



ARTICLE

Large-Scale Multi-Objective Optimization Algorithm Based on Weighted Overlapping Grouping of Decision Variables

Liang Chen¹, Jingbo Zhang¹, Linjie Wu¹, Xingjuan Cai^{1,2,*} and Yubin Xu¹

¹Shanxi Key Laboratory of Big Data Analysis and Parallel Computing, Taiyuan University of Science and Technology, Taiyuan, 030024, China

²School of State Key Laboratory of Novel Software Technology, Nanjing University, Nanjing, 210008, China

*Corresponding Author: Xingjuan Cai. Email: xingjuancai@163.com

Received: 26 December 2023 Accepted: 22 February 2024 Published: 16 April 2024

ABSTRACT

The large-scale multi-objective optimization algorithm (LSMOA), based on the grouping of decision variables, is an advanced method for handling high-dimensional decision variables. However, in practical problems, the interaction among decision variables is intricate, leading to large group sizes and suboptimal optimization effects; hence a large-scale multi-objective optimization algorithm based on weighted overlapping grouping of decision variables (MOEAWOD) is proposed in this paper. Initially, the decision variables are perturbed and categorized into convergence and diversity variables; subsequently, the convergence variables are subdivided into groups based on the interactions among different decision variables. If the size of a group surpasses the set threshold, that group undergoes a process of weighting and overlapping grouping. Specifically, the interaction strength is evaluated based on the interaction frequency and number of objectives among various decision variables. The decision variable with the highest interaction in the group is identified and disregarded, and the remaining variables are then reclassified into subgroups. Finally, the decision variable with the strongest interaction is added to each subgroup. MOEAWOD minimizes the interactivity between different groups and maximizes the interactivity of decision variables within groups, which contributed to the optimized direction of convergence and diversity exploration with different groups. MOEAWOD was subjected to testing on 18 benchmark large-scale optimization problems, and the experimental results demonstrate the effectiveness of our methods. Compared with the other algorithms, our method is still at an advantage.

KEYWORDS

Decision variable grouping; large-scale multi-objective optimization algorithms; weighted overlapping grouping; direction-guided evolution

1 Introduction

Large-Scale Multi-Objective Optimization Problems (LSMOPs), applied in territories like engineering design and logistics [1–4], share the conflicting objectives characteristic of Multi-Objective Optimization Problems (MOPs) [5–7]. However, their high-dimensional decision variables necessitate specialized strategies for effective resolution. In addressing MOPs, the optimization of one objective frequently results in the degradation of others. Multi-Objective Evolutionary Algorithms (MOEAs)



address this by finding a group of non-dominated solutions, effectively balancing these objectives to meet various requirements [8,9]. For a minimizing problem that involves m objectives, it is symbolically represented as $F(X) = \min(F_1(X), F_2(X), \dots, F_n(X)), \forall F(X_1) < F(X_2)$, then it is called that X_1 dominates X_2 [10]. If at least one objective in X_1 and X_2 is such that neither can be superior to the other, then X_1 and X_2 are considered non-dominated solutions [11–13]. This implies that each solution has its unique advantages, necessitating the retention of all such solutions [14–16]. Therefore, the fundamental goal of MOPs is to find a set of mutually non-dominated Pareto solutions. MOEAs like nondominated sorting genetic algorithm III (NSGA-III) [17] have shown effectiveness in these problems [18–21]. However, the decision variables in engineering design, genetic biology, and other fields usually reach thousands or even hundreds of thousands recently [22–24]. These problems are characterized by high nonlinearity, complexity, and uncertainty, which make the traditional optimization algorithms inefficient and perform degradation, and present a great challenge [11,25,26].

MOPs that encompass more than 100 dimensions of decision variables are referred to as LSMOPs [27]. Algorithms addressing LSMOPs must consider how the substantial dimensionality of decision variables impacts problem-solving [28]. This is crucial because the decision space expands exponentially with high dimensionality, rendering existing MOEAs less effective in solving LSMOPs [29]. Traditional MOEAs are generally committed to an excellent solution selection strategy, i.e., selecting the better and more appropriate solution when multiple individuals are at the same Pareto level during the evolutionary iteration process. However, the conventional MOEAs fail to construct a more excellent individual in the LSMOPs, and the huge decision space cannot be explored through basic operations like crossover and mutation, which often results in the algorithms finding suboptimal solutions within a limited local space [30–32]. In environmental selection among poor-quality suboptimal solutions, the chosen individuals usually provide no significant aid in problem-solving [18]. Consequently, this situation compels researchers to develop exceptional individuals that are capable of guiding the evolution of the entire population and explore the expansive decision space. Currently, LSMOAs are typically categorized into three broad groups [1].

The first category encompasses MOEAs that are based on the reduction of decision space, primarily concentrating on reducing the dimensionality of decision variables. This approach transforms LSMOPs into more manageable, non-large-scale optimization tasks [33]. It is crucial that offspring generated from reduced-dimension parent variables can be mapped back to the initial decision space for function evaluation, facilitating optimization without navigating the high-dimensional space. However, this approach often leads to local optima rather than global optima [34]. Common strategies here involve problem transformation and dimensionality reduction.

The second category involves MOEAs with novel search strategies, distinct from decision space partitioning and dimensionality reduction [27]. Recent MOEAs address LSMOPs directly through innovative search strategies that enhance traditional MOEA operators, including crossover and mutation [35]. These approaches focus on generating superior offspring without complex manipulation of the vast decision space [36–38]. Typical strategies include new crossover operators and innovative probabilistic models. However, while typically effective for specific problems, these algorithms often lack generalizability.

The third category comprises MOEAs that focus on decision variable grouping, employing a classic divide-and-conquer strategy to tackle LSMOPs [39,40]. Given the abundance of decision variables in LSMOPs, specific strategies for grouping and alternately optimizing each group are widely applied. Common strategies encompass random grouping, differential grouping, and decision variable analysis.

Currently, most of the grouping-based evolutionary algorithms are random grouping, which divides the decision variables into several groups of equal size [41]. This random partitioning strategy increases the probability of simultaneous optimization of interacting variables and makes the population converge faster [42]. However, it completely fails to take into account the interaction between decision variables and easily leads to local optima. The differential grouping strategy analyzes the relationship between the decision variables and achieves better convergence, but fails to take into account the diversity of the population at the same time [43]. Hence, a decision variable analysis (DVA) approach is used by MOEA/DVA to categorize decision variables into distance, position, and mixed variables [39]. A more precise grouping method is employed to differentiate diversity variables from convergence variables, while the DVA is utilized to address the challenge of insufficient population diversity within the differential grouping method [44].

However, the size of the subgroups remains large due to the intricacy of the decision variables in LSMOPs. For example, LSMOPs with 1000 dimensions of decision variables are grouped in such a way that there are still more than 100 decision variables in a group, in which case the subproblem is still an LSMOPs. Reference [40] proposed the concept of overlapping grouping, which divides the oversized group into multiple subgroups that can contain the same decision variables, so that the group size can be reduced while taking into account that the decision variables with direct interaction relationships can be optimized at the same time. However, the decision variables with the highest number of direct interaction relationships are chosen for division during the process of overlapping grouping, but this approach overlooks the intensity of these interaction relationships. The determination of interaction relationships is random and may have a large error with the real situation between decision variables. Based on these problems, this paper proposes MOEAWOD with the following contributions:

- A weighted overlapping grouping strategy enhances decision variable grouping effectiveness. This strategy involves subdividing larger groups after interaction analysis on convergent variables, using interaction frequency and targets as subdivision weights. This method strengthens intra-group interactions and weakens inter-group ones, showing better effectiveness in experiments.
- For optimizing post-grouped decision variables, directional information is used to guide non-dominated solutions toward more promising directions. Diversity is explored when directed by two non-dominated individuals, and convergence is expanded when directed by both non-dominated and dominated individuals.
- The experimental findings reveal that the convergence and diversity of the proposed MOEAWOD algorithm surpass those of other algorithms.

This paper is structured as follows: The related work is shown in [Section 2](#). [Section 3](#) elaborates on the specifically proposed MOEAWOD. [Section 4](#) discusses the experimental process, results, and analysis. Finally, [Section 5](#) concludes the paper and outlines prospects for future research.

2 Related Work

One straightforward method to address LSMOPs involves employing the divide-and-conquer concept. This approach entails grouping decision variables and optimizing them alternately. LSMOPs based on decision variable grouping are typically classified into three categories: random grouping, differential grouping, and decision variable analysis.

The random grouping method enhances the probability of simultaneous optimization of interacting variables, resulting in expedited population convergence [41]. Antonio et al. [42] introduced

a MOEA named Cooperative Coevolutionary Generalized Differential Evolution 3 (CCGDE3) to tackle LSMOPs, incorporating randomization within a cooperative coevolutionary framework. It builds on the generalized differential evolution 3 (GDE3), and decision variables are randomly partitioned into subgroups of equal size to increase the probability of optimizing interactive variables concurrently. However, its performance is susceptible to the predetermined determination of population number and grouping mode, potentially leading to a decline in algorithmic effectiveness. Song et al. [45] introduced a stochastic coevolutionary dynamic grouping strategy, yielding favorable test results. Relying solely on randomness to enhance the probability of simultaneous optimization of interacting variables may result in local optimization of the population. Additionally, the size of the group can also influence the evolutionary effectiveness of the population.

