

## Research Article

# A Decomposition-Based Harmony Search Algorithm for Multimodal Multiobjective Optimization

Wei Xu , Weifeng Gao , and Qianlong Dang

School of Mathematics and Statistics, Xidian University, Xi'an 710126, China

Correspondence should be addressed to Weifeng Gao; gaoweifeng2004@126.com

Received 18 March 2022; Accepted 21 April 2022; Published 25 May 2022

Academic Editor: Xueyi Wang

Copyright © 2022 Wei Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multimodal multiobjective optimization problem (MMOP) is a special kind of multiobjective optimization problem (MOP) with multimodal characteristics, where multiple different Pareto optimal sets (PSs) map to the same Pareto optimal front (PF). To handle MMOPs, a decomposition-based harmony search algorithm (called MOEA/D-HSA) is devised. In MOEA/D-HSA, multiple individuals who are assigned to the same weight vector form a subpopulation for finding multiple different PSs. Then, an environmental selection method based on greedy selection is designed to dynamically adjust the subpopulation scale for keeping the population diversity. Finally, the modified harmony search algorithm and elite learning strategy are utilized to balance the diversity and convergence of the population. Experimental results on the CEC 2019 test suite reveal that MOEA/D-HSA has superior performance than a few state-of-the-art algorithms.

## 1. Introduction

MOPs with multiple objectives that require to be optimized simultaneously are frequently occurring in real-world applications [1, 2]. In general, an MOP is defined as follows [3]:

$$\begin{aligned} \min_{X \in \Omega} F(X) &= (f_1(X), \dots, f_m(X))^T, \\ x_i &\in (x_i^l, x_i^u), \quad i = 1, 2, \dots, n, \end{aligned} \quad (1)$$

where  $\Omega$  refers to the search space and  $X = (x_1, \dots, x_n)^T \in \Omega$  denotes an  $n$ -dimensional decision variable;  $f_j(X)$  indicates the  $j$ -th ( $j = 1, 2, \dots, m$ ) objective, and  $m$  refers to the amount of objectives;  $x_i^u$  and  $x_i^l$  are the upper and lower bounds of  $x_i$ , respectively. In an MOP, a solution  $X$  is said to dominate another solution  $Y$ , if and only if  $\forall j \in \{1, 2, \dots, m\}$ ,  $f_j(X) \leq f_j(Y)$ , and  $\exists k \in \{1, 2, \dots, m\}$ ,  $f_k(X) < f_k(Y)$ . If there exists no solution that dominates  $X$ , then  $X$  is called a Pareto optimal solution [4, 5]. The set of all Pareto optimal solutions is called PS [6], which is denoted as  $X^*$ . The image of  $X^*$  in the objective space is termed as PF [7].

A large number of MOPs with multimodal properties exist in practical applications [8], such as scheduling

problem [9], epsilon-efficient solutions problem [10], rocket engine problem [11], and 0/1 knapsack problem [12]. This category of problems is termed MMOPs in [13]. In MMOPs, multiple different PSs map to the same PF. As depicted in Figure 1, an MMOP with two different Pareto regions (i.e., Region1 and Region2) in the decision space is presented. In Figure 1, all Pareto regions map to the same Pareto front (i.e., PF) in the objective space. For example,  $A_1 \in \text{Region1}$  and  $A_2 \in \text{Region2}$  correspond to the same point A and  $A_1 \neq A_2$ . All PSs should be discovered in handling MMOPs.

In the past few years, numerous multimodal multiobjective evolutionary algorithms (MMOEA) have been developed to tackle MMOPs, which can be grouped into the following types:

- (i) Pareto-based MMOEAs: DN-NSGA-II [13], Niching-CMA [14], and Omni-optimizer [15] are the three typical algorithms in this class. DN-NSGA-II proposed a niching method to deal with MMOPs, in which the crowding distance is adopted to find multiple PSs. Unlike DN-NSGA-II, an omni-optimizer calculated the crowding distance in both the decision and objective spaces to locate the

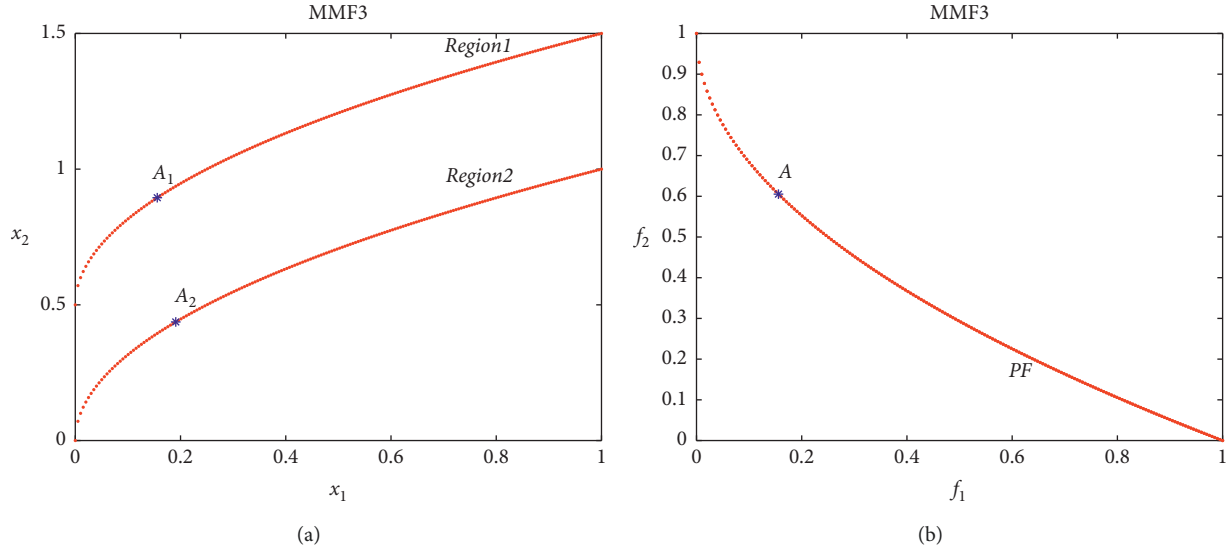


FIGURE 1: Example of MMF3 with two PSs to illustrate MMOP.

equivalent PSs. In Niching-CMA, a niche function is constructed to replace the crowding distance in the omni-optimizer by summing the distances of the individuals in the two spaces. Experimental results indicated that Niching-CMA can improve the diversity of decision space. Subsequently, a method similar to the omni-optimizer is proposed to deal with MMOPs (named MO\_Ring\_PSO\_SCD) [16], where the ring topology and diversity measure are utilized to keeping the population diversity. In addition, Liu et al. [17] proposed a penalty density strategy (called CPDEA) for solving MMOPs. In CPDEA, the individual density is calculated by the transformed Euclidean distance and used as the selection criteria to search for the equivalent PSs. Then, they proposed an MMOEA based on archives and recombination strategy to find multiple PSs (called TriMOEA-TA&R) [18]. Fan et al. [19] devised a zoning search approach (ZS) to tackle MMOPs, where the whole space is classified into multiple subspaces and existing MMOEAs are executed in each subspace. The results revealed that ZS helps to increase the property of MMOEAs in solving MMOPs. Inspired by this, Fan et al. proposed a zoning search approach with adaptive resource allocating [20] for balanced and imbalanced MMOPs. Lin et al. [21] presented a clustering-based MMOEA that is able to efficiently locate local PSs. Zhang et al. [22] devised a knee-based approach to deal with MMOPs, where a multicriteria decision strategy is introduced to search for equivalent PS regions. Afterwards, the repulsive force between isotropic magnetic individuals is introduced into MMOPs to locate different PSs by Zhang et al. [23, 24]. In addition, inspired by online learning [25, 26]. Li et al. [27] devised a differential evolution based on

reinforcement learning to handle MMOPs. However, these algorithms generally adopt Pareto-based method to search for equivalent PSs, so they cannot deal with large-scale MMOPs.

