.9000	7.1510e-25
	15	24	3.1516e-1 (4.03e-1)	1.3122e-3 (1.51e-5) +	1.2501e-3 (5.10e-5) +	4.3183e-2 (3.22e-2) +	2.1287e+2 (9.16e+1) −	1.9917e-3 (1.56e-5) +	1.2542e-1 (1.39e-1) +	5.2884e-2 (5.76e-2) +	2.1534e+0 (4.73e+0) −	
	Friedman Ranking		6.9000	1.8500	1.1500	4.9500	8.8500	3.0000	5.4000	5.1500	7.7500	1.0692e-27
DTLZ6	3	12	2.5364e-3 (6.02e-5)	2.5034e-3 (1.94e-5) ≈	2.6861e-3 (2.86e-4) −	4.3832e-3 (1.01e-4) −	1.3642e-2 (3.13e-4) −	2.5144e-3 (1.96e-5) ≈	1.1619e-2 (2.26e-3) −	5.2321e-3 (1.52e-3) −	2.7440e-3 (1.00e-4) −	
	Friedman Ranking		2.5500	1.9500	3.5000	6.4000	8.8000	2.2000	8.1500	6.6500	4.8000	1.8892e-27
	10	19	2.7552e-1 (1.83e-1)	7.6127e-2 (3.35e-2) +	9.2567e-2 (3.35e-2) +	2.0282e-1 (6.48e-2) ≈	9.6282e+0 (8.54e-1) −	3.8550e-2 (9.21e-3) +	9.0768e-2 (2.44e-2) +	1.4829e-1 (3.45e-2) +	2.5279e-1 (1.06e-1) ≈	
	Friedman Ranking		6.7000	2.7000	3.4000	6.3500	9.0000	1.2000	3.3500	5.3500	6.9500	8.9940e-25
	15	24	3.2358e-1 (2.36e-1)	1.5241e-1 (1.42e-1) +	1.6610e-1 (1.21e-1) +	1.3767e-1 (4.96e-2) +	1.0376e+1 (1.46e-3) −	4.6218e-2 (9.97e-3) +	8.9320e-2 (1.86e-2) +	1.9493e-1 (7.65e-2) +	2.8633e-1 (1.56e-1) ≈	
	Friedman Ranking		6.5000	4.0500	4.2500	4.4000	9.0000	1.2000	3.1000	5.8500	6.6500	2.5423e-20
DTLZ7	3	22	3.6842e-2 (1.21e-3)	3.6045e-2 (5.58e-4) +	3.6258e-2 (7.28e-4) ≈	9.6044e-2 (6.71e-2) −	8.1947e-1 (1.69e+0) −	4.1047e-2 (1.51e-3) −	2.8563e-1 (2.08e-1) −	7.3754e-2 (7.70e-2) ≈	4.6641e-2 (4.33e-2) ≈	
	Friedman Ranking		3.5000	2.4000	2.6500	7.3500	7.8000	5.7500	8.3500	3.6500	3.5500	6.6990e-22
	10	29	7.9233e+0 (1.10e+1)	7.8251e-1 (1.84e-2) +	9.3793e-1 (5.49e-2) +	9.4406e-1 (3.16e-2) +	3.5875e+0 (5.85e-1) ≈	1.0164e+0 (1.38e-1) +	1.6321e+0 (3.09e-1) ≈	8.0826e-1 (1.60e-1) +	9.9129e-1 (2.64e-1) +	
	Friedman Ranking		7.4000	1.9500	4.4000	4.0500	8.6500	4.9000	7.3500	2.0500	4.2500	1.1150e-17
	15	34	2.2191e+1 (2.57e+1)	2.0009e+0 (2.00e-1) +	1.4916e+0 (1.03e-1) +	1.5596e+0 (5.45e-2) +	4.0304e+1 (1.22e+1) −	1.9053e+0 (2.46e-1) +	1.6009e+0 (5.22e-1) +	1.8288e+0 (8.04e-1) +	7.0227e+0 (1.22e+0) ≈	
	Friedman Ranking		7.4500	5.4500	2.4000	3.3000	8.7500	4.8000	2.3000	3.1000	7.4500	3.1287e-23
SDTLZ1	3	7	3.0756e-2 (3.42e-4)	3.0871e-2 (3.98e-4) ≈	3.3340e-2 (2.71e-3) −	3.3681e-2 (1.15e-3) −	3.6736e-2 (6.17e-4) −	3.1342e-2 (6.82e-4) −	1.7142e-1 (5.54e-2) −	3.7324e-2 (1.50e-3) −	5.8965e-2 (2.84e-2) −	
	Friedman Ranking		1.6000	1.7000	4.5500	4.7500	6.1500	2.8500	8.9500	6.5000	7.9500	7.3772e-28
	10	14	5.0054e+0 (6.76e-1)	4.8262e+0 (8.01e-1) ≈	3.6666e+0 (1.27e+0) +	1.3628e+1 (6.74e+0) −	4.0337e+3 (2.57e+3) −	8.0984e+0 (2.71e+0) −	2.4261e+1 (1.14e+1) −	2.1552e+1 (1.39e+1) −	3.0383e+0 (1.10e+0) +	
	Friedman Ranking		3.3500	3.2500	2.0000	6.2500	9.0000	5.3000	7.3500	6.9500	1.5500	5.3562e-27
	15	19	6.1205e+1 (1.45e+1)	8.8192e+1 (2.09e+1) −	2.5260e+2 (4.56e+1) −	3.0574e+2 (3.29e+1) −	2.0698e+5 (8.62e+4) −	1.6280e+2 (4.87e+1) −	6.7916e+2 (4.17e+2) −	5.9903e+1 (5.72e+1) ≈	6.7456e+1 (2.74e+1) ≈	
	Friedman Ranking		2.1500	3.3500	6.2500	6.9500	9.0000	4.9500	7.6500	2.2500	2.4500	8.5397e-27
IDTLZ1	3	7	1.3954e-2 (6.66e-4)	1.3800e-2 (1.76e-4) ≈	1.4133e-2 (2.40e-4) −	1.6952e-2 (4.49e-4) −	3.2885e-2 (7.30e-4) −	1.4228e-2 (3.14e-4) −	3.2321e-2 (6.24e-3) −	1.6546e-2 (1.38e-3) −	2.1141e-2 (5.90e-3) −	
	Friedman Ranking		1.9000	1.7000	3.0500	5.8000	8.5000	3.4500	8.2500	5.6000	6.7500	5.9709e-27
	10	14	8.2828e-2 (1.39e-3)	8.8249e-2 (5.35e-3) −	8.4248e-2 (4.45e-3) ≈	2.6348e-1 (2.88e-1) −	3.7631e-1 (9.10e-1) −	8.7213e-2 (1.06e-3) −	3.7631e-1 (9.10e-1) −	8.7213e-2 (1.06e-3) −	1.0997e-1 (3.17e-3) −	
	Friedman Ranking		1.4000	3.4500	2.0500	8.5000	6.8500	3.4000	6.2000	8.3000	4.8500	1.5200e-27
	15	19	1.0091e-1 (2.82e-3)	1.0203e-1 (1.76e-3) −	1.0383e-1 (3.32e-3) −	1.7533e-1 (1.44e-2) −	1.4906e-1 (4.42e-3) −	1.1923e-1 (2.88e-3) −	1.2277e-1 (2.47e-3) −	1.5857e-1 (5.61e-3) −	1.1354e-1 (7.90e-3) −	
	Friedman Ranking		1.4500	2.0000	2.5500	8.7500	7.1000	5.1000	5.8000	8.1000	4.1500	6.9466e-29
Modified IDTLZ1	3	12	2.6927e-2 (1.29e-4)	2.7435e-2 (1.61e-4) −	2.9231e-2 (7.97e-4) −	2.7936e-2 (2.05e-4) −	6.5815e-2 (8.34e-4) −	2.7704e-2 (1.52e-4) −	7.6140e-2 (2.56e-2) −	3.4283e-2 (2.76e-3) −	2.7667e-2 (4.98e-4) −	
	Friedman Ranking		1.0500	2.5000	5.8500	4.6000	8.5000	3.5500	8.5000	7.0000	3.4500	3.7674e-28
	10	19	1.7585e-1 (4.23e-2)	1.7184e-1 (9.83e-4) +	1.7227e-1 (2.53e-3) +	2.8097e-1 (4.99e-2) −	2.1298e-1 (5.50e-3) −	1.7513e-1 (1.12e-3) +	2.2939e-1 (5.82e-3) −	2.7490e-1 (3.50e-2) −	1.8444e-1 (8.99e-3) −	
	Friedman Ranking		1.4000	2.4500	2.6000	8.3000	5.9500	4.0500	7.1500	8.4000	4.7000	6.4563e-26
	15	24	2.0062e-1 (1.21e-3)	2.0285e-1 (1.65e-3) −	2.1357e-1 (6.36e-3) −	2.8925e-1 (3.72e-2) −	2.9165e-1 (3.81e-3) −	2.1523e-1 (3.48e-3) −	2.5761e-1 (4.05e-3) −	3.7532e-1 (3.84e-2) −	2.3667e-1 (1.37e-2) −	
	Friedman Ranking		1.2000	1.8000	3.4000	7.2000	7.8000	3.7500	5.9500	8.9500	4.9500	2.1088e-29
IDTLZ2	3	12	3.4765e-2 (3.67e-4)	3.4779e-2 (3.94e-4) ≈	3.8044e-2 (1.16e-3) −	4.0955e-2 (1.21e-3) −	4.9632e-2 (2.58e-3) −	3.5665e-2 (3.74e-4) −	5.4856e-2 (3.80e-3) −	4.1287e-2 (1.90e-3) −	3.7027e-2 (1.91e-3) −	
	Friedman Ranking		1.7500	1.6500	4.8500	6.3500	8.1000	3.0500	8.8500	6.4500	3.9500	8.8813e-28
	10	19	2.8233e-1 (3.29e-3)	2.9231e-1 (3.25e-3) −	2.5619e-1 (4.70e-3) +	5.3328e-1 (5.75e-2) −	3.3802e-1 (3.20e-3) −	3.1485e-1 (2.24e-3) −	3.2712e-1 (5.60e-3) −	2.8167e-1 (9.84e-3) ≈	2.6135e-1 (1.21e-2) +	
	Friedman Ranking		3.6000	4.7500	1.2500	9.0000	7.9500	6.0000	7.0500	3.5000	1.9000	2.5234e-29
	15	24	3.6566e-1 (5.11e-3)	3.5327e-1 (4.18e-3) +	3.3108e-1 (9.39e-3) +	7.1177e-1 (4.30e-2) −	4.3922e-1 (2.59e-3) −	4.3023e-1 (7.47e-3) −	4.5188e-1 (7.48e-3) −	4.3877e-1 (2.71e-2) −	3.9892e-1 (2.49e-2) −	
	Friedman Ranking		2.9000	2.0000	1.1000	9.0000	6.3500	5.3000	7.7500	6.1000	4.5000	4.7742e-28
CDTLZ2	3	12	2.2437e-2 (3.57e-4)	2.1614e-2 (2.80e-4) +	2.9909e-2 (8.94e-4) −	2.4865e-2 (1.12e-3) −	2.7192e-2 (9.74e-4) −	2.1917e-2 (4.31e-4) +	8.8160e-2 (5.39e-2) −	3.4361e-2 (2.68e-3) −	2.2811e-2 (2.32e-3) ≈	
	Friedman Ranking		3.4000	1.6500	7.0000	4.8000	5.9500	2.3000	8.9500	8.0000	2.9500	1.2229e-27
	10	19	1.0998e-1 (1.06e-2)	1.0484e-1 (5.58e-3) +	7.2769e-2 (2.54e-3) +	8.9495e-2 (2.74e-2) +	4.0455e-1 (2.44e-1) −	4.8435e-2 (2.57e-3) +	7.6787e-2 (2.07e-2) +	4.4728e-2 (5.21e-3) +	2.7902e-2 (4.31e-3) +	
	Friedman Ranking		7.4000	6.8500	4.9000	6.0000	8.8500	2.7500	4.9000	2.3500	1.0000	1.0886e-26
	15	24	6.2088e-2 (1.48e-2)	1.2135e-1 (1.38e-2) −	5.1763e-2 (2.87e-3) +	3.8560e-2 (1.76e-2) +	1.2481e+0 (3.82e-1) −	3.7589e-2 (3.48e-3) +	7.2160e-2 (1.55e-2) −	2.4184e-2 (2.40e-3) +	3.7807e-2 (6.37e-3) +	
	Friedman Ranking		5.9500	8.0000	5.2500	2.7500	9.0000	3.1000	6.6000	1.1000	3.2500	5.3229e-28
SDTLZ2	3	12	7.0972e-2 (5.67e-4)	7.2506e-2 (8.87e-4) −	7.4410e-2 (1.20e-3) −	7.1135e-2 (7.25e-4) ≈	8.8893e-2 (1.64e-3) −	7.0378e-2 (1.07e-3) +	1.2466e-1 (1.19e-2) −	7.8138e-2 (1.49e-3) −	7.5226e-2 (9.68e-4) −	
	Friedman Ranking		2.2000	3.7500	5.1500	2.4500	8.0000	1.7000	9.0000	7.0000	5.7500	1.6600e-28
	10	19	1.4313e+1 (1.12e+0)	1.4264e+1 (1.28e+0) ≈	2.6810e+1 (1.23e+0) −	3.3509e+1 (1.68e+0) −	3.7251e+2 (1.09e+2) −	2.9521e+1 (3.08e+0) −	1.5870e+1 (1.47e+0) −	1.4404e+1 (3.83e+0) ≈	2.5549e+1 (3.95e+0) −	
	Friedman Ranking		2.2500	2.5000	5.8000	7.9000	9.0000	6.7000	3.4500	1.8000	5.6000	1.3290e-27
	15	24	5.6875e+2 (3.25e+2)	5.1731e+2 (1.79e+2) ≈	2.1929e+2 (1.51e+2) +	8.3673e+2 (1.04e+2) −	1.3922e+4 (2.64e+0) −	8.1444e+2 (1.52e+2) −	3.3314e+2 (4.83e+1) +	3.7430e+1 (4.41e+1) +	3.7571e+2 (1.69e+2) +	
	Friedman Ranking		5.4500	5.1500	2.4000	7.2500	9.0000	7.2000	3.7000	1.0500	3.8000	7.2969e-26
DTLZ2BZ	3	12	1.8995e-2 (4.70e-4)	1.8507e-2 (1.87e-4) +	1.9896e-2 (1.96e-4) −	1.8987e-2 (2.28e-4) ≈	3.2629e-2 (1.24e-3) −	1.8668e-2 (2.58e-4) +	4.0609e-2 (2.29e-3) −	1.9092e-2 (2.08e-4) ≈	1.8706e-2 (3.99e-4) ≈	
	Friedman Ranking		4.0000	1.9000	6.9000	4.4000	8.0000	2.6000	9.0000	5.2500	2.9500	9.5246e-25
	10	19	1.2952e-1 (5.82e-3)	1.1623e-1 (2.91e-3) +	1.1574e-1 (2.42e-3) +	1.2431e-1 (4.32e-3) +	4.8264e-1 (8.94e-3) −	1.1404e-1 (3.03e-3) +	1.2027e-1 (3.99e-3) +	1.6519e-1 (6.73e-3) −	1.8499e-1 (1.09e-2) −	
	Friedman Ranking		5.7000	2.3500	2.5000	5.0500	9.0000	1.8000	3.6000	7.1000	7.9000	1.2789e-27
	15	24	1.5264e-1 (7.18e-3)	1.2409e-1 (2.91e-3) +	1.2846e-1 (4.22e-3) +	1.5164e-1 (4.20e-3) ≈	5.0506e-1 (7.61e-5) −	1.3037e-1 (4.74e-3) +	1.3738e-1 (5.65e-3) +	1.5933e-1 (4.62e-3) −	1.9003e-1 (1.37e-2) −	
	Friedman Ranking		5.8000	1.5000	2.4000	5.5500	9.0000	2.4000	3.7000	6.6500	8.0000	2.3012e-28
CDTLZ3	3	12	2.3933e-2 (1.02e-3)	2.2765e-2 (1.03e-3) +	2.9691e-2 (1.59e-3) −	2.5825e-2 (1.49e-3) −	9.6279e+2 (4.30e+3) −	2.3381e-2 (1.16e-3) ≈	1.2224e-1 (8.79e-2) −	3.5517e-2 (3.65e-3) −	3.4459e-2 (1.45e-2) −	
	Friedman Ranking		2.8500	1.7500	5.9500	4.2000	6.3500	2.1500	8.0000	7.5500	5.4000	9.7795e-24
	10	19	1.1837e-1 (8.45e-2)	6.1826e-2 (1.23e-2) +	7.5438e+3 (8.52e+3) −	7.6351e-2 (5.36e-3) +	8.8939e+3 (1.33e+4) −	4.2092e-2 (2.98e-3) +	2.9123e-1 (9.42e-2) −	5.0344e-2 (5.95e-3) +	3.0343e-1 (1.64e-1) −	
	Friedman Ranking		5.1500	3.1500	7.7000	3.9000	8.5500	1.1000	6.6500	2.3000	6.5000	4.3795e-26
	15	24	4.5522e-2 (1.15e-2)	4.8156e-2 (8.34e-3) ≈	9.1679e+4 (2.25e+5) −	2.2832e-2 (3.90e-3) +	1.4400e+8 (4.42e+8) −	3.8927e-2 (3.97e-3) +	3.1838e-1 (1.03e-1) −	2.9994e-2 (3.35e-3) +	2.8164e-1 (4.16e-1) −	
	Friedman Ranking		4.2500	4.4000	7.8000	1.1000	8.8500	3.5000	6.9000	2.0000	6.2000	8.6017e-28
ISDTLZ3	3	12	1.6000e-1 (4.34e-3)	1.5975e-1 (7.01e-3) ≈	1.8557e-1 (7.71e-3) −	1.7528e-1 (5.27e-3) −	2.2025e-1 (1.10e-2) −	1.6296e-1 (1.90e-3) −	2.5894e-1 (2.74e-2) −	1.8329e-1 (1.22e-2) −	2.0788e-1 (1.07e-1) −	
	Friedman Ranking		2.0500	2.0000	6.0500	4.7500	7.9000	2.7000	8.8500	5.5500	5.1500	1.0555e-23
	10	19	3.0076e+1 (2.85e+0)	3.6883e+1 (2.24e+0) −	2.9033e+1 (2.88e+0) ≈	4.8792e+1 (7.22e+0) −	8.3540e+1 (5.04e+0) −	3.0878e+1 (1.68e+0) ≈	3.9497e+1 (4.86e+0) −	2.2847e+1 (1.88e+0) +	5.3300e+1 (8.23e+0) −	
	Friedman Ranking		3.0500	5.2500	2.7000	7.3000	9.0000	3.3000	5.8000	1.0000	7.6000	1.0535e-28
	15	24	9.2165e+2 (2.08e+2)	1.1075e+3 (1.08e+2) −	8.5141e+2 (2.51e+2) ≈	1.2942e+3 (1.84e+2) −	2.7140e+3 (2.02e+2) −	9.7803e+2 (9.65e+1) ≈	1.3282e+3 (3.77e+2) −	6.4200e+2 (1.90e+2) +	1.4151e+3 (3.06e+2) −	
	Friedman Ranking		3.2500	5.1500	2.9500	6.5500	9.0000	3.7500	6.1500	1.4500	6.7500	8.6855e-22
CSDTLZ4	3	12	5.5230e-1 (9.31e-1)	1.4560e-1 (1.76e-3) +	1.9576e-1 (2.09e-1) +	3.0612e-1 (3.49e-1) ≈	1.0015e+0 (1.41e+0) −	3.3991e-1 (8.98e-1) +	1.6861e+0 (1.69e+0) −	4.5237e-1 (4.03e-1) +	1.4913e-1 (2.67e-3) +	
	Friedman Ranking		4.2000	3.2500	5.0500	3.2500	7.6000	1.5000	8.3000	6.7500	4.6500	3.7272e-19
	10	19	3.6901e+1 (5.45e+0)	2.6273e+1 (1.34e+0) +	4.9245e+1 (3.89e+0) −	2.8579e+1 (7.64e+0) +	4.2221e+1 (6.70e+0) −	6.3034e+1 (4.95e+0) −	3.0467e+1 (2.58e+0) +	5.6686e+1 (3.84e+0) −	3.2018e+1 (2.40e+0) +	
	Friedman Ranking		4.6000	1.4500	6.8500	2.7000	5.8500	8.9000	3.0500	8.0500	3.5500	2.0487e-26
	15	24	1.5958e+3 (4.51e+2)	5.1863e+2 (7.74e+1) +	7.5333e+2 (1.51e+2) +	6.4743e+2 (2.73e+2) +	1.7252e+3 (7.75e+2) ≈	1.6407e+3 (3.03e+2) ≈	6.4391e+2 (6.50e+1) +	1.4850e+3 (4.97e+2) ≈	6.3942e+2 (5.79e+1) +	
	Friedman Ranking		7.4500	1.9500	4.5000	2.9500	7.1500	7.5500	3.2500	6.8500	3.3500	1.6439e-19
+/−/≈				25/15/11	23/24/4	15/27/9	2/47/2	19/26/6	15/35/1	20/24/7	13/30/8	
Average Ranking			3.8460	3.2696	4.3107	5.1872	7.8950	4.0264	6.4323	4.9088	5.1235	

Table 11: Statistical results (means and standard deviations) of the obtained HV values on the DTLZ test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
DTLZ1	3	7	8.4417e-1 (2.85e-4)	8.4334e-1 (3.97e-4) −	8.4219e-1 (4.37e-4) −	8.4364e-1 (8.88e-4) −	7.9597e-1 (1.87e-1) −	8.4107e-1 (8.65e-4) −	5.9460e-1 (3.13e-2) −	8.4118e-1 (1.13e-3) −	7.5857e-1 (6.40e-2) −	
	Friedman Ranking		8.8000	7.3000	5.7500	7.7500	2.9500	4.6500	1.1000	4.7000	2.0000	1.4605e-28
	10	14	9.9924e-1 (2.73e-4)	9.9662e-1 (2.61e-3) −	9.9595e-1 (1.26e-2) −	9.9693e-1 (2.57e-3) −	0.0000e+0 (0.00e+0) −	9.9715e-1 (1.12e-3) −	8.3232e-1 (6.75e-2) −	9.9957e-1 (3.39e-5) +	7.2331e-1 (8.25e-2) −	
	Friedman Ranking		7.8000	5.2000	6.5500	5.5000	1.0000	5.0000	2.9000	8.9500	2.1000	1.0289e-27
	15	19	9.9581e-1 (3.13e-3)	9.9508e-1 (3.72e-3) ≈	9.1692e-1 (2.16e-1) ≈	9.6590e-1 (2.16e-2) −	0.0000e+0 (0.00e+0) −	9.9256e-1 (2.81e-3) −	8.9655e-1 (3.60e-2) −	9.9870e-1 (2.73e-3) +	8.8754e-1 (4.38e-2) −	
	Friedman Ranking		7.0500	6.7000	6.3500	4.1500	1.0000	5.9500	2.7000	8.4000	2.7000	3.0927e-24
DTLZ2	3	12	5.6390e-1 (4.06e-4)	5.6334e-1 (8.22e-4) −	5.5796e-1 (1.42e-3) −	5.6461e-1 (2.71e-4) +	5.4700e-1 (1.47e-3) −	5.6182e-1 (7.75e-4) −	5.3496e-1 (1.29e-2) −	5.6174e-1 (7.37e-4) −	5.5279e-1 (2.11e-3) −	
	Friedman Ranking		7.7000	7.2000	3.9500	8.9000	2.0000	5.6000	1.0000	5.6000	3.0500	2.1775e-29
	10	19	9.6623e-1 (7.59e-4)	9.5072e-1 (5.23e-3) −	9.6632e-1 (5.74e-4) ≈	9.5708e-1 (2.09e-2) ≈	9.9937e-2 (4.04e-2) −	9.3300e-1 (4.51e-3) −	9.2684e-1 (5.52e-3) −	9.6657e-1 (2.43e-4) ≈	8.7468e-1 (1.69e-2) −	
	Friedman Ranking		7.3000	5.2000	7.1500	7.4000	1.0000	3.9500	3.3500	7.6500	2.0000	7.8776e-26
	15	24	9.1369e-1 (1.54e-2)	9.5709e-1 (2.20e-2) +	9.9364e-1 (5.03e-4) +	9.0474e-1 (5.08e-2) ≈	8.8445e-2 (8.47e-3) −	8.8313e-1 (2.46e-2) −	9.6460e-1 (3.87e-3) +	9.8901e-1 (7.32e-4) +	9.6905e-1 (5.48e-3) +	
	Friedman Ranking		3.4000	5.5000	9.0000	3.3000	1.0000	2.5500	5.9500	8.0000	6.3000	6.6593e-28
DTLZ3	3	12	5.6193e-1 (2.49e-3)	5.6013e-1 (2.54e-3) −	5.5806e-1 (2.93e-3) −	5.5712e-1 (4.75e-3) −	3.6243e-1 (2.48e-1) −	5.5060e-1 (3.78e-3) −	4.6636e-1 (4.70e-2) −	5.3969e-1 (1.68e-2) −	5.1515e-1 (2.45e-2) −	
	Friedman Ranking		8.6000	7.7500	6.7500	6.7000	2.1500	4.9000	1.7000	3.9000	2.5500	1.8147e-26
	10	19	8.5970e-1 (2.66e-1)	8.1284e-1 (4.50e-2) −	0.0000e+0 (0.00e+0) −	9.2066e-1 (4.40e-2) ≈	4.9985e-2 (4.64e-2) −	8.8118e-1 (1.77e-2) +	6.5750e-1 (6.49e-2) −	9.5153e-1 (7.03e-3) ≈	6.0144e-1 (8.64e-2) −	
	Friedman Ranking		7.6750	5.1500	1.2500	7.4000	1.8250	6.2500	3.8000	8.2000	3.4500	1.3409e-26
	15	24	9.5353e-1 (6.51e-2)	8.1706e-1 (6.62e-2) −	0.0000e+0 (0.00e+0) −	7.6229e-1 (9.25e-2) −	9.0810e-3 (2.80e-2) −	7.8812e-1 (2.13e-2) −	6.6317e-1 (6.87e-2) −	8.9425e-1 (2.16e-1) −	7.9357e-1 (5.57e-2) −	
	Friedman Ranking		8.4500	6.1000	1.4500	5.1000	1.5500	5.6000	3.3500	7.7500	5.6500	1.8994e-24
DTLZ4	3	12	5.3145e-1 (7.92e-2)	5.5167e-1 (4.81e-2) +	5.5751e-1 (1.16e-3) +	5.4271e-1 (6.72e-2) +	4.3661e-1 (1.08e-1) −	5.0695e-1 (1.26e-1) −	4.2943e-1 (1.30e-1) −	4.7165e-1 (1.56e-1) −	5.5489e-1 (1.97e-3) +	
	Friedman Ranking		7.2500	6.4500	4.7500	8.1000	2.2000	5.9000	1.8000	4.5500	4.0000	8.7269e-18
	10	19	9.5519e-1 (4.16e-3)	9.6238e-1 (2.94e-3) +	9.6663e-1 (2.32e-3) +	9.4183e-1 (1.74e-2) −	9.4024e-1 (8.02e-3) −	9.4997e-1 (4.29e-3) −	9.4434e-1 (3.04e-3) −	9.6638e-1 (5.62e-4) +	9.2295e-1 (6.89e-3) −	
	Friedman Ranking		5.8500	6.9000	8.6000	3.7500	2.9500	4.5500	3.1000	8.1500	1.1500	7.3436e-26
	15	24	9.7150e-1 (2.43e-2)	9.9068e-1 (6.99e-4) +	9.3113e-1 (6.52e-2) ≈	9.7864e-1 (9.22e-3) ≈	9.4386e-1 (4.88e-2) −	9.5817e-1 (9.12e-3) −	9.8140e-1 (1.53e-3) +	9.9000e-1 (9.72e-4) +	9.8802e-1 (9.40e-4) +	
	Friedman Ranking		4.5500	8.6000	3.4000	4.5000	2.6000	2.1000	4.6000	7.8500	6.8000	2.2549e-20
DTLZ5IM	3	12	1.9999e-1 (1.85e-4)	2.0024e-1 (1.70e-4) +	1.9986e-1 (3.87e-4) ≈	1.9804e-1 (1.47e-4) −	1.9081e-1 (1.17e-3) −	1.9920e-1 (9.49e-4) −	1.8412e-1 (6.07e-3) −	1.9835e-1 (8.14e-4) −	2.0025e-1 (1.65e-4) +	
	Friedman Ranking		6.4000	8.5000	6.2500	3.4000	1.9500	5.1500	1.0500	4.0500	8.2500	1.3809e-27
	10	19	4.9462e-2 (4.09e-2)	1.0079e-1 (2.93e-4) +	1.0094e-1 (2.27e-4) +	9.4151e-2 (4.25e-3) +	5.0995e-2 (4.31e-2) ≈	1.0061e-1 (3.27e-4) +	9.4411e-2 (2.35e-3) +	9.6006e-2 (1.67e-3) +	8.6829e-3 (1.67e-2) −	
	Friedman Ranking		3.3250	7.7500	8.3000	4.7500	2.1500	7.2000	4.6500	5.1500	1.7250	2.3634e-22
	15	24	2.1756e-2 (3.42e-2)	9.5379e-2 (3.08e-4) +	9.5396e-2 (2.99e-4) +	9.4107e-2 (7.51e-4) +	1.0366e-2 (2.79e-2) ≈	9.5113e-2 (5.19e-4) +	9.2530e-2 (9.27e-4) +	8.2243e-2 (2.48e-2) +	6.0179e-3 (1.67e-2) ≈	
	Friedman Ranking		2.8250	8.2000	8.0500	5.8000	1.8500	7.2500	4.4000	4.6250	2.0000	4.0372e-25
DTLZ6	3	12	2.0007e-1 (9.82e-5)	2.0020e-1 (2.64e-4) +	2.0012e-1 (2.00e-4) +	1.9821e-1 (1.83e-4) −	1.9013e-1 (3.53e-4) −	2.0005e-1 (1.50e-4) ≈	1.8815e-1 (3.38e-3) −	1.9757e-1 (1.34e-3) −	2.0021e-1 (1.27e-4) +	
	Friedman Ranking		6.0500	8.3000	7.1000	3.6000	1.6500	5.9500	1.3500	3.4000	7.6000	1.0612e-26
	10	19	7.7237e-2 (3.33e-2)	8.9996e-2 (9.87e-3) +	8.6767e-2 (2.02e-2) +	2.1033e-2 (2.68e-2) −	0.0000e+0 (0.00e+0) −	8.6080e-2 (1.51e-2) +	9.3408e-2 (1.42e-3) +	9.1156e-2 (5.83e-4) +	3.8572e-2 (4.41e-2) −	
	Friedman Ranking		4.4250	6.2500	6.2000	3.3500	1.2000	6.8000	7.9500	5.7000	3.1250	1.1150e-17
	15	24	8.1861e-2 (2.80e-2)	7.3252e-2 (3.76e-2) ≈	6.8527e-2 (3.99e-2) ≈	9.2000e-2 (2.78e-4) +	0.0000e+0 (0.00e+0) −	8.3263e-2 (1.21e-2) ≈	9.1807e-2 (8.18e-4) +	8.6377e-2 (2.03e-2) ≈	6.3653e-2 (4.28e-2) ≈	
	Friedman Ranking		4.6000	5.4000	4.8000	7.9000	1.2000	5.2000	7.0000	5.0250	3.8750	1.6059e-13
DTLZ7	3	22	2.7975e-1 (8.52e-4)	2.8094e-1 (4.37e-4) +	2.7986e-1 (4.52e-4) ≈	2.6536e-1 (1.17e-2) −	2.1449e-1 (9.43e-2) −	2.7933e-1 (5.70e-4) −	2.3226e-1 (2.23e-2) −	2.7259e-1 (1.49e-2) ≈	2.7670e-1 (8.72e-3) −	
	Friedman Ranking		7.0500	8.7500	6.7000	3.0000	2.0000	5.6000	1.4500	5.8000	4.6500	8.5572e-24
	10	29	1.0884e-1 (7.53e-2)	1.2400e-1 (1.18e-2) ≈	1.1245e-1 (1.97e-2) ≈	1.3930e-1 (1.72e-2) ≈	1.5249e-1 (5.20e-2) +	1.8760e-1 (5.26e-3) +	1.8484e-1 (9.59e-3) +	1.4924e-1 (1.81e-2) ≈	1.7631e-1 (1.06e-2) +	
	Friedman Ranking		3.8500	2.5500	1.8500	3.7000	5.6500	8.3000	7.8500	4.3000	6.9500	6.3740e-21
	15	34	7.1433e-2 (6.26e-2)	9.5226e-2 (4.82e-2) ≈	8.4370e-2 (3.22e-2) ≈	1.1012e-1 (3.82e-3) ≈	0.0000e+0 (0.00e+0) −	1.5397e-1 (2.60e-3) +	1.3261e-1 (4.16e-3) +	4.1675e-2 (2.71e-2) ≈	1.5134e-1 (5.47e-3) +	
	Friedman Ranking		4.1250	4.9000	3.7500	4.9500	1.1750	8.5500	6.5000	2.7000	8.3500	4.0010e-24
SDTLZ1	3	7	8.4257e-1 (3.33e-4)	8.4310e-1 (3.05e-4) +	8.3984e-1 (3.23e-3) −	8.3909e-1 (1.45e-3) −	8.3780e-1 (6.62e-4) −	8.4022e-1 (1.03e-3) −	6.0838e-1 (5.28e-2) −	8.3207e-1 (2.57e-3) −	7.8376e-1 (4.71e-2) −	
	Friedman Ranking		8.0000	8.8000	6.1500	5.3500	4.5500	6.0500	1.0000	3.1000	2.0000	1.2546e-27
	10	14	9.9597e-1 (8.64e-4)	9.9658e-1 (1.23e-3) +	9.7822e-1 (4.39e-3) −	2.0389e-1 (3.02e-1) −	1.4876e-2 (6.65e-2) −	9.8652e-1 (3.60e-3) −	8.6591e-1 (5.17e-2) −	5.1417e-2 (2.16e-1) −	7.1523e-1 (1.08e-1) −	
	Friedman Ranking		8.3000	8.7000	6.0500	2.7500	1.6000	6.9500	4.8500	1.9500	3.8500	4.0894e-29
	15	19	9.9528e-1 (1.97e-3)	9.9726e-1 (2.11e-3) +	7.1553e-2 (8.34e-2) −	9.7790e-1 (7.47e-3) −	0.0000e+0 (0.00e+0) −	9.9025e-1 (5.05e-3) −	8.9602e-1 (5.42e-2) −	7.7804e-1 (7.66e-2) −	9.0836e-1 (4.87e-2) −	
	Friedman Ranking		8.0500	8.7000	1.8250	5.9500	1.1750	7.2500	4.4000	3.2000	4.4500	2.9069e-29
IDTLZ1	3	7	2.2292e-1 (2.18e-3)	2.2364e-1 (5.23e-4) +	2.2230e-1 (5.56e-4) −	2.1744e-1 (1.14e-3) −	1.8293e-1 (1.56e-3) −	2.2321e-1 (7.91e-4) ≈	1.8659e-1 (1.07e-2) −	2.1389e-1 (3.34e-3) −	2.0673e-1 (1.09e-2) −	
	Friedman Ranking		7.5500	8.4500	6.2500	4.7500	1.2500	7.6000	1.9000	3.9500	3.3000	1.9506e-27
	10	14	3.3237e-7 (3.91e-7)	3.3726e-7 (5.01e-7) ≈	4.1661e-7 (2.66e-7) ≈	1.0701e-8 (7.16e-9) ≈	3.2186e-7 (1.34e-7) ≈	2.9062e-7 (4.97e-7) ≈	1.0000e-7 (3.08e-7) −	2.1709e-8 (2.14e-8) ≈	4.2675e-7 (2.11e-7) ≈	
	Friedman Ranking		5.0000	4.4750	6.4500	4.4000	6.2000	4.1250	2.8000	4.5500	7.0000	4.4643e-06
	15	19	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0) ≈	1.9364e-11 (8.08e-11) ≈	4.4650e-13 (3.53e-13) +	3.0827e-12 (8.94e-13) +	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	9.2223e-13 (2.93e-13) +	3.7566e-12 (1.59e-11) ≈	
	Friedman Ranking		3.4000	3.4000	4.0000	7.0000	8.8500	3.4000	3.4000	7.6000	3.9500	6.4563e-26
Modified IDTLZ1	3	12	2.2335e-1 (2.84e-4)	2.2421e-1 (1.28e-4) +	2.2131e-1 (1.21e-3) −	2.2381e-1 (1.64e-4) +	1.8299e-1 (9.79e-4) −	2.2407e-1 (1.06e-4) +	1.7688e-1 (2.23e-2) −	2.1304e-1 (3.57e-3) −	2.2167e-1 (6.44e-4) −	
	Friedman Ranking		6.0500	8.8000	4.3000	7.1500	1.4000	8.0000	1.6000	3.0000	4.7000	1.4821e-29
	10	19	5.4820e-7 (3.32e-7)	1.3629e-7 (3.34e-7) −	4.2752e-7 (2.27e-7) ≈	1.2697e-7 (1.00e-7) −	5.2162e-7 (3.64e-8) ≈	3.5748e-7 (7.19e-7) −	0.0000e+0 (0.00e+0) −	9.2784e-8 (3.83e-8) −	4.3916e-7 (3.56e-7) ≈	
	Friedman Ranking		7.2250	2.9750	6.3500	4.5250	7.5000	3.8250	2.0250	4.5500	6.0250	4.3179e-14
	15	24	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	2.0442e-14 (6.30e-14) ≈	4.0151e-12 (5.34e-13) +	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	1.7270e-13 (1.94e-13) +	0.0000e+0 (0.00e+0) ≈	
	Friedman Ranking		3.9500	3.9500	3.9500	4.4000	9.0000	3.9500	3.9500	7.9000	3.9500	7.7231e-29
IDTLZ2	3	12	5.3942e-1 (5.24e-4)	5.3976e-1 (3.87e-4) +	5.3689e-1 (8.08e-4) −	5.3337e-1 (1.06e-3) −	5.2121e-1 (1.90e-3) −	5.3975e-1 (3.62e-4) +	5.1499e-1 (3.35e-3) −	5.3481e-1 (1.63e-3) −	5.3632e-1 (2.55e-3) −	
	Friedman Ranking		7.5000	8.4000	5.4000	3.3000	2.0000	7.9500	1.0000	4.3500	5.1000	9.7758e-28
	10	19	7.5680e-5 (9.06e-6)	6.4695e-5 (1.04e-5) −	1.9858e-4 (2.49e-5) +	9.3852e-6 (9.50e-6) −	2.5191e-4 (1.09e-5) +	7.1115e-5 (1.45e-5) ≈	5.4643e-5 (1.68e-5) −	2.9233e-4 (3.52e-5) +	4.5380e-4 (3.55e-5) +	
	Friedman Ranking		4.3500	3.3500	6.0000	1.0000	7.1500	3.8000	2.5000	7.8500	9.0000	1.2849e-28
	15	24	0.0000e+0 (0.00e+0)	6.7620e-9 (3.02e-8) ≈	4.7324e-8 (2.49e-8) +	1.3041e-10 (3.25e-10) +	2.9479e-7 (8.50e-9) +	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	1.3720e-7 (4.26e-8) +	1.4571e-7 (9.24e-8) +	
	Friedman Ranking		2.4750	2.6750	6.1000	4.9500	8.9000	2.4750	2.4750	7.4500	7.5000	2.4075e-29
CDTLZ2	3	12	9.6357e-1 (1.37e-4)	9.6378e-1 (7.83e-5) +	9.6168e-1 (4.06e-4) −	9.6260e-1 (4.55e-4) −	9.6118e-1 (4.91e-4) −	9.6335e-1 (1.81e-4) −	8.8283e-1 (5.48e-2) −	9.5875e-1 (7.11e-4) −	9.4853e-1 (5.25e-3) −	
	Friedman Ranking		8.0500	8.8500	4.8000	5.9500	4.2500	7.1000	1.0000	3.0000	2.0000	5.4840e-30
	10	19	1.0000e+0 (1.52e-6)	9.9999e-1 (5.47e-6) −	1.0000e+0 (2.73e-15) +	9.9965e-1 (4.24e-4) −	7.3305e-1 (2.68e-1) −	9.9999e-1 (1.18e-5) −	9.8747e-1 (8.24e-3) −	9.9942e-1 (7.38e-4) −	9.9345e-1 (3.81e-3) −	
	Friedman Ranking		7.9250	6.5500	8.7250	5.3500	1.1000	6.0000	2.2000	4.4500	2.7000	1.0460e-27
	15	24	1.0000e+0 (0.00e+0)	1.0000e+0 (1.78e-6) −	1.0000e+0 (8.56e-13) ≈	9.9996e-1 (8.54e-5) −	5.0421e-2 (7.41e-2) −	9.9996e-1 (4.09e-5) −	9.9353e-1 (4.92e-3) −	9.9837e-1 (6.37e-4) −	9.9021e-1 (6.08e-3) −	
	Friedman Ranking		8.5500	6.7000	8.4000	6.2000	1.0000	5.1500	2.7500	4.0000	2.2500	9.2970e-30
SDTLZ2	3	12	5.6320e-1 (6.91e-4)	5.6285e-1 (7.78e-4) ≈	5.6109e-1 (7.16e-4) −	5.6200e-1 (6.02e-4) −	5.4661e-1 (1.68e-3) −	5.6118e-1 (1.47e-3) −	5.3230e-1 (1.89e-2) −	5.5394e-1 (2.31e-3) −	5.5266e-1 (2.53e-3) −	
	Friedman Ranking		8.3500	7.9000	5.7000	6.9500	2.1000	6.0500	1.0000	3.7000	3.2500	2.7375e-27
	10	19	9.2042e-1 (6.70e-3)	9.5013e-1 (2.33e-3) +	9.5740e-1 (1.81e-3) +	8.9570e-1 (4.75e-3) −	1.3449e-1 (1.35e-1) −	9.0056e-1 (1.19e-2) −	9.2831e-1 (5.14e-3) +	8.1539e-1 (2.14e-2) −	8.8441e-1 (1.60e-2) −	
	Friedman Ranking		6.0500	8.0000	9.0000	4.0000	1.0000	4.5000	6.8500	2.0000	3.6000	2.9243e-29
	15	24	9.2736e-1 (3.24e-2)	5.2416e-1 (4.12e-1) −	9.5718e-1 (1.13e-2) +	8.9974e-1 (9.25e-3) −	9.0876e-2 (1.46e-4) −	8.8287e-1 (1.83e-2) −	9.6438e-1 (4.43e-3) +	6.1044e-1 (8.77e-2) −	9.6762e-1 (6.64e-3) +	
	Friedman Ranking		5.9000	3.2000	7.2500	4.7000	1.3500	3.9500	7.8500	2.5000	8.3000	2.3673e-24
DTLZ2BZ	3	12	2.4442e-1 (7.06e-4)	2.4622e-1 (6.77e-4) +	2.4520e-1 (3.94e-4) +	2.4654e-1 (3.10e-4) +	2.3514e-1 (1.14e-3) −	2.4609e-1 (6.33e-4) +	2.3005e-1 (1.74e-3) −	2.4419e-1 (7.16e-4) ≈	2.4337e-1 (1.22e-3) −	
	Friedman Ranking		4.6000	7.7500	6.1500	8.2500	1.9500	7.4500	1.0500	4.2000	3.6000	1.0817e-26
	10	19	1.9304e-1 (6.18e-3)	1.9593e-1 (4.62e-3) ≈	2.0024e-1 (4.27e-3) +	1.8704e-1 (4.57e-3) −	4.7283e-2 (4.73e-3) −	2.1367e-1 (1.23e-3) +	2.1462e-1 (3.04e-3) +	1.8438e-1 (5.67e-3) −	1.9389e-1 (6.08e-3) ≈	
	Friedman Ranking		4.5500	5.5500	6.5500	3.0000	1.0000	8.4000	8.6000	2.6000	4.7500	9.1031e-27
	15	24	1.5360e-1 (7.32e-3)	1.3284e-1 (1.02e-2) −	1.7179e-1 (5.07e-3) +	1.6069e-1 (1.29e-2) +	4.2698e-2 (3.56e-5) −	2.0986e-1 (4.58e-3) +	1.8425e-1 (6.85e-3) +	1.1434e-1 (7.84e-3) −	1.8615e-1 (1.30e-2) +	
	Friedman Ranking		4.3500	3.0000	5.9500	5.0000	1.0000	9.0000	7.3000	2.1000	7.3000	1.8042e-28
CDTLZ3	3	12	9.6247e-1 (7.18e-4)	9.6271e-1 (1.08e-3) ≈	9.6125e-1 (9.17e-4) −	9.6207e-1 (6.78e-4) −	8.0222e-1 (3.51e-1) −	9.6237e-1 (8.05e-4) ≈	8.2214e-1 (1.22e-1) −	9.5765e-1 (1.12e-3) −	9.3970e-1 (2.06e-2) −	
	Friedman Ranking		7.3000	7.9500	5.4500	6.6000	3.6000	7.3500	1.2000	3.3500	2.2000	1.5161e-24
	10	19	9.7327e-1 (1.20e-1)	9.9913e-1 (1.56e-3) +	9.9968e-2 (3.08e-1) −	1.0000e+0 (4.57e-6) +	3.6783e-2 (1.45e-1) −	9.9991e-1 (8.76e-5) +	8.9255e-1 (7.38e-2) −	1.0000e+0 (1.17e-6) +	7.8594e-1 (1.93e-1) −	
	Friedman Ranking		8.5000	5.2000	1.7000	7.2000	1.5000	5.9000	3.6500	8.1000	3.2500	2.0090e-28
	15	24	1.0000e+0 (1.13e-6)	9.9905e-1 (1.25e-3) −	4.9977e-2 (2.24e-1) −	9.9935e-1 (3.79e-4) −	0.0000e+0 (0.00e+0) −	9.9993e-1 (6.11e-5) −	8.9681e-1 (8.01e-2) −	9.9994e-1 (8.01e-5) −	8.3648e-1 (2.24e-1) −	
	Friedman Ranking		9.0000	5.6000	1.7500	5.5000	1.5000	7.1500	3.5000	7.6500	3.3500	1.3208e-28
ISDTLZ3	3	12	5.3391e-1 (2.22e-3)	5.3678e-1 (3.04e-3) +	5.3312e-1 (1.91e-3) ≈	5.3164e-1 (1.88e-3) −	5.1940e-1 (2.83e-3) −	5.3377e-1 (8.43e-4) ≈	5.0966e-1 (5.99e-3) −	5.3075e-1 (2.41e-3) −	5.2320e-1 (2.28e-2) −	
	Friedman Ranking		6.8500	8.3000	6.1500	5.1000	2.1500	6.6500	1.1500	4.6000	4.0500	9.1963e-21
	10	19	2.5003e-5 (1.82e-5)	3.5566e-5 (1.13e-5) +	5.9079e-5 (5.27e-5) ≈	1.9678e-5 (2.26e-5) ≈	2.5130e-4 (1.29e-5) +	7.7991e-5 (2.58e-5) +	6.2908e-5 (2.71e-5) +	6.5583e-6 (3.93e-6) −	2.4420e-4 (7.21e-5) +	
	Friedman Ranking		3.6000	4.0500	4.6000	2.7000	8.5500	6.1000	5.3000	1.6500	8.4500	2.1542e-22
	15	24	8.7511e-10 (1.14e-9)	4.7565e-9 (2.13e-8) +	1.7577e-9 (3.19e-9) ≈	5.6376e-9 (5.08e-9) +	3.1221e-7 (2.11e-8) +	1.0853e-8 (1.17e-8) ≈	0.0000e+0 (0.00e+0) −	4.0503e-11 (1.87e-11) −	8.9383e-8 (4.09e-8) +	
	Friedman Ranking		4.6000	1.9750	5.0000	6.1500	9.0000	5.1750	1.6500	3.6000	7.8500	2.8636e-24
CSDTLZ4	3	12	4.8566e-1 (1.33e-1)	5.6315e-1 (6.28e-4) ≈	5.5040e-1 (4.78e-2) +	5.1919e-1 (8.85e-2) ≈	4.0418e-1 (1.66e-1) −	5.3876e-1 (1.05e-1) ≈	3.2652e-1 (1.80e-1) −	4.7081e-1 (1.04e-1) −	5.5504e-1 (1.92e-3) +	
	Friedman Ranking		6.2000	8.1000	5.6000	6.7500	2.3000	6.6000	1.6000	3.3000	4.5500	6.6910e-19
	10	19	9.2104e-1 (1.53e-2)	9.5161e-1 (5.00e-3) +	9.6331e-1 (6.69e-3) +	8.8117e-1 (3.59e-2) −	9.3916e-1 (1.09e-2) +	8.8353e-1 (1.33e-2) −	9.4448e-1 (2.82e-3) +	9.1310e-1 (1.67e-2) ≈	9.2453e-1 (6.35e-3) ≈	
	Friedman Ranking		4.2500	7.9500	8.8000	1.8000	5.9500	1.6000	6.7000	3.6000	4.3500	6.7591e-26
	15	24	9.7591e-1 (1.93e-2)	9.8807e-1 (1.10e-3) +	9.9175e-1 (1.95e-3) +	9.1958e-1 (1.97e-2) −	9.8418e-1 (7.33e-3) +	8.9348e-1 (2.06e-2) −	9.8123e-1 (1.38e-3) ≈	8.9776e-1 (2.21e-2) −	9.8808e-1 (8.59e-4) +	
	Friedman Ranking		4.7000	7.1500	8.8000	2.7500	6.3500	1.6500	4.8000	1.7500	7.0500	4.1350e-26
+/ − / ≈				25/14/12	18/17/16	12/29/10	9/38/4	13/26/12	14/33/4	13/29/9	15/28/8	
Average Ranking			6.1225	6.3745	5.7088	5.1465	3.0642	5.6500	3.4823	4.9029	4.5480	

Table 12: Statistical results (means and standard deviations) of the obtained IGD+ values on the WFG test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	M	D	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
WFG1	3	12	9.7817e-2 (6.79e-3)	9.5726e-2 (6.30e-3) ≈	9.5794e-2 (3.85e-3) ≈	1.1455e-1 (3.15e-2) −	9.0944e-1 (1.68e-1) −	9.7952e-2 (1.33e-2) ≈	3.7045e-1 (2.99e-2) −	1.7671e-1 (4.37e-2) −	9.3555e-2 (4.38e-3) +	
	Friedman Ranking		3.6000	3.0000	3.4500	5.5500	9.0000	3.1000	8.0000	6.9500	2.3500	6.4687e-24
	10	19	6.2733e-1 (7.50e-2)	5.5947e-1 (1.03e-1) ≈	5.2654e-1 (2.30e-2) +	1.2062e+0 (2.83e-1) −	1.2798e+0 (2.39e-1) −	3.7488e-1 (2.73e-2) +	5.4610e-1 (5.25e-2) +	4.4848e-1 (7.54e-2) +	5.7943e-1 (2.57e-2) +	
	Friedman Ranking		6.2500	4.7500	3.8500	8.3500	8.6500	1.0500	4.4500	2.3500	5.3000	3.0061e-25
	15	24	6.8043e-1 (1.54e-1)	5.9947e-1 (1.04e-1) ≈	5.2295e-1 (7.84e-2) +	7.1999e-1 (7.70e-2) ≈	4.2076e+0 (4.37e+0) −	3.9643e-1 (6.32e-2) +	7.9688e-1 (9.96e-2) −	4.3721e-1 (7.45e-2) +	8.3235e-1 (6.16e-2) −	
	Friedman Ranking		5.3500	4.4000	3.1000	5.5500	9.0000	1.6000	6.6000	1.9500	7.4500	7.2427e-25
WFG2	3	12	1.0115e-1 (2.03e-3)	1.0179e-1 (2.79e-3) ≈	9.4665e-2 (2.12e-3) +	1.0234e-1 (3.97e-3) ≈	1.2064e-1 (4.66e-3) −	1.0018e-1 (2.42e-3) ≈	1.3601e-1 (2.19e-2) −	1.0072e-1 (4.40e-3) ≈	7.9830e-2 (3.87e-3) +	
	Friedman Ranking		4.9000	5.4000	2.3500	5.3000	8.3000	4.4000	8.7000	4.6500	1.0000	4.7365e-24
	10	19	8.4200e-1 (1.33e-1)	9.9207e-1 (1.00e-1) −	6.1561e-1 (3.17e-2) +	1.1474e+0 (4.28e-1) −	1.5426e+1 (3.50e+0) −	5.7445e-1 (3.71e-2) +	5.9284e-1 (8.80e-2) +	3.6490e-1 (1.94e-2) +	4.9263e-1 (1.04e-1) +	
	Friedman Ranking		6.3000	7.4500	4.3500	7.1500	9.0000	3.6500	3.5500	1.0500	2.5000	1.8065e-27
	15	24	1.0623e+0 (4.90e-1)	1.3388e+0 (4.66e-1) −	7.8774e-1 (9.87e-2) ≈	6.3767e-1 (1.20e-1) +	2.5023e+1 (5.19e+0) −	8.9428e-1 (2.25e-1) ≈	7.7911e-1 (2.02e-1) +	5.4958e-1 (1.68e-1) +	1.3186e+0 (3.29e-1) −	
	Friedman Ranking		5.4000	7.1000	4.0000	2.2000	9.0000	4.8000	3.7500	1.6000	7.1500	8.9940e-25
WFG3	3	12	7.3519e-2 (9.30e-3)	5.7434e-2 (4.91e-3) +	9.2767e-2 (1.03e-2) −	4.3558e-2 (2.76e-3) +	5.2119e-2 (3.10e-3) +	4.9062e-2 (7.05e-3) +	2.8439e-2 (1.81e-3) +	7.2681e-2 (7.04e-3) ≈	7.1479e-2 (8.75e-3) ≈	
	Friedman Ranking		7.0500	4.7500	8.9500	2.4000	3.7500	3.3000	1.0000	6.9500	6.8500	2.3631e-27
	10	19	2.1140e+0 (2.86e-1)	1.5132e+0 (2.47e-1) +	1.9710e+0 (2.20e-1) +	2.1027e+0 (4.40e-1) ≈	1.0000e+1 (6.31e-5) −	1.1463e+0 (1.99e-1) +	1.8184e-1 (2.24e-2) +	2.4432e+0 (6.01e-1) −	1.0159e+0 (2.98e-1) +	
	Friedman Ranking		6.5000	4.1000	5.8000	6.3500	9.0000	2.7500	1.0000	7.0000	2.5000	1.5469e-26
	15	24	3.6423e+0 (4.77e-1)	2.6174e+0 (5.30e-1) +	3.2326e+0 (4.20e-1) +	3.1927e+0 (7.23e-1) +	1.5000e+1 (1.07e-4) −	3.2784e+0 (5.76e-1) +	2.7988e-1 (4.49e-2) +	3.6828e+0 (5.18e-1) ≈	1.4950e+0 (8.98e-1) +	
	Friedman Ranking		6.4000	3.6500	5.2000	5.4000	9.0000	5.5000	1.0000	6.6000	2.2500	5.9416e-23
WFG4	3	12	1.2586e-1 (7.30e-4)	1.2746e-1 (1.56e-3) −	1.3701e-1 (2.40e-3) −	1.2929e-1 (2.52e-3) −	2.1640e-1 (8.45e-3) −	1.2596e-1 (9.72e-4) ≈	2.1468e-1 (1.51e-2) −	1.3748e-1 (2.00e-3) −	1.3641e-1 (2.63e-3) −	
	Friedman Ranking		1.6500	2.9500	6.1000	3.7500	8.5000	1.7500	8.5000	6.1500	5.6500	9.6515e-28
	10	19	2.8347e+0 (2.83e-2)	2.5608e+0 (3.70e-2) +	2.7795e+0 (3.50e-2) +	2.5049e+0 (5.17e-2) +	1.4680e+1 (3.86e+0) −	2.7064e+0 (4.10e-2) +	2.7667e+0 (5.97e-2) +	2.8180e+0 (4.73e-2) ≈	2.8163e+0 (7.63e-2) ≈	
	Friedman Ranking		7.1000	1.7500	5.1000	1.2500	9.0000	3.4000	5.0500	6.3500	6.0000	3.3288e-25
	15	24	5.4696e+0 (1.76e-1)	5.0455e+0 (1.90e-1) +	4.7113e+0 (1.25e-1) +	5.2867e+0 (3.46e-1) +	2.5039e+1 (1.22e+0) −	6.1915e+0 (2.55e-1) −	5.0177e+0 (9.46e-2) +	6.0117e+0 (2.31e-1) −	5.4122e+0 (3.53e-1) ≈	
	Friedman Ranking		5.4000	3.0000	1.1500	4.1500	9.0000	7.5500	2.8000	7.2500	4.7000	3.4804e-26
WFG5	3	12	1.6563e-1 (6.93e-4)	1.6941e-1 (2.05e-3) −	1.7131e-1 (1.83e-3) −	1.7134e-1 (3.20e-3) −	2.1967e-1 (5.32e-3) −	1.6551e-1 (2.10e-3) ≈	2.4371e-1 (8.92e-3) −	1.7285e-1 (2.08e-3) −	1.7483e-1 (2.44e-3) −	
	Friedman Ranking		1.6500	3.4500	4.7000	4.7000	8.0000	1.5500	9.0000	5.5000	6.4500	1.1604e-26
	10	19	2.8406e+0 (3.08e-2)	2.5483e+0 (2.75e-2) +	2.8511e+0 (3.75e-2) ≈	2.6716e+0 (9.91e-2) +	1.5660e+1 (8.23e-1) −	2.6983e+0 (2.55e-2) +	2.8539e+0 (5.41e-2) ≈	2.8723e+0 (3.97e-2) −	2.9797e+0 (1.81e-1) −	
	Friedman Ranking		4.9000	1.1000	5.6500	2.5000	9.0000	2.6500	5.7500	6.0500	7.4000	8.9940e-25
	15	24	5.6705e+0 (1.11e-1)	4.6403e+0 (1.34e-1) +	4.7581e+0 (1.18e-1) +	5.1903e+0 (1.50e-1) +	2.5494e+1 (9.49e-10) −	5.8113e+0 (1.85e-1) −	5.0743e+0 (1.07e-1) +	6.0609e+0 (8.00e-2) −	5.1995e+0 (3.45e-1) +	
	Friedman Ranking		6.2000	1.2500	1.8000	4.3500	9.0000	6.7500	3.5500	7.8500	4.2500	2.6493e-28
WFG6	3	12	1.8053e-1 (1.27e-2)	1.8570e-1 (2.42e-2) ≈	1.8896e-1 (1.61e-2) −	1.9335e-1 (2.45e-2) −	2.6196e-1 (7.23e-2) −	1.8160e-1 (1.94e-2) ≈	2.7722e-1 (1.69e-2) −	1.9755e-1 (1.53e-2) −	1.7929e-1 (1.61e-2) ≈	
	Friedman Ranking		3.4000	3.9500	4.5500	4.5500	7.1500	3.9000	8.5000	5.6000	3.4000	2.0732e-11
	10	19	2.9077e+0 (2.58e-2)	2.5548e+0 (4.56e-2) +	2.8987e+0 (4.59e-2) ≈	2.5540e+0 (6.74e-2) +	1.1832e+1 (3.05e+0) −	2.7312e+0 (2.99e-2) +	2.8021e+0 (6.31e-2) +	2.9071e+0 (4.87e-2) ≈	3.3266e+0 (6.03e-1) −	
	Friedman Ranking		6.0500	1.6000	5.8500	1.4000	9.0000	3.0500	4.2000	6.3500	7.5000	1.1619e-27
	15	24	5.5374e+0 (1.44e-1)	4.6559e+0 (1.27e-1) +	4.5524e+0 (2.71e-1) +	5.0457e+0 (7.23e-1) +	2.3457e+1 (2.62e+0) −	5.9630e+0 (2.42e-1) −	5.0111e+0 (1.03e-1) +	5.8963e+0 (2.78e-1) −	6.0342e+0 (1.49e+0) ≈	
	Friedman Ranking		5.6500	1.9000	1.6500	3.3000	9.0000	7.1000	3.9000	6.9000	5.6000	8.0710e-25
WFG7	3	12	1.2677e-1 (8.71e-4)	1.2808e-1 (1.13e-3) −	1.3885e-1 (2.05e-3) −	1.3279e-1 (2.70e-3) −	1.8344e-1 (6.05e-3) −	1.2613e-1 (1.74e-3) ≈	2.3386e-1 (2.15e-2) −	1.3571e-1 (1.63e-3) −	1.3139e-1 (2.62e-3) −	
	Friedman Ranking		1.8500	2.8500	6.9000	4.7500	8.0000	1.5000	9.0000	5.7500	4.4000	1.6389e-28
	10	19	2.8752e+0 (3.96e-2)	2.7860e+0 (5.69e-2) +	2.7810e+0 (3.66e-2) +	2.5364e+0 (6.80e-2) +	1.5935e+1 (8.62e-4) −	2.7290e+0 (4.30e-2) +	2.5497e+0 (1.05e-1) +	2.8604e+0 (4.91e-2) ≈	3.3446e+0 (6.48e-1) −	
	Friedman Ranking		6.9000	4.7000	4.4500	1.5000	9.0000	3.6000	1.5500	6.4500	6.8500	7.1131e-26
	15	24	5.5321e+0 (6.79e-1)	5.7879e+0 (1.96e-1) ≈	4.5192e+0 (1.41e-1) +	5.6597e+0 (5.33e-1) ≈	2.5321e+1 (7.58e-1) −	6.1932e+0 (2.54e-1) −	4.8308e+0 (1.82e-1) +	5.9765e+0 (2.59e-1) ≈	6.4966e+0 (1.38e+0) ≈	
	Friedman Ranking		4.4500	5.0500	1.2000	4.4500	9.0000	6.9000	2.1000	6.1000	5.7500	4.0608e-22
WFG8	3	12	2.3091e-1 (2.33e-3)	2.2061e-1 (2.82e-3) +	2.4780e-1 (2.97e-3) −	2.4376e-1 (3.05e-3) −	2.8503e-1 (7.60e-3) −	2.2384e-1 (4.16e-3) +	3.4299e-1 (1.26e-2) −	2.4481e-1 (2.83e-3) −	2.4611e-1 (8.58e-3) −	
	Friedman Ranking		2.9500	1.2000	6.2500	4.9000	8.0000	1.9000	9.0000	5.3000	5.5000	7.8647e-28
	10	19	2.9749e+0 (9.21e-2)	2.8604e+0 (1.08e-1) +	3.0743e+0 (1.09e-1) −	3.8846e+0 (5.76e-1) −	1.5049e+1 (1.20e+0) −	3.0712e+0 (4.76e-2) −	3.4461e+0 (1.04e-1) −	2.9263e+0 (7.26e-2) ≈	5.0652e+0 (2.94e-1) −	
	Friedman Ranking		2.9000	1.6500	4.0500	6.8000	9.0000	4.2500	6.2500	2.1500	7.9500	1.3809e-27
	15	24	5.4490e+0 (5.43e-1)	5.4796e+0 (2.41e-1) ≈	4.9563e+0 (1.52e-1) +	8.3950e+0 (9.54e-1) −	2.4636e+1 (1.04e+0) −	6.6942e+0 (3.21e-1) −	5.9735e+0 (1.64e-1) −	6.0636e+0 (4.27e-1) −	9.6510e+0 (7.38e-1) −	
	Friedman Ranking		2.6500	2.7000	1.3000	7.1000	9.0000	5.9000	4.1000	4.4000	7.8500	9.5288e-28
WFG9	3	12	1.4376e-1 (2.63e-3)	1.6098e-1 (5.45e-2) ≈	1.4980e-1 (2.40e-3) −	1.6514e-1 (3.93e-2) −	2.2733e-1 (6.95e-2) −	1.4302e-1 (5.78e-3) ≈	2.2058e-1 (1.13e-2) −	1.4986e-1 (3.25e-3) −	1.4377e-1 (3.84e-3) ≈	
	Friedman Ranking		2.5000	3.2500	5.4500	6.3500	8.1000	2.5500	8.6500	5.3500	2.8000	2.4610e-22
	10	19	2.9265e+0 (4.71e-2)	3.0384e+0 (1.45e-1) −	2.9155e+0 (2.93e-2) ≈	3.1928e+0 (1.80e-1) −	1.5564e+1 (1.15e+0) −	2.8863e+0 (6.66e-2) +	3.0922e+0 (7.07e-2) −	2.8045e+0 (6.06e-2) +	3.6666e+0 (2.88e-1) −	
	Friedman Ranking		3.7000	4.9500	3.4500	6.3000	9.0000	2.4500	6.1000	1.1500	7.9000	1.7689e-26
	15	24	6.1232e+0 (2.64e-1)	5.8982e+0 (3.29e-1) +	4.7401e+0 (2.05e-1) +	5.8552e+0 (5.64e-1) +	2.5616e+1 (5.68e-2) −	6.4625e+0 (3.76e-1) −	5.5572e+0 (1.64e-1) +	5.7702e+0 (1.57e-1) +	6.3261e+0 (4.57e-1) ≈	
	Friedman Ranking		5.8500	4.7000	1.0000	4.1500	9.0000	7.2000	2.6500	4.1500	6.3000	6.2910e-23
+/ − / ≈				13/6/8	14/8/5	11/12/4	1/26/0	12/7/8	14/12/1	6/13/8	7/12/8	
Average Ranking			4.7222	3.5777	4.1351	4.6111	8.5351	3.8555	5.1370	5.1074	5.3185	