To avoid this situation, Cao et al. [43] incorporated a differential grouping strategy to identify interactions among decision variables. This strategy resolves the interaction problem by simultaneously optimizing interrelated decision variables, thus fostering a closer alignment between theoretical and practical optimization. Nevertheless, the performance improvement of the differential grouping strategy comes at the expense of increased computational complexity, as it requires analyzing the interaction between decision variables in pairs. To mitigate computational demands, Li et al. [46] implemented a fast interdependent identification method that selectively ignored certain unnecessary checks between decision variables. While the differential grouping technique enhances the convergence of the population, it overlooks population diversity in LSMOPs. This oversight may result in the evolution of the population towards a solution set characterized by overly singular traits.

To circumvent the issue of poor diversity, Ma et al. [39] proposed MOEA/DVA, which uses a grouping method of DVA to treat convergence variables and diversity variables separately. Within this approach, variables are categorized into position variables, distance variables, and mixed variables. Following an interactive analysis, decision variables that notably influence convergence are grouped for joint optimization. Subsequently, variables influencing diversity are optimized independently, yielding improved results. Although MOEA/DVA is very effective in dealing with LSMOPs, it is computationally costly due to the evaluation of the interaction of a wide range of decision variables. Moreover, due to the particularity of large-scale, the interaction between decision variables is intricate, and the grouping after DVA processing may still be too large, which will still be an LSMOPs.

To address the issue of declining evolutionary performance in large groups, we propose a MOEAWOD based on DVA to conduct weighted overlapping grouping for such instances. Following the interaction analysis, the most interactive decision variables are allocated to different groups as overlapping parts. This enables the simultaneous optimization of interacting decision variables, with the added advantage that the size of the group does not influence the evolutionary performance of the population.

3 Proposed MOEAWOD

3.1 Motivation

Random grouping of decision variables fails to account for their interactive relationships, depending solely on randomness to enhance the probability of jointly optimizing interacting variables [41]. This oversight might result in the population towards local optima within the decision space [43]. The interaction between decision variables is characterized by how variables x_i and x_j jointly influence the behavior of the objective value, such as its monotonicity or convexity [47,48]. In summary, x_i alone cannot determine the directional change of the objective function $f(x)$, the value of x_j must also be considered. Formally, an interaction between variables x_i and x_j implies at least one set of

values a_1, a_2, b_1, b_2 that meet specific conditions [46].

$$\begin{cases} f(X) |_{x_i=a_2, x_j=b_1} < f(X) |_{x_i=a_1, x_j=b_1} \\ f(X) |_{x_i=a_1, x_j=b_2} < f(X) |_{x_i=a_2, x_j=b_2} \end{cases} \quad (1)$$

This illustrates that optimizing decision variables i and j together in one group can more accurately and efficiently optimize the original objective function.

Current group-based LSMOAs primarily utilize random and differential grouping. However, these methods do not effectively balance solution diversity and convergence. Consequently, decision variable analysis-based strategies are used to achieve more effective grouping.

In LSMOPs characterized by a high number of decision variables, even after categorizing them into convergent and diverse types, the resulting groups from differential grouping may still be excessively large. For example, when the LSMOPs problem exceeds 1000 dimensions, the decision variables are intricately related, and the group size obtained could be larger than 100, which is still an LSMOPs and remains difficult to process. Addressing these challenges, this study proposes a weighted overlapping grouping strategy. For larger groups, weighted overlapping is implemented, distributing highly interactive variables across subgroups to maximize intra-group and minimize inter-group interactions.

3.2 The Proposed Overall Framework of MOEAWOD

Algorithm 1 and Fig. 1 comprehensively describe the MOEAWOD framework. Initially, decision variables undergo control analysis, where each position is fixed except one, to determine its influence on the convergence of objective function, categorizing them into convergent and diverse variables. Then interaction analysis between pairs of convergent variables to assess their combined effect on the objective. Subsequently, interacting variables are grouped using a weighted overlapping approach for simultaneous optimization. Finally, the grouped variables are alternately optimized.

Algorithm 1: Main framework of MOEAWOD

Input: Maximum number of times a function evaluates to a value FE_{\max} , Population size N , Threshold for group size α , The number of direction vectors r

Output: Population P

1. $P, V \leftarrow$ Initialization (N)
 2. $ConverVariable (CV), DiverVariable (DV) \leftarrow$ DecisionVariablesAnalysis (P)
 3. $subcomponents \leftarrow$ WeightedOverlappGrouping (CV, α) \rightarrow see Algorithm 2
 4. **while** $FE_s < FE_{\max}$
 5. $Q \leftarrow$ Preselection (P)
 6. $Offspring \leftarrow$ DirectionGuidedEvolution (CV, P, Q, V, r) \rightarrow see Algorithm 3
 7. $P \leftarrow$ EnvironmentalSelection ($P, Offspring, N$)
 8. **end**
-

After initializing the population for LSMOPs (the parameters are set in the experimental setup section), decision variables undergo perturbation analysis. Each variable is fixed and individually perturbed within its space, creating a set of individuals differing only in the perturbed position. Diversity variables (DV), not affecting convergence, are randomly grouped for optimization, whereas convergent variables undergo weighted overlapping grouping. LSMOPs are transformed into manageable non-large-scale tasks for collaborative optimization after grouping.

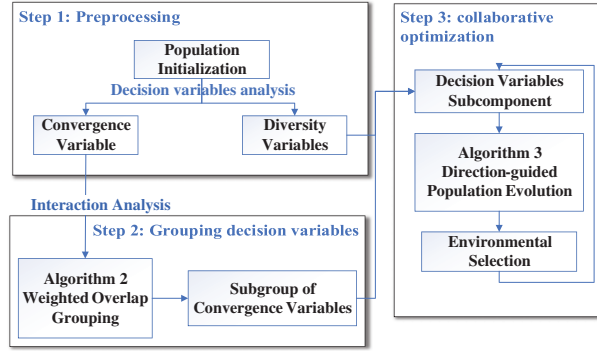


Figure 1: Algorithm framework diagram

3.3 Method of Weighted Overlapping Grouping of Decision Variables

After identifying interactions between pairs of convergent decision variables, those with direct interactions are grouped. If a group becomes too large, a weighted overlapping grouping strategy is applied for further division. To ascertain interactions between pairs of convergent variables, an individual is randomly selected to study the interaction between positions x_i and x_j , with random values assigned within their decision spaces. In the original population, the value of individual x^k is given by $x_i^k = a_1, x_j^k = b_1$. Values a_2 and b_2 are randomly selected within the bounds of x_i and x_j . Subsequently, calculate the function value $F(x^k)|_{x_i=a_2}, F(x^k)|_{x_j=b_2}, F(x^k)|_{x_i=a_2, x_j=b_2}$, where $F(x^k)|_{x_i=a_2, x_j=b_2} = F(x_1^k, \dots, x_{i-1}^k, a_2, x_{i+1}^k, \dots, x_{j-1}^k, b_2, x_{j+1}^k, \dots, x_n^k)$. The correlation between decision variables x_i and x_j concerning the objective function F is assessed through an interaction judgment formula, which is expressed as follows:

$$\begin{aligned} \Delta_i F|_{x_j=b_1} &\leftarrow F(X^k)|_{x_i=a_2, x_j=b_1} - F(X^k)|_{x_i=a_1, x_j=b_1} \\ \Delta_i F|_{x_j=b_2} &\leftarrow F(X^k)|_{x_i=a_2, x_j=b_2} - F(X^k)|_{x_i=a_1, x_j=b_2} \end{aligned} \quad (2)$$

Hence, it follows that the decision variables x_i and x_j exhibit interactive characteristics in circumstances where $\Delta_i F|_{x_j=b_1} * \Delta_i F|_{x_j=b_2} < 0$ is applicable. This implies that the monotonicity of the objective function F is jointly determined by x_i and x_j . In single-objective problems, the interaction between decision variables is directly discernible. However, in LSMOPs, it is assumed that at least one of the m objectives demonstrates a correlation between variables i and j , indicates their interaction. In LSMOPs, optimizing groups of connected decision variables improves results. Yet, the complexity and quantity of these variables can lead to oversized groups, maintaining the large-scale challenge.

The links between decision variables in LSMOPs are complex, and dividing subgroups based on interactions will lead to oversized subgroups, but disregarding interactions will segment the indivisible problem and result in not getting the genuine optimal solution. Therefore, a weighted overlapping grouping decision variable division strategy is proposed to address the issue of excess decision variables with interconnections, and the grouping protects the integrity of the indivisible problem. As shown in Algorithm 2.

Algorithm 2: WeightedOverlapGrouping

Input: CV, α

Output: Combined subcomponents (CS)

(Continued)

Algorithm 2: (continued)

```

1. for  $p = 1:M-1$ 
2.   for  $q = p+1:M$ 
3.     for  $r = 1:NIA$ 
4.        $link, fitness \leftarrow \text{FitnessEvaluate}(CV)$ 
5.     end for
6.   end for
7. end for
8.  $divided = \text{false}$ 
9. while  $\sim \text{all}(divided)$  do
10.   $x_0 \leftarrow \text{find}(divided = \text{false})$ 
11.   $x \leftarrow \text{Interconnected}(x_0)$ 
12.  if  $\text{size}(x) > \alpha$ 
13.     $k \leftarrow \text{MaximumInteractivity}(x)$ 
14.     $S_0 \leftarrow \text{Ignoring the role of the decision variable } k \text{ and regrouping}$ 
15.     $S \leftarrow \text{Decision variable } k \text{ is added to each new set } S_0$ 
16.    end if
17.   $CS \leftarrow CS \cup S$ 
18. end while

```

In the interaction analysis of convergent variables due to the huge decision space, multiple interactions are analyzed to learn the real interaction situation and to avoid failing to detect the real interaction situation due to the absence of interactions in some subspaces. The number of detected interactions is used as the weight of the interactions between the variables, and the higher number of interactions indicates a closer connection between the two variables. At the same time, two variables in the multi-objective problem have an interaction with one or more objectives, thus the number of interacting objectives is used as the second weight to indicate the intensity of the interaction (Steps 1–4). Next, decision variables are segmented into weighted overlapping groups according to interactions (Steps 8–16). Grouping directly or indirectly related decision variables together (Steps 8–11). Weighted overlapping grouping is applied to group sizes larger than the threshold α , which maintains the connections between variables within the subgroups while refining the subgroups (Steps 12–16). After neglecting the variable k with the most interactive performance and then regrouping the remaining variables efficiently by interaction, finally, k is added as an overlap within each regrouped group.

To demonstrate the weighted overlapping group strategy, we apply it to the Benchmark MOP proposed by Walking Fish Group (WFG) with a setup of 10 decision variables and 3 objectives. Fig. 2a shows how other methods group directly connected variables in WFG6, for comparison. In the test problem, x_1 and x_2 are diversity variables that are not subject to interaction analysis. The diagram reveals that x_3 has direct connections with x_5 and x_6 , and indirect connections with other variables through intermediary decision variables. This linkage via intermediaries often results in overly large groups, such as assigning the seven variables from x_3 to x_{10} into the same group, which can reduce optimization performance. By omitting variables with the strongest interactions, we refine the decision variables: x_3, x_4, x_5, x_6 and x_8 from Subgroup 1; x_7 and x_9 , Subgroup 2, as shown in Fig. 2b. Reintegrating x_{10} into each subgroup ensures a balance between subgroup size and internal connectivity. The example uses 10 decision variables to effectively demonstrate the weighted overlapping grouping. This approach is highly practical for LSMOPs.

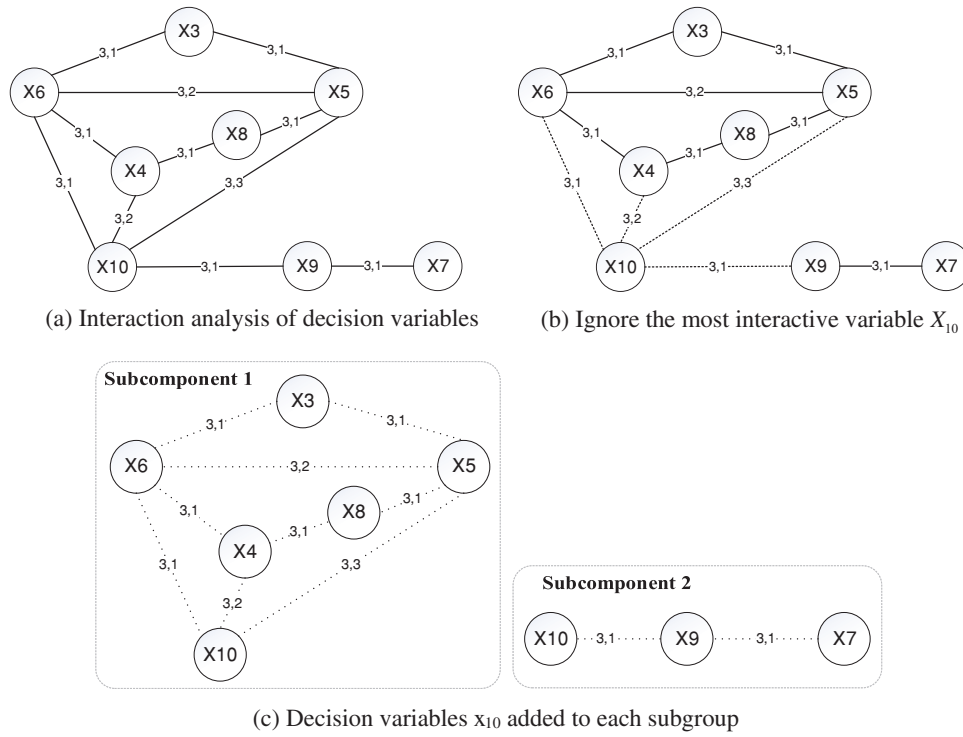


Figure 2: Weighted overlapping grouping

3.4 The Optimization Method for Grouping Decision Variables

After weighted overlapping grouping of decision variables, subgroups are collaboratively optimized using a direction-guided evolutionary method. This involves preselecting individuals to inherit directional cues, steering them toward optimal evolution in their respective subspaces, and focusing on both convergence and diversity. To improve convergence in less cohesive populations, the direction shifts from dominant to non-dominant solutions, and to enhance diversity, it moves between non-dominant solutions.

Algorithm 3: Direction-guided evolutionary algorithms

Input: CS, P, V, r

Output: *Offspring*

1. $F_m, N_d = \text{NDsort}(P)$
 2. **if** $N_d > r$
 3. $Q_0 \leftarrow F_1 \cup \dots \cup F_{m-1}; P \leftarrow F_m$
 4. **else if**
 5. $Q_0 \leftarrow F_1; P \leftarrow F_2 \cup \dots \cup F_{m-1}$
 6. **end if**
 7. $z \leftarrow \text{Calculate the ideal point}(P)$
 8. **for** $i = 1: |P|$
 9. **for** $j = 1: |V|$
 10. $\text{COS}_\theta \leftarrow \text{Cosines of reference vector } j \text{ and individual } i$
-

(Continued)

Algorithm 3: (continued)

```

11.   end for
12.    $k = \arg \max(COS_\theta)$ 
13.   end for
14.   for  $j = 1: |V|$ 
15.      $k = \arg \min(APD)$ 
16.   end for
17.    $Q \leftarrow Q_0 \cup p_k$ 
18.    $p_s \leftarrow \text{random}(Q)$ 
19.   if  $N_d < r$ 
20.      $R \leftarrow$  Selecting  $r$  dominated solutions at random from the population  $Q$ 
21.   else
22.      $R \leftarrow$  Selecting  $r$  non-dominated solutions at random from the population  $Q$ 
23.   for  $j = 1: r$ 
24.      $d_j \leftarrow p_s - r_i$ 
25.      $s_j = \gamma * d_j + p_s$ 
26.      $Offspring \leftarrow Offspring \cup s_j$ 
27.   end for

```

The algorithm includes two main steps: preselecting individuals for directional information inheritance (Steps 1–17), and evolving non-dominance individuals using direction cues from preselected and non-dominance individuals (Steps 18–25). Pre-selection is primarily done to improve parental population diversity in order to facilitate more effective evolution, i.e., individuals of the parent that balance diversity and convergence can lead to better overall evolution. Angle-penalized distance APD to assess individual diversity and convergence as a way to select a few less convergent but well-distributed solutions (Steps 13–15). During evolution, pre-selected individuals keep the population from reaching a local optimum and pre-selected non-dominated individuals allow convergence to be guaranteed, while the reference vector approach selects dominated individuals to ensure diversity. If the number of dominant solutions is below threshold r , indicating good diversity, all dominant solutions are chosen to enhance population convergence (Steps 19–22). Otherwise, selection from non-dominant solutions expands diversity. Directional information is then added to preselected individuals, completing the evolution process, in which step size γ is determined randomly from a normal distribution (Steps 23–27). A balance in step size is crucial; too large causes instability and local optima, while too small results in low efficiency. Thus, using the average length of the non-dominant solutions and direction vectors as the evolutionary step size is more effective.

It is observed in Fig. 3 in which X_3 is a dominated solution and X_1 and X_2 are non-dominated individuals. When there are many non-dominant solutions, suggesting room for convergence improvement, the direction shifts from dominance solution X_3 to non-dominance X_2 . Conversely, a scarcity of non-dominant solutions, indicating low diversity, leads to a directional shift from non-dominance X_2 to X_1 , enhancing diversity potential.

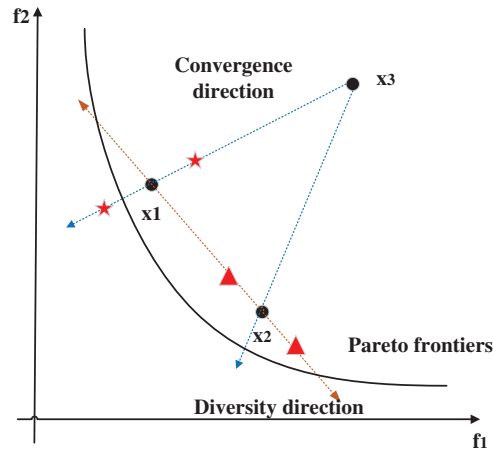


Figure 3: Direction-guided evolutionary

3.5 Discussions

The core operations of the proposed MOEAWOD algorithm encompass two key components: the first involves weighted overlapping grouping, and the second entails directed population evolution. In weighted overlap grouping, the convergence and diversity of decision variables are initially analyzed. Subsequently, each bit of the n -dimensional variable is classified after a constant number of perturbations, resulting in a time complexity of $O(n)$. Following the analysis, the interaction between each pair of convergent variables is examined to calculate the weight, and the time complexity is $O(n^2)$. Subsequently, the group is overlapped according to the weight. The procedure involves selecting the most interactive variable each time, and regrouping the group until the group size meets the specified threshold, and the time complexity is $O(\log n)$. In the process of direction-guided population evolution, preselected individuals exhibiting both diversity and convergence accept directional information for evolution. Subsequently, directional information is added to the preselected individuals, and the time complexity of both operations is $O(1)$.

In summary, the time complexity for decision variable grouping is $O(n^2)$. For LSMOPs, an increase in the dimension of decision variables results in an exponential expansion of the decision space. While weighted overlapping grouping introduces a computational complexity of $O(n^2)$, it significantly reduces the decision space compared to the original, enabling faster local convergence. Moreover, the combination of variables enhances the exploration of the potential optimization space more effectively.

4 Experiments

4.1 Experimental Setup

This study evaluates the performance of the proposed MOEAWOD through the utilization of two distinct test sets: the large-scale multi-objective optimization set LSMOPs and the continuous multi-objective set WFG. LSMOPs, comprising LSMOP1-LSMOP9, feature complex decision variable interactions, non-uniform variable grouping, and intricate fitness landscapes [39]. WFG, containing WFG1-WFG9, includes problems with complex Pareto fronts and optimal sets, suitable for assessing overall performance in diverse multi-objective scenarios [49]. We compare our algorithm with other advanced LSMOAs to validate its practical problem-solving effectiveness.

Different types of advanced LSMOAs were selected for comparison, including methods based on decision variable grouping and those reconstructing LSMOPs. Key comparison algorithms are the Decision Variable Clustering-based Evolutionary Algorithm (LMEA) [50], Grouped and Linked Polynomial Mutation Operator (GLMO) [51], Decision Variable Analysis and Reconstruction-based Multi-objective Evolutionary Algorithm (LERD) [52], Fast Large-scale Evolutionary Algorithm (FLEA) guided by reference [53]. All algorithms were sourced from PlatEMO 4.2, a platform encompassing various multi-objective algorithms, including those for large-scale problems [54]. The experiments were conducted on an Intel i5 PC with a 2.4 GHz CPU, Windows 10 OS, and implemented in Matlab-R2021b. Parameters are detailed in the following Table 1. Decision variables were perturbed 20 times for analysis, and categorized into convergence and diversity variables. For convergence variable grouping, the interaction was set at 6 times to assess interactivity strength. The grouping threshold was set at 60, with weighted overlapping grouping applied when the number of directly or indirectly connected variables exceeded this threshold.

Table 1: Parameter settings

Algorithm	Parameter settings
MOEAWOD	Number of decision variable perturbations: 20; Number of interactivity analyses: 6; Weighted overlapping grouping thresholds: 60
LMEA	Decision variables clustering selective solution number: 5; Number of selected solutions for decision variable interaction analysis 5; Number of perturbations per solution for clustering: 50
GLMO	The optimization method: NSGA-III; Grouping method: ordered; The number of variable Groups: 4
LERD	The number of sampling points: 3; DVA optimized population size: 10; Number of DVA optimization iterations: 20
FLEA	Population sizes: 100; Number of reference points: \sqrt{N}

In multi-objective problems with multiple outcome objectives, performance cannot be measured solely by objective values. Consequently, the overall performance of the algorithm is evaluated using the Inverted Generational Distance plus (IGD^+) metric [55–57]. IGD^+ effectively evaluates the convergence and diversity of the population. The formula is as follows:

$$IGD^+(P) = \frac{1}{|Z|} \sum_{j=1}^{|Z|} \min_{\vec{p} \in P} d(\vec{p}_i, \vec{z}_j) \quad (3)$$

where $d_{IGD^+}(\vec{p}, \vec{z}) = \sqrt{\sum_{k=1}^m (\max\{p_k - z_k, 0\})^2}$ (minimization problems). Z denotes the reference points on the Pareto front of the specific problem, while P representing the results obtained from the algorithm. IGD^+ evaluates the quality of solutions by calculating, the average distance from each reference point to the nearest point in the dominated region by the solution set. When the IGD^+ value is low, it indicates a high degree of similarity between P and Z , reflecting better convergence and distribution of P .

4.2 Experimental Results and Analysis

Initially, the performance of the algorithm is assessed on LSMOPs test problems, using the IGD⁺ metric to evaluate the convergence, diversity, Pareto optimality, and overall distribution characteristics of the populations generated by the proposed evolutionary algorithm. Table 2 compares the performance of five algorithms on the LSMOPs test series with 300, 500, and 1000 decision variables and 3 objectives. It shows the average IGD⁺ values and standard deviations for each algorithm over 30 independent runs on each test case. The Wilcoxon rank-sum test was employed for statistical tests with a significance level of 5%. Bold text indicates the best solutions among all compared algorithms. In the table, ‘+’ denotes IGD⁺ metrics better than the MOEAWORD algorithm, ‘-’ indicates inferior solutions, and ‘=’ signifies similar results to the proposed algorithm.

Table 2: IGD⁺ obtained across LSMOPs test suites

Problem	D	LMEA	GLMO	LERD	FLEA	MOEAWORD
LSMOP1	300	2.6974e-1 (3.07e-1) -	2.6922e-1 (2.40e-2) -	3.2887e-1 (3.79e-2) -	5.6008e-1 (1.17e-1) -	2.1357e-1 (1.69e-2)
	500	9.3841e+0 (3.79e+0) -	2.9052e-1 (2.12e-2) -	4.4200e-1 (3.28e-2) -	5.0573e-1 (1.13e-1) -	8.5119e-2 (3.11e-3)
	1000	8.3417e+0 (4.61e+0) -	3.1414e-1 (8.67e-3) -	4.1036e-1 (1.55e-2) -	4.7241e-1 (9.58e-2) -	5.0282e-2 (1.69e-3)
LSMOP2	300	8.3712e-2 (7.32e-3) =	8.6105e-2 (2.39e-2) -	8.9839e-2 (1.99e-3) -	8.5844e-2 (9.32e-3) =	8.5185e-2 (2.71e-3)
	500	7.2870e-2 (6.19e-3) -	6.5551e-2 (1.58e-2) +	6.8155e-2 (8.54e-4) -	7.3285e-2 (5.90e-3) -	6.6761e-2 (1.22e-3)
	1000	5.7138e-2 (3.84e-2) =	5.8561e-2 (1.84e-3) -	5.0424e-2 (3.58e-4) -	5.7603e-2 (5.07e-3) -	4.7844e-2 (4.36e-4)
LSMOP3	300	3.4808e+0 (2.75e+0) -	7.7982e-1 (3.00e-1) =	2.5349e+0 (2.22e+0) =	6.6545e-1 (2.77e-3) +	1.1219e+0 (8.56e-1)
	500	5.0339e+2 (5.52e+2) -	8.7373e-1 (4.88e-1) +	8.4136e+0 (1.60e+0) -	7.5083e-1 (4.69e-1) +	1.5994e+0 (1.16e-1)
	1000	2.3233e+2 (3.08e+2) -	6.6667e-1 (1.61e-6) +	7.7720e+0 (2.77e+0) -	6.6499e-1 (2.99e-3) +	9.2805e-1 (4.28e-2)
LSMOP4	300	1.5131e-1 (5.81e-2) -	2.0904e-1 (5.70e-2) -	1.9563e-1 (3.70e-3) -	1.9282e-1 (1.18e-2) -	1.1852e-1 (3.76e-3)
	500	2.1253e-1 (1.37e-2) -	1.5783e-1 (4.68e-2) -	1.4203e-1 (4.79e-3) -	1.4108e-1 (6.21e-3) -	7.4304e-2 (3.41e-3)
	1000	1.2695e-1 (1.46e-2) -	1.2339e-1 (5.37e-3) -	9.5312e-2 (3.54e-3) -	9.0997e-2 (5.39e-3) -	4.8426e-2 (1.84e-3)
LSMOP5	300	4.9799e+0 (4.11e+0) -	1.4527e-1 (3.32e-2) =	9.2454e-2 (1.68e-3) +	7.9752e-1 (4.33e-1) -	1.3380e-1 (1.12e-2)
	500	1.3991e+1 (3.47e+0) -	1.7148e-1 (2.94e-2) +	9.2290e-2 (1.48e-3) +	7.1865e-1 (4.46e-1) -	2.0241e-1 (4.52e-3)
	1000	9.1705e+0 (5.13e+0) -	1.9859e-1 (1.53e-2) -	9.3379e-2 (4.60e-3) -	6.5353e-1 (3.75e-1) -	8.7888e-2 (7.35e-3)

(Continued)

Table 2 (continued)

Problem	D	LMEA	GLMO	LERD	FLEA	MOEAWOD
LSMOP6	300	1.8092e+3 (3.93e+3) –	1.1378e+0 (4.22e–3) +	1.0111e+0 (1.80e–1) +	7.7495e+0 (3.70e+1) –	7.4127e+0 (2.99e+1)
	500	1.1527e+4 (1.74e+4) =	1.1022e+0 (1.77e–1) +	1.4918e+0 (5.64e–1) +	6.5899e+0 (3.08e+1) +	1.4041e+1 (1.57e+1)
	1000	5.2197e+3 (8.50e+3) =	1.2000e+0 (1.09e–3) +	5.2548e+0 (1.41e+1) –	9.0289e+0 (3.70e+1) –	3.2162e+0 (2.79e+0)
LSMOP7	300	1.6453e+0 (8.22e–1) –	7.0798e–1 (2.53e–2) –	5.1814e–1 (9.27e–4) +	8.1483e–1 (6.44e–2) –	5.4132e–1 (1.67e–2)
	500	1.1908e+2 (5.47e+2) –	6.5432e–1 (1.28e–2) +	5.2182e–1 (1.00e–2) +	7.2593e–1 (5.61e–2) =	1.1389e+0 (8.39e–1)
	1000	8.0443e–1 (1.62e–1) –	2.8070e–1 (2.34e–3) +	5.1951e–1 (3.80e–4) +	6.4701e–1 (3.35e–2) –	6.1828e–1 (2.38e–1)
LSMOP8	300	1.5945e–1 (1.07e–1) =	1.2143e–1 (3.64e–2) =	1.0097e–1 (5.26e–3) +	3.0994e–1 (1.44e–2) –	1.3621e–1 (4.55e–2)
	500	2.4880e–1 (1.32e–1) –	1.1993e–1 (3.41e–2) =	9.8846e–2 (5.25e–3) +	3.0550e–1 (4.98e–3) –	1.2776e–1 (5.19e–3)
	1000	1.1970e–1 (1.62e–1) –	1.4094e–1 (5.48e–3) –	8.3998e–2 (6.11e–3) –	2.9691e–1 (5.20e–3) –	6.3485e–2 (2.73e–3)
LSMOP9	300	1.5210e+0 (1.81e+0) =	1.1875e+0 (1.95e–1) –	9.6628e–1 (1.09e–1) –	1.0594e+0 (4.88e–1) –	5.3104e–1 (8.59e–2)
	500	1.1710e+2 (5.18e+1) –	1.0537e+0 (1.47e–1) –	1.0550e+0 (1.65e–1) –	9.5137e–1 (4.70e–1) –	3.4384e–1 (1.72e–2)
	1000	9.5989e+1 (6.78e+1) –	9.8228e–1 (3.68e–5) –	9.1195e–1 (2.14e–1) –	1.1797e+0 (3.42e–1) –	1.0077e–1 (4.93e–3)
+/-/=		0/21/6	9/14/4	9/17/1	4/21/2	

The LSMOPs test problems, designed for complex large-scale multi-objective optimization, effectively reflect the performance of an algorithm in large-scale scenarios. The IGD⁺ values of five algorithms on LSMOPs with a 3-dimensional objective and decision variables set at 300, 500, and 1000 dimensions are shown in the table below. In this table and in the following ones, ‘D’ denotes the dimensions of the decision variable.

As Table 2 shows, in the LSMOPs test suites, the MOEAWOD secured the best results in 12 out of the tests, indicated by black bold text. Following that, LERD in 7 and LMEA in only 1, but none matched the overall performance of MOEAWOD. When the decision variable is set at 1000 dimensions, MOEAWOD achieves the optimal solution 6 times in the LSMOPs test suite. This performance is notably superior to other comparison algorithms, providing evidence of its effectiveness in addressing LSMOPs. Notably, in LSMOP1 with 500 and 1000 decision variables, the IGD⁺ of MOEAWOD was significantly lower than its competitors. However, its performance in LSMOP6 with 500 decision variables was weaker, likely due to insufficient analysis of decision variable convergence and less accurate capture of variable relationships in weighted overlapping grouping. Despite this, the overall superiority of MOEAWOD in LSMOP test suites over LMEA, GLMO, LERD, and FLEA validates the effectiveness of the proposed weighted overlapping group-based MOEAs in handling LSMOPs.

In addition, the performance of the algorithm is evaluated on the WFG test suite, characterized by complex variable interactions and more frequent interactions between different variables. Table 3 compares the performance of the five algorithms across WFG test cases with 300, 500, and 1000 decision variables and 3 objectives.

Table 3: IGD⁺ obtained across WFGs test suites

Problem	D	LMEA	GLMO	LERD	FLEA	MOEAWORD
WFG1	300	1.0199e+0 (4.73e-1) -	1.4208e+0 (3.31e-2) -	1.1997e+0 (7.05e-2) -	1.1905e+0 (1.83e-1) -	6.4433e-1 (1.10e-1)
	500	7.1906e-1 (1.01e-1) -	1.4935e+0 (2.02e-2) -	1.2933e+0 (8.52e-2) -	1.1190e+0 (2.52e-1) -	5.5973e-1 (1.81e-1)
	1000	1.0904e+0 (1.23e-1) -	1.5329e+0 (1.41e-2) -	1.3338e+0 (7.71e-2) -	1.0982e+0 (2.99e-1) -	6.3016e-1 (9.02e-2)
WFG2	300	3.3143e-1 (5.75e-1) -	2.7695e-1 (1.86e-2) -	1.6528e-1 (1.85e-2) =	3.4815e-1 (5.02e-2) -	1.6531e-1 (4.43e-2)
	500	1.6650e-1 (8.27e-3) +	3.3616e-1 (1.88e-2) +	1.7765e-1 (1.87e-2) +	3.5137e-1 (4.51e-2) +	4.0484e-1 (3.04e-2)
	1000	1.8607e-1 (1.34e-2) +	4.2186e-1 (2.46e-2) +	1.8132e-1 (1.58e-2) +	2.5823e-1 (6.69e-2) +	5.7114e-1 (2.58e-3)
WFG3	300	9.7846e-2 (8.35e-3) +	3.5014e-1 (1.57e-2) -	1.9358e-1 (1.91e-2) +	6.2458e-2 (2.02e-2) +	2.9803e-1 (1.84e-2)
	500	1.6174e-1 (5.93e-3) +	4.8746e-1 (1.74e-2) -	1.6301e-1 (2.38e-2) +	4.8682e-2 (1.58e-2) +	3.5100e-1 (2.53e-2)
	1000	1.8414e-1 (7.34e-3) +	6.1968e-1 (3.15e-2) -	1.7616e-1 (1.31e-2) +	4.9273e-2 (1.44e-2) +	4.0908e-1 (4.52e-3)
WFG4	300	1.7287e-1 (3.07e-1) -	2.6688e-1 (8.00e-3) -	1.9349e-1 (1.36e-2) -	1.7557e-1 (2.72e-2) -	1.5268e-1 (6.90e-3)
	500	7.6302e-2 (1.47e-3) +	3.1084e-1 (1.14e-2) -	1.8359e-1 (1.48e-2) -	1.7476e-1 (3.75e-2) =	1.5623e-1 (1.07e-2)
	1000	7.7948e-2 (2.63e-3) +	3.5719e-1 (8.63e-3) -	1.9065e-1 (1.59e-2) -	1.6024e-1 (8.44e-3) =	1.6224e-1 (7.32e-5)
WFG5	300	1.4354e-1 (1.42e-2) +	2.0634e-1 (1.70e-2) =	1.7751e-1 (6.94e-3) =	1.8150e-1 (1.37e-2) =	1.8634e-1 (3.55e-2)
	500	2.4931e-1 (4.59e-2) -	2.1125e-1 (2.08e-2) -	1.7594e-1 (6.37e-3) -	1.8049e-1 (1.58e-2) -	1.7166e-1 (6.85e-2)
	1000	1.5343e-1 (2.19e-2) =	1.9278e-1 (1.32e-2) -	1.7685e-1 (5.35e-3) -	1.7856e-1 (1.38e-2) -	1.4874e-1 (1.63e-2)
WFG6	300	8.0985e-2 (1.47e-3) +	3.4475e-1 (1.23e-2) -	1.2229e-1 (9.21e-3) +	1.5931e-1 (2.53e-2) =	1.6667e-1 (2.53e-2)
	500	8.2918e-2 (6.91e-3) +	4.3767e-1 (1.41e-2) -	1.2295e-1 (2.15e-2) +	1.4236e-1 (1.11e-2) +	1.7248e-1 (8.83e-3)
	1000	9.5659e-2 (1.47e-2) =	5.4671e-1 (8.48e-3) -	1.2011e-1 (1.23e-2) -	1.4367e-1 (1.35e-2) -	9.5262e-2 (6.02e-4)

(Continued)

Table 3 (continued)

Problem	D	LMEA	GLMO	LERD	FLEA	MOEAWOD
WFG7	300	3.4690e-1 (5.65e-2) -	3.1046e-1 (1.17e-2) -	1.4960e-1 (6.02e-3) +	1.4759e-1 (9.74e-3) +	2.2856e-1 (2.36e-2)
	500	3.5427e-1 (7.55e-2) -	3.8793e-1 (1.16e-2) -	1.4374e-1 (4.48e-3) +	1.3985e-1 (1.45e-2) +	2.1793e-1 (1.16e-2)
	1000	3.4745e-1 (1.44e-1) -	5.1909e-1 (1.65e-2) -	1.5636e-1 (9.22e-3) +	1.4998e-1 (1.19e-2) +	2.0697e-1 (1.70e-2)
WFG8	300	1.8238e-1 (4.40e-2) +	2.9948e-1 (5.85e-3) -	2.4075e-1 (1.98e-2) =	3.1516e-1 (5.78e-2) -	2.4599e-1 (2.52e-2)
	500	1.7192e-1 (5.68e-3) +	3.4992e-1 (8.27e-3) -	2.1740e-1 (1.37e-2) +	2.2292e-1 (3.90e-2) +	2.4753e-1 (9.42e-3)
	1000	2.2155e-1 (1.29e-2) -	4.2018e-1 (1.15e-2) -	2.2165e-1 (6.97e-3) -	2.3424e-1 (7.04e-2) -	1.8006e-1 (4.94e-4)
WFG9	300	5.9226e-1 (1.15e-1) -	2.9353e-1 (7.98e-2) -	1.6029e-1 (3.36e-2) -	1.7515e-1 (1.00e-1) -	1.0993e-1 (1.42e-2)
	500	7.1619e-1 (6.38e-2) -	2.3443e-1 (1.29e-1) -	1.6817e-1 (2.87e-2) -	1.8658e-1 (1.04e-1) -	1.1767e-1 (5.94e-2)
	1000	7.4727e-1 (1.02e-1) -	3.2435e-1 (2.16e-1) -	1.8772e-1 (6.32e-2) -	1.4514e-1 (1.76e-2) -	9.5544e-2 (4.79e-4)
+/-/=		12/13/2	2/24/1	11/13/3	10/13/4	

In these tests, the MOEAWOD algorithm achieved the best results in 11 cases, while LMEA excelled in 8 cases. The WFG tests feature complex variable interactions, with more frequent interactions among different variables. The clustering approach of LMEA for variable grouping yielded good results, but overall, MOEAWOD showed superior optimization performance. The effectiveness of MOEAWOD in handling strong variable interactions outperformed LMEA, GLMO, LERD, and FLEA, highlighting its advantage in LSMOPs.

Fig. 4 displays the final populations optimized by the comparative algorithms on LSMOP1, involving 500 decision variables and 3,000,000 iterations. GLMO, and LERD achieved objective values around 1 but lacked uniform distribution and diversity. LMEA and FLEA were farther from the Pareto optimal reference, indicating poorer convergence, with the solutions of FLEA being particularly concentrated, showing limited diversity. MOEAWOD demonstrated superior performance, maintaining objective values close to 1 and exhibiting a more uniform distribution. However, there was some deviation from the true Pareto optimal reference. Such results may be attributed to errors in the analysis of the interaction among decision variables. In Fig. 5, the LSMOP8 test with 300 variables and 3 objectives, GLMO and LERD lack diversity, LMEA and FLEA struggle with convergence, and the solutions of FLEA concentrated between 5–15 in one objective, indicating poor convergence in that area. MOEAWOD algorithm maintains convergence on several objectives with decent results, the overall distribution of solutions is uniform, and the convergence and diversity are superior to several other algorithms.

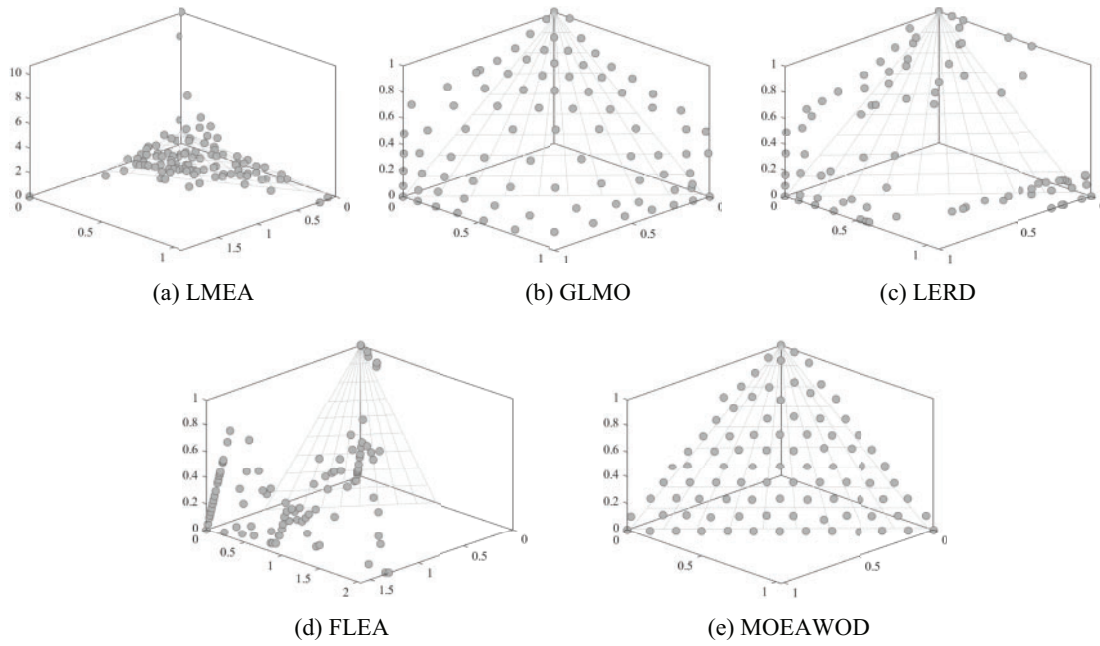


Figure 4: Effectiveness of comparison algorithms in LSMOP1 test problem with 500-dimensional 3-objective

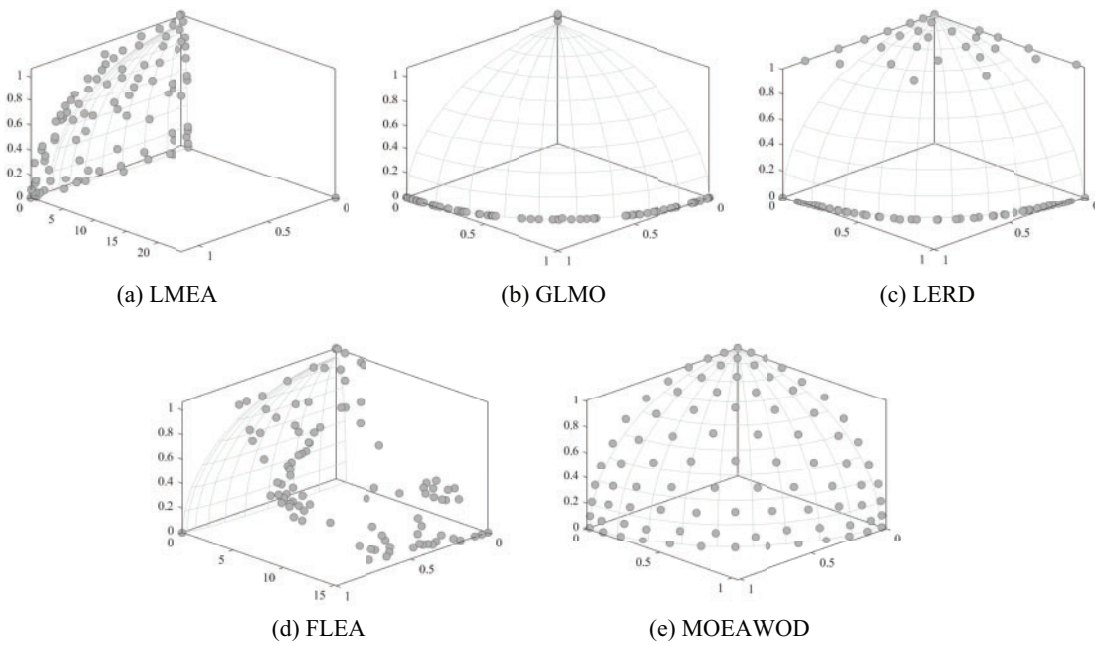


Figure 5: Effectiveness of comparison algorithm in the LSMOP8 test problem with 300-dimensional 3-objective

Fig. 6 illustrates the progression of IGD^+ values for different algorithms applied to two LSMOP test problems. Owing to the poor convergence performance exhibited by certain algorithms in LSMOPs, 100-dimensional LSMOP tests were employed to facilitate a more distinct comparison. The decision variable is 100-dimensional, the objective number is 3, and the maximum number of function evaluations is 1,000,000. Several comparison algorithms experienced varying degrees of rebound during convergence, indicating instability. MOEAWOD, however, showed stable convergence in both tests, continually evolving until convergence with noticeably superior IGD^+ values compared to others, demonstrating the most stable convergence performance and optimal IGD^+ results on both problems.