- (ii) Decomposition-based MMOEAs: like MOEA/D [28], an MMOP is decomposed into a few simple subproblems through a group of evenly distributed vectors in decomposition-based MMOEAs. In [29], the diversity maintenance strategy combining penalty boundary intersection technique [28] and two distance metrics is introduced into the decision space for locating equivalent PSs. Later, a similar concept is suggested to handle the large-scale optimization problems in [30]. Tanabe et al. [31] designed a modified MOEA/D for MMOPs, which employs the addition and deletion operators to search for equivalent PSs. In [32], this concept was expanded as a framework to increase the diversity of MMOEAs in the decision space. In [33], an evolutionary algorithm based on the graph Laplacian is presented to tackle MMOPs, which employs the decomposition technique in two spaces to obtain equivalent Pareto solutions. Although the above algorithms can obtain multiple different PSs, they are unable to balance the population diversity in the decision and objective spaces well.
- (iii) Indicator-based MMOEAs: the indicator-based approaches utilize the chosen indicators to guide population converge to PF. Li et al. [34] proposed a weighted indicator-based MMOEA (termed MMEA-WI), which devised an indicator to keep the diversity in the decision space and introduced a convergence archive for approximating the true PF. In [35], a Niching indicator is designed to solve MMOPs, where individuals' fitness values are calculated to keep the diversity of PSs. In summary, the

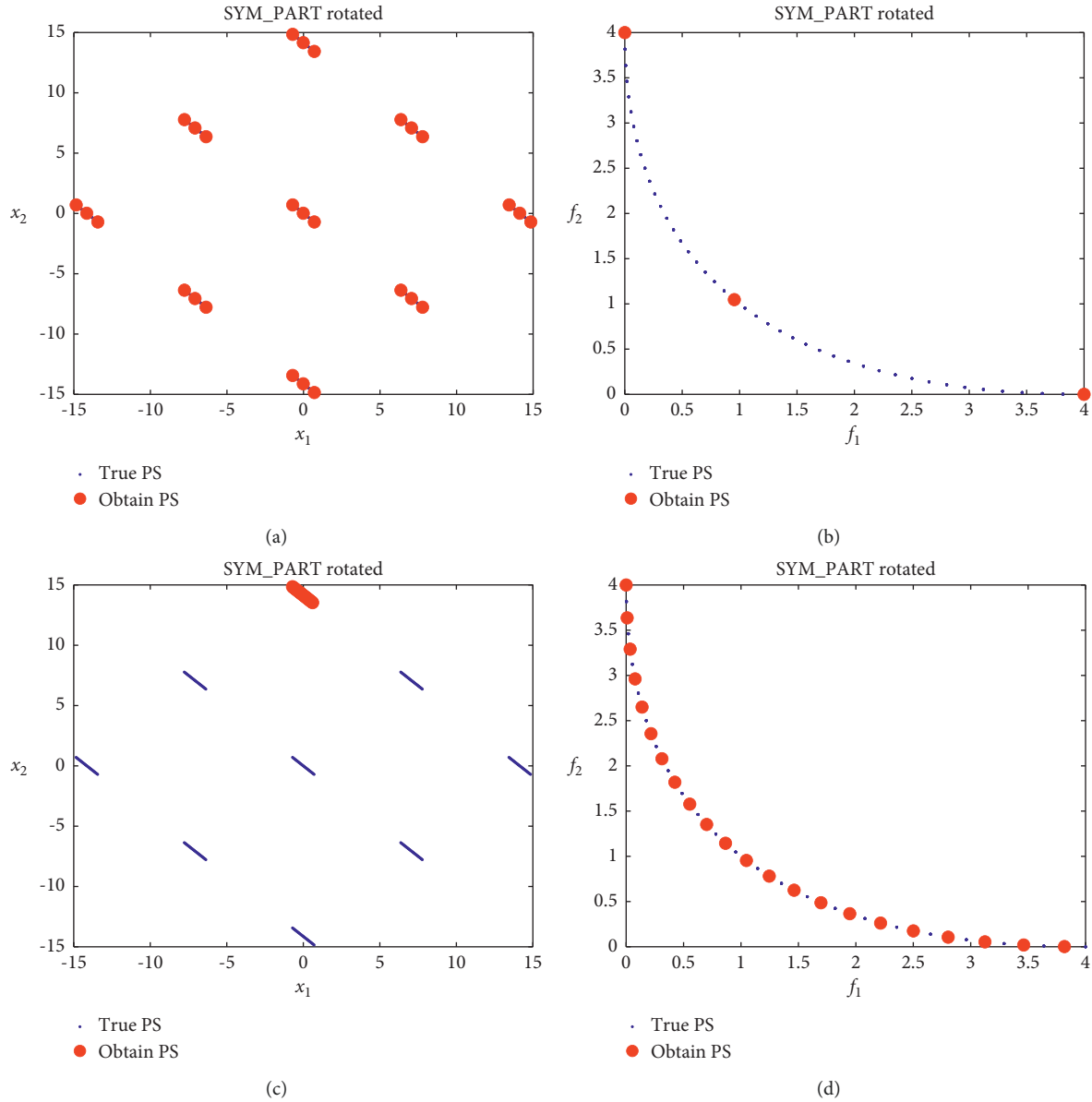


FIGURE 2: Example of SYM-PART rotated with nine PSs to explain the importance of diversity.

indicator-based MMOEAs usually adopt fitness indicator to keep the diversity in the decision space, but the calculation of indicator wastes a large amount of computing resources, especially for large-scale MMOPs:

To sum up, although the decomposition-based MMOEAs can find multiple equivalent PSs in dealing with MMOPs, they are unable to well balance the diversity in the decision and objective spaces. In other words, they increase the diversity in the decision space at the cost of the performance in the objective space. To deal with this issue, a decomposition-based harmony search algorithm (named MOEA/D-HSA) is presented. In MOEA/D-HSA, multiple individuals who are assigned to the same weight vector form a subpopulation after an MMOP is decomposed. Then, an environmental selection method is used for guiding the individuals in the

subpopulation to find the equivalent PSs. Finally, the modified harmony search algorithm is utilized to generate the offspring, which helps to balance the relationship of diversity and convergence. At the same time, an elite learning strategy is employed to prevent premature convergence. The test results on CEC 2019 validate the property of MOEA/D-HSA.

## 2. Materials and Methods

**2.1. Motivation.** The diversity of decision and objective spaces is crucial in solving MMOPs. In the following, we take SYM-PART rotated with nine distinct PSs as an example to explain the importance of the diversity in these two spaces. As depicted in Figures 2(a) and 2(b), all PSs are located in the decision space, but only three points are found on PF. The

reason is that the algorithm pays more attention to the diversity in the decision space. For Figures 2(c) and 2(d), although the obtained PF has well-diversity and well-convergence, only one PS is located. When an algorithm is only concerned about the diversity in the objective space, the situation in Figures 2(c) and 2(d) may appear. Therefore, the following issues need to be considered for preventing situations in Figure 2.

- (1) How to preserve the diversity in the objective space? This problem has been widely studied in multi-objective evolutionary algorithms (MOEAs). The classic methods include the weighted Tchebycheff approach [28], nondominated sorting method [36], and the fitness indicator technology [37, 38]. The weighted Tchebycheff approach is selected to deal with this problem. The reason is that the whole objective space is classified into multiple subspaces by a group of evenly distributed weight vectors in the weighted Tchebycheff approach. Thus, it can keep the diversity in this space. Besides, for balancing the exploration and exploitation of the algorithm, a modified harmony search algorithm is adopted to generate the offspring, where the algorithm parameters such as harmony memory size (HMS), harmony memory consideration rate (HMCR), and pitch adjusting rate (PAR) are dynamically adjusted.
- (2) How to preserve the diversity in the decision space? For handling this problem, the framework of subpopulation is adopted to search for different PSs, i.e., each weight vector corresponds to multiple individuals instead of one. As shown in Figure 3, an MMOP is decomposed into a few simple subproblems by a group of evenly distributed weight vectors  $W_j$  ( $j = 1, 2, \dots, i$ ), and each weight vector is given to three distinct individuals to shape a subpopulation (i.e., the red, yellow, and blue dots indicate distinct individuals who are assigned to the same weight vector). In this way, the individuals corresponding to the same weight vector search toward the equivalent PSs. Furthermore, preserving the diversity of the subpopulation is vital in MMOEAs. Therefore, an environmental selection method is designed to separate individuals belonging to the same PS in the subpopulation.

**2.2. The Proposed Algorithm.** A decomposition-based harmony search algorithm for MMOPs, termed MOEA/D-HSA, is presented. In MOEA/D-HSA, first, initialize population  $P$ , weight vector  $W$ , and subpopulation  $T_i$  ( $i = 1, 2, \dots, NP$ ) (lines 1–6). Then, multiple individuals that are assigned to the same weight vector form a subpopulation (lines 7–9). Next, an offspring  $\mu$  is generated by a modified harmony search algorithm and elite learning strategy (line 13). Lastly, the subpopulation is dynamically adjusted by an environmental selection method, which helps to preserve the diversity of the decision space (line 14). Algorithm 1 depicts the framework of MOEA/D-HSA.

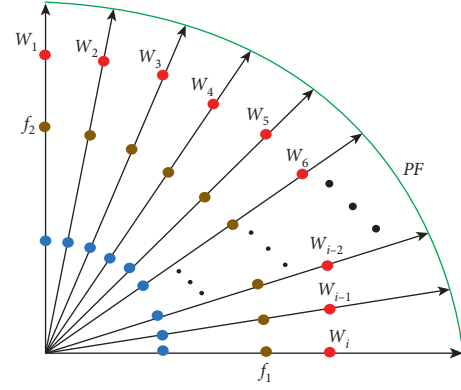


FIGURE 3: Illustration of the subpopulation framework.

**2.3. Reproduction.** For balancing the diversity and convergence of the population, a modified harmony search algorithm [39] shown in Algorithm 2 is used to generate the offspring. In Algorithm 2, first, initialize the algorithm parameters (i.e., HMS, HMCR, PAR, and BW). Then, parameters HMS, HMCR, and PAR are updated as follows [40, 41]:

$$HMS = HMS_{\max} - \left\lfloor \frac{FES}{\text{MaxFES}} \right\rfloor \times (HMS_{\max} - HMS_{\min}), \quad (2)$$

$$HMCR = HMCR_{\max} - \frac{(HMCR_{\max} - HMCR_{\min})}{\text{MaxFES}} \cdot FES, \quad (3)$$

$$PAR = PAR_{\min} + \frac{(PAR_{\max} - PAR_{\min})}{\text{MaxFES}} \cdot FES, \quad (4)$$

where  $\text{MaxFES}$  is the maximum number of evaluation functions,  $FES$  is the current number of evaluation functions,  $HMS \in [HMS_{\min}, HMS_{\max}]$ ,  $HMCR \in [HMCR_{\min}, HMCR_{\max}]$  and  $PAR \in [PAR_{\min}, PAR_{\max}]$ . With the help of adaptive parameters, the proposed algorithm has more extensive exploration ability in the early phase of evolution and nice exploitation ability in the late phase of evolution. It means that MOEA/D-HSA can well balance the relationship of diversity and convergence. Next, the harmony memory (HM) containing HMS individuals is constructed. Finally, the offspring  $\mu$  is generated.

**2.4. The Elite Learning Strategy.** For avoiding premature convergence, the elite learning strategy is employed to generate the offspring  $\mu$ , which can increase the global search ability and help the algorithm to jump out of local optimal location.

$$x_{\text{new},j} = x_{\text{best},j} + \text{Gauss}(0, pr^2) \times (x_j^u - x_j^l) \cdot \text{if} \cdot \text{rand} < pr, \quad (5)$$

where  $\text{Gauss}(0, pr^2)$  is a Gaussian distribution with mean 0 and standard deviation  $pr$ ,  $x_{\text{best},j}$  is the  $j$ -th dimension variable of the best individual in  $HM$ , and  $x_j^u$  and  $x_j^l$  are the

**Input:** population size  $NP$ , decision variable dimension  $n$ , algorithm parameters:  $HMS$ ,  $HMCR$ ,  $PAR$ , bandwidth ( $BW$ ),  $pr$ ,  $t$ , the maximum number of evaluation functions  $MaxFES$ ,

**Output:** the set  $TP$ .

- (1) Initialize population  $P$  and weight vector  $W$ ;
- (2) Calculate the fitness  $Fit$  and ideal point  $Z^*$ ;
- (3) Set  $FES = NP$ ;
- (4) for  $i = 1:NP$  do
- (5)   Set  $T_i = []$ ;
- (6) end for
- (7) for  $i = 1:NP$  do
- (8)    $T_j = T_j \cup X_i$ , where  $j = \min\{g^{tch}(X_i|W_1, Z^*), \dots, g^{tch}(X_i|W_{NP}, Z^*)\}$ ;
- (9) end for
- (10) while  $FES < MaxFES$  do
- (11)   Set  $T = T_1 \cup \dots \cup T_{NP}$ ;
- (12)   for  $i = 1:|T|$  do
- (13)     Generate offspring  $\mu$  by **Algorithm 2** and elite learning strategy;
- (14)     Update ideal point  $Z^*$  and subpopulation  $T_j$  by **Algorithm 3**, where  $j = \min\{g^{tch}(\mu|W_1, Z^*), \dots, g^{tch}(\mu|W_{NP}, Z^*)\}$ ;
- (15)      $FES = FES + 1$ ;
- (16)   end for
- (17) end while
- (18)  $TP \leftarrow$  all Pareto optimal solutions in  $T_1 \cup \dots \cup T_{NP}$ ;

ALGORITHM 1: MOEA/D-HSA.

**Input:** archive  $EXA$ , individual  $X_i$ , algorithm parameters:  $HMS$ ,  $HMCR$ ,  $PAR$ ,  $BW$ , the maximum number of evaluation functions  $MaxFES$ , the current number of evaluation functions  $FES$ , decision variable dimension  $n$ .

**Output:** an offspring  $\mu$ .

- (1) Update the parameters  $HMS$ ,  $HMCR$ , and  $PAR$  using equations (2)–(4)
- (2) Select the nearest  $HMS$  individuals to  $X_i$  in  $EXA$  and put them into  $HM$
- (3) for  $j = 1:n$  do
- (4)   if  $\text{rand} \leq HMCR$ , then
- (5)      $x_{\text{new},j} \in \{x_{1,j}, x_{2,j}, \dots, x_{HMS,j}\}$
- (6)     if  $\text{rand} \leq PAR$ , then
- (7)        $x_{\text{new},j} = x_{\text{new},j} + \text{rand} \cdot BW$
- (8)     end if
- (9)   else
- (10)     $x_{\text{new},j} = x_j^l + \text{rand} \cdot (x_j^u - x_j^l)$ ;
- (11) end for
- (12)  $\mu = (x_{\text{new},1}, \dots, x_{\text{new},n})^T$ ;

ALGORITHM 2: Reproduction.

upper and lower bounds of  $x_j$ ,  $\text{rand} \in (0, 1)$ . In the early phase of the algorithm, the larger  $pr$  value makes the offspring have a greater mutation range that is helpful for global search. In the late phase of the algorithm, small mutation range is helpful for local search. Therefore,  $pr$  is updated as follows:

$$pr = pr_{\max} - \frac{(pr_{\max} - pr_{\min})}{\text{MaxFES}} FES, \quad (6)$$

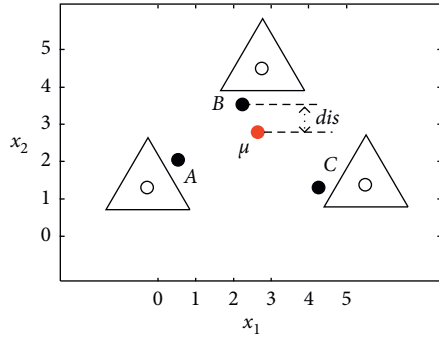
where  $\text{MaxFES}$  is the maximum number of evaluation functions,  $FES$  is the current number of evaluation functions,  $pr_{\max}$  and  $pr_{\min}$  are the upper and lower bounds of  $pr$ , and  $pr$  decreases as the iteration increases.

**2.5. Environmental Selection.** For MOEA/D-HSA, it is vital to keep the diversity of subpopulations in the decision space. To reach this goal, the environmental selection method presented in Algorithm 3 is employed to separate individuals belonging to the same PS in the subpopulation. In Algorithm 3, when the subpopulation scale exceeds  $t$ , the individual in the subpopulation that is nearest to the offspring  $\mu$  is compared with  $\mu$  by the weighted Tchebycheff approach and the better individuals are saved. Otherwise,  $\mu$  into the next generation. In the following, we take the subpopulation  $T_j$  as an example to explain the working mechanism of environmental selection. As depicted in Figure 4, an MMOP has three

**Input:** offspring  $\mu$ , Subpopulation  $T_j$ , algorithm parameter  $t$ .  
**Output:** subpopulation  $T_j$ .

- (1) if  $|T_j| + 1 > t$ , then
- (2) Select the nearest individual  $X$  to  $\mu$  in  $T_j$
- (3) if  $g^{tch}\{\mu|W_j, Z^*\} < g^{tch}\{X|W_j, Z^*\}$ , then
- (4)  $T_j = T_j \setminus X$  and  $T_j = T_j \cup \mu$
- (5) end if
- (6) else
- (7)  $T_j = T_j \cup \mu$
- (8) end if

ALGORITHM 3: Environmental Selection

FIGURE 4: Example of subpopulation  $T_j$  to illustrate the working mechanism of environmental selection.

equivalent PSs, and all individuals within the triangle are Pareto optimal solutions. In Figure 4, the subpopulation  $T_j$  consists of four different individuals (i.e., A, B, C, and offspring  $\mu$ ). When the subpopulation scale is larger than  $t$ , the individual in the subpopulation  $T_j$  that is nearest to the offspring  $\mu$  is compared with  $\mu$  (here, B is the closest individual to  $\mu$ ), and the preferable one is preserved. Otherwise,  $\mu$  into the subpopulation  $T_j$ . By this way, MOEA/D-HSA can find multiple different PSs.

**2.6. Complexity Analysis.** MOEA/D-HSA consists of the following components: initialization subpopulation, reproduction operation, and the environmental selection. The complexity of initialization subpopulation and environmental selection is  $O(NP^2)$ . The complexity of the reproduction operation is  $O(n \times NP)$ , where  $n$  is the decision variable dimension. In summary, considering  $n < NP$ , the complexity of MOEA/D-HSA is  $O(NP^2)$ .

### 3. Results and Discussion

**3.1. Performance Metric and Benchmark Test Functions.** In our experiment, 22 MMOPs from CEC 2019 competition [42] is utilized to examine the property of MMOEAs. In addition, inverted generational distance in the objective space (IGDF) [43, 44] and the Pareto set proximity (PSP) [16] are utilized to assess the property of algorithms. The

larger the PSP is, the better the property of algorithm is. IGDF is just the opposite. They are defined as follows:

(1) IGDF

$$IGDF = \sum_{Y \in PF} \frac{\min_{X \in pf} ED(X, Y)}{|PF|}, \quad (7)$$

where  $ED(X, Y)$  denotes the Euclidean distance between  $X$  and  $Y$ , PF and pf refer to the true PF and the obtained PF, respectively. IGDF can assess the diversity and convergence of the obtained solutions in the objective space.

(2) PSP

$$PSP = \frac{CR}{IGDX}, \quad (8)$$

$$IGDX = \sum_{Y \in PS} \frac{\min_{X \in ps} ED(X, Y)}{|PS|},$$

where IGDX is inverted generational distance in the decision space, PS and ps represent the true PS and the obtained PS, respectively, CR denotes the cover ratio between ps and PS, and the details refer to [16],  $ED(X, Y)$  is the Euclidean distance between  $X$  and  $Y$ . PSP can reflect the convergence and diversity of the population in the decision space.

**3.2. Competitive Algorithms and Parameter Setup.** To test the efficiency of the algorithm, MOEA/D-HAS is compared with five up-to-date MMOEAs: DN-NSGA-II [13], TriMOEA-TA&R [18], LORD [33], MOEA/D [28], and MO\_Ring\_PSO\_SCD [16]. For a fair comparison, the population size is set to 800, all algorithms are independently implemented 25 times, and the maximum function evaluations are set to 80000 [16]. In addition, the parameter setting of all competitors is consistent with their original references. For MOEA/D-HSA, according to [40, 41, 45], the algorithm parameters are  $HMCR_{\max} = 0.9$ ,  $HMCR_{\min} = 0.2$ ,  $PAR_{\max} = 1$ ,  $PAR_{\min} = 0.6$ ,  $HMS_{\max} = 5$ ,  $HMS_{\min} = 2$ ,  $BW = 0.1$ ,  $pr_{\max} = 0.2$ ,  $pr_{\min} = 0.05$ , and  $t = 5$ .

In further, the Wilcoxon's rank-sum test [46] and Friedman's test [47] are utilized to draw statistically reliable conclusions with significance level  $\alpha = 0.05$ . The symbols "+", "-", and "=" denote that MOEA/D-HSA is distinctly

TABLE 1: Experimental results of six algorithms for PSP.

	MMOEA/D-HSA	MO_Ring_PSO_SCD	TriMOEA-TA&R	DN-NSGA-II	LORD	MOEA/D
MMF1	<b>86.39</b>	65.76(+)	29.72(+)	47.67(+)	70.04(+)	8.62(+)
MMF1_e	<b>6.21</b>	5.43(+)	1.31(+)	0.81(+)	2.39(+)	0.75(+)
MMF1_z	<b>118.21</b>	90.04(+)	29.92(+)	48.32(+)	57.44(+)	8.36(+)
MMF2	<b>155.33</b>	105.86(+)	73.04(+)	68.85(+)	145.32(+)	4.31(+)
MMF3	<b>182.72</b>	138.59(+)	31.29(+)	84.24(+)	169.81(+)	5.74(+)
MMF4	<b>176.86</b>	112.72(+)	85.43(+)	38.46(+)	136.69(+)	2.19(+)
MMF5	<b>48.96</b>	33.71(+)	17.71(+)	14.83(+)	30.91(+)	3.86(+)
MMF6	<b>54.89</b>	35.39(+)	21.45(+)	18.46(+)	33.20(+)	4.76(+)
MMF7	<b>150.19</b>	108.12(+)	60.71(+)	96.55(+)	129.81(+)	8.27(+)
MMF8	<b>60.11</b>	46.20(+)	9.95(+)	17.01(+)	13.94(+)	0.11(+)
MMF9	567.62	331.47(+)	344.58(+)	36.57(+)	<b>648.92(-)</b>	0.96(+)
MMF10	0.71	<b>6.16(-)</b>	0.21(+)	0.13(+)	0.11(+)	0.07(+)
MMF11	0.59	<b>3.07(-)</b>	0.63(=)	0.46(+)	0.76(-)	0.11(+)
MMF12	0.52	<b>4.10(-)</b>	0.16(+)	0.32(+)	0.54(=)	0.14(+)
MMF13	1.92	<b>2.31(-)</b>	1.65(+)	1.77(+)	1.72(+)	0.97(+)
MMF14	<b>28.32</b>	26.95(+)	27.41(+)	14.99(+)	20.59(+)	7.75(+)
MMF14_a	<b>23.31</b>	23.01(=)	21.27(+)	11.70(+)	15.43(+)	9.39(+)
MMF15	4.65	<b>6.07(-)</b>	3.14(+)	3.37(+)	0.94(+)	0.68(+)
MMF15_a	4.81	<b>6.09(-)</b>	3.64(+)	4.10(+)	3.64(+)	3.63(+)
SYM_PART simple	21.61	20.67(+)	<b>76.69(-)</b>	0.69(+)	43.32(-)	0.01(+)
SYM_PART rotated	<b>19.19</b>	18.23(+)	13.66(+)	2.95(+)	17.97(+)	0.01(+)
Omni-test	11.47	7.16(+)	14.48(-)	1.09(+)	<b>26.17(-)</b>	0.03(+)
Sum up	+ / = / -	15 / 1 / 6	19 / 1 / 2	22 / 0 / 0	17 / 1 / 4	22 / 0 / 0

TABLE 2: Experimental results of six algorithms for IGDF.

	MMOEA/D-HSA	MO_Ring_PSO_SCD	TriMOEA-TA&R	DN-NSGA-II	LORD	MOEA/D
MMF1	7.442 e - 04	9.938 e - 04(+)	2.368 e - 03(+)	8.638 e - 04(+)	6.226 e - 04(-)	6.562 e - 04(-)
MMF1_e	1.882 e - 03	2.948 e - 03(+)	2.467 e - 03(+)	9.669 e - 04(-)	6.817 e - 04(-)	2.068 e - 03(+)
MMF1_z	6.461 e - 04	9.501 e - 04(+)	2.362 e - 03(+)	7.424 e - 04(+)	6.494 e - 04(+)	5.188 e - 04(-)
MMF2	3.449 e - 03	5.188 e - 03(+)	4.366 e - 03(+)	<b>1.165 e - 03(-)</b>	1.758 e - 03(-)	6.826 e - 04(-)
MMF3	2.917 e - 03	3.805 e - 03(+)	4.037 e - 03 (+)	7.288 e - 04(-)	1.576 e - 03(-)	6.729 e - 04(-)
MMF4	<b>4.578 e - 04</b>	8.801 e - 04 (+)	3.107 e - 02(+)	7.881 e - 04(+)	5.319 e - 04(+)	4.781 e - 04(+)
MMF5	<b>6.162 e - 04</b>	9.423 e - 04(+)	4.021 e - 03 (+)	8.161 e - 04(+)	6.203 e - 04(+)	6.289 e - 04(+)
MMF6	<b>5.757 e - 04</b>	9.071 e - 04(+)	2.252 e - 03 (+)	8.218 e - 04(+)	5.811 e - 04(+)	6.296 e - 04(+)
MMF7	<b>5.738 e - 04</b>	9.158 e - 04 (+)	3.131 e - 03(+)	1.017e-03(+)	5.778e-04(+)	5.380 e - 04(+)
MMF8	8.018 e - 04	1.311 e - 03 (+)	2.467 e - 03(+)	8.668 e - 04(+)	7.408 e - 04(-)	5.290 e - 04(-)
MMF9	8.235 e - 03	3.658 e - 03 (-)	6.178 e - 02(+)	3.444 e - 03(-)	2.489 e - 03(-)	9.047 e - 03(+)
MMF10	<b>1.901 e - 01</b>	1.909 e - 01(+)	2.263 e - 01(+)	1.917 e - 01(+)	1.914 e - 01(+)	1.943 e - 01(+)
MMF11	9.668 e - 02	<b>7.584 e - 02(-)</b>	1.574e-01(+)	9.036 e - 02(-)	8.717 e - 02(-)	9.768 e - 02(+)
MMF12	8.332 e - 02	<b>6.189 e - 02(-)</b>	8.460 e - 02(+)	8.398 e - 02(+)	8.352 e - 02(+)	8.379 e - 02(+)
MMF13	1.503 e - 01	<b>8.269 e - 02(-)</b>	2.420 e - 01(+)	1.458 e - 01(=)	1.593 e - 01(+)	1.520 e - 01(+)
MMF14	<b>5.517 e - 02</b>	5.867 e - 02 (+)	8.526 e - 02(+)	6.745 e - 02(+)	5.598 e - 02(+)	6.870 e - 2(+)
MMF14_a	5.594 e - 02	<b>5.385 e - 02 (=)</b>	7.541 e - 02(+)	7.882 e - 02(+)	5.410 e - 02(=)	6.936 e - 02(+)
MMF15	1.779 e - 01	<b>1.530 e - 01(-)</b>	2.058 e - 01(+)	1.801 e - 01(+)	1.844 e - 01(+)	1.941 e - 01(+)
MMF15_a	1.791 e - 01	<b>1.553 e - 01(-)</b>	1.954 e - 01(+)	1.907 e - 01(+)	1.840 e - 01(+)	1.928 e - 01(+)
SYM_PART simple	1.048 e - 02	9.703 e - 03 (-)	4.051 e - 02(+)	<b>3.074 e - 03(-)</b>	5.698 e - 03(-)	8.199 e - 03(-)
SYM_PART rotated	1.032 e - 02	1.179 e - 02 (+)	1.407 e - 02(+)	<b>3.837 e - 03(-)</b>	6.151 e - 03(-)	8.199 e - 03(-)
Omni-test	1.264 e - 02	1.796 e - 02 (+)	8.205 e - 03(-)	<b>2.881 e - 03(-)</b>	4.022 e - 03(-)	4.208 e - 03(-)
Sum up	+ / = / -	14 / 1 / 7	21 / 0 / 1	13 / 1 / 8	11 / 2 / 10	14 / 0 / 8

better, worse, and not much different from the other competitor, respectively.

**3.3. Comparison with Other Algorithms.** The experimental results about *PSP* are shown in Table 1. As indicated in Table 1, MOEA/D-HSA achieves the best *PSP* on 13 test

functions and defeats DN-NSGA-II and MOEA/D on all test functions. MO\_Ring\_PSO\_SCD gets the maximum *PSP* on six test functions out of 22. LORD acquires the best performance on two test functions. TriMOEA-TA&R gets the maximum *PSP* on one test function. Although MOEA/D-HSA is not the best on some test functions, the difference is appreciably small and acceptable. Therefore, the

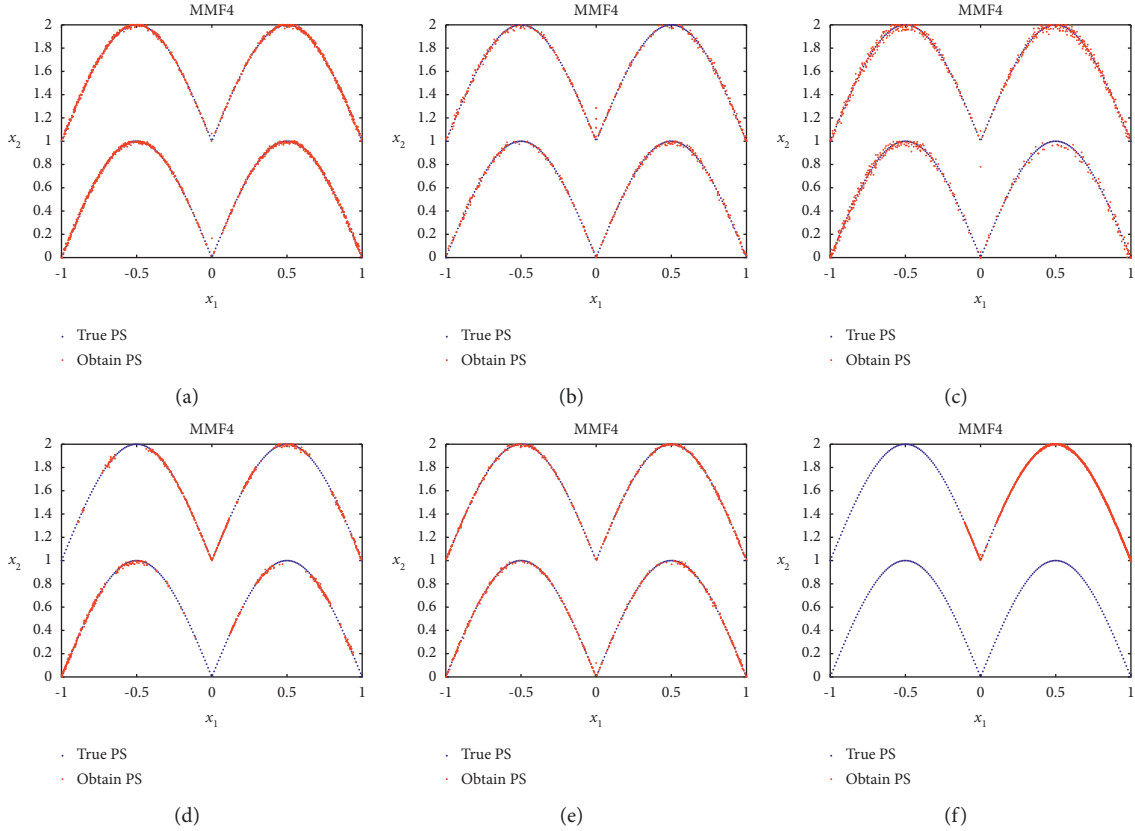


FIGURE 5: The population distributions of the true PSs and the final solutions obtained by different MMOEAs for MMF4 with four equivalent PSs, (a) MOEA/D-HSA, (b) MO\_Ring\_PSO\_SCD, (c) TriMOEA-TA&R, (d) DN-NSGA-II, (e) LORD, and (f) MOEA/D.

experimental results suggest that MOEA/D-HSA can find multiple PSs with well-distribution and well-convergence.

The experimental results in IGDF are listed in Table 2. In Table 2, MOEA/D-HSA gets the smallest IGDF on six test functions and defeats TriMOEA-TA&R on most test functions. Moreover, MO\_Ring\_PSO\_SCD is the best on six test functions out of 22. DN-NSGA-II acquires the best IGDF on four test functions. MOEA/D and LORD get the minimum IGDF on three test functions. In conclusion, it can be concluded that MOEA/D-HSA is not only outperforms the competitors on most test functions in IGDF but also can well approximate the true PF.

**3.4. Visual Comparison.** The final population distributions of all algorithms are shown in Figure 5 for MMF4 and Figure 6 for SYM\_PART rotated. As illustrated in Figure 5, each algorithm except MOEA/D is able to locate all PS regions, but TriMOEA-TA&R has poor convergence in the decision space, and DN-NSGA-II achieves the least number of Pareto optimal solutions on each PS region. In addition, compared with other algorithms, not only MOEA/D-HSA can locate more Pareto optimal solutions on each PS but also the obtained PSs have well-diversity and well-convergence.

Besides, for SYM\_PART rotated with nine equivalent PSs, as depicted in Figure 6 that all algorithms have good convergence in the decision space, but MOEA/D and DN-NSGA-II cannot locate all PSs. Furthermore,

MO\_Ring\_PSO\_SCD and MOEA/D-HSA have similar properties on SYM\_PART rotated. And, compared with the remaining competitors, MOEA/D-HSA not only can search more Pareto optimal solutions on each PS but also show good convergence. In summary, based on visual comparison, MOEA/D-HSA can find multiple PSs with good distribution and convergence.

To further verify the effectiveness of each component, two MOEA/D-HSA variants are tested. SS-variant-1 indicates MOEA/D-HSA without the elite learning strategy. SS-variant-2 represents MOEA/D-HSA deletes an individual with the largest Tchebycheff value in the subpopulation instead of using the environmental selection operation. Table 3 records the average rankings of PSP obtained by MOEA/D-HSA and MOEA/D-HSA variants on all test functions. As depicted in Table 3, MOEA/D-HSA owns the best ranking. Therefore, it can be concluded that each component of the proposed algorithm can collaborate with the other components to enhance its overall performance.

**3.5. Parameter Analysis.** All parameters except  $t$  are adaptively adjusted, so only the subpopulation size  $t$  is discussed in this section. MOEA/D-HSA with  $t = \{2, 3, 4, 5, 6\}$  is tested on all problems, and Table 4 records their rankings of the Friedman test. In Table 4, it can be seen that the property of MOEA/D-HSA is the best when  $t$  is set to 5. Therefore, the subpopulation size  $t$  is set to 5 in the proposed algorithm.



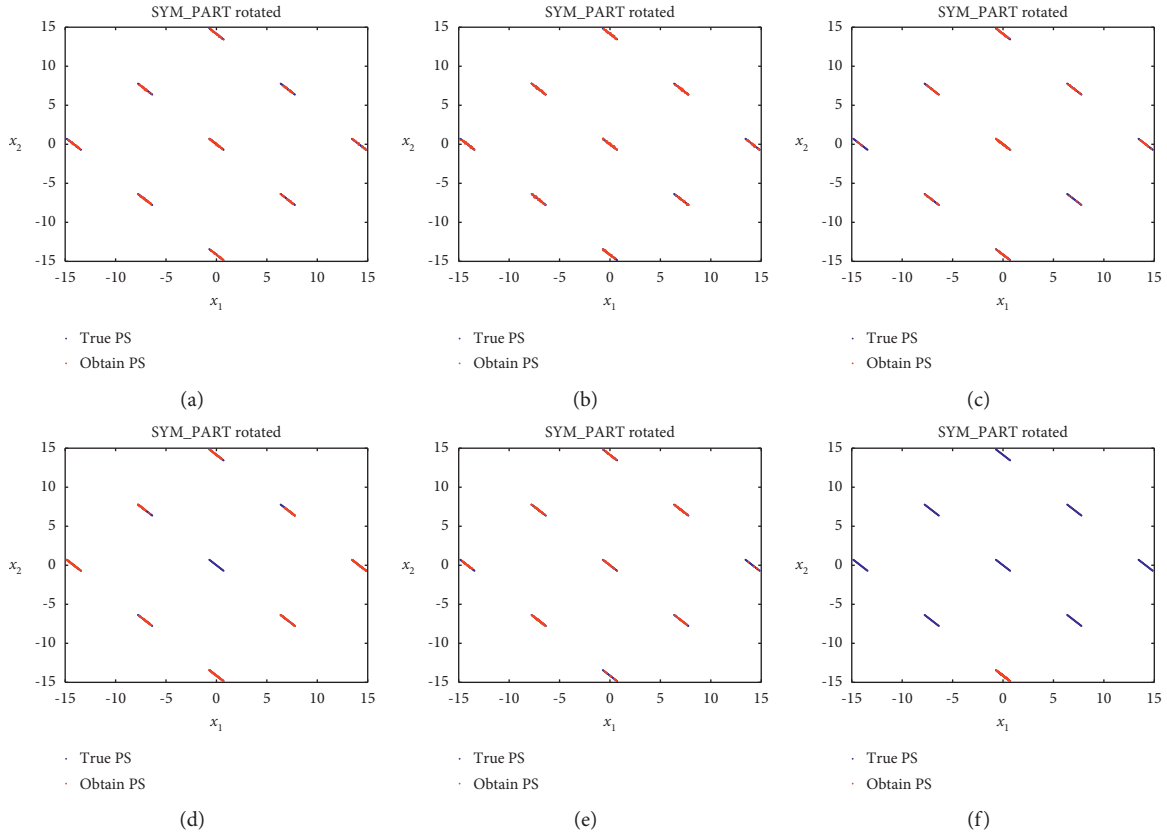


FIGURE 6: The population distributions of the true PSs and the final solutions obtained by different MMOEAs for SYM\_PART rotated with nine equivalent PSs, (a) MOEA/D-HSA, (b) MO\_Ring\_PSO\_SCD, (c) TriMOEA-TA&R, (d) DN-NSGA-II, (e) LORD, and (f) MOEA/D. Effectiveness analysis of MOEA/D-HSA.

TABLE 3: Average rankings of MOEA/D-HSA and MOEA/D-HSA variants by the Friedman test in PSP.

Algorithm	Rank
MOEA/D-SS	<b>1.5909</b>
SS-variant-1	2.4545
SS-variant-2	1.9545

TABLE 4: Average rankings of MOEA/D-HSA with different parameter settings by the Friedman test in PSP.

Algorithm	Rank
$t = 2$	3.5
$t = 3$	3.1818
$t = 4$	3.1818
$t = 5$	<b>2.1364</b>
$t = 6$	3

TABLE 5: Normalized wall-clock time of six algorithms.

	MMOEA/D-HSA	MO_Ring_PSO_SCD	TriMOEA-TA&R	DN-NSGA-II	MOEA/D	LORD
MMF1	0.0056	0.0169(+)	<b>0.0045(-)</b>	0.0220(+)	0.0049(-)	1.0000(+)
MMF1_e	0.0111	0.0317(+)	<b>0.0046(-)</b>	0.0213(+)	0.0057(-)	1.0000(+)
MMF1_z	0.0051	0.0133(+)	<b>0.0036(-)</b>	0.0107(+)	0.0037(-)	1.0000(+)
MMF2	0.0071	0.0339(+)	<b>0.0039(-)</b>	0.0437(+)	0.0059(-)	1.0000(+)
MMF3	0.0071	0.0327(+)	<b>0.0039(-)</b>	0.0383(+)	0.0061(-)	1.0000(+)
MMF4	0.0040	0.0124(+)	<b>0.0027(-)</b>	0.0072(+)	0.0037(-)	1.0000(+)

TABLE 5: Continued.

	MMOEA/D-HSA	MO_Ring_PSO_SCD	TriMOEA-TA&R	DN-NSGA-II	MOEA/D	LORD
MMF5	0.0059	0.0181(+)	<b>0.0039</b> (-)	0.0219(+)	0.0049(-)	1.0000(+)
MMF6	0.0053	0.0168(+)	<b>0.0036</b> (-)	0.0204(+)	0.0047(-)	1.0000(+)
MMF7	0.0038	0.0115(+)	<b>0.0034</b> (-)	0.0093(+)	0.0036(=)	1.0000(+)
MMF8	0.0042	0.0160(+)	<b>0.0025</b> (-)	0.0088(+)	0.0041(=)	1.0000(+)
MMF9	0.0035	0.0102(+)	<b>0.0031</b> (-)	0.0043(+)	0.0028(-)	1.0000(+)
MMF10	0.0044	0.0107(+)	0.0029(-)	0.0048(+)	0.0028(-)	1.0000(+)
MMF11	0.0028	0.0088(+)	0.0028(=)	0.0034(+)	0.0026(=)	1.0000(+)
MMF12	0.0042	0.0171(+)	<b>0.0034</b> (-)	0.0080(+)	0.0037(-)	1.0000(+)
MMF13	0.0049	0.0126(+)	<b>0.0036</b> (-)	0.0050(+)	0.0040(-)	1.0000(+)
MMF14	0.0655	0.1534(+)	0.0500(-)	0.0603(-)	0.0259(-)	1.0000(+)
MMF14_a	0.0803	0.1813(+)	0.0453(-)	0.0762(-)	0.0309(-)	1.0000(+)
MMF15	0.0385	0.1573(+)	0.0535(+)	0.0612(+)	0.0262(-)	1.0000(+)
MMF15_a	0.0471	0.1925(+)	0.0618(+)	0.0791(+)	0.0342(-)	1.0000(+)
SYM_PART simple	0.0079	0.0422(+)	<b>0.0044</b> (-)	0.0172(+)	0.0061(-)	1.0000(+)
SYM_PART rotated	0.0103	0.0502(+)	<b>0.0037</b> (-)	0.0283(+)	0.0066(-)	1.0000(+)
Omni-test	0.0060	0.0240(+)	<b>0.0015</b> (-)	0.0102(+)	0.0042(-)	1.0000(+)
Sum up	+/-/-	22/0/0	2/1/19	20/0/2	0/3/19	22/0/0

3.6. *Analysis of Running Time.* With the aim of comparing running time, all algorithms run 25 times on each test function, and the normalized wall-clock time of all algorithms is given in Table 5. As revealed in Table 5, TriMOEA-TA&R and MOEA/D have the shortest running time on 16 and 6 test functions, respectively. And, LORD takes the longest running time on all test functions. In addition, compared with the remaining algorithms, MMOEA/D-HSA achieves the shortest running time on most test functions. It can be concluded that the running time of MOEA/D-SS is not the shortest, but the difference is acceptable.

## 4. Conclusions

A simple but effective method MOEA/D-HSA is developed to tackle MMOPs. In MOEA/D-HSA, first, multiple individuals are assigned to the same weight vector to form a subpopulation to find equivalent PSs. Then, an environmental selection based on greedy selection is used to dynamically adjust the subpopulation size. Finally, a modified harmony search algorithm and elite learning strategy are utilized for balancing the diversity and convergence of the population. The experimental results prove that MOEA/D-HSA is superior to selected popular algorithms with respect to the IGDF and PSP values.

Besides, in this article, the dimension of the test problems is low. It is a meaningful study to develop MOEA/D-HSA on more complicated problems or the real applications [48, 49], such as the neural architecture search problems.

## Data Availability

Data and code can be made available upon request to 915847903@qq.com.

## Conflicts of Interest

The authors declare that there are no conflicts of interest.

## Acknowledgments

This work was supported in part by the National Nature Science Foundation of China under Grant 61772391 and 62106186, in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2022JQ-670, and in part by the Fundamental Research Funds for the Central Universities under Grant QTZX22047.

## References

- [1] F. Kudo, T. Yoshikawa, and T. Furuhashi, "A Study on Analysis of Design Variables in Pareto Solutions for Conceptual Design Optimization Problem of Hybrid Rocket Engine," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 2558–2562, New Orleans, LA, USA, June 2011.
- [2] J. Michalek, R. Choudhary, and P. Papalambros, "Architectural layout design optimization," *Engineering Optimization*, vol. 34, no. 5, pp. 461–484, 2002.
- [3] X. He, C. Fei, Y. Liu, K. Yang, and X. Ji, "Multi-objective Longitudinal Decision-Making for Autonomous Electric Vehicle: A Entropy-Constrained Reinforcement Learning Approach," in *Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–6, Rhodes, Greece, September 2020.
- [4] G. Li, Q. Zhang, and Z. Wang, "Evolutionary Competitive Multitasking Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 26, 2022.
- [5] X. Zheng, T. Gong, X. Li, and X. Lu, "Generalized scene classification from small-scale datasets with multitask learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2021.
- [6] G. Li and Q. Zhang, "Multiple penalties and multiple local surrogates for expensive constrained optimization," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 4, pp. 769–778, 2021.
- [7] G. Li, Q. Zhang, Q. Lin, and W. Gao, "A three-level radial basis function method for expensive optimization," *IEEE Transactions on Cybernetics*, vol. 99, pp. 1–12, 2021.

- [8] L. Jinjian, A. Doniec, J. Boonaert et al., "A New Integrated Multimodal Multi-Criteria Route Planning Method in Bus Network Considering Passengers's Walking Behavior," in *Proceedings of the 2018 3rd IEEE International Conference On Intelligent Transportation Engineering (ICITE)*, pp. 13–17, Singapore, September 2018.
- [9] Y. Han, D. Gong, Y. Jin, and Q. Pan, "Evolutionary multi-objective blocking lot-streaming flow shop scheduling with machine breakdowns," *IEEE Transactions on Cybernetics*, vol. 49, no. 1, pp. 184–197, 2019.
- [10] O. Schütze, M. Vasile, and C. A. C. Coello, "Computing the set of epsilon-efficient solutions in multiobjective space mission design," *Journal of Aerospace Computing, Information, and Communication*, vol. 8, no. 3, pp. 53–70, 2011.
- [11] X. Shan, Y. Li, X. Chen, Y. Wu, Y. He, and W. Li, "Investigations on internal ballistic characteristics of pasty propellant rocket engine," *International Journal of Aerospace Engineering*, vol. 2021, no. 8, pp. 1–14, Article ID 9952209, 2021.
- [12] A. Jaszkiwicz, "On the performance of multiple-objective genetic local search on the 0/1 knapsack problem - a comparative experiment," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 4, pp. 402–412, 2002.
- [13] J. Liang, C. Yue, and B. Qu, "Multimodal Multi-Objective Optimization: A Preliminary Study," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 2454–2461, Vancouver, BC, Canada, July 2016.
- [14] O. M. Shir, M. Preuss, B. Naujoks, and M. Emmerich, "Enhancing Decision Space Diversity in Evolutionary Multi-objective algorithms," in *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization*, pp. 95–109, Nantes, France, April 2009.
- [15] K. Deb and S. Tiwari, "Omni-optimizer: a generic evolutionary algorithm for single and multi-objective optimization," *European Journal of Operational Research*, vol. 185, no. 3, pp. 1062–1087, 2008.
- [16] C. Yue, B. Qu, and J. Liang, "A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 5, pp. 805–817, 2018.
- [17] Y. Liu, H. Ishibuchi, G. G. Yen, Y. Nojima, and N. Masuyam, "Handling imbalance between convergence and diversity in the decision space in evolutionary multimodal multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 3, pp. 551–565, 2020.
- [18] Y. Liu, G. G. Yen, and D. Gong, "A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 4, pp. 660–674, 2019.
- [19] Q. Fan and X. Yan, "Solving multimodal multiobjective problems through zoning search," *IEEE Transactions on Systems*, vol. 51, 2019.
- [20] Q. Fan and O. K. Ersoy, "Zoning search with adaptive resource allocating method for balanced and imbalanced multimodal multi-objective optimization," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 6, pp. 1163–1176, 2021.
- [21] Q. Lin, W. Lin, Z. Zhu, M. Gong, J. Li, and C. A. C. Coello, "Multimodal Multi-Objective Evolutionary Optimization with Dual Clustering in Decision and Objective Spaces," *IEEE Transactions on Evolutionary Computation*, vol. 25, 2020.
- [22] K. Zhang, C. Shen, J. He, and G. G. Yen, "Knee based multimodal multi-objective evolutionary algorithm for decision making," *Information Sciences*, vol. 544, pp. 39–55, 2021.
- [23] K. Zhang, M. S. Chen, X. Xu, and G. G. Yen, "Multi-objective evolution strategy for multimodal multi-objective optimization," *Applied Soft Computing*, vol. 101, 2021.
- [24] K. Zhang, C. Shen, W. Hu, G. G. Yen, Z. Xu, and J. He, "Two-stage double niched evolution strategy for multimodal multi-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 99, p. 1, 2021.
- [25] L. Ma, S. Cheng, and Y. Shi, "Enhancing learning efficiency of brain storm optimization via orthogonal learning design," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6723–6724, 2020.
- [26] X. Zheng, X. Chen, X. Lu, and B. Sun, "Unsupervised change detection by cross-resolution difference learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2021.
- [27] Z. Li, L. Shi, C. Yue, Z. Shang, and B. Qu, "Differential evolution based on reinforcement learning with fitness ranking for solving multimodal multiobjective problems," *Swarm and Evolutionary Computation*, vol. 49, pp. 24–244, 2019.
- [28] Q. Zhang and H. Li, "MOEA/D: a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2008.
- [29] C. Hu and H. Ishibuchi, "Incorporation of a Decision Space Diversity Maintenance Mechanism into MOEA/D for Multimodal Multi-Objective Optimization," in *Proceedings of the Genetic And Evolutionary Computation Conference Companion*, pp. 1898–1901, Kyoto, Japan, July 2018.
- [30] Y. Peng and H. Ishibuchi, "A Decomposition-Based Large-Scale Multimodal Multi-Objective Optimization Algorithm," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1–8, Glasgow, UK, July 2020.
- [31] R. Tanabe and H. Ishibuchi, "A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization," in *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pp. 249–261, Coimbra, Portugal, September 2018.
- [32] R. Tanabe and H. Ishibuchi, "A framework to handle multimodal multiobjective optimization in decomposition-based evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 4, pp. 720–734, 2020.
- [33] M. Pal and S. Bandyopadhyay, "Decomposition in decision and objective space for multi-modal multi-objective optimization," *Swarm and Evolutionary Computation*, vol. 62, no. 1, 2021.
- [34] W. H. Li, T. Zhang, R. Wang, and H. Ishibuchi, "Weighted Indicator Based Evolutionary Algorithm for Multimodal Multi-Objective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 6, 2021.
- [35] R. Tanabe and H. Ishibuchi, "A niching indicator-based multimodal many-objective optimizer," *Swarm and Evolutionary Computation*, vol. 49, pp. 134–146, 2019.
- [36] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: nsga-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [37] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pp. 832–842, Birmingham, UK, September 2004.
- [38] L. Li, G. G. Yen, A. Sahoo, L. Chang, and T. Gu, "On the estimation of pareto front and dimensional similarity in many-objective evolutionary algorithm," vol. 563, pp. 371–400, Information Sciences, 2021.

- [39] B. Qu, G. Li, Q. Guo, L. Yan, X. Z. Chai, and Z. Q. Guo, "A Niching Multi-Objective harmony Search Algorithm for Multimodal Multi-Objective Problems," in *Proceedings of the 2019 IEEE Congress on Evolutionary Computation*, Wellington, New Zealand, June 2019.
- [40] X. Li, K. Qin, B. Zeng, L. Gao, and L. Wang, "A dynamic parameter controlled harmony search algorithm for assembly sequence planning," *International Journal of Advanced Manufacturing Technology*, vol. 9, no. 2, pp. 3399–3411, 2017.
- [41] B. Y. Qu, P. N. Suganthan, and S. Das, "A distance-based locally informed particle swarm model for multimodal optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 3, pp. 387–402, 2013.
- [42] J. Liang, B. Qu, D. W. Gong, and C. T. Yue, "Problem Definitions and Evaluation Criteria for the CEC 2019 Special Session on Multimodal Multiobjective Optimization," *Computational Intelligence Laboratory*, Zhengzhou University, 2019.
- [43] A. Aimin Zhou, Q. Yaochu Jin, and Y. Jin, "Approximating the set of Pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1167–1189, 2009.
- [44] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, "Performance assessment of multiobjective optimizers: an analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, 2003.
- [45] J. Liang, Q. Guo, C. Yue, B. Qu, and K. Yu, "A Self-Organizing Multiobjective Particle Swarm Optimization Algorithm for Multimodal Multiobjective Problems," in *Proceedings of the International Conference on Swarm Intelligence*, Shanghai, China, June 2018.
- [46] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometric Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.
- [47] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Publications of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1939.
- [48] Z. Zhang, S. Liu, Y. Yang, and Y. Bai, "A link-adding strategy for improving robustness and traffic capacity in large-scale wireless sensor networks," *Cluster Computing*, vol. 22, no. 3, pp. 7687–7694, 2019.
- [49] X. Zheng, B. Wang, X. Du, and X. Lu, "Mutual attention inception network for remote sensing visual question answering," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2021.