51 total test instances. The other algorithms also have strong performances on one or more instances. For example, iRVEA obtains good performances in 17 instances in terms of IGD+ and on 14 instances in terms of the HV. The performances of MOEA/D-AWA, MOEA/D-PaS and RPEA are very different from those of the other algorithms since MOEA/D-AWA only obtains 8 good IGD+ values and 6 good HV values, MOEA/D-PaS only obtains 2 good IGD+ values and 5 good HV values, and RPEA only obtains 1 good IGD+ value and 5 good HV values. MOEA/D-URAW performs better on the IGD+ indicator than on the HV indicator because it obtains 12 good IGD+ values but 6 good HV values. In contrast to that of MOEA/D-UARW, the population obtained by DEA-GNG is more inclined to the HV indicator. For RVEAa, its results on the whole DTLZ test suite are ideal, especially in high-dimensional instances.

5.7. Comparisons on WFG Test Suite

We also test the nine decomposition-based algorithms with regard to solving the WFG test suite as shown in Tables 12 and 13. In this test suite, AdaW

Table 13: Statistical results (means and standard deviations) of the obtained HV values on the WFG test instances, where the best mean among the algorithms for each instance is highlighted in a gray background.

Problem	*M*	*D*	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG	P-value
WFG1	3	12	9.3834e-1 (4.52e-3)	9.4357e-1 (4.20e-3) +	9.4222e-1 (2.29e-3) +	9.3035e-1 (2.27e-2) −	5.7159e-1 (7.60e-2) −	9.3834e-1 (3.87e-3) ≈	8.5299e-1 (1.44e-2) −	8.8591e-1 (2.27e-2) −	9.3095e-1 (2.94e-3) −	
	Friedman Ranking		6.0500	8.4500	7.9000	5.5000	1.0000	6.4500	2.1000	2.9500	4.6000	2.2547e-26
	10	19	9.9916e-1 (9.75e-4)	9.9994e-1 (7.03e-5) +	9.9952e-1 (1.54e-4) ≈	9.8442e-1 (3.77e-2) ≈	9.9411e-1 (1.17e-2) −	9.9974e-1 (1.32e-4) +	9.9721e-1 (5.95e-4) −	9.8432e-1 (3.59e-2) −	9.9229e-1 (2.59e-3) −	
	Friedman Ranking		5.7000	8.3500	5.9500	6.6500	3.4000	7.2000	2.9000	3.4000	1.4500	9.9836e-21
	15	24	9.7358e-1 (4.60e-2)	9.9955e-1 (1.44e-4) +	9.9932e-1 (3.60e-4) ≈	1.0000e+0 (3.81e-6) +	8.5096e-1 (1.84e-1) −	9.9981e-1 (1.40e-4) +	9.9883e-1 (4.12e-4) ≈	9.9163e-1 (2.35e-2) ≈	9.9329e-1 (1.58e-3) +	
	Friedman Ranking		4.4000	6.4000	6.0000	9.0000	1.3500	7.7500	4.6000	3.2500	2.2500	1.6300e-25
WFG2	3	12	9.3420e-1 (7.37e-4)	9.3644e-1 (6.07e-4) +	9.3562e-1 (7.78e-4) +	9.2209e-1 (3.24e-3) −	9.2480e-1 (2.99e-3) −	9.3395e-1 (1.27e-3) ≈	8.8665e-1 (1.10e-2) −	9.2421e-1 (2.01e-3) −	9.2843e-1 (2.32e-3) −	
	Friedman Ranking		6.6500	8.8000	8.0000	2.5500	3.6000	6.5500	1.0000	3.1000	4.7500	2.0905e-28
	10	19	9.9273e-1 (1.96e-3)	9.9879e-1 (4.74e-4) +	9.9558e-1 (1.23e-3) +	9.9773e-1 (1.29e-3) +	2.6668e-1 (1.93e-1) −	9.9903e-1 (5.03e-4) +	9.8382e-1 (5.31e-3) −	9.8729e-1 (1.81e-3) −	9.8391e-1 (4.61e-3) −	
	Friedman Ranking		5.0500	8.1000	5.9500	7.4000	1.0000	8.4500	2.9000	3.5500	2.6000	1.5177e-28
	15	24	9.9307e-1 (1.65e-3)	9.9628e-1 (1.37e-3) +	9.9494e-1 (1.65e-3) +	9.9609e-1 (1.88e-3) +	2.1315e-1 (1.74e-1) −	9.9624e-1 (2.16e-3) +	9.8713e-1 (7.18e-3) −	9.7874e-1 (8.23e-3) −	9.7750e-1 (9.97e-3) −	
	Friedman Ranking		5.1000	7.6000	6.8000	7.3000	1.0000	7.5500	4.1500	2.6500	2.8500	3.1323e-24
WFG3	3	12	3.9299e-1 (3.93e-3)	3.9791e-1 (2.31e-3) +	3.7748e-1 (6.07e-3) −	4.0320e-1 (1.84e-3) +	4.0437e-1 (1.23e-3) +	4.0230e-1 (3.03e-3) +	4.1251e-1 (7.53e-4) +	3.8654e-1 (4.41e-3) −	3.9389e-1 (3.70e-3) ≈	
	Friedman Ranking		3.7500	4.7500	1.1000	6.9500	7.5000	6.4000	9.0000	2.0000	3.5500	2.8792e-28
	10	19	2.8085e-3 (5.86e-3)	0.0000e+0 (0.00e+0) −	0.0000e+0 (0.00e+0) −	9.0348e-2 (1.15e-2) +	9.0145e-2 (6.48e-4) +	5.0740e-3 (8.96e-3) ≈	1.4507e-1 (9.11e-3) +	0.0000e+0 (0.00e+0) −	0.0000e+0 (0.00e+0) −	
	Friedman Ranking		3.9500	3.2000	3.2000	7.2000	7.8000	4.2500	9.0000	3.2000	3.2000	4.6321e-28
	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) ≈	4.6833e-2 (2.60e-2) +	0.0000e+0 (0.00e+0) ≈	2.1122e-2 (2.40e-2) +	0.0000e+0 (0.00e+0) ≈	0.0000e+0 (0.00e+0) ≈	
	Friedman Ranking		4.2000	4.2000	4.2000	4.2000	8.4000	4.2000	7.2000	4.2000	4.2000	5.8086e-24
WFG4	3	12	5.6280e-1 (5.56e-4)	5.6157e-1 (1.10e-3) −	5.5558e-1 (1.29e-3) −	5.5830e-1 (1.35e-3) −	5.0983e-1 (6.60e-3) −	5.5983e-1 (7.54e-4) −	5.3452e-1 (9.35e-3) −	5.5210e-1 (1.84e-3) −	5.5181e-1 (3.33e-3) −	
	Friedman Ranking		8.8000	8.1500	5.0000	6.1000	1.0500	6.8500	1.9500	3.5500	3.5500	1.8196e-29
	10	19	9.3123e-1 (2.60e-3)	9.1506e-1 (1.07e-2) −	9.4305e-1 (3.67e-3) +	9.5369e-1 (1.56e-3) +	1.5089e-1 (1.85e-1) −	9.4615e-1 (2.29e-3) +	8.9068e-1 (8.84e-3) −	9.1045e-1 (4.35e-3) −	8.2788e-1 (9.79e-3) −	
	Friedman Ranking		5.9000	4.7500	7.3000	8.9500	1.0000	7.7500	3.0000	4.3500	2.0000	7.9559e-30
	15	24	9.5152e-1 (5.78e-3)	8.6824e-1 (1.84e-2) −	9.7913e-1 (2.48e-3) +	9.6312e-1 (9.85e-3) +	1.1385e-1 (6.32e-2) −	9.4992e-1 (6.53e-3) ≈	9.4091e-1 (5.98e-3) −	9.2281e-1 (1.08e-2) −	8.8421e-1 (2.13e-2) −	
	Friedman Ranking		6.7000	2.4000	9.0000	7.6500	1.0000	6.2000	5.3000	4.1000	2.6500	1.3611e-28
WFG5	3	12	5.2208e-1 (8.91e-4)	5.1827e-1 (2.40e-3) −	5.1945e-1 (1.01e-3) −	5.1387e-1 (4.25e-3) −	4.8225e-1 (4.62e-3) −	5.1638e-1 (3.12e-3) −	4.9791e-1 (4.36e-3) −	5.1589e-1 (1.64e-3) −	5.0521e-1 (2.69e-3) −	
	Friedman Ranking		9.0000	6.7500	7.4000	4.9500	1.0500	5.5500	2.0500	5.3500	2.9000	1.1604e-26
	10	19	8.7216e-1 (2.22e-3)	8.6351e-1 (2.96e-3) −	8.9412e-1 (1.49e-3) +	8.8008e-1 (4.53e-3) +	9.5218e-2 (3.85e-2) −	8.7407e-1 (3.61e-3) +	8.0513e-1 (7.38e-3) −	8.7084e-1 (2.50e-3) ≈	7.8008e-1 (1.45e-2) −	
	Friedman Ranking		6.0000	4.0500	9.0000	7.7500	1.0000	6.6500	3.0000	5.5500	2.0000	4.0034e-29
	15	24	8.8280e-1 (3.36e-3)	8.3363e-1 (9.07e-3) −	9.1734e-1 (4.27e-4) +	8.8195e-1 (3.21e-3) ≈	7.9015e-2 (2.81e-9) −	8.7420e-1 (7.75e-3) −	8.3128e-1 (1.04e-2) −	8.9017e-1 (4.75e-3) +	8.3714e-1 (9.23e-3) −	
	Friedman Ranking		6.6000	2.8500	9.0000	6.3000	1.0000	5.2500	2.9000	7.8500	3.2500	1.3699e-28
WFG6	3	12	5.0961e-1 (1.05e-2)	5.0776e-1 (1.97e-2) ≈	5.0739e-1 (1.37e-2) ≈	5.0138e-1 (1.99e-2) ≈	4.6492e-1 (6.29e-2) ≈	5.0802e-1 (1.64e-2) ≈	4.8449e-1 (1.22e-2) −	4.9680e-1 (1.21e-2) −	5.0755e-1 (1.36e-2) ≈	
	Friedman Ranking		5.9500	5.7500	5.6500	5.0500	4.9000	5.5500	2.3000	4.1000	5.7500	3.2053e-04
	10	19	8.4414e-1 (1.49e-2)	8.5185e-1 (2.77e-2) ≈	8.6695e-1 (1.94e-2) +	8.6253e-1 (3.82e-2) ≈	2.1484e-1 (1.45e-1) −	8.6240e-1 (1.85e-2) +	8.3261e-1 (2.59e-2) ≈	8.3460e-1 (1.81e-2) ≈	7.5820e-1 (3.45e-2) −	
	Friedman Ranking		5.5500	6.0500	7.3500	6.7000	1.0000	6.7000	4.9000	4.7500	2.0000	8.9940e-25
	15	24	8.6514e-1 (2.16e-2)	8.1701e-1 (3.67e-2) −	8.8843e-1 (1.74e-2) +	8.7776e-1 (3.52e-2) ≈	1.4155e-1 (7.35e-2) −	8.5793e-1 (3.29e-2) ≈	8.6564e-1 (2.81e-2) ≈	8.5540e-1 (2.38e-2) ≈	8.3091e-1 (5.68e-2) −	
	Friedman Ranking		5.8000	3.0000	8.2000	6.4000	1.0000	5.4000	5.7500	5.1500	4.3000	3.1010e-16
WFG7	3	12	5.6212e-1 (5.98e-4)	5.6197e-1 (9.57e-4) ≈	5.5667e-1 (1.03e-3) −	5.5591e-1 (1.79e-3) −	5.3057e-1 (3.44e-3) −	5.6087e-1 (8.63e-4) −	5.2987e-1 (1.28e-2) −	5.5451e-1 (1.24e-3) −	5.5271e-1 (1.71e-3) −	
	Friedman Ranking		8.5000	8.2000	5.5500	5.0000	1.4500	7.2500	1.5500	4.2500	3.2500	2.7530e-28
	10	19	9.3248e-1 (2.79e-3)	8.9368e-1 (1.07e-2) −	9.5306e-1 (1.98e-3) +	9.5428e-1 (1.22e-3) +	9.0836e-2 (1.40e-4) −	9.4425e-1 (2.92e-3) +	9.1233e-1 (8.99e-3) −	9.2319e-1 (2.58e-3) −	8.3334e-1 (2.40e-2) −	
	Friedman Ranking		6.0000	3.1500	8.4000	8.6000	1.0000	7.0000	4.0500	4.8000	2.0000	8.9940e-25
	15	24	9.5134e-1 (1.98e-2)	7.5134e-1 (3.87e-2) −	9.8696e-1 (1.23e-3) +	9.4329e-1 (4.38e-2) ≈	9.9454e-2 (3.84e-2) −	9.4370e-1 (9.93e-3) ≈	9.5036e-1 (7.26e-3) ≈	9.0977e-1 (1.08e-2) −	9.0635e-1 (2.92e-2) −	
	Friedman Ranking		6.3000	2.0000	9.0000	6.7000	1.0000	5.9000	6.6500	3.6000	3.8500	2.7133e-26
WFG8	3	12	4.8211e-1 (1.15e-3)	4.8325e-1 (2.55e-3) ≈	4.7246e-1 (2.06e-3) −	4.7149e-1 (1.91e-3) −	4.5078e-1 (4.69e-3) −	4.8142e-1 (2.61e-3) ≈	4.4535e-1 (4.53e-3) −	4.7122e-1 (2.17e-3) −	4.6605e-1 (6.27e-3) −	
	Friedman Ranking		7.9000	8.3500	5.0500	4.7500	1.9000	7.7500	1.1500	4.5500	3.6000	4.2012e-27
	10	19	8.7736e-1 (2.65e-2)	8.0909e-1 (2.18e-2) −	8.4154e-1 (7.34e-2) ≈	7.7304e-1 (2.28e-2) −	1.0118e-1 (1.44e-2) −	8.7598e-1 (4.66e-3) ≈	7.3012e-1 (1.94e-2) −	8.2155e-1 (7.75e-2) −	7.1936e-1 (2.52e-2) −	
	Friedman Ranking		7.9500	5.7000	6.4000	4.4500	1.0000	7.7500	2.8500	6.3500	2.5500	1.4504e-23
	15	24	8.8789e-1 (4.27e-2)	7.5030e-1 (3.56e-2) −	9.0735e-1 (8.07e-2) ≈	8.3579e-1 (2.04e-2) −	9.9101e-2 (1.49e-2) −	9.2024e-1 (4.73e-3) +	7.8652e-1 (1.88e-2) −	7.6811e-1 (1.12e-1) −	7.8453e-1 (3.86e-2) −	
	Friedman Ranking		7.4500	2.9500	7.5000	5.9500	1.0000	8.0500	4.0000	4.1000	4.0000	1.8267e-22
WFG9	3	12	5.4054e-1 (2.59e-3)	5.2605e-1 (3.95e-2) −	5.3875e-1 (2.12e-3) −	5.1829e-1 (2.86e-2) −	4.8457e-1 (5.27e-2) −	5.3521e-1 (4.42e-3) −	5.1625e-1 (5.08e-3) −	5.3261e-1 (3.04e-3) −	5.3309e-1 (3.54e-3) −	
	Friedman Ranking		8.3500	7.0500	7.4500	3.0500	1.3000	5.9500	2.2000	4.8500	4.8000	1.0421e-23
	10	19	7.9568e-1 (3.12e-2)	6.9929e-1 (4.53e-2) −	8.7256e-1 (1.56e-2) +	7.7029e-1 (9.49e-2) ≈	8.9165e-2 (5.10e-2) −	8.0708e-1 (3.73e-2) ≈	8.0221e-1 (8.00e-3) ≈	8.2687e-1 (1.45e-2) +	7.6865e-1 (3.28e-2) −	
	Friedman Ranking		5.2000	2.6500	8.9000	4.9500	1.0000	6.0500	5.3500	7.1500	3.7500	9.3122e-22
	15	24	7.2705e-1 (4.66e-2)	6.4518e-1 (4.26e-2) −	8.8432e-1 (1.55e-2) +	7.7820e-1 (8.02e-2) +	6.4266e-2 (7.19e-3) −	7.3133e-1 (6.41e-2) ≈	8.2577e-1 (8.88e-3) +	7.6921e-1 (4.05e-2) +	8.1546e-1 (7.62e-2) +	
Friedman Ranking			3.9500	2.4500	8.9500	5.8000	1.0000	4.2000	6.7000	4.9500	7.0000	1.6159e-23
+/ − / ≈				7/15/5	14/7/6	10/9/8	3/23/1	10/5/12	4/18/5	3/19/5	2/22/3	
Average Ranking			6.1759	5.4111	6.8222	6.1425	2.1740	6.4666	4.0185	4.3574	3.4314	

Table 14: Wilcoxon's rank sum test for the DMEA-WUA and the other compared algorithms on all test instances.

	Test Suite	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG
IGD+(+/−/≈)	ZDT	1/2/3	0/6/0	0/5/1	0/6/0	0/6/0	0/6/0	0/5/1	0/6/0
	VNT	0/1/3	1/3/0	0/4/0	0/4/0	0/2/2	0/4/0	1/2/1	0/2/2
	UF	0/1/9	2/3/5	0/7/3	4/5/1	0/6/4	1/7/2	0/8/2	2/0/8
	DPF	5/5/5	3/11/1	1/11/3	0/15/0	4/6/5	2/13/0	1/12/2	1/7/7
	DTLZ	25/15/11	23/24/4	15/27/9	2/47/2	19/26/6	15/35/1	20/24/7	13/30/8
	WFG	13/6/8	14/8/5	11/12/4	1/26/0	12/7/8	14/12/1	6/13/8	7/12/8
	Total	44/30/39	43/55/15	27/66/20	7/103/3	35/53/25	32/77/4	28/64/21	23/57/33
HV(+/−/≈)	ZDT	2/3/1	0/5/1	2/2/2	0/6/0	0/6/0	0/5/1	1/4/1	1/4/1
	VNT	3/1/0	1/1/2	2/2/0	1/3/0	2/2/0	0/4/0	1/2/1	1/2/1
	UF	1/0/9	1/3/6	1/6/3	4/5/1	1/5/4	0/5/5	0/8/2	1/1/8
	DPF	5/4/6	5/6/4	1/11/3	0/15/0	4/1/10	1/10/4	2/12/1	5/5/5
	DTLZ	25/14/12	18/17/16	12/29/10	9/38/4	13/26/12	14/33/4	13/29/9	15/28/8
	WFG	7/15/5	14/7/6	10/9/6	3/23/1	10/5/12	4/18/5	4/19/5	2/22/3
	Total	43/37/33	39/39/35	28/58/24	17/90/6	30/45/38	19/75/19	21/73/19	24/63/26

performs best in terms of IGD+, and iRVEA performs the best in terms of the HV. MOEA/D-URAW also performs very well, and it is followed by our algorithm. It can be found from these two tables that our algorithm performs well on the low-dimensional instances, especially for problems with concave scaled Pareto fronts, e.g., the 3-objective WFG4-WFG9 instances. The algorithms that perform best on WFG1 and WFG2 are AdaW and MOEA/D-URAW in terms of the HV indicator and RVEAa in terms of the IGD+ indicator. RPEA has an absolute advantage over the other algorithms on the WFG3 problem, because it obtains the best values in terms of both the IGD+ and HV indicators. For the WFG4-WFG9 problems, iRVEA performs better than the other algorithms in terms of the HV indicator. However, all nine algorithms except MOEA/D-PaS and DEA-GNG have comparable performances in terms of IGD+. On the whole, MOEA/D-PaS is weaker than the other algorithms in terms of the IGD+ indicator, while DEA-GNG performs poorly in terms of the HV indicator.

5.8. Statistical Results

In this section, we summarize the statistical analysis results for all test instances. In this paper, two significant analysis methods, Wilcoxon's rank sum test and Friedman's rank test, are provided. Wilcoxon's rank sum test mainly evaluates the significant differences between our algorithm and the other compared algorithms, while Friedman's rank test mainly evaluates the significant differences over the entire multiple comparisons.

As shown in Table 14, the proportions of all instances on which our DMEA-WUA is significantly better than AdaW, iRVEA, MOEA/D-AWA, MOEA/D-PaS, MOEA/D-URAW, RPEA, RVEAa, DEA-GNG are 30/113,

Table 15: Friedman's rank test for the nine algorithms on all test instances.

Indicator	Test Suite	DMEA-WUA	AdaW	iRVEA	MOEADAWA	MOEADPaS	MOEADURAW	RPEA	RVEAa	DEAGNG
IGD+	ZDT	2.0416	2.2750	5.2375	4.8375	8.1000	4.8833	7.4500	5.3580	4.8166
	VNT	2.5375	2.9000	5.1375	6.8250	8.8000	3.6125	6.9875	4.1500	4.0500
	UF	3.8150	4.1050	4.8800	5.8200	5.2900	5.3500	5.8250	5.8600	4.0550
	DPF	3.0433	2.7033	4.8500	6.0000	8.8033	2.9066	5.9633	6.5866	4.1433
	DTLZ	3.8460	3.2696	4.3107	5.1872	7.8950	4.0264	6.4323	4.9088	5.1235
	WFG	4.7222	3.5777	4.1351	4.6111	8.5351	3.8555	5.1370	5.1074	5.3185
	Average Ranking	3.3342	3.1384	4.7584	5.5468	7.9039	4.1057	6.2991	5.3284	4.5844
HV	ZDT	7.2166	7.3916	5.0875	5.3958	2.0916	3.7583	3.0250	4.9500	6.0833
	VNT	5.6125	6.4625	6.1125	5.3750	3.1125	5.2750	3.1625	4.6000	5.2875
	UF	6.0250	5.8675	5.0650	4.5575	4.8350	4.7625	4.0350	4.1425	5.7100
	DPF	6.3900	6.7733	6.1283	3.9366	1.3533	6.7500	4.3666	3.5300	5.7716
	DTLZ	6.1225	6.3745	5.7088	5.1465	3.0642	5.6500	3.4823	4.9029	4.5480
	WFG	6.1759	5.4111	6.8222	6.1425	2.1740	6.4666	4.0185	4.3574	3.4314
	Average Ranking	6.2570	6.3800	5.8207	5.0923	2.7717	5.4437	3.6816	4.4138	5.1386

55/113, 66/113, 103/113, 53/113, 77/113, 64/113 and 57/113 according to the IGD+ indicator. In contrast, the proportions of all instances on which our DMEA-WUA is significantly poorer than AdaW, iRVEA, MOEA/D-AWA, MOEA/D-PaS, MOEA/D-URAW, RPEA, RVEAa, and DEA-GNG are 44/113, 43/113, 27/113, 7/113, 35/113, 32/113, 28/113 and 23/113, respectively, according to the IGD+ indicator. For the HV indicator, the proportions of all instances on which our DMEA-WUA is significantly better than AdaW, iRVEA, MOEA/D-AWA, MOEA/D-PaS, MOEA/D-URAW, RPEA, RVEAa, and DEA-GNG are 37/113, 39/113, 58/113, 90/113, 45/113, 75/113, 73/113 and 63/113, respectively. In contrast, the proportions of the whole instances on which our DMEA-WUA is significantly poorer than AdaW, iRVEA, MOEA/D-AWA, MOEA/D-PaS, MOEA/D-URAW, RPEA, RVEAa, and DEA-GNG are 43/113, 39/113, 28/113, 17/113, 30/113, 19/113, 21/113 and 24/113, respectively. Therefore, it can be seen from these numerical results that our algorithm is only inferior to AdaW in terms of IGD+ and HV indicators but better than the other algorithms on the whole test instances in terms of IGD+. In addition, for the HV indicator, our algorithm has the same performance as iRVEA and is better than other algorithms.

Table 15 shows Friedman's rank test for the IGD+ and HV results. For the IGD+ indicator, the comprehensive ranking of all algorithms from high to low is AdaW, DMEA-WUA, MOEA/D-URAW, DEA-GNG, iRVEA, RVEAa, MOEA/D-AWA, RPEA and MOEA/D-PaS. For the HV indicator, the comprehensive ranking of all algorithms from high to low is AdaW, DMEA-WUA, iRVEA, MOEA/D-URAW, DEA-GNG, MOEA/D-AWA, RVEAa, RPEA, MOEA/D-PaS. Therefore, regardless of whether the IGD+ or the HV, the performances of our algorithm and AdaW are very competitive, and

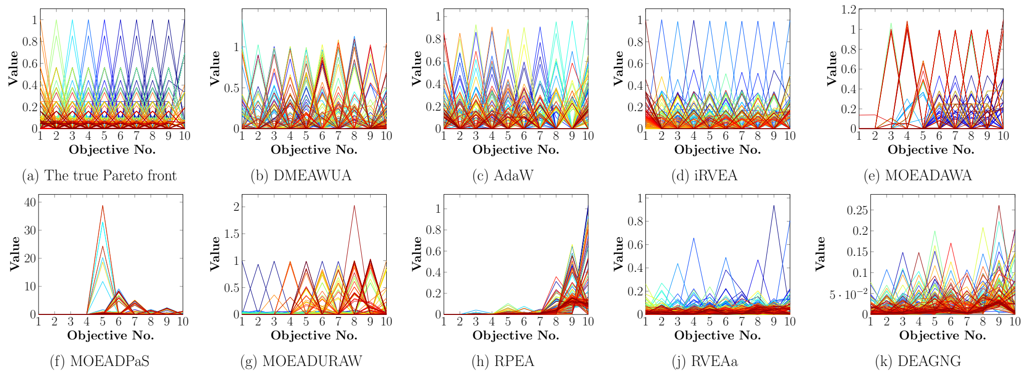


Figure 6: The parallel coordinates of the final solution sets on the 10-objective CDTLZ2 instance.

both are better than the other seven decomposition-based algorithms.

5.9. Visualized Comparison

In this section, we try to intuitively understand the performances of the compared algorithms on different problems. Particularly, we use scatter plots to describe the low-dimensional problems and parallel coordinates to describe the high-dimensional problems.

Fig. 6 plots the final solution sets obtained by all algorithms on the 10-objective CDTLZ2 instance, which has a convex Pareto optimal curve, and the range of each objective is $[0, 1]$. As shown, the populations of all algorithms except MOEA/D-PaS can converge to the true Pareto front, but their distributions are very different. For example, the population distributions of our DMEA-WUA and AdaW are obviously more uniform than those of the other algorithms; the populations obtained by iRVEA, RVEAa and DEA-GNG are mostly concentrated in the middle of the Pareto front; and the populations obtained by MOEA/D-AWA, MOEA/D-URAW and RPEA are mostly located in the 5-10 dimensional regions of the Pareto front.

Fig. 7 plots the final solution sets obtained by all algorithms on the 10-objective DPF4 instance, which has a degenerate Pareto front. As shown, our DMEA-WUA and MOEA/D-URAW obtain the solution sets that are closest to the true Pareto front. the populations obtained by AdaW, iRVEA and DEA-GNG have no solutions in part of the first dimension. The search capabilities of MOEA/D-AWA and RVEAa are weaker than the previous algorithms, because they have large numbers of overlapping solutions. Fi-

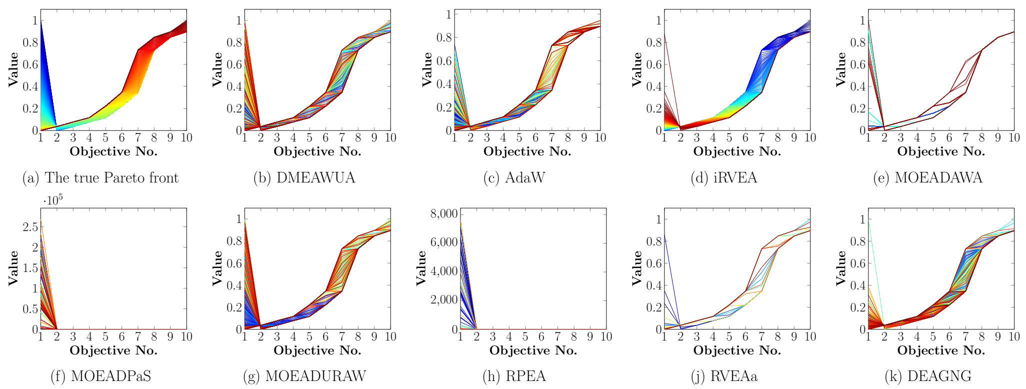


Figure 7: The parallel coordinates of the final solution sets on the 10-objective DPF4 instance.

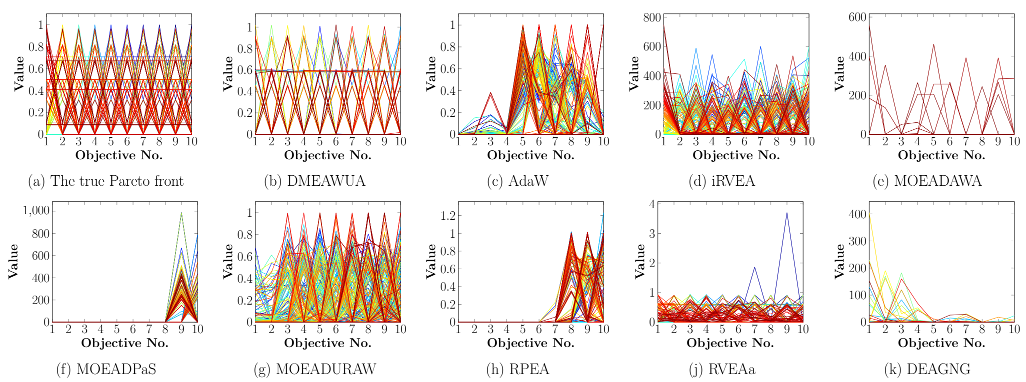
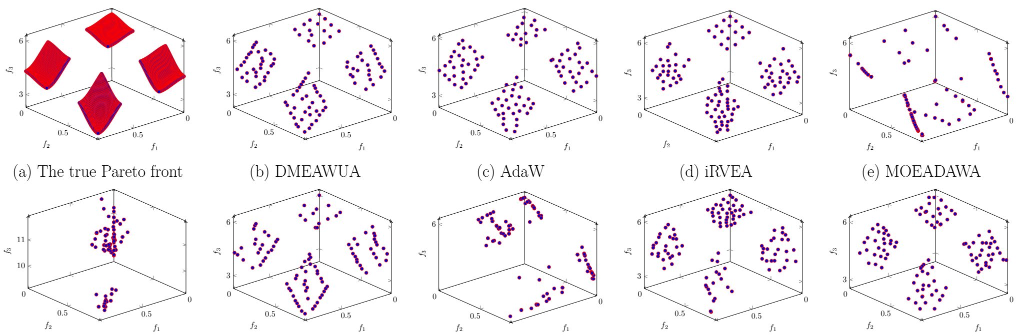


Figure 8: The parallel coordinates of the final solution sets on the 10-objective DTLZ3 instance.

nally, the final populations of MOEA/D-PaS and RPEA are obviously not convergent.

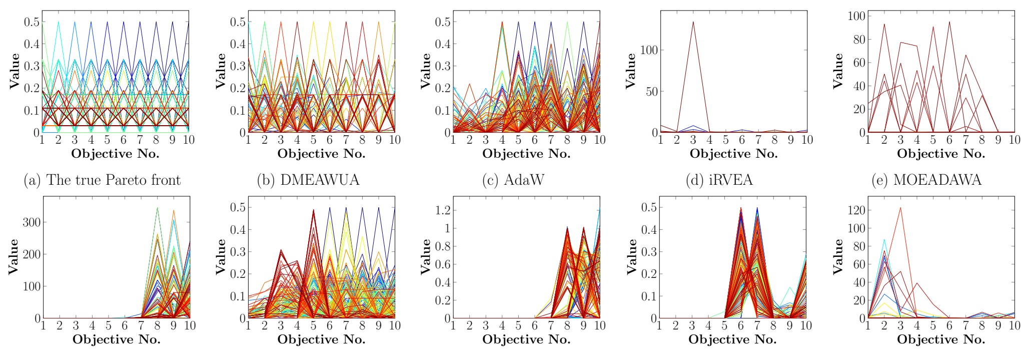
Fig. 8 plots the final solution sets obtained by all algorithms on the 10-objective DTLZ3 instance, which has a concave Pareto optimal curve, and the range of each objective is $[0, 1]$. As shown, there are five algorithms, e.g., DMEA-WUA, AdaW, MOEA/D-URAW and RVEAa, that obtain populations with good convergence. However, among these five algorithms, only the populations of our DMEA-WUA and RVEAa can cover the whole Pareto front. Therefore, this instance reflects that our algorithm is very competitive when dealing with convex problems.

Fig. 9 illustrates the final solution sets of all algorithms on the 3-objective



(a) The true Pareto front (b) DMEAWUA (c) AdaW (d) iRVEA (e) MOEADAWA
(f) MOEADPaS (g) MOEADURAW (h) RPEA (j) RVEAa (k) DEAGNG

Figure 9: The scatter plots of the final solution sets on the 3-objective DTLZ7 instance.

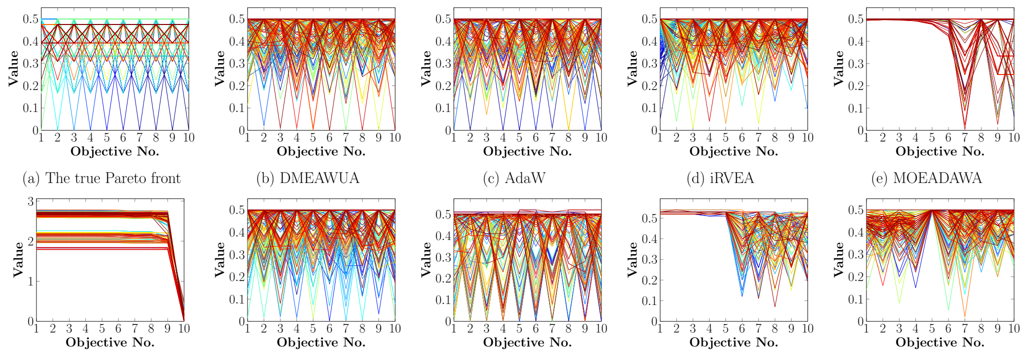


(a) The true Pareto front (b) DMEAWUA (c) AdaW (d) iRVEA (e) MOEADAWA
(f) MOEADPaS (g) MOEADURAW (h) RPEA (j) RVEAa (k) DEAGNG

Figure 10: The parallel coordinates of the final solution sets on the 10-objective DTLZ1 instance.

DTLZ7 instance to reflect the performances of these algorithms in solving discontinuous optimization problems. As shown, the true Pareto front of the 3-objective DTLZ7 consists of four segments, but the populations of the MOEA/D-AWA, MOEA/D-PaS and RPEA algorithms cannot cover the whole Pareto front. For the other algorithms, the distribution maintenance abilities of MOEA/D-URAW and RVEAa for this instance are far less than that of our algorithms. These results demonstrate the effectiveness of our algorithm for solving low-dimensional discontinuous optimization problems.

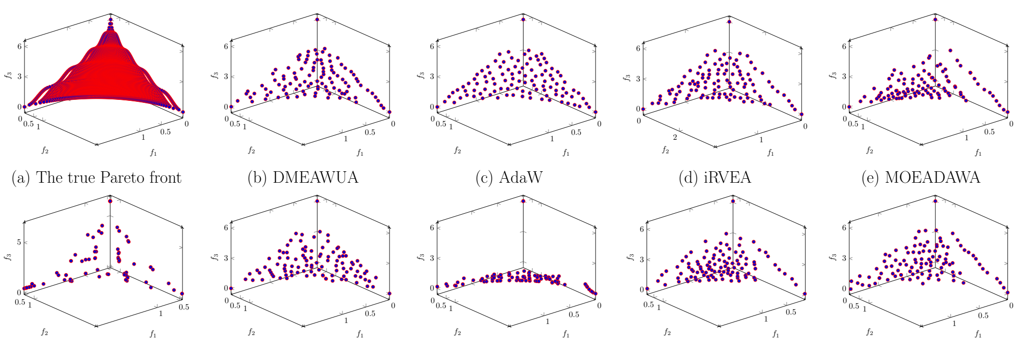
Figs. 10 and 11 show the final population distributions of all algorithms for the optimization problems with a linear Pareto front and an inverted Pareto front, respectively. As shown, DTLZ1 and IDTLZ1 have value ranges of $[0, 0.5]$ intervals for each objective, but their Pareto fronts are reversed. For the 10-objective DTLZ1 instance, only our DMEA-WUA obtains a set of



(a) The true Pareto front (b) DMEAWUA (c) AdaW (d) iRVEA (e) MOEADAWA

(f) MOEADPaS (g) MOEADURAW (h) RPEA (j) RVEAa (k) DEAGNG

Figure 11: The parallel coordinates of the final solution sets on the 10-objective IDTLZ1 instance.



(a) The true Pareto front (b) DMEAWUA (c) AdaW (d) iRVEA (e) MOEADAWA

(f) MOEADPaS (g) MOEADURAW (h) RPEA (j) RVEAa (k) DEAGNG

Figure 12: The scatter plots of the final solution sets on the 3-objective WFG1 instance.

solutions that covers the whole Pareto front. AdaW and MOEA/D-URAW also have a good distribution, but the values for the first to third objectives are obviously not fully covered. For the other algorithms, the populations of these algorithms either do not converge or only converge to local optimal regions. For the 10-objective IDTLZ1 instance, our algorithm still has a good distribution near the true Pareto front, and the same is true for AdaW, MOEA/D-URAW and RPEA. The distributed searches of MOEA/D-AWA, RVEAa and DEA-GNG in this instance are limited, while MOEA/D-PaS has no convergence at all.

For the problems with mixed Pareto fronts, we provide the 3-objective WFG1 instance as a visual comparison. It can be observed that the populations of all algorithms cannot cover the whole Pareto front. Even so, there are obvious differences in the search capabilities of different algorithms with

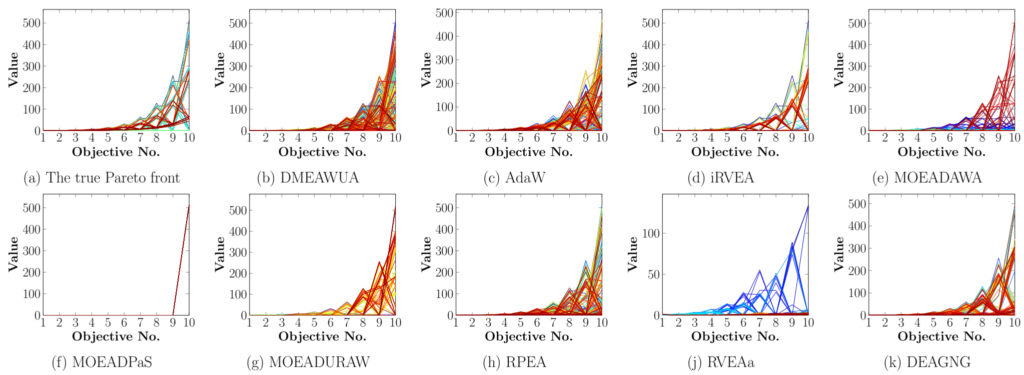


Figure 13: The parallel coordinates of the final solution sets on the 10-objective SDTLZ2 instance.

respect to this instance. For example, the population obtained by AdaW is significantly better than other algorithms in terms of uniformity; the solution sets of MOEA/D-PaS and RPEA are concentrated in some local areas, and their distributions are clearly not as good as thoe of the other algorithms. Although the DMEA-WUA and iRVEA are inferior to AdaW in terms of uniformity, they are still very advantageous compared to the other algorithms.

Finally, Fig. 13 uses the population distribution of each algorithm for the 10-objective SDTLZ2 instance to evaluate their abilities to optimize problems with badly-scaled Pareto fronts. For the SDTLZ2 problem, each objective f_i is multiplied by a factor 2^{i-1}, so the value range for the objective is $[0, 2^{i-1}]$. As shown in Fig. 11, all algorithms except MOEA/D-PaS and RVEAa can obtain populations whose distributions are the same as the shape of the true Pareto front.

In short, although the numerical indicators (e.g., the IGD+ and the HV) of the DMEA-WUA on some problems are inferior to those of some other algorithms, it can be found through the visual comparisons that our algorithm can still achieve relatively ideal results when optimizing problems with different Pareto fronts.

6. Conclusion

For decomposition-based MOEAs, designing an adaptive weight generation mechanism to accommodate problems with different Pareto fronts is one of the important but difficult topics in the field of evolutionary computation.

A feasible method is to adjust the weights adaptively during the optimization process. We consider the adjustment of weights to be an exploration operation, and the evolution of the population in the direction of the weights is an exploration operation. In this paper, a novel exploration versus exploitation model is proposed to balance the evolution of the population with the updating of the weights. In the exploration phase, four basic operations are provided, including weight generation, weight deletion, weight addition and weight replacement. These steps are performed in an orderly manner to ensure that the weights are consistent with the Pareto front of the problem. In the exploration phase, a new selection mechanism is provided, and it is inclined toward the nondominated solution that is closest to the weight in each subregion.

Furthermore, considering that most existing periodic-based adaptive weighting methods cannot achieve a balance between exploration and exploitation for various problems, we propose a stability matrix to control the transition between exploration and exploitation. This matrix reflects the evolutionary trend of the population. Once the population has stabilized, our algorithm stops conducting exploitation and turns to exploring new areas. In this way, our algorithm does not need to adjust the weight update period according to the characteristics of the problems, and this greatly improves the applicability of the algorithm.

The performance of the proposed algorithm is verified through a comprehensive experimental study. The results obtained on six test suites show that our algorithm significantly outperforms other state-of-the-art algorithms in most instances. In the future, we will further study how to use the current population instead of the archive to adjust weights, mainly by considering that the adjustments to solutions in the archive set will incur expensive time overhead.

7. Acknowledgments

This work was supported in part by the National Outstanding Youth Science Program of the National Natural Science Foundation of China under Grant 61625202; the International (Regional) Cooperation and Exchange Program of the National Natural Science Foundation of China under Grant 61860206011; the Program of the National Natural Science Foundation of China under Grants 61876061 and 61876164; and the Postgraduate Scientific Research Innovation Project of Hunan under Grant CX20190309.

References

[1] A. Mohammadi, H. Asadi, S. Mohamed, K. Nelson, S. Nahavandi, Openga, a c++ genetic algorithm library, in: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017, pp. 2051–2056. doi:10.1109/SMC.2017.8122921.

[2] Y. Liu, N. Zhu, K. Li, M. Li, J. Zheng, K. Li, An angle dominance criterion for evolutionary many-objective optimization, Information Sciences 509 (2020) 376–399. doi:10.1016/j.ins.2018.12.078.

[3] J. Luo, X. Huang, Y. Yang, X. Li, Z. Wang, J. Feng, A many-objective particle swarm optimizer based on indicator and direction vectors for many-objective optimization, Information Sciences 514 (2020) 166–202. doi:10.1016/j.ins.2019.11.047.

[4] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 712–731. doi:10.1109/TEVC.2007.892759.

[5] S. Liu, Q. Yu, Q. Lin, K. C. Tan, An adaptive clustering-based evolutionary algorithm for many-objective optimization problems, Information Sciences 537 (2020) 261–283. doi:10.1016/j.ins.2020.03.104.

[6] H. Ishibuchi, H. Masuda, Y. Tanigaki, Y. Nojima, Modified distance calculation in generational distance and inverted generational distance, in: Evolutionary Multi-Criterion Optimization, Springer International Publishing, Cham, 2015, pp. 110–125. doi:10.1007/978-3-319-15892-1_8.

[7] Q. Gu, H. Chen, L. Chen, X. Li, N. N. Xiong, A many-objective evolutionary algorithm with reference points-based strengthened dominance relation, Information Sciences 554 (2021) 236–255. doi:10.1016/j.ins.2020.12.025.

[8] Y. Tian, R. Cheng, X. Zhang, F. Cheng, Y. Jin, An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility, IEEE Transactions on Evolutionary Computation 22 (4) (2018) 609–622. doi:10.1109/TEVC.2017.2749619.

[9] M. Zhang, L. Wang, W. Li, B. Hu, D. Li, Q. Wu, Many-objective evolutionary algorithm with adaptive reference vector, Information Sciences 563 (2021) 70–90. doi:10.1016/j.ins.2021.01.015.

[10] H. Li, J. Sun, Q. Zhang, Y. Shui, Adjustment of weight vectors of penalty-based boundary intersection method in MOEA/D, in: Evolutionary Multi-Criterion Optimization, Springer International Publishing, Cham, 2019, pp. 91–100. doi:10.1007/978-3-030-12598-1_8.

[11] 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. doi:10.1109/TEVC.2013.2281535.

[12] M. Li, X. Yao, What weights work for you? adapting weights for any pareto front shape in decomposition-based evolutionary multiobjective optimisation, Evolutionary computation 28 (2) (2020) 227–253. doi:10.1162/evco_a_00269.

[13] R. Behmanesh, I. Rahimi, A. H. Gandomi, Evolutionary many-objective algorithms for combinatorial optimization problems: A comparative study, Archives of Computational Methods in Engineering (2020) 1–16.

[14] B. Derbel, A. Liefooghe, Q. Zhang, H. Aguirre, K. Tanaka, Multi-objective local search based on decomposition, in: J. Handl, E. Hart, P. R. Lewis, M. López-Ibáñez, G. Ochoa, B. Paechter (Eds.), Parallel Problem Solving from Nature – PPSN XIV, Springer International Publishing, Cham, 2016, pp. 431–441. doi:10.1007/978-3-319-45823-6_40.

[15] W. xiang Wang, K. shun Li, X. zhen Tao, F. hui Gu, An improved moea/d algorithm with an adaptive evolutionary strategy, Information Sciences 539 (2020) 1–15. doi:10.1016/j.ins.2020.05.082.

[16] Z. Wang, Q. Zhang, A. Zhou, M. Gong, L. Jiao, Adaptive replacement strategies for MOEA/D, IEEE Transactions on Cybernetics 46 (2) (2016) 474–486. doi:10.1109/TCYB.2015.2403849.

[17] 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. doi:10.1109/TEVC.2013.2281534.

[18] M. Wu, K. Li, S. Kwong, Q. Zhang, J. Zhang, Learning to decompose: A paradigm for decomposition-based multiobjective optimization, IEEE Transactions on Evolutionary Computation 23 (3) (2019) 376–390. doi:10.1109/TEVC.2018.2865931.

[19] J. Siwei, C. Zhihua, Z. Jie, O. Yew-Soon, Multiobjective optimization by decomposition with pareto-adaptive weight vectors, in: 2011 Seventh International Conference on Natural Computation, Vol. 3, 2011, pp. 1260–1264. doi:10.1109/ICNC.2011.6022367.

[20] F. Gu, Y. Cheung, Self-organizing map-based weight design for decomposition-based many-objective evolutionary algorithm, IEEE Transactions on Evolutionary Computation 22 (2) (2018) 211–225. doi:10.1109/TEVC.2017.2695579.

[21] M. Li, S. Yang, X. Liu, A performance comparison indicator for pareto front approximations in many-objective optimization, in: Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, New York, NY, USA, 2015, pp. 703–710. doi:10.1145/2739480.2754687.

[22] M. Asafuddoula, H. K. Singh, T. Ray, An enhanced decomposition-based evolutionary algorithm with adaptive reference vectors, IEEE Transactions on Cybernetics 48 (8) (2018) 2321–2334. doi:10.1109/TCYB.2017.2737519.

[23] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable test problems for evolutionary multiobjective optimization, in: Evolutionary multiobjective optimization, Springer, 2005, pp. 105–145. doi:10.1007/1-84628-137-7_6.

[24] X. Ma, Y. Yu, X. Li, Y. Qi, Z. Zhu, A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms, IEEE Transactions on Evolutionary Computation 24 (4) (2020) 634–649. doi:10.1109/TEVC.2020.2978158.

[25] Y. Liu, H. Ishibuchi, N. Masuyama, Y. Nojima, Adapting reference vectors and scalarizing functions by growing neural gas to handle irregular pareto fronts, IEEE Transactions on Evolutionary Computation 24 (3) (2020) 439–453. doi:10.1109/TEVC.2019.2926151.

[26] Y. Tian, X. Zhang, R. Cheng, C. He, Y. Jin, Guiding evolutionary multiobjective optimization with generic front modeling, IEEE Transactions on Cybernetics 50 (3) (2020) 1106–1119. doi:10.1109/TCYB.2018.2883914.

[27] H. Ge, M. Zhao, L. Sun, Z. Wang, G. Tan, Q. Zhang, C. P. Chen, A many-objective evolutionary algorithm with two interacting processes: Cascade clustering and reference point incremental learning, IEEE Transactions on Evolutionary Computation 23 (4) (2018) 572–586.

[28] 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. doi:10.1109/TEVC.2016.2587808.

[29] X. He, Y. Zhou, Z. Chen, Q. Zhang, Evolutionary many-objective optimization based on dynamical decomposition, IEEE Transactions on Evolutionary Computation 23 (3) (2019) 361–375. doi:10.1109/TEVC.2018.2865590.

[30] H. Liu, L. Chen, Q. Zhang, K. Deb, Adaptively allocating search effort in challenging many-objective optimization problems, IEEE Transactions on Evolutionary Computation 22 (3) (2018) 433–448. doi:10.1109/TEVC.2017.2725902.

[31] R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, A reference vector guided evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 20 (5) (2016) 773–791. doi:10.1109/TEVC.2016.2519378.

[32] X. Cai, Z. Mei, Z. Fan, A decomposition-based many-objective evolutionary algorithm with two types of adjustments for direction vectors, IEEE Transactions on Cybernetics 48 (8) (2018) 2335–2348. doi:10.1109/TCYB.2017.2737554.

[33] L. R. C. de Farias, P. H. M. Braga, H. F. Bassani, A. F. R. Araújo, MOEA/D with uniformly randomly adaptive weights, in: Proceedings of the genetic and evolutionary computation conference, Association

for Computing Machinery, New York, NY, USA, 2018, pp. 641–648. doi:10.1145/3205455.3205648.

[34] F. Vafaee, G. Turán, P. C. Nelson, T. Y. Berger-Wolf, Balancing the exploration and exploitation in an adaptive diversity guided genetic algorithm, in: 2014 IEEE Congress on Evolutionary Computation (CEC), 2014, pp. 2570–2577. doi:10.1109/CEC.2014.6900257.

[35] K. Tan, S. Chiam, A. Mamun, C. Goh, Balancing exploration and exploitation with adaptive variation for evolutionary multi-objective optimization, European Journal of Operational Research 197 (2) (2009) 701–713. doi:10.1016/j.ejor.2008.07.025.

[36] C. Segura, C. A. Coello Coello, E. Segredo, A. H. Aguirre, A novel diversity-based replacement strategy for evolutionary algorithms, IEEE Transactions on Cybernetics 46 (12) (2016) 3233–3246. doi:10.1109/TCYB.2015.2501726.

[37] K. A. De Jong, W. M. Spears, A formal analysis of the role of multi-point crossover in genetic algorithms, Annals of mathematics and Artificial intelligence 5 (1) (1992) 1–26. doi:10.1007/BF01530777.

[38] M. Črepinšek, S.-H. Liu, M. Mernik, Exploration and exploitation in evolutionary algorithms: A survey, ACM Computing Surveys 45 (3) (2013) 35:1–35:33. doi:10.1145/2480741.2480752.

[39] P. J. Rousseeuw, C. Croux, Alternatives to the median absolute deviation, Journal of the American Statistical association 88 (424) (1993) 1273–1283. doi:10.1080/01621459.1993.10476408.

[40] 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. doi:10.1109/TEVC.2016.2592479.

[41] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, X. Yao, A benchmark test suite for evolutionary many-objective optimization, Complex & Intelligent Systems 3 (1) (2017) 67–81. doi:10.1007/s40747-017-0039-7.

[42] S. Huband, P. Hingston, L. Barone, L. While, A review of multi-objective test problems and a scalable test problem toolkit, IEEE Transactions on Evolutionary Computation 10 (5) (2006) 477–506. doi:10.1109/TEVC.2005.861417.

[43] R. Vlennet, C. Fonteix, I. Marc, Multicriteria optimization using a genetic algorithm for determining a Pareto set, International Journal of Systems Science 27 (2) (1996) 255–260. doi:10.1080/00207729608929211.

[44] L. Zhen, M. Li, R. Cheng, D. Peng, X. Yao, Multiobjective test problems with degenerate pareto fronts, arXiv preprint arXiv:1806.02706 (2018).

[45] Q. Liu, Y. Jin, M. Heiderich, T. Rodemann, Adaptation of reference vectors for evolutionary many-objective optimization of problems with irregular pareto fronts, in: 2019 IEEE Congress on Evolutionary Computation (CEC), 2019, pp. 1726–1733. doi:10.1109/CEC.2019.8790214.

[46] R. Wang, Q. Zhang, T. Zhang, Decomposition-based algorithms using pareto adaptive scalarizing methods, IEEE Transactions on Evolutionary Computation 20 (6) (2016) 821–837. doi:10.1109/TEVC.2016.2521175.

[47] Y. Liu, D. Gong, X. Sun, Y. Zhang, Many-objective evolutionary optimization based on reference points, Applied Soft Computing 50 (2017) 344 – 355.

[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. doi:10.1109/MCI.2017.2742868.

[49] J. Derrac, S. García, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm and Evolutionary Computation 1 (1) (2011) 3 – 18.

[50] T. Eftimov, P. Korošec, B. K. Seljak, A novel approach to statistical comparison of meta-heuristic stochastic optimization algorithms using deep statistics, Information Science 417 (2017) 186–215.