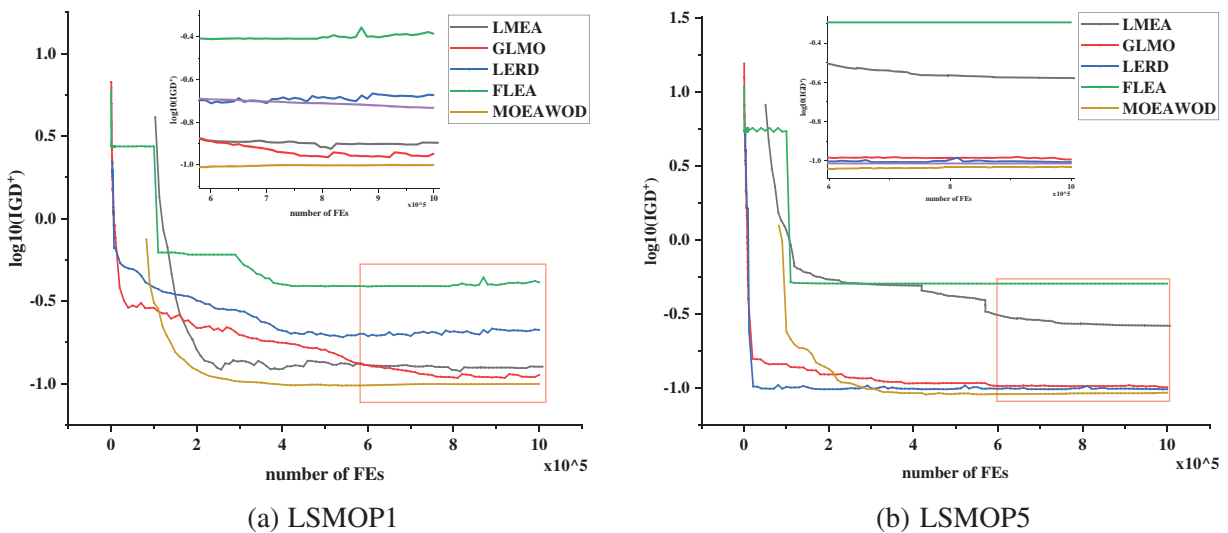


Figure 6: Evolution of IGD^+ metrics for the contrasting algorithms on the LSMOP test problem

5 Conclusion

This paper addresses the inefficiency of optimization algorithms in LSMOPs due to high decision variable dimensions by proposing MOEAWOD. Currently, group-based LSMOAs typically factor in decision variable interactions for grouping but overlook the intricate nature of variable interactions in LSMOPs. Having over 100 decision variables in a group still qualifies as an LSMOP, contradicting the original intent of divide-and-conquer.

The proposed MOEAWOD assesses the interactivity of decision variables using two weights and designates the most interactive decision variables to each group as overlapping elements. This strengthens the interactivity within each decision variable group while simultaneously diminishing the interactivity between different groups. Subsequently, each group underwent optimization employing a direction-guided population evolution strategy. The experimental results on various LSMOPs benchmark test suites demonstrate the remarkable convergence and diversity of the proposed MOEAWOD algorithm. The superiority of its IGD^+ index over other algorithms substantiates its effectiveness.

Nevertheless, MOEAWOD still exhibits certain limitations that warrant further investigation in future studies. Grouping based on weights acquired through DVA of random individuals before population evolution may result in interaction information that only represents the local subspace of the analyzed individuals. This could result in an inaccurate analysis of decision variables, limited to the partial space rather than the entire huge decision variable space. Additionally, the reliance on a limited

number of evaluations for the weight measurement in the DVA process may introduce errors that deviate from the actual situation, potentially impacting the performance of LSMOAs. Future work will delve into the dynamic analysis of decision variables, and DVA will be conducted based on the stage of population evolution to attain precise variable interactions. Additionally, there is consideration given to designing reasonable weights to achieve a more accurate evaluation of interaction performance.

Acknowledgement: None.

Funding Statement: This work was supported in part by the Central Government Guides Local Science and Technology Development Funds (Grant No. YDZJSX2021A038); in part by the National Natural Science Foundation of China under (Grant No. 61806138); in part by the China University Industry-University-Research Collaborative Innovation Fund (Future Network Innovation Research and Application Project) (Grant 2021FNA04014).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Liang Chen, Jingbo Zhang, Linjie Wu; data collection: Jingbo Zhang, Yubin Xu, Xingjuan Cai; analysis and interpretation of results: Liang Chen, Yubin Xu, Xingjuan Cai; draft manuscript preparation: Liang Chen, Jingbo Zhang, Linjie Wu, Xingjuan Cai. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The dataset used to support the results of this study is available at <https://GitHub.com/BIMK/PlatEMO/releases/tag/v4.2>.

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

References

1. Tian, Y., Si, L., Zhang, X., Cheng, R., He, C. et al. (2021). Evolutionary large-scale multi-objective optimization: A survey. *ACM Computing Surveys*, 54(8), 1–34.
2. Sonmez, R., Aminbakhsh, S., Atan, T. (2020). Activity uncrashing heuristic with noncritical activity rescheduling method for the discrete time-cost trade-off problem. *Journal of Construction Engineering Management*, 146(8), 04020084.
3. Khoshoo, B., Blank, J., Pham, T. Q., Deb, K., Foster, S. N. (2023). Optimal design of electric machine with efficient handling of constraints and surrogate assistance. *Engineering Optimization*, 56(2), 274–292.
4. Azad, S. K., Aminbakhsh, S. (2022). ϵ -constraint guided stochastic search with successive seeding for multi-objective optimization of large-scale steel double-layer grids. *Journal of Building Engineering*, 46, 103767. <https://doi.org/10.1016/j.jobbe.2021.103767>
5. Yan, X., Zuo, H., Hu, C., Gong, W., Sheng, V. S. (2023). Load optimization scheduling of chip mounter based on hybrid adaptive optimization algorithm. *Complex System Modeling and Simulation*, 3(1), 1–11.
6. Yao, B., Chen, C., Shan, W., Yu, B. (2022). Artificial leaf-vein network optimisation algorithm for urban transportation network design. *International Journal of Bio-Inspired Computation*, 20(4), 256–268.
7. Zheng, Z., Wu, S., Huang, Q., Yang, J. (2022). Research on localisation algorithm of large irregular workpiece for industrial robot. *International Journal of Computing Science and Mathematics*, 15(1), 30–42.
8. Wang, F., Huang, Z., Wang, S. (2023). I ϵ + LGEA a learning-guided evolutionary algorithm based on I ϵ + indicator for portfolio optimization. *Complex System Modeling and Simulation*, 3(3), 191–201.

9. Wu, L., Wu, D., Zhao, T., Cai, X., Xie, L. (2023). Dynamic multi-objective evolutionary algorithm based on knowledge transfer. *Information Sciences*, 636, 118886.
10. Ren, H., Zhang, D. (2023). Non-destructive diagnosis of knee osteoarthritis based on sparse coding of MRI. *International Journal of Computing Science and Mathematics*, 18(3), 276–285.
11. Suman, S., Chatterjee, D., Mohanty, R. (2023). Power quality improvement for microgrid-connected PV-based converters under partial shading conditions using mixed optimisation algorithms. *International Journal of Bio-Inspired Computation*, 21(3), 123–136.
12. Chen, G., Guo, Y., Huang, M., Gong, D., Yu, Z. (2022). A domain adaptation learning strategy for dynamic multiobjective optimization. *Information Sciences*, 606, 328–349.
13. Cai, X., Lan, Y., Zhang, Z., Wen, J., Cui, Z. et al. (2021). A many-objective optimization based federal deep generation model for enhancing data processing capability in IoT. *IEEE Transactions on Industrial Informatics*, 19(1), 561–569.
14. Cui, Z., Zhao, T., Wu, L., Qin, A., Li, J. (2023). Multi-objective cloud task scheduling optimization based on evolutionary multi-factor algorithm. *IEEE Transactions on Cloud Computing*, 11(4), 3685–3699.
15. Cai, X., Wu, L., Zhao, T., Wu, D., Zhang, W. et al. (2024). Dynamic adaptive multi-objective optimization algorithm based on type detection. *Information Sciences*, 654, 119867.
16. Verma, P., Parouha, R. P. (2022). Solving systems of nonlinear equations with real world problems using an advanced hybrid algorithm. *International Journal of Bio-Inspired Computation*, 19(4), 238–252.
17. Deb, K., Jain, H. (2013). 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), 577–601.
18. Shu, Z., Song, A., Wu, G., Pedrycz, W. (2023). Variable reduction strategy integrated variable neighborhood search and NSGA-II hybrid algorithm for emergency material scheduling. *Complex System Modeling and Simulation*, 3(2), 83–101.
19. Zhang, J., Gong, B., Waqas, M., Tu, S., Chen, S. (2023). Many-objective optimization based intrusion detection for in-vehicle network security. *IEEE Transactions on Intelligent Transportation Systems*, 24(12), 15051–15065.
20. Cui, Z., Jin, Y., Zhang, Z., Xie, L., Chen, J. (2023). An interval multi-objective optimization algorithm based on elite genetic strategy. *Information Sciences*, 648, 119533.
21. Filipkowska, M. S. (2022). Two combined methods for the global solution of implicit semilinear differential equations with the use of spectral projectors and Taylor expansions. *International Journal of Computing Science and Mathematics*, 15(1), 1–29.
22. Bai, H., Fan, T., Niu, Y., Cui, Z. (2022). Multi-UAV cooperative trajectory planning based on many-objective evolutionary algorithm. *Complex System Modeling and Simulation*, 2(2), 130–141.
23. Shen, X., Pan, H., Ge, Z., Chen, W., Song, L. et al. (2023). Energy-efficient multi-trip routing for municipal solid waste collection by contribution-based adaptive particle swarm optimization. *Complex System Modeling and Simulation*, 3(3), 202–219.
24. Guo, K., Bai, Z., Ma, Z. (2023). Application of particle swarm optimisation algorithm in manipulator compliance control. *International Journal of Computing Science and Mathematics*, 18(2), 113–127.
25. Deng, L., Di, Y., Song, L., Gong, W. (2023). LTCSD: A large-scale tri-particle competitive swarm optimizer based on decomposition for multiobjective optimization. *Applied Intelligence*, 53(20), 24034–24055.
26. Li, W., Chen, Y., Cai, Q., Wang, C., Huang, Y. et al. (2022). Dual-stage hybrid learning particle swarm optimization algorithm for global optimization problems. *Complex System Modeling and Simulation*, 2(4), 288–306.

27. Li, B., Zhang, Y., Yang, P., Yao, X., Zhou, A. (2023). A two-population algorithm for large-scale multi-objective optimization based on fitness-aware operator and adaptive environmental selection. *IEEE Transactions on Evolutionary Computation*. <https://doi.org/10.1109/TEVC.2023.3296488>
28. Ma, M., Wang, W., Li, X. (2023). Multi-population artificial bee colony algorithm based on the nearest neighbour partition. *International Journal of Computing Science and Mathematics*, 18(3), 235–244.
29. Liu, S., Lin, Q., Li, J., Tan, K. C. (2023). A survey on learnable evolutionary algorithms for scalable multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 27(6), 1941–1961.
30. Qiao, K., Liang, J., Qu, B., Yu, K., Yue, C. et al. (2022). Differential evolution with level-based learning mechanism. *Complex System Modeling and Simulation*, 2(1), 35–58.
31. Hu, Z., Fang, C., Wang, Z., Tseng, S. M., Dong, M. (2023). Many-objective optimization based-content popularity prediction for cache-assisted cloud-edge-end collaborative IoT networks. *IEEE Internet of Things Journal*, 11(1), 1190–1200.
32. Gong, J., Yan, X., Hu, C. (2022). An ensemble-surrogate assisted cooperative particle swarm optimisation algorithm for water contamination source identification. *International Journal of Bio-Inspired Computation*, 19(3), 169–177.
33. Xie, F., Li, L., Li, L., Huang, Y., He, Z. (2023). A decomposition-based multi-objective Jaya algorithm for lot-streaming job shop scheduling with variable sublots and intermingling setting. *Expert Systems with Applications*, 228, 120402.
34. Wang, Z., Pei, Y., Li, J. (2023). A survey on search strategy of evolutionary multi-objective optimization algorithms. *Applied Sciences*, 13(7), 4643.
35. Sander, F., Zille, H., Mostaghim, S. (2018). Transfer strategies from single-to multi-objective grouping mechanisms. *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 729–736. New York, NY, USA, Association for Computing Machinery.
36. Yang, X., Zou, J., Yang, S., Zheng, J., Liu, Y. (2021). A fuzzy decision variables framework for large-scale multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 27(3), 445–459.
37. Wang, S. T., Zheng, J. H., Liu, Y., Zou, J., Yang, S. X. (2023). An extended fuzzy decision variables framework for solving large-scale multiobjective optimization problems. *Information Sciences*, 643, 119221.
38. Li, L., Li, Y., Lin, Q., Liu, S., Zhou, J. et al. (2023). Neural net-enhanced competitive swarm optimizer for large-scale multiobjective optimization. *IEEE Transactions on Cybernetics*. <https://doi.org/10.1109/TCYB.2023.3287596>
39. Ma, X., Liu, F., Qi, Y., Wang, X., Li, L. et al. (2016). A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables. *IEEE Transactions on Evolutionary Computation*, 20(2), 275–298.
40. Gao, M., Feng, X., Yu, H., Li, X. (2023). A large-scale multiobjective evolutionary algorithm with overlapping decomposition and adaptive reference point selection. *Applied Intelligence*, 53, 21576–21605. <https://doi.org/10.1007/s10489-023-04596-3>
41. Chen, H., Cheng, R., Wen, J., Li, H., Weng, J. (2020). Solving large-scale many-objective optimization problems by covariance matrix adaptation evolution strategy with scalable small subpopulations. *Information Sciences*, 509, 457–469.
42. Antonio, L. M., Coello, C. A. C. (2013). Use of cooperative coevolution for solving large scale multiobjective optimization problems. *Proceedings of the 2013 IEEE Congress on Evolutionary Computation*, pp. 2758–2765. Cancun, Mexico, IEEE.
43. Cao, B., Zhao, J., Gu, Y., Ling, Y., Ma, X. (2020). Applying graph-based differential grouping for multiobjective large-scale optimization. *Swarm Evolutionary Computation*, 53, 100626.
44. Li, Y., Li, L., Tang, H., Lin, Q., Ming, Z. et al. (2023). Redefined decision variable analysis method for large-scale optimization and its application to feature selection. *Swarm Evolutionary Computation*, 82, 101360.

45. Song, A., Yang, Q., Chen, W. N., Zhang, J. (2016). A random-based dynamic grouping strategy for large scale multi-objective optimization. *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*, pp. 468–475. Vancouver, BC, Canada, IEEE.
46. Li, M., Wei, J. (2018). A cooperative co-evolutionary algorithm for large-scale multi-objective optimization problems. *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 1716–1721. New York, NY, USA, Association for Computing Machinery.
47. Guo, Y., Ge, S., Huang, Y., Zhang, Y., Jiang, E. et al. (2023). Low-carbon routing based on improved artificial bee colony algorithm for electric trackless rubber-tyred vehicles. *Complex System Modeling and Simulation*, 3(3), 169–190.
48. George, S. S., Pramila, R. S. (2023). Improved whale social optimisation algorithm and deep fuzzy clustering for optimal and QoS-aware load balancing in cloud computing. *International Journal of Bio-Inspired Computation*, 22(1), 40–52.
49. Huband, S., Hingston, P., Barone, L., While, L. (2006). A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10(5), 477–506.
50. Zhang, X., Tian, Y., Cheng, R., Jin, Y. (2016). A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 22(1), 97–112.
51. Zille, H. (2019). *Large-scale multi-objective optimisation: New approaches and a classification of the state-of-the-art (Dissertation)*. Otto von Guericke University, Magdeburg, Germany.
52. He, C., Cheng, R., Li, L., Tan, K. C., Jin, Y. (2022). Large-scale multiobjective optimization via reformulated decision variable analysis. *IEEE Transactions on Evolutionary Computation*, 28(1), 47–61.
53. Li, L., He, C., Cheng, R., Li, H., Pan, L. et al. (2022). A fast sampling based evolutionary algorithm for million-dimensional multiobjective optimization. *Swarm Evolutionary Computation*, 75, 101181.
54. Tian, Y., Zhu, W., Zhang, X., Jin, Y. (2023). A practical tutorial on solving optimization problems via PlatEMO. *Neurocomputing*, 518, 190–205.
55. Ishibuchi, H., Imada, R., Masuyama, N., Nojima, Y. (2019). Comparison of hypervolume, IGD and IGD+ from the viewpoint of optimal distributions of solutions. *Proceedings of the Evolutionary Multi-Criterion Optimization: 10th International Conference*, pp. 332–345. East Lansing, MI, USA, Springer.
56. Guo, Y., Chen, G., Jiang, M., Gong, D., Liang, J. (2022). A knowledge guided transfer strategy for evolutionary dynamic multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 27(6), 1750–1764.
57. Chen, G., Guo, Y., Wang, Y., Liang, J., Gong, D. et al. (2023). Evolutionary dynamic constrained multiobjective optimization: Test suite and algorithm. *IEEE Transactions on Evolutionary Computation*. <https://doi.org/10.1109/TEVC.2023.3313689>