![mathematics logo] *mathematics*

MDPI

Article

Handling Irregular Many-Objective Optimization Problems via Performing Local Searches on External Archives

Lining Xing [1] ![ORCID], **Rui Wu** [2], **Jiaxing Chen** [2] and **Jun Li** [3,*]

1   School of Electronic Engineering, Xidian University, Xi'an 710071, China
2   Inner Mongolia Institute of Dynamical Machinery, Hohhot 010010, China
3   School of Management, Hunan Institute of Engineering, Xiangtan 411104, China
*   Correspondence: 08036@hnie.edu.cn

Abstract: Adaptive weight-vector adjustment has been explored to compensate for the weakness of the evolutionary many-objective algorithms based on decomposition in solving problems with irregular Pareto-optimal fronts. One essential issue is that the distribution of previously visited solutions likely mismatches the irregular Pareto-optimal front, and the weight vectors are misled towards inappropriate regions. The fact above motivated us to design a novel many-objective evolutionary algorithm by performing local searches on an external archive, namely, LSEA. Specifically, the LSEA contains a new selection mechanism without weight vectors to alleviate the adverse effects of inappropriate weight vectors, progressively improving both the convergence and diversity of the archive. The solutions in the archive also feed back the weight-vector adjustment. Moreover, the LSEA selects a solution with good diversity but relatively poor convergence from the archive and then perturbs the decision variables of the selected solution one by one to search for solutions with better diversity and convergence. At last, the LSEA is compared with five baseline algorithms in the context of 36 widely-used benchmarks with irregular Pareto-optimal fronts. The comparison results demonstrate the competitive performance of the LSEA, as it outperforms the five baselines on 22 benchmarks with respect to metric hypervolume.

Keywords: evolutionary computation; many-objective optimization; irregular Pareto-optimal fronts; local search; external archive

MSC: 97M40; 97P30

1. Introduction

Optimization problems deriving from many fields often involve multiple conflicting objective functions. For example, scheduling workflows for cloud platforms needs to simultaneously minimize economical cost and completion time [1–3]. In general, reducing economical cost means renting cloud resources with lower configuration, thereby prolonging task execution time and workflow completion time. Thus, the two optimization objectives of scheduling cloud workflows conflict. In addition, accuracy and robustness are two conflicting objectives for neural-network-architecture searching [4]. These problems are termed multi-objective optimization problems (MOPs), which are mathematically formulated as:

$$\begin{cases} \text{Min} & \vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \cdots, f_m(\vec{x})], \\ \text{S.t.} & \vec{x} \in \Omega, \end{cases} \quad (1)$$

where $\vec{x} = (x_1, x_2, \cdots, x_n)$ is the decision vector of n variables, and $\vec{f}(\vec{x})$ represents the objective vector of m objectives $f_j(\vec{x})$, $j \in \{1, 2, \cdots, m\}$. $\Omega \subseteq \mathbb{R}^n$ is the feasible search space. MOPs with more than three objectives, i.e., $m > 3$, are commonly termed many-objective optimization problems (MaOPs). Due to the conflicts between optimization objectives, no single solution can optimize multiple objectives simultaneously. Instead, there is a set of

compromise solutions which is called the Pareto-optimal solution set in the decision space and the Pareto-optimal front in the objective space.

With the attractive feature of creating a set of solutions in each iteration, evolutionary algorithms have been intensively explored to resolve MaOPs. Over the past two decades, researchers and engineers have proposed and developed numerous evolutionary many-objective optimization algorithms (EMOAs) to resolve MaOPs from various domains [5–8]. According to the environmental selection framework, most of the established EMOAs can be roughly classified into three categories: Pareto-dominance-based, indicator-based, and decomposition-based EMOAs. Regarding the algorithms based on Pareto dominance [9–11], they partition the combined population into different non-dominated levels (L_1, L_2, and so on). Then, solutions in each level are selected one at a time, starting from L_1, until the number of selected solutions equals the population size or the time exceeds the pre-defined limit. Next, a secondary metric, e.g., crowding distance, is used to sort solutions in the last accepted level. The indicator-based EMOAs [8] attempt to quantify the population using one indicator, such as Hypervolume [12] or inverted generational distance [13]. For the EMOAs based on decomposition (EMOA/Ds) [14,15], they first employ Das and Dennis's systematic approach [16] to initialize a set of uniformly-distributed weight vectors within a simplex, and then leverage these weight vectors to convert one single MaOP into a series of subproblems, which are solved in a cooperative way.

Among the three categories of EMOAs, the decomposition-based ones are commonly recognized as an effective way to solve MaOPs by substantially relieving selection-pressure-loss issues in Pareto-dominance-based EMOAs and the high computing overhead of calculating indicator values in indicator-based EMOAs [15]. The EMOA/Ds perform well on MaOPs with regular Pareto-optimal fronts, e.g., simplex-like shapes. However, they perform poorly on MaOPs with irregular Pareto-optimal fronts, such as discontinuous, degenerate, inverted, badly-scaled, or strongly concave/convex shapes [17–19]. In this case, some weight vectors cannot intersect with the Pareto-optimal fronts and become invalid, resulting in a waste of search resources allocated to these subproblems defined by the invalid weight vectors. Then, the capacity of EMOAs to solve MaOPs with irregular Pareto-optimal fronts deteriorated. For example, the RVEA and its variants [20,21], each solution is associated with a weight vector with the smallest acute angle, and at most one solution is reserved for a weight vector. For confronting the irregular Pareto-optimal fronts, some weight vectors of these algorithms are associated with multiple solutions, while many weight vectors are not associated with solutions [22,23]. This causes the weight vectors without associated solutions to become invalid, and the number of output solutions obtained by these algorithms is far less than the population size.

Researchers and engineers have recently tailored various algorithms to make a sound trade-off between convergence and diversity for EMOA/Ds in solving MaOPs with irregular Pareto-optimal fronts [17,24]. Most algorithms in this direction can be classified into the following three categories. In the first category, they adjust the distribution of weight vectors by using the solutions in the current population or archive. For instance, Li et al. explored the distribution of solutions in the archive for weight generation, weight addition, and weight deletion [25]. Liu et al. employed the objective vectors of solutions in the current population to adjust the weight vectors by adding angle thresholds [26]. Algorithms falling into the second category intend to learn the distribution of weight vectors by performing machine learning on visited solutions [27,28]. For instance, self-organizing mapping techniques and growing neural gas networks are employed to assist weight-vector adjustment [29–31]. For the above two categories of algorithms, the weight-vector adjustment depends on the distribution of visited solutions, which is often inconsistent with the irregular Pareto-optimal fronts. In this scenario, weight vectors will be misled into inappropriate areas, wasting computation resources. In the third category, the algorithm replaces inactive weight vectors by generating new ones. For instance, Jain et al. added new weight vectors around the active ones when the active number was less than the population size [32]. Cheng et al. suggested randomly generating additional weight vectors to compensate for

invalid ones [20]. Elarbi et al. suggested adding multiple normal-boundary intersection directions to approximate irregular Pareto-optimal fronts [33]. However, facing the objective space of exponential growth, the weight vectors added by the third category of algorithms are likely ineffective.

In sum, the algorithms based on visited solutions mislead weight vectors to unsuitable areas, and algorithms that do not refer to visited solutions likely add ineffective weight vectors. Thus, adjusting the weight vectors for unknown shapes of irregular Pareto-optimal fronts is a dilemma. What is worse, when solving MaOPs with strongly convex or concave Pareto-optimal fronts, is the solutions do not inevitably converge to the intersection of the weight vectors and the Pareto-optimal fronts. This fact further challenges the EMOA/Ds to balance diversity and convergence.

To cope with the above-mentioned issues of the weight vector adjustment-based algorithms, this paper suggests adding an external archive, and presents a selection mechanism without weight vectors to maintain this archive. The core idea of this selection mechanism is to associate each new solution with a previously preserved solution with the minimum distance, and preserves the new solution only when it has better convergence and diversity than the associated solution. Moreover, it performs local searches on solutions with better diversity and relatively poor convergence to further strengthen convergence and diversity. Of course, the solutions in the archive also feed back the weight-vector adjustment.

This paper is organized as follows: Section 2 designs the proposed LSEA, followed by the experimental verification in Section 3. Section 4 provides the conclusion and two promising research directions.

2. Algorithm Design

This section designs a local search-assisted external archive mechanism to improve the EMOA/Ds using weight-vector adjustment to solve many-objective optimization problems with irregular Pareto-optimal fronts.

2.1. Main Framework of LSEA

The many-objective optimization problems with irregular Pareto-optimal fronts still pose challenges for the multi-objective evolutionary community. Although the weight-vector-adaptive adjustment can improve EMOA/Ds' performance to a certain extent, the adjustment of the weight vectors heavily depends on the visited solutions, unexpectedly misleading the weight vectors to unsuitable search areas. Considering this, we propose a novel many-objective evolutionary algorithm by performing local searches on an external archive. This algorithm embraces an external archive to maintain a well-converged and well-distributed population, and further strengthens convergence and diversity by conducting local searches on solutions with better diversity and relatively poor convergence. Its main framework is summarized as Algorithm 1.

As shown in Algorithm 1, the pivotal inputs of the LSEA are an optimization problem, the population size, and the termination condition. With the above inputs, the LSEA begins with initializing a set of weight vectors (Line 1) and the neighborhood of each weight vector (Line 2). B_i records the indexes of the T closest weight vectors of the i-th weight vector, where T refers to the number of the weight vectors in the neighborhood. If $b \in B_i$, the b-th subproblem is termed as a neighbor of the i-th subproblem. Then, it randomly generates N solutions to construct the initial population P and the external archive EA (Lines 3–4). Additionally, the number of used evaluation times ETc and the ideal point $\vec{z}^*$ are updated (Lines 5–6).

The LSEA follows the mainstream framework of the EMOA/Ds using weight-vector adjustment. Its main loop includes offspring population generation, weight-vector adjust-

ment, and maintenance of the external archive.

Algorithm 1: Main framework of LSEA.

Input: A problem; population size N; maximum evaluation times ET_m;
Output: The external archive EA;

1   $W \leftarrow$ Generate a set of weight vectors uniformly;
2   $B \leftarrow$ Initialize the neighborhood for each weight vector;
3   $P \leftarrow$ Generate a population randomly;
4   $EA \leftarrow P$;
5   $ET_c \leftarrow N$;
6   $\vec{z}^* \leftarrow \min_{\vec{p} \in P} f_j(\vec{p}), j \in \{1, 2, \cdots, m\}$;
7   **while** $ET_c < ET_m$ **do**
8     $Q \leftarrow \varnothing$;
9   **for** $i = 1 \to N$ **do**
10       $\vec{p} \leftarrow$ Offspring-creation(P, B_i);
11       $ET_c \leftarrow ET_c + 1$;
12       $\vec{z}^* \leftarrow \min\{\vec{z}_j^*, f_j(\vec{p})\}, j \in \{1, 2, \cdots, m\}$;
13   **for** $b \in B_i$ **do**
14   **if** $f^{tch}(P_b|W_b, \vec{z}^*) > f^{tch}(\vec{p}|W_b, \vec{z}^*)$ **then**
15           $P_b \leftarrow \vec{p}$;
16       $Q \leftarrow Q \cup \{\vec{p}\}$;
17     $W \leftarrow$ AdjustWeightVector($P \cup EA$, W, N);
18     $EA \leftarrow$ MaintainExternalArchive(EA, Q, N);

Before the offspring population generation, a set Q is initialized to record the offspring solutions (Line 8). Then, the subproblems are iterated one by one to generate new solutions (Line 9). In this process, the solutions belonging to each subproblem and its neighborhood are randomly selected, and an existing variation operator, such as differential evolution [34,35], cuckoo search [36,37], or krill herd optimization [38], is used on these selected solutions. Then, the number of used evaluation times ETc and the ideal point are updated (Lines 11–12). After that, the new solution $\vec{p}$ will be used to update the solutions belonging to the i-th subproblem and its neighborhood (Lines 13–15). If the fitness of $\vec{p}$ is better than that of compared solutions (Line 14), it will replace them. Note that P_b and W_b, respectively, represent the solution and weight vector corresponding to the b-th subproblem. The EMOA/D and its variants generally calculate the fitness of a solution using the following three approaches [15]: penalty-based boundary intersection, weighted sum, and Tchebycheff. Among them, the Tchebycheff-based approach is more prevalent in solving many-objective optimization problems [39,40]. The Tchebycheff value of solution $\vec{x}$ to weight vector W_i can be calculated as:

$$f^{tch}(\vec{x}|W_i, \vec{z}^*) = \max_{j \in \{1, 2, \cdots, m\}} \{w_j \cdot |f_j(\vec{x}) - \vec{z}_j^*|\}, \tag{2}$$

where $\vec{z}^* = (z_1^*, z_2^*, \cdots, z_m^*)$ is the ideal point and $W_i = \{w_1, w_2, \cdots, w_m\}$ indicates the i-th weight vector.

For the weight-vector adjustment approach (Line 17), designing a new one is not the focus of this paper. We directly use the growing neural gas network-based weight-vector adjustment approach [30]. In this approach, the growing neural gas network is employed to dynamically learn the topological structures of the irregular Pareto-optimal fronts. Then, the nodes of the growing neural gas network are adopted to adjust the weight vectors.

For maintaining the external archive, all the new solutions in this iteration are employed to update the external archive (Line 18), which is summarized in Algorithm 2. After the LSEA meets the termination condition, it outputs the external archive EA.

2.2. Maintain External Archive

Adjusting weight vectors based on the visited solutions is a promising way to compensate for EMOA/Ds' deficiency in solving MaOPs with irregular Pareto-optimal fronts. Unfortunately, the distribution of obtained solutions often cannot well approximate the irregular Pareto-optimal fronts, especially in the early stage of the evolutionary search. In this scenario, the weight vectors are always misled to the unsuitable search area, which will seriously inhibit the potential of the EMOA/Ds to balance diversity and convergence. To address the aforementioned issue, this paper designs a new selection mechanism without weight vectors to maintain a well-converged and well-diversified external archive. Moreover, the proposed algorithm performs local searches on solutions with good diversity but relatively poor convergence to further strengthen convergence and diversity. The main steps of Function MaintainExternalArchive() is summarized in Algorithm 2. Before detailing this function, we define the convergence and diversity for solutions as follows.

The L_p-metric is employed to evaluate the convergence of solutions, which is calculated as follows.

$$C(\vec{x}) = \left(\sum_{j=1}^{m} |f_j(\vec{x}) - \vec{z}_j^*|\right)^{\frac{1}{p}}. \tag{3}$$

Similarly to the literature [41–43], the distance of two solutions $\vec{x}_1$ and $\vec{x}_2$ is defined as:

$$Dis(\vec{x}_1, \vec{x}_2) = \sum_{j=1}^{m} \frac{\max\{f_j(\vec{x}_1), f_j(\vec{x}_2)\} - \min\{f_j(\vec{x}_1), f_j(\vec{x}_2)\}}{1 + \max\{f_j(\vec{x}_1), f_j(\vec{x}_2)\}}. \tag{4}$$

Relative to population P, the diversity of a solution $\vec{x}$ is defined as:

$$D(\vec{x}|P) = \min_{\vec{p} \in P} Dis(\vec{x}, \vec{p}). \tag{5}$$

As illustrated in Algorithm 2, the primary inputs of Function MaintainExternal Archive() are: the archive, a new population, the population size, and the ideal point. This function can be roughly divided into two stages: environmental selection and local search.

The first stage strives to maintain a well-converged and well-diversified archive as follows. The dominated solutions in the combined population $EA \bigcup Q$ are first removed (Line 1) to ensure that only non-dominated solutions are retained in the archive. Then, the non-dominated solutions in the archive and the offspring population are selected (Lines 2–3). After that, if the number of solutions in the archive is less than the population size (Line 4), then the solution with the minimum ratio of convergence to diversity (Line 8) is moved from Q to EA one by one until the population size is reached or the set Q become empty. Next, each remaining solution in the Q will be associated with a solution with the minimum distance (Line 15), which is defined in (4). These two solutions will compete for survival. If the remaining solution in the Q is better than the associated solution with respect to both convergence and diversity (Line 16), it will replace the associated solution (Lines 17–18).

The second stage (Lines 19–25) attempts to perform local searches on solutions with good diversity but relatively poor convergence, avoiding the elimination of solutions with better diversity due to insufficient convergence. During this stage, a solution is first selected from the archive by applying roulette-wheel selection based on the sum of diversity and convergence values, which are, respectively, defined in (5) and (3) (Line 19). Note that the symbol n in line 21 denotes the number of decision variables. Afterward, all decision variables of the selected solution are perturbed one by one to reproduce new solutions (Line 22). If a new solution is better than the previous solution with respect to both convergence

and diversity (Line 23), the new solution will be retained (Line 24).

Algorithm 2: Function **MaintainExternalArchive**().

Input: Current archive EA; offspring population Q; population size N; ideal point $\vec{z}^*$;
Output: A new archive EA;

1 $Q' \leftarrow$ Select non-dominated solutions from $EA \bigcup Q$;
2 $EA \leftarrow EA \bigcap Q'$;
3 $Q \leftarrow Q \bigcap Q'$;
4 **while** $|EA| < N$ & $|Q| > 0$ **do**
5      $\vec{q}^* \leftarrow \varnothing$;
6      $tFit \leftarrow \infty$;
7   **for** $\vec{q} \in Q$ **do**
8         $Fit \leftarrow C(\vec{q})/D(\vec{q}|EA)$;
9   **if** $Fit < tFit$ **then**
10           $\vec{q}^* \leftarrow \vec{q}$;
11           $tFit \leftarrow Fit$;
12     $EA \leftarrow EA \bigcup \{\vec{q}^*\}$;
13     $Q \leftarrow Q \setminus \{\vec{q}^*\}$;
14 **for** $\vec{q} \in Q$ **do**
15     $\vec{p}^* \leftarrow \arg\min_{\vec{p} \in EA} Dis(\vec{p}, \vec{q})$;
16   **if** $D(\vec{q}|EA) >= D(\vec{p}^*|EA)$ & $C(\vec{q}) <= C(\vec{p}^*)$ **then**
17        $EA \leftarrow EA \bigcup \{\vec{q}\}$;
18        $EA \leftarrow EA \setminus \{\vec{p}^*\}$;
19 $\vec{p} \leftarrow$ Select a solution from EA;
20 $EA \leftarrow EA \setminus \{\vec{p}\}$;
21 **for** $i = 1 \rightarrow n$ **do**
22     $\vec{q} \leftarrow$ Generate a solution by disturbing the i-th variable of $\vec{p}$;
23   **if** $D(\vec{q}|EA) >= D(\vec{p}|EA)$ & $C(\vec{q}) <= C(\vec{p})$ **then**
24        $\vec{p} \leftarrow \vec{q}$;
25 $EA \leftarrow EA \bigcup \{\vec{p}\}$;

Figure 1 provides an illustrative example to explain the local searches for a solution with good diversity but relatively poor convergence. In Figure 1, each solid blue point corresponds to a solution. We can observe that the solution within the red circle poses good diversity; that is, it is in a sparse region. However, its convergence is relatively poor, and it is likely to be discarded when it is dominated by the offsprings of better-converged solutions. Based on the policy of the local search, it will select this solution and perturbs all decision variables one by one to reproduce new solutions to explore the sparse region. This way, both the diversity and convergence of the population are strengthened.
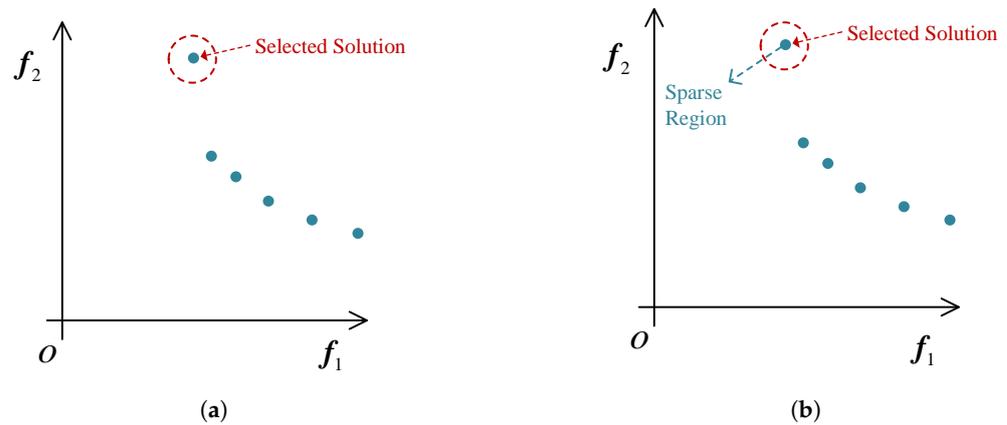
Figure 1. An illustrative example of the local search. (**a**) select a solution with good diversity, (**b**) explore the sparse region by the local search.

3. Experimental Studies

This section provides the experimental settings and analyzes the comparison results to verify the performance of the proposed LSEA.

3.1. Experimental Settings

Competitors: Five existing EMOAs, i.e., MOEA/D-CWV [44], DEA-GNG [30], MaOEA/IGD [13], NSGA-III [45], and MOEA/D-PaS [46], are chosen for comparison experiments. For these five baseline EMOAs, they were implemented on MATLAB and exposed to the platform PlatEMO 3.4 [47]. Unless otherwise specified, these baseline EMOAs directly adopt the defaulted parameter settings in the platform.

Benchmark functions: Cheng et al. [48] designed the test suite MaF1–MaF15 by considering various characteristics of real-world applications, e.g., irregular Pareto-optimal fronts, complex landscapes, and large-scale decision variables. In detail, the functions MaF1–MaF9 have irregular Pareto-optimal front shapes, and the other functions mainly reflect the characteristics of MaOPs with complex Pareto sets and large-scale decision variables. Since this paper tries to resolve MaOPs with irregular Pareto-optimal fronts, we selected functions MaF1–MaF9 to perform the experiments and set the number of optimization objectives for the selected functions as 5, 10, 13, and 15, respectively. After configuring a benchmark function with a specific objective number, we term it a test instance, e.g., 10-objective MaF1.

Population Size: For fairness, the population size of the six algorithms is set according to the number of optimization objectives. Specifically, they are 210, 230, 182, and 240 for optimization problems with 5, 10, 13, and 15 objectives, respectively.

Termination Condition: Similar to published papers [30,49,50], we resorted to the maximum number of function evaluations as the termination condition and set it to 1×10^4 for each test instance.

Performance Metrics: The hypervolume (HV) [51] and inverted generational distance (IGD) [52] are two common metrics for assessing the quality of a solution set concerning convergence and diversity. In the experiment, they were employed to measure the quality of solution sets obtained by the six algorithms.

(1) The metric HV represents the volume of space composed of a reference vector and a non-dominated solution set in the objective space. Assuming that the reference vector is $\vec{v} = \{v_1, \cdots, v_m\}$ and the solution set is P, the HV value of P can be estimated as below:

$$HV(P, \vec{v}) = L(\bigcup_{\vec{p} \in P} [f_1(\vec{p}), v_1] \times \cdots \times [f_m(\vec{p}), v_m]), \tag{6}$$

where $L(\cdot)$ refers to the Lebesgue measure. In the experiments, the reference vector of a test instance is set as 1.5 times its nadir point. According to (6), it can be derived that a

larger HV value of a solution set implies the corresponding algorithm being better in both convergence and diversity.

(2) Suppose V^* is a set of vectors evenly distributed over the Pareto-optimal front of an MaOP. The IGD value of a solution set P can be estimated as below:

$$IGD(P, V^*) = \frac{\sum_{\vec{v} \in V^*} \min_{\vec{p} \in \mathbb{P}} dist(\vec{v}, \vec{f}(\vec{p}))}{|V^*|}, \tag{7}$$

where $dist(\vec{v}, \vec{f}(\vec{p}))$ refers to the Euclidean distance between vector $\vec{v}$ and objective vector $\vec{f}(\vec{p})$ of solution $\vec{p}$. According to (7), it can be derived that the solution set with a smaller *IGD* value implies the corresponding algorithm being better in both convergence and diversity. In the experiments, around 40,000 vectors are evenly sampled from the Pareto-optimal front of each test instance.

In each test instance, the six algorithms were repeated 30 times, and their average values and standard deviation for the two metrics are reported. The best average value is highlighted with a gray background. In addition, the Wilcoxon rank-sum test with a significance of 5% was employed to identify the significant difference between the LSEA and each competitor. The signs $-$, $+$, and $\approx$ indicate that the competitor performs significantly worse, better, and similarly to the LSEA, respectively.

All the experiments were conducted on a workstation, which was Windows Server 2016, 256 GB RAM, and Intel(R) Gold 6226R CPUs.

3.2. Experimental Results and Analysis

With respect to metric HV, Table 1 summarizes the comparison results of the six algorithms, i.e., MOEA/D-CWV, DEA-GNG, MaOEA/IGD, NSGA-III, MOEA/D-PaS, and LSEA, on the thirty-six test instances (MaF1–MaF9 with 5, 10, 13, and 15 objectives). In addition, a triple in the last row of this table represents the number of test instances that the corresponding baseline is significantly inferior to, significantly superior to, and similar to the LSEA. For example, the triple 25/6/5 in Table 1 represents that the MOEA/D-CWV is significantly inferior to the LSEA in 25 test instances, significantly superior to the LSEA in 6 test instances, and similar to the LSEA in 5 test instances.

Table 1. HV values (average and standard deviation) obtained by the LSEA and the five baselines on MaF1–MaF9 with 5, 10, 13, and 15 objectives. Highest average HV value on each test instance is highlighted with a gray background.

MOPs	m	MOEA/D-CWV	DEA-GNG	MaOEA/IGD	NSGA-III	MOEA/D-PaS	LSEA
MaF1	5	9.9944×10^{-2} (2.86×10^{-4}) $-$	1.2990×10^{-1} (4.34×10^{-3}) $-$	3.3668×10^{-2} (4.10×10^{-4}) $-$	1.1240×10^{-1} (2.25×10^{-3}) $-$	9.9839×10^{-2} (3.04×10^{-4}) $-$	1.4049×10^{-1} (9.54×10^{-4})
	10	2.9382×10^{-4} (9.21×10^{-6}) $-$	1.3093×10^{-3} (6.86×10^{-5}) $-$	1.7340×10^{-4} (1.28×10^{-5}) $-$	5.6650×10^{-4} (3.15×10^{-5}) $-$	7.0192×10^{-4} (4.10×10^{-5}) $-$	1.6532×10^{-3} (3.93×10^{-5})
	13	4.2211×10^{-6} (5.62×10^{-7}) $-$	5.5170×10^{-5} (1.04×10^{-5}) $-$	5.9935×10^{-6} (1.93×10^{-7}) $-$	2.7776×10^{-5} (2.40×10^{-6}) $-$	1.8881×10^{-5} (1.08×10^{-6}) $-$	8.3325×10^{-5} (1.16×10^{-5})
	15	5.4239×10^{-7} (5.17×10^{-8}) $-$	7.7658×10^{-6} (1.84×10^{-6}) $-$	6.9000×10^{-7} (2.81×10^{-8}) $-$	3.6474×10^{-6} (3.20×10^{-7}) $-$	2.3402×10^{-6} (2.36×10^{-7}) $-$	1.1451×10^{-5} (3.16×10^{-6})
MaF2	5	4.1010×10^{-1} (1.39×10^{-3}) $-$	4.3082×10^{-1} (1.19×10^{-2}) $-$	3.4550×10^{-1} (7.76×10^{-2}) $-$	4.4355×10^{-1} (3.61×10^{-3}) $-$	4.1573×10^{-1} (1.22×10^{-3}) $-$	4.5270×10^{-1} (2.65×10^{-3})
	10	1.9747×10^{-1} (1.60×10^{-3}) $-$	3.9456×10^{-1} (9.60×10^{-3}) $-$	3.6316×10^{-1} (9.96×10^{-3}) $-$	4.4021×10^{-1} (6.01×10^{-3}) $+$	2.1070×10^{-1} (8.45×10^{-3}) $-$	4.2898×10^{-1} (5.35×10^{-3})
	13	1.7913×10^{-1} (8.24×10^{-3}) $-$	3.9394×10^{-1} (9.15×10^{-3}) $-$	3.1243×10^{-1} (2.63×10^{-2}) $-$	4.3522×10^{-1} (4.90×10^{-3}) $+$	1.9692×10^{-1} (8.55×10^{-5}) $-$	4.1855×10^{-1} (9.97×10^{-3})
	15	1.7812×10^{-1} (6.89×10^{-3}) $-$	3.9446×10^{-1} (1.19×10^{-2}) $-$	3.0678×10^{-1} (1.28×10^{-2}) $-$	4.4328×10^{-1} (4.39×10^{-3}) $+$	1.9680×10^{-1} (5.26×10^{-5}) $-$	4.2810×10^{-1} (5.86×10^{-3})

Table 1. *Cont.*

MOPs	m	MOEA/D-CWV	DEA-GNG	MaOEA/IGD	NSGA-III	MOEA/D-PaS	LSEA
MaF3	5	9.1135×10^{-1} (9.69×10^{-2}) −	9.8054×10^{-1} (2.13×10^{-2}) −	1.0602×10^{-1} (1.63×10^{-1}) −	9.9986×10^{-1} (1.19×10^{-5}) +	9.8676×10^{-1} (1.58×10^{-2}) ≈	9.9955×10^{-1} (1.46×10^{-4})
	10	1.8740×10^{-1} (1.66×10^{-1}) −	7.9315×10^{-1} (1.91×10^{-1}) −	3.4905×10^{-1} (3.73×10^{-1}) −	6.2445×10^{-1} (4.65×10^{-1}) −	6.6542×10^{-2} (2.31×10^{-1}) −	9.9975×10^{-1} (1.25×10^{-4})
	13	2.1550×10^{-1} (1.60×10^{-1}) −	8.5703×10^{-1} (5.35×10^{-2}) −	3.5008×10^{-1} (3.40×10^{-1}) −	9.8437×10^{-1} (1.66×10^{-2}) −	0.0000×10^{0} (0.00×10^{0}) −	9.9958×10^{-1} (2.02×10^{-4})
	15	1.8838×10^{-1} (1.66×10^{-1}) −	9.0308×10^{-1} (7.00×10^{-2}) −	4.7280×10^{-1} (3.75×10^{-1}) −	8.9614×10^{-1} (2.83×10^{-1}) −	0.0000×10^{0} (0.00×10^{0}) −	9.9983×10^{-1} (1.02×10^{-4})
MaF4	5	2.1499×10^{-1} (4.51×10^{-4}) −	3.1611×10^{-1} (1.45×10^{-2}) ≈	1.5186×10^{-2} (1.09×10^{-2}) −	2.9706×10^{-1} (4.25×10^{-2}) −	2.1834×10^{-1} (1.47×10^{-3}) −	3.2297×10^{-1} (3.20×10^{-3})
	10	2.2594×10^{-3} (5.09×10^{-4}) −	1.1243×10^{-2} (1.49×10^{-3}) +	3.0327×10^{-4} (1.97×10^{-4}) −	8.9303×10^{-3} (3.23×10^{-4}) −	8.1359×10^{-3} (2.53×10^{-4}) −	1.0364×10^{-2} (4.88×10^{-4})
	13	6.6933×10^{-5} (2.31×10^{-5}) −	8.3993×10^{-4} (1.55×10^{-4}) ≈	2.6118×10^{-6} (4.04×10^{-6}) −	1.0803×10^{-3} (3.42×10^{-5}) +	5.9928×10^{-4} (1.92×10^{-4}) −	8.1597×10^{-4} (5.11×10^{-5})
	15	1.2400×10^{-5} (5.10×10^{-6}) −	1.6756×10^{-4} (3.54×10^{-5}) ≈	1.0684×10^{-6} (7.62×10^{-7}) −	2.5890×10^{-4} (1.15×10^{-5}) +	1.3625×10^{-4} (8.46×10^{-6}) −	1.6980×10^{-4} (9.53×10^{-6})
MaF5	5	6.0621×10^{-1} (2.22×10^{-1}) −	9.4818×10^{-1} (3.60×10^{-3}) +	9.2256×10^{-1} (1.38×10^{-2}) −	9.6023×10^{-1} (2.08×10^{-4}) +	9.5050×10^{-1} (2.20×10^{-3}) +	9.3376×10^{-1} (1.20×10^{-3})
	10	8.2644×10^{-1} (1.25×10^{-1}) −	9.9635×10^{-1} (5.00×10^{-4}) +	9.4846×10^{-1} (2.30×10^{-2}) −	9.9864×10^{-1} (4.17×10^{-5}) +	9.9762×10^{-1} (1.68×10^{-4}) +	9.9416×10^{-1} (2.90×10^{-4})
	13	4.6692×10^{-1} (1.40×10^{-1}) −	9.9932×10^{-1} (1.43×10^{-4}) +	8.4928×10^{-1} (1.06×10^{-1}) −	9.9987×10^{-1} (1.24×10^{-5}) +	9.9952×10^{-1} (2.34×10^{-4}) +	9.9879×10^{-1} (1.49×10^{-4})
	15	5.1025×10^{-1} (1.31×10^{-1}) −	9.9984×10^{-1} (4.61×10^{-5}) +	9.4498×10^{-1} (1.87×10^{-2}) −	9.9998×10^{-1} (4.54×10^{-6}) +	9.9989×10^{-1} (4.16×10^{-5}) +	9.9951×10^{-1} (1.85×10^{-4})
MaF6	5	3.6042×10^{-1} (3.93×10^{-4}) −	3.9291×10^{-1} (3.79×10^{-4}) ≈	3.1271×10^{-1} (9.46×10^{-2}) −	3.8492×10^{-1} (1.57×10^{-3}) −	3.6908×10^{-1} (2.09×10^{-3}) −	3.9295×10^{-1} (7.60×10^{-4})
	10	3.3333×10^{-1} (1.25×10^{-8}) −	2.3356×10^{-1} (1.49×10^{-1}) −	2.5027×10^{-1} (1.51×10^{-1}) −	2.9251×10^{-1} (2.67×10^{-2}) −	3.3313×10^{-1} (2.25×10^{-4}) −	3.4732×10^{-1} (3.05×10^{-4})
	13	3.3333×10^{-1} (1.13×10^{-8}) −	9.3038×10^{-2} (1.26×10^{-1}) −	2.7794×10^{-1} (1.30×10^{-1}) −	5.2199×10^{-2} (8.53×10^{-2}) −	$0.0000 \times 10^{+0}$ $(0.00 \times 10^{+0})$ −	3.3987×10^{-1} (5.63×10^{-3})
	15	3.3333×10^{-1} (9.68×10^{-9}) −	6.6126×10^{-2} (1.20×10^{-1}) −	2.4986×10^{-1} (1.51×10^{-1}) −	1.0683×10^{-2} (3.70×10^{-2}) −	$0.0000 \times 10^{+0}$ $(0.00 \times 10^{+0})$ −	3.3859×10^{-1} (3.72×10^{-3})
MaF7	5	4.5432×10^{-1} (3.16×10^{-2}) −	5.2343×10^{-1} (3.67×10^{-3}) −	4.4304×10^{-1} (3.78×10^{-2}) −	5.3033×10^{-1} (2.10×10^{-3}) −	5.0364×10^{-1} (3.49×10^{-2}) −	5.4277×10^{-1} (3.11×10^{-3})
	10	2.2761×10^{-1} (1.68×10^{-1}) −	4.2564×10^{-1} (2.05×10^{-2}) ≈	3.1528×10^{-1} (4.08×10^{-2}) −	4.7757×10^{-1} (1.57×10^{-3}) +	4.1073×10^{-1} (1.76×10^{-2}) −	4.4004×10^{-1} (1.06×10^{-2})
	13	4.3710×10^{-2} (3.90×10^{-2}) −	3.7934×10^{-1} (3.63×10^{-2}) +	1.0065×10^{-1} (4.98×10^{-2}) −	3.4274×10^{-1} (9.83×10^{-2}) ≈	8.6801×10^{-2} (7.76×10^{-2}) −	3.5740×10^{-1} (1.09×10^{-2})
	15	6.3234×10^{-2} (4.65×10^{-2}) ≈	3.8706×10^{-1} (9.05×10^{-3}) ≈	3.1697×10^{-2} (1.69×10^{-2}) ≈	2.7707×10^{-1} (7.49×10^{-2}) ≈	2.5971×10^{-2} (5.61×10^{-2}) ≈	3.2815×10^{-1} (1.52×10^{-2})
MaF8	5	2.5366×10^{-1} (3.77×10^{-4}) −	3.1472×10^{-1} (1.71×10^{-3}) −	1.6996×10^{-1} (8.33×10^{-3}) −	3.0070×10^{-1} (2.97×10^{-3}) −	1.6025×10^{-2} (3.47×10^{-2}) −	3.2164×10^{-1} (2.28×10^{-3})
	10	1.1594×10^{-3} (2.80×10^{-3}) −	6.7903×10^{-2} (7.09×10^{-4}) −	2.1569×10^{-2} (2.76×10^{-3}) −	5.5293×10^{-2} (1.46×10^{-3}) −	1.1359×10^{-4} (3.93×10^{-4}) −	7.0693×10^{-2} (2.69×10^{-4})
	13	1.3891×10^{-3} (1.95×10^{-3}) −	2.2276×10^{-2} (2.24×10^{-4}) −	5.0363×10^{-3} (1.09×10^{-3}) −	1.8205×10^{-2} (1.22×10^{-3}) −	2.5077×10^{-4} (8.69×10^{-4}) −	2.3271×10^{-2} (1.64×10^{-4})
	15	3.8518×10^{-4} (7.11×10^{-4}) −	1.1125×10^{-2} (2.68×10^{-4}) −	2.2940×10^{-3} (6.29×10^{-4}) −	9.5006×10^{-3} (4.34×10^{-4}) −	5.5675×10^{-4} (1.86×10^{-3}) −	1.1679×10^{-2} (1.02×10^{-4})
MaF9	5	4.7966×10^{-1} (3.94×10^{-3}) −	4.4942×10^{-1} (7.64×10^{-2}) −	3.5350×10^{-1} (5.45×10^{-2}) −	4.7587×10^{-1} (4.28×10^{-2}) −	5.5184×10^{-1} (8.28×10^{-3}) −	5.8329×10^{-1} (1.17×10^{-3})
	10	3.5589×10^{-3} (6.66×10^{-3}) −	6.5076×10^{-2} (1.21×10^{-2}) −	3.3146×10^{-2} (9.68×10^{-3}) −	6.9721×10^{-2} (1.19×10^{-2}) −	1.0099×10^{-1} (3.72×10^{-4}) −	1.1326×10^{-1} (4.08×10^{-4})
	13	6.4587×10^{-4} (1.54×10^{-3}) −	2.8560×10^{-2} (1.02×10^{-2}) −	5.6675×10^{-3} (6.86×10^{-3}) −	3.2285×10^{-2} (2.93×10^{-3}) −	5.4966×10^{-3} (1.97×10^{-4}) −	4.8543×10^{-2} (2.47×10^{-4})
	15	2.6913×10^{-4} (6.95×10^{-4}) −	1.8232×10^{-2} (2.83×10^{-3}) −	3.0516×10^{-3} (3.18×10^{-3}) −	1.6899×10^{-2} (1.92×10^{-3}) −	1.9960×10^{-3} (4.74×10^{-5}) −	2.5110×10^{-2} (1.25×10^{-4})
−/+/≈		35/0/1	25/6/5	35/0/1	23/11/2	30/4/2	

The experimental results in Table 1 illustrate that LSEA has strong competitiveness in solving MaOPs with irregular Pareto-optimal fronts, as it performed best on 22 out of

36 test instances by metric HV. By comparison, MOEA/D-CWV, DEA-GNG, MaOEA/IGD, NSGA-III, and MOEA/D-PaS, respectively, performed best in 0, 3, 0, 11, and 0 instances for metric HV. In addition, the LSEA was significantly superior in MaF1, MaF3, MaF6, MaF8, and MaF9 by all considered objective measures. These comparison results illustrate that the proposed LSEA substantially improved over the five baselines in resolving MaOPs with inverted, concave, disconnected, and degenerate Pareto-optimal fronts.

Although the proposed LSEA and the DEA-GNG all update the weight vectors according to the distribution of visited solutions learned by a growing neural gas network, the LSEA outperforms the DEA-GNG in most test instances. Unlike the DEA-GNG, the LSEA embodies a heuristic selection mechanism and a local search to maintain a well-converged and well-diversified archive, which also feeds back the weight-vector adjustment. The comparison results between LSEA and DEA-GNG show that the proposed mechanism in this paper has superior performance in resolving irregular Pareto-optimal fronts. In particular, the selection mechanism for the external archive in the LSEA only preserves these solutions with better convergence and diversity, effectively avoiding the adverse effects of inappropriate weight vectors. In addition, the local search for solutions with better diversity and relatively poor convergence is conducive to further strengthening convergence and diversity.

The MOEA/D-CWV is a recent weight-vector adjustment approach that dynamically adopts the intermediate objective vectors of the visited solutions to control the distribution of the weight vectors. Its HV values on 35 out of 36 test instances are significantly inferior to those of the proposed LSEA. One essential reason is that the distribution of the visited solutions being inconsistent with the Pareto-optimal fronts misleads the weight vectors to unsuitable areas, which unavoidably impairs the capability of MOEA/D-CWV in balance convergence and diversity. In the LSEA, the non-weight-vector selection mechanism for maintaining archives can alleviate the above drawbacks.

The NSGA-III is a classical baseline integrating Pareto-dominance-based and weight-vector-based environmental selection mechanisms. The HV values obtained by NSGA-III on 25 out of 36 test instances are lower than those of LSEA. The victory of NSGA-III over LSEA on test instances derived from badly-scaled MaF4 and MaF5. The primary reason is that the NSGA-III employs a hyperplane-based normalization procedure to obtain the intercepts along each objective dimension, mitigating the adverse effects of badly-scales among different objectives of MaF4 and MaF5. For MaOPs with other types of Pareto-optimal fronts, the proposed LSEA has strong advantages. Compared with MaOEA/IGD and MOEA/D-PaS, the LSEA poses an overwhelming advantage, as it performed significantly better than them on 30 out of 36 test instances; the MOEA/D-PaS significantly outperformed the LSEA on five test instances derived from MaF5.

It is reasonable to conclude that the proposed LSEA has a superior overall performance than MOEA/D-CWV, DEA-GNG, MaOEA/IGD, NSGA-III, and MOEA/D-PaS with respect to the metric HV. When compared to the five baselines, the main contribution of LSEA is to add an external archive to preserve solutions with better diversity and convergence, gradually approximating the Pareto-optimal fronts. The superior overall performance of LSEA on these 36 test instances demonstrates the effectiveness of the proposed mechanism in solving MaOPs with various Pareto-optimal fronts.

Regarding the metric IGD, the average values and standard deviations of the six algorithms on 5-, 10-, 13-, and 15-objective test instances are summarized in Table 2. By comparing the proposed LESA with the five baselines one by one, we can see that the LESA yielded significantly lower IGD values than MOEA/D-CWV, DEA-GNG, MaOEA/IGD, NSGA-III, and MOEA/D-PaS on 35, 23, 34, 25, and 32 out of the 36 test instances, respectively. These results again illustrate the superior performance of the proposed method in handling MaOPs with various Pareto-optimal fronts. By comparing the results between metrics HV and IGD, we can see that there exist differences in the ordering of the six algorithms. Take the 5-objective MaF1 as an example. The LSEA is significantly better than the DEA-GNG with respect to the metric HV, whereas the opposite is true with respect to the metric

IGD. Although both metrics, HV and IGD, can simultaneously reflect the convergence and diversity of a solution set, their calculation methods are quite different. Specifically, the metric HV is compatible with Pareto dominance—i.e., the comparison result between two solution sets is always consistent with the Pareto-domination-based comparison result— whereas the metric IGD is not.

Table 2. IGD values (average and standard deviation) obtained by the LSEA and the five baselines on the MaF1-MaF9 with 5, 10, 13, and 15 objectives. The lowest average IGD value on each test instance is highlighted with a gray background.

MOPs	m	MOEA/D-CWV	DEA-GNG	MaOEA/IGD	NSGA-III	MOEA/D-PaS	LSEA
MaF1	5	2.2571×10^{-1} (8.88×10^{-6}) −	1.0184×10^{-1} (2.26×10^{-3}) +	2.7503×10^{-1} (1.78×10^{-3}) −	1.8731×10^{-1} (7.36×10^{-3}) −	2.2891×10^{-1} (8.92×10^{-5}) −	1.0518×10^{-1} (1.32×10^{-3})
	10	3.3502×10^{-1} (1.43×10^{-3}) −	2.4214×10^{-1} (5.80×10^{-3}) −	3.5705×10^{-1} (4.11×10^{-3}) −	2.7816×10^{-1} (5.64×10^{-3}) −	2.7747×10^{-1} (4.99×10^{-3}) −	2.1725×10^{-1} (1.77×10^{-3})
	13	5.3133×10^{-1} (3.98×10^{-2}) −	3.0101×10^{-1} (2.53×10^{-2}) −	3.2469×10^{-1} (1.38×10^{-2}) −	2.4220×10^{-1} (4.39×10^{-3}) +	3.1440×10^{-1} (1.18×10^{-2}) −	2.5685×10^{-1} (6.60×10^{-3})
	15	5.6522×10^{-1} (2.29×10^{-2}) −	3.2560×10^{-1} (1.99×10^{-2}) −	4.1628×10^{-1} (6.30×10^{-3}) −	3.2140×10^{-1} (4.71×10^{-3}) −	3.8627×10^{-1} (1.02×10^{-2}) −	2.9037×10^{-1} (3.34×10^{-3})
MaF2	5	1.5809×10^{-1} (8.91×10^{-4}) −	9.3391×10^{-2} (6.13×10^{-3}) ≈	2.4635×10^{-1} (1.60×10^{-1}) −	1.1233×10^{-1} (2.57×10^{-3}) −	2.3429×10^{-1} (7.53×10^{-3}) −	9.1487×10^{-2} (1.26×10^{-3})
	10	8.6779×10^{-1} (2.49×10^{-3}) −	3.9263×10^{-1} (3.48×10^{-2}) −	4.0510×10^{-1} (1.12×10^{-2}) −	2.1820×10^{-1} (2.65×10^{-2}) −	8.4936×10^{-1} (1.44×10^{-2}) −	1.5851×10^{-1} (1.80×10^{-3})
	13	9.1204×10^{-1} (1.15×10^{-2}) −	4.7460×10^{-1} (3.66×10^{-2}) −	4.3217×10^{-1} (1.91×10^{-2}) −	1.8220×10^{-1} (1.88×10^{-2}) +	8.8818×10^{-1} (4.69×10^{-5}) −	2.0778×10^{-1} (1.43×10^{-2})
	15	9.2279×10^{-1} (9.56×10^{-3}) −	4.5420×10^{-1} (4.25×10^{-2}) −	4.6660×10^{-1} (1.74×10^{-2}) −	1.6833×10^{-1} (7.14×10^{-3}) +	8.9755×10^{-1} (5.21×10^{-5}) −	2.2303×10^{-1} (1.16×10^{-2})
MaF3	5	3.0579×10^{-1} (1.74×10^{-1}) −	1.5297×10^{-1} (7.66×10^{-2}) ≈	$7.7677 \times 10^{+0}$ $(8.51 \times 10^{+0})$ −	6.9770×10^{-2} (4.46×10^{-4}) +	1.6804×10^{-1} (9.36×10^{-2}) ≈	8.9238×10^{-2} (6.74×10^{-3})
	10	$5.5843 \times 10^{+0}$ $(1.05 \times 10^{+1})$ −	4.2611×10^{-1} (2.13×10^{-1}) −	$2.4675 \times 10^{+0}$ $(3.30 \times 10^{+0})$ −	$5.4408 \times 10^{+0}$ $(1.36 \times 10^{+1})$ −	$9.6254 \times 10^{+3}$ $(1.21 \times 10^{+4})$ −	1.0013×10^{-1} (2.82×10^{-3})
	13	$2.4819 \times 10^{+0}$ $(2.56 \times 10^{+0})$ −	3.0701×10^{-1} (4.92×10^{-2}) −	$1.5762 \times 10^{+0}$ $(1.37 \times 10^{+0})$ −	2.0418×10^{-1} (5.38×10^{-2}) −	$2.9818 \times 10^{+8}$ $(7.24 \times 10^{+8})$ −	7.7562×10^{-2} (1.73×10^{-3})
	15	$3.6582 \times 10^{+0}$ $(3.75 \times 10^{+0})$ −	2.5794×10^{-1} (6.00×10^{-2}) −	$1.6449 \times 10^{+0}$ $(3.13 \times 10^{+0})$ −	6.3113×10^{-1} $(1.39 \times 10^{+0})$ −	$1.7318 \times 10^{+8}$ $(5.64 \times 10^{+8})$ −	1.0745×10^{-1} (3.10×10^{-3})
MaF4	5	$3.9584 \times 10^{+0}$ (2.30×10^{-3}) −	$2.6085 \times 10^{+0}$ (5.13×10^{-1}) −	$2.1113 \times 10^{+1}$ $(2.31 \times 10^{+0})$ −	$2.5966 \times 10^{+0}$ (7.26×10^{-1}) −	$3.8914 \times 10^{+0}$ (1.13×10^{-1}) −	$2.2408 \times 10^{+0}$ (1.60×10^{-1})
	10	$2.1338 \times 10^{+2}$ $(1.52 \times 10^{+1})$ −	$9.1320 \times 10^{+1}$ $(1.57 \times 10^{+1})$ ≈	$4.3475 \times 10^{+2}$ $(2.59 \times 10^{+2})$ −	$1.0043 \times 10^{+2}$ $(6.65 \times 10^{+0})$ −	$1.0653 \times 10^{+2}$ $(3.45 \times 10^{+0})$ −	$8.8093 \times 10^{+1}$ $(1.13 \times 10^{+1})$
	13	$1.6614 \times 10^{+3}$ $(1.36 \times 10^{+2})$ −	$5.6217 \times 10^{+2}$ $(9.62 \times 10^{+1})$ +	$5.6394 \times 10^{+3}$ $(1.92 \times 10^{+3})$ −	$6.5076 \times 10^{+2}$ $(6.96 \times 10^{+1})$ ≈	$3.2266 \times 10^{+3}$ $(8.20 \times 10^{+3})$ −	$7.1901 \times 10^{+2}$ $(1.07 \times 10^{+2})$
	15	$6.1102 \times 10^{+3}$ $(4.48 \times 10^{+2})$ −	$1.6252 \times 10^{+3}$ $(4.50 \times 10^{+2})$ +	$1.7830 \times 10^{+4}$ $(5.60 \times 10^{+3})$ −	$2.4390 \times 10^{+3}$ $(2.33 \times 10^{+2})$ +	$3.8316 \times 10^{+3}$ $(2.49 \times 10^{+2})$ −	$2.9311 \times 10^{+3}$ $(4.69 \times 10^{+2})$
MaF5	5	$1.2147 \times 10^{+1}$ $(7.84 \times 10^{+0})$ −	$1.7301 \times 10^{+0}$ (2.87×10^{-2}) +	$2.7769 \times 10^{+0}$ (2.89×10^{-1}) +	$2.0839 \times 10^{+0}$ (7.93×10^{-4}) +	$2.3925 \times 10^{+0}$ (1.20×10^{-1}) +	$3.0127 \times 10^{+0}$ (8.69×10^{-2})
	10	$1.4508 \times 10^{+2}$ $(3.17 \times 10^{+1})$ −	$5.2517 \times 10^{+1}$ $(1.74 \times 10^{+0})$ +	$2.9066 \times 10^{+2}$ $(9.67 \times 10^{+0})$ −	$8.3923 \times 10^{+1}$ (9.07×10^{-1}) +	$8.1024 \times 10^{+1}$ $(3.56 \times 10^{+0})$ +	$1.0744 \times 10^{+2}$ $(5.52 \times 10^{+0})$
	13	$1.9188 \times 10^{+3}$ $(3.73 \times 10^{+2})$ −	$3.9859 \times 10^{+2}$ $(2.04 \times 10^{+1})$ +	$2.1260 \times 10^{+3}$ $(1.68 \times 10^{+0})$ −	$5.9210 \times 10^{+2}$ $(6.83 \times 10^{+0})$ +	$1.1559 \times 10^{+3}$ $(5.99 \times 10^{+1})$ −	$8.2484 \times 10^{+2}$ $(4.56 \times 10^{+1})$
	15	$8.1065 \times 10^{+3}$ $(6.61 \times 10^{+3})$ −	$1.2578 \times 10^{+3}$ $(7.57 \times 10^{+1})$ +	$7.2447 \times 10^{+3}$ $(1.25 \times 10^{+0})$ −	$1.8003 \times 10^{+3}$ $(2.40 \times 10^{+1})$ +	$3.5520 \times 10^{+3}$ $(3.31 \times 10^{+2})$ ≈	$3.3784 \times 10^{+3}$ $(8.43 \times 10^{+2})$
MaF6	5	1.6569×10^{-1} (2.88×10^{-6}) −	2.6181×10^{-3} (7.91×10^{-5}) −	5.9087×10^{-1} (1.07×10^{-1}) −	1.9151×10^{-2} (4.03×10^{-3}) −	1.2003×10^{-1} (2.64×10^{-2}) −	2.1521×10^{-3} (3.93×10^{-5})
	10	7.4209×10^{-1} (6.98×10^{-9}) −	2.9395×10^{-1} (2.76×10^{-1}) −	6.1506×10^{-1} (1.79×10^{-1}) −	3.3622×10^{-1} (5.42×10^{-2}) −	7.4221×10^{-1} (1.26×10^{-4}) −	1.9433×10^{-3} (2.78×10^{-5})
	13	7.4209×10^{-1} (6.29×10^{-9}) −	$2.2383 \times 10^{+0}$ $(3.06 \times 10^{+0})$ −	6.5163×10^{-1} (1.53×10^{-1}) −	6.0317×10^{-1} (2.03×10^{-1}) −	$2.4674\text{e}{+2}$ $(4.70 \times 10^{+0})$ −	2.6573×10^{-2} (8.34×10^{-2})
	15	7.4209×10^{-1} (5.39×10^{-9}) −	$7.0657 \times 10^{+0}$ $(1.63 \times 10^{+1})$ −	7.0315×10^{-1} (1.19×10^{-1}) −	$1.0623 \times 10^{+0}$ (4.31×10^{-1}) −	$2.5036 \times 10^{+2}$ (3.79×10^{-2}) −	9.7062×10^{-3} (2.71×10^{-2})

Table 2. *Cont.*

MOPs	m	MOEA/D-CWV	DEA-GNG	MaOEA/IGD	NSGA-III	MOEA/D-PaS	LSEA
MaF7	5	9.5302×10^{-1} (3.94×10^{-1}) −	2.1641×10^{-1} (1.10×10^{-2}) +	8.2024×10^{-1} (6.75×10^{-1}) −	2.7419×10^{-1} (1.05×10^{-2}) −	5.6868×10^{-1} (3.15×10^{-1}) −	2.2791×10^{-1} (4.57×10^{-3})
	10	$1.1289 \times 10^{+1}$ $(7.63 \times 10^{+0})$ −	8.8720×10^{-1} (1.16×10^{-1}) +	$1.4360 \times 10^{+0}$ (5.01×10^{-2}) −	$1.0156 \times 10^{+0}$ (6.15×10^{-2}) −	$3.7380 \times 10^{+0}$ (7.77×10^{-1}) −	9.3593×10^{-1} (2.96×10^{-2})
	13	$2.0031 \times 10^{+1}$ $(1.79 \times 10^{+0})$ −	$3.5781 \times 10^{+0}$ $(1.06 \times 10^{+0})$ −	$2.0050 \times 10^{+0}$ (1.99×10^{-1}) −	$2.1802 \times 10^{+0}$ (6.03×10^{-1}) −	$1.9243 \times 10^{+1}$ $(5.05 \times 10^{+0})$ −	$1.3667 \times 10^{+0}$ (6.16×10^{-2})
	15	$2.2924 \times 10^{+1}$ $(3.66 \times 10^{+0})$ ≈	$5.0657 \times 10^{+0}$ $(1.06 \times 10^{+0})$ ≈	$2.0684 \times 10^{+0}$ (1.90×10^{-1}) ≈	$2.6705 \times 10^{+0}$ (6.87×10^{-1}) ≈	$3.0924 \times 10^{+1}$ $(1.14 \times 10^{+1})$ ≈	$1.1887 \times 10^{+1}$ $(2.06 \times 10^{+1})$
MaF8	5	3.2300×10^{-1} (1.70×10^{-3}) −	8.9701×10^{-2} (3.11×10^{-3}) −	6.3120×10^{-1} (4.70×10^{-2}) −	1.5590×10^{-1} (1.26×10^{-2}) −	$9.6021 \times 10^{+0}$ $(8.71 \times 10^{+0})$ −	7.6439×10^{-2} (8.62×10^{-4})
	10	$4.3291 \times 10^{+0}$ $(2.41 \times 10^{+0})$ −	1.5532×10^{-1} (1.13×10^{-2}) −	$1.2912 \times 10^{+0}$ (8.47×10^{-2}) −	4.5024×10^{-1} (4.80×10^{-2}) −	$1.6009 \times 10^{+1}$ $(1.46e+1)$ −	1.1823×10^{-1} (1.44×10^{-3})
	13	$3.6341 \times 10^{+0}$ $(1.95 \times 10^{+0})$ −	1.9495×10^{-1} (1.17×10^{-2}) −	$1.6276 \times 10^{+0}$ (1.09×10^{-1}) −	4.4571×10^{-1} (1.10×10^{-1}) −	$1.5099 \times 10^{+1}$ $(8.19 \times 10^{+0})$ −	1.5517×10^{-1} (3.06×10^{-3})
	15	$3.6607 \times 10^{+0}$ $(1.88 \times 10^{+0})$ −	1.9304×10^{-1} (2.57×10^{-2}) −	$1.7547 \times 10^{+0}$ (1.47×10^{-1}) −	4.0050×10^{-1} (7.08×10^{-2}) −	$1.9356 \times 10^{+1}$ $(1.96 \times 10^{+1})$ −	1.4683×10^{-1} (2.49×10^{-3})
MaF9	5	4.0919×10^{-1} (9.54×10^{-3}) −	4.1060×10^{-1} (2.79×10^{-1}) −	4.9266×10^{-1} (1.54×10^{-1}) −	3.0970×10^{-1} (1.25×10^{-1}) −	2.1014×10^{-1} (2.68×10^{-2}) −	7.5243×10^{-2} (7.68×10^{-4})
	10	$6.5611 \times 10^{+0}$ $(3.57 \times 10^{+0})$ −	6.0301×10^{-1} (1.63×10^{-1}) −	$1.0792 \times 10^{+0}$ (2.05×10^{-1}) −	5.2745×10^{-1} (1.59×10^{-1}) −	2.4386×10^{-1} (3.02×10^{-3}) −	1.5527×10^{-1} (4.67×10^{-3})
	13	$1.8372 \times 10^{+1}$ $(1.50 \times 10^{+1})$ −	5.8637×10^{-1} (3.73×10^{-1}) −	$5.5421 \times 10^{+0}$ $(4.53 \times 10^{+0})$ −	4.1982×10^{-1} (7.77×10^{-2}) −	$2.4404 \times 10^{+0}$ (3.72×10^{-2}) −	1.4748×10^{-1} (2.34×10^{-3})
	15	$1.6395 \times 10^{+1}$ $(8.85 \times 10^{+0})$ −	3.6453×10^{-1} (1.40×10^{-1}) −	$4.1857 \times 10^{+0}$ $(4.63 \times 10^{+0})$ −	4.0156×10^{-1} (9.71×10^{-2}) −	$2.7329 \times 10^{+0}$ (2.38×10^{-2}) −	1.3699×10^{-1} (1.67×10^{-3})
−/+/≈		35/0/1	23/9/4	34/1/1	25/9/2	32/2/2	

To intuitively compare the LSEA with the five baselines, we resort to the parallel coordinates plot [53] to illustrate their populations with the median HV values among 30 runs on 10-objective MaF1, MaF6, and MaF8, as shown in Figures 2–4. The parallel coordinates plot maps m-dimensional vectors in a 2-dimensional graph with m parallel axes. The value of each dimension of a vector is converted to the vertices on the corresponding axis, and then a polyline connecting these m vertices represents this vector. Generally, a population is considered to have well convergence if it falls within the range of the Pareto-optimal front in the parallel coordinates plot. A population is considered to have good diversity if it spreads over the whole range of the Pareto-optimal front in the parallel coordinates plot.

The 10-objective MaF1 is the representative of MaOPs with inverted Pareto-optimal fronts. The value range of each dimension in 10-objective MaF1 is from 0 to 1. In Figure 2, the populations of the six algorithms fall in the range of 0 to 1, meaning they all have good convergence. This is because the landscape of MaF1 very simple, and it is easy to find the Pareto-optimal solutions. However, the inverted Pareto front of MaF1 brings difficulties to the environmental selection of EMOAs. For the baselines, MOEA/D-CWV, MaOEA/IGD, NSGA-III, and MOEA/D-PaS, the values in many objective dimensions are not less than 0.8, which means their diversity is poor. Among the five baselines, the DEA-GNG performed the best in both convergence and diversity. By comparing Figure 2b,f, we can observe that the DEA-GNG and the LSEA have good convergence, and the LSEA has much better diversity, especially for the first six objectives. The visual advantage of the LSEA concerning convergence and diversity in Figure 2 is consistent with its better metric values than the other five baselines.
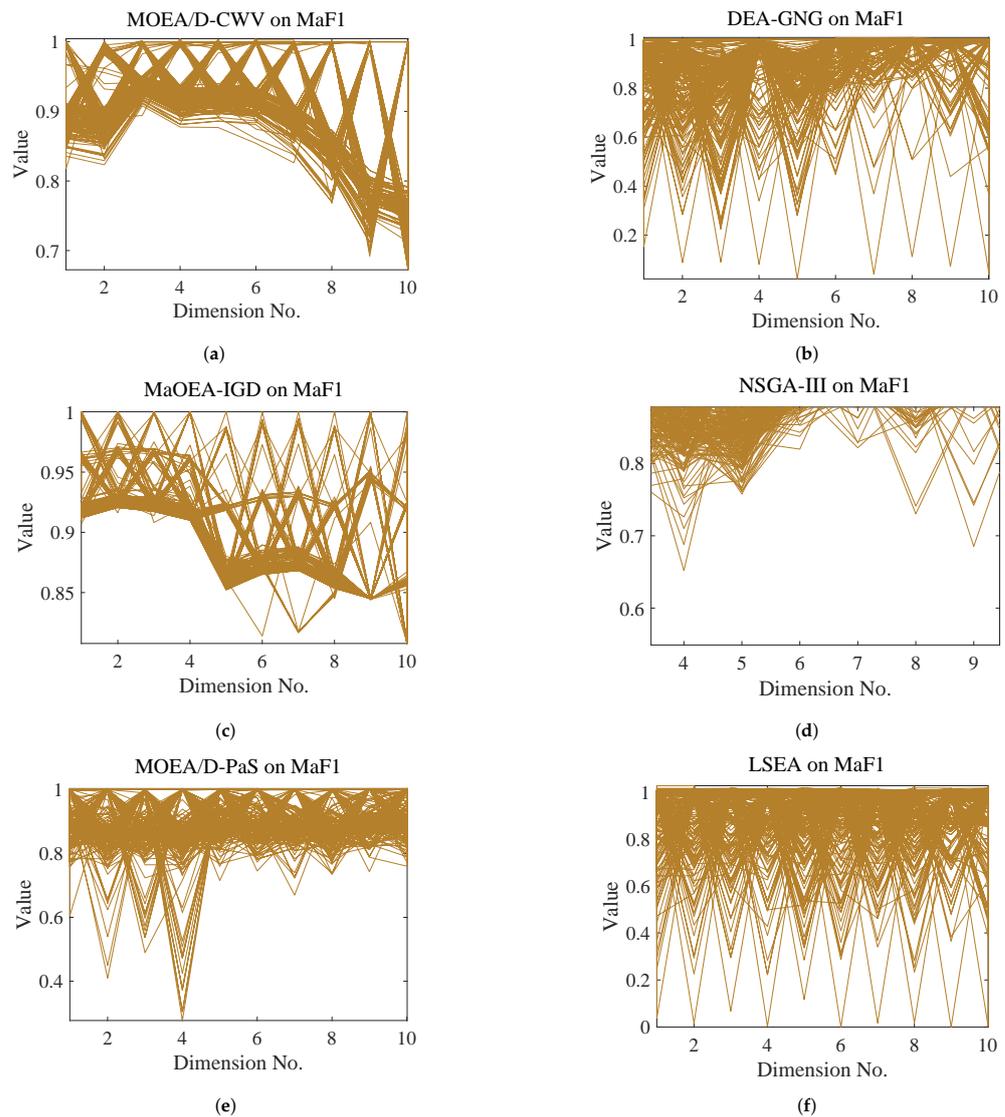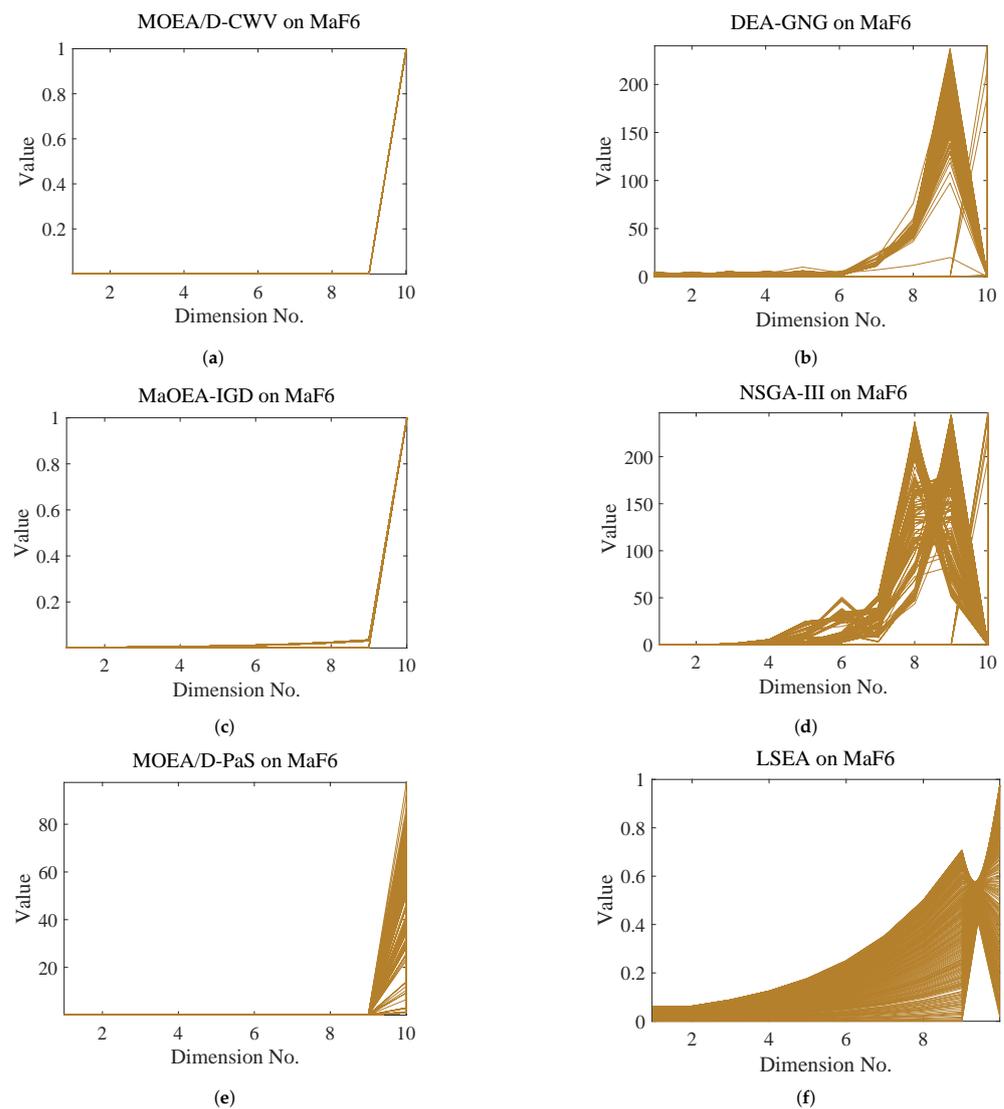
Figure 2. Distributions of populations obtained by the six algorithms in the 10-objective MaF1. (**a**) output population of MaOEA-IGD, (**b**) output population of DEA-GNG, (**c**) output population of MaOEA-IGD, (**d**) output population of NSGA-III, (**e**) output population of MOEA/D-PaS, (**f**) output population of LSEA.

The function MaF6 has a degenerate Pareto-optimal front. The lower bound of its Pareto-optimal front is 0 in each dimension, and the upper bounds in ten objectives are 4.0.0625, 0.0625, 0.0884, 0.1250, 0.1768, 0.2500, 0.3536, 0.5000, 0.7071, and 1.0, respectively. As illustrated in Figure 3, the values of most solutions obtained by DEA-GNG and NSGA-III on the ninth objective are far greater than 0.7071, or even as high as 240, meaning the output populations of these two baselines are far from converging to the Pareto-optimal front. Similarly, the MOEA/D-PaS is far from convergent on the tenth objective. Although MOEA/D-CWV and MaOEA/IGD exhibit good convergence, their output populations converged to very small areas, which means their diversity is very poor. Compared with the five baselines, the proposed LSEA poses outstanding performance in balancing convergence and diversity when solving MaOPs with degenerate Pareto-optimal fronts.

Figure 3. Distributions of populations obtained by the six algorithms on 10-objective MaF6. (**a**) output population of MaOEA-IGD, (**b**) output population of DEA-GNG, (**c**) output population of MaOEA-IGD, (**d**) output population of NSGA-III, (**e**) output population of MOEA/D-PaS, (**f**) output population of LSEA.

The function MaF8 is a classical Pareto-box problem whose Pareto-optimal domains are invariably 2D manifold in the decision spaces. The true Pareto-optimal solutions of MaF8 are evenly distributed within the polyhedron, as illustrated by the curves in Figure 4. From the distributions of the decision variables in Figure 4a, we can observe that the MOEA/D-CWV converges to one point outside the polygon, which means its poor convergence and diversity. As shown in Figure 4c,d, MaOEA/IGD, and NSGA-III pose good convergence by pushing all the solutions into the polygon. However, their diversity is poor, as the output solutions gather in a small area and do not evenly cover the interior of the polygon. The solutions obtained by MOEA/D-PaS are far from converging to the polyhedron. Compared with these five baselines, the proposed LSEA performs better in both convergence and diversity.
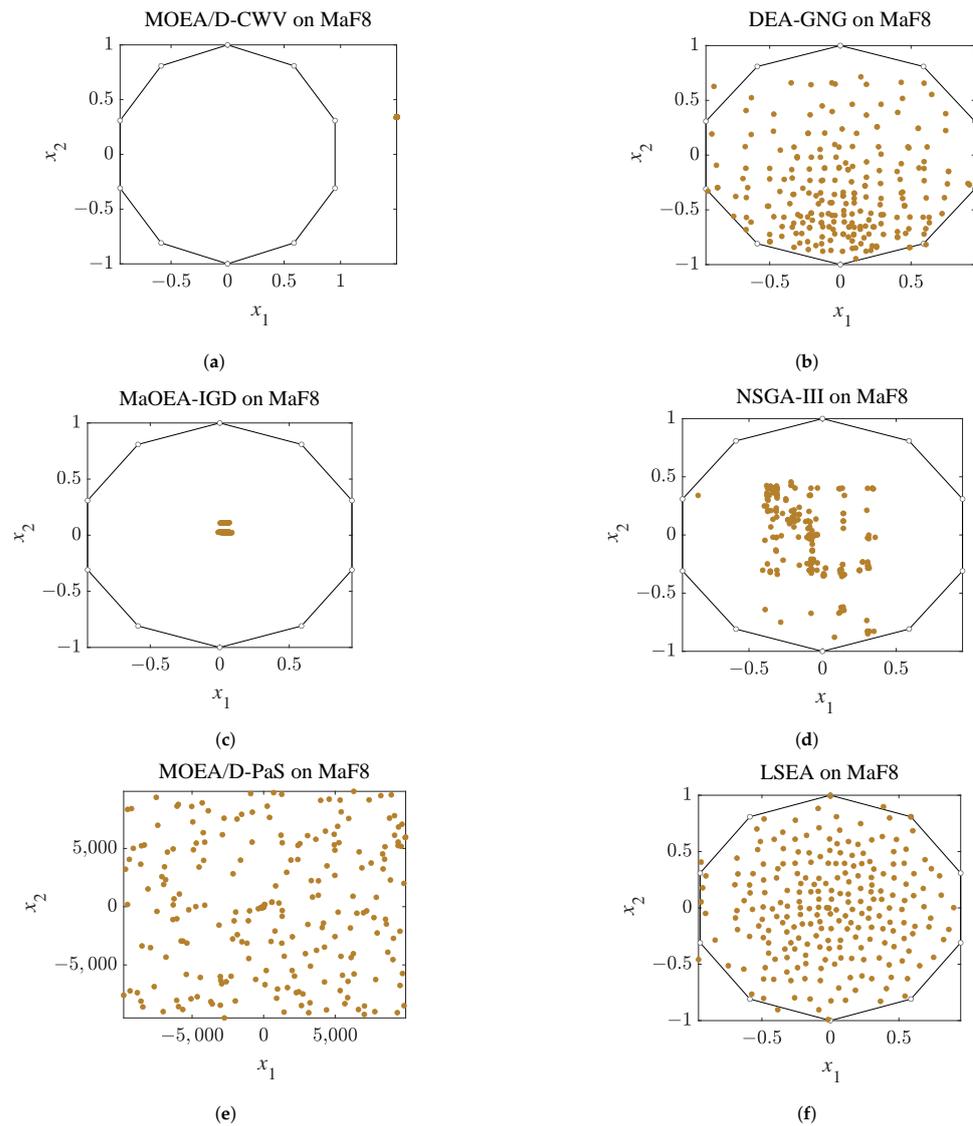
Figure 4. Distributions of populations in decision space of the six algorithms on 10-objective MaF8. (**a**) output population of MaOEA-IGD, (**b**) output population of DEA-GNG, (**c**) output population of MaOEA-IGD, (**d**) output population of NSGA-III, (**e**) output population of MOEA/D-PaS, (**f**) output population of LSEA.

3.3. Performance Improvement of Local Search

To distinguish the performance improvement of the local search mechanism, we constructed two variants of the proposed LSEA, denoted as LSEA-non-LS and LSEA-ran-LS. LSEA-non-LS was constructed by removing the local search mechanism from the LSEA. Unlike the LSEA, which selects a solution with better diversity and relatively poor convergence for local searches, its variant LSEA-ran-LS treats all solutions in the archive equally and selects one randomly. In the context of four test instances, i.e., 10-objective MaF1, MaF2, MaF3, and MaF4, this section compares the changes of the IGD values of LSEA, LSEA-non-LS, and LSEA-ran-LS during their evolution, as shown in Figure 5.

It can be observed in Figure 5 that the IGD values of the LSEA decrease faster than those of its two variants, especially in the early stage of the evolutionary search. This phenomenon demonstrates that the proposed local search mechanism proposed in this paper can effectively accelerate the populations toward the Pareto-optimal fronts. In addition, in the test instance 10-objective MaF2, the IGD values obtained by the LSEA were always lower than those of its two variants. This illustrates that the proposed local search

mechanism can explore solutions with better convergence and diversity. By comparing LSEA-non-LS and LSEA-ran-LS, we found that the IGD value of the LSEA-ran-LS was smaller than that of LSEA-non-LS in most cases. This further demonstrates the performance improvement of the local search mechanism. The comparison results between LSEA and LSEA-ran-LS demonstrate that selecting solutions for local search according to the sum of their diversity and convergence is also beneficial to performance improvement.
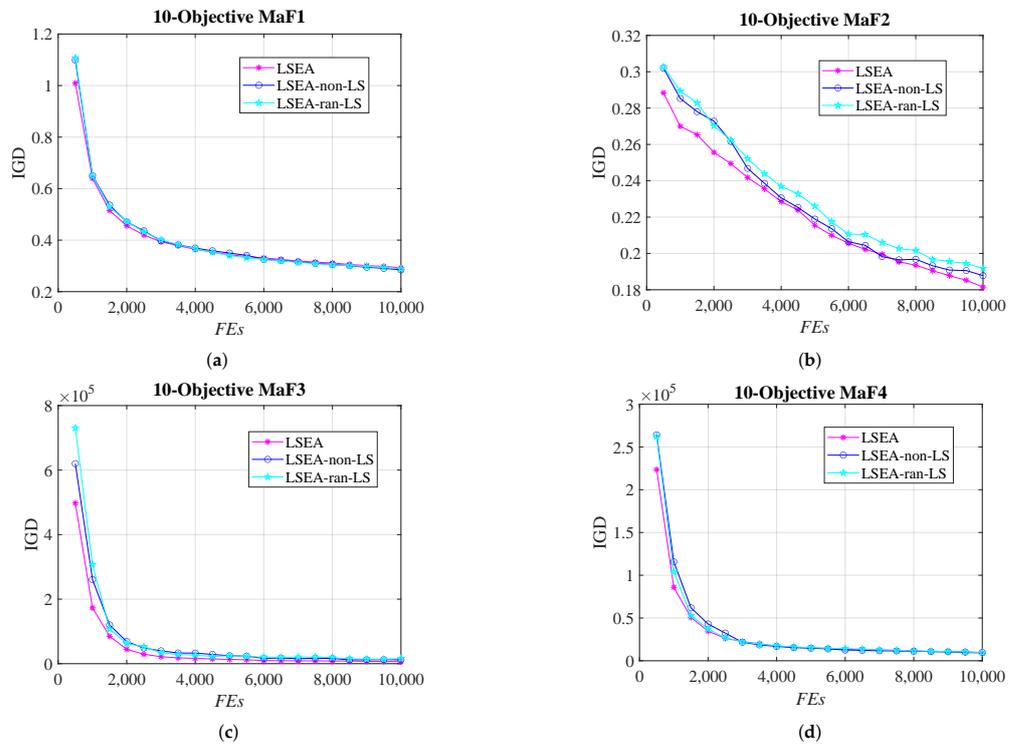
Figure 5. Changes in the IGD values of LSEA, LSEA-non-LS, and LSEA-ran-LS, where *FEs* denotes the number of function evaluations. (**a**) on 10-objective MaF1, (**b**) on 10-objective MaF2, (**c**) on 10-objective MaF3, (**d**) on 10-objective MaF4.

4. Conclusions and Future Directions

This paper suggests adding an external archive for EMOA/Ds using weight-vector adjustment to give full play to their advantages, and we presented a selection mechanism without weight vectors to progressively improve both the convergence and diversity of the archive. Moreover, the proposed algorithm performed local searches on solutions with good diversity but relatively poor convergence to further strengthen convergence and diversity. In the proposed LSEA, once a new solution is reproduced, it will be associated with a solution with the minimum distance. Only the new solution has better convergence and diversity; it will replace the associated solution, progressively enlarging the diversity and convergence of the archive to well approximate irregular Pareto-optimal fronts. To examine the effectiveness of the proposed mechanism, it was compared with five baselines on 36 test instances with various Pareto-optimal front shapes, such as disconnected, degenerate, inverted, and strongly convex/concave. Numerical comparison results corroborate the superior performance of the LSEA in tackling MaOPs with irregular Pareto-optimal fronts: it significantly outperformed all the five baselines on 22 test instances concerning the metric HV.

Although LSEA possesses competitive advantages in solving MaOPs with various Pareto-optimal fronts, it still has two shortcomings. On the one hand, the validation of the LSEA was based on extensive numerical experiments, lacking rigorous theoretical proof. On the other hand, the LSEA has no significant advantages in tackling MaOPs with badly-scaled Pareto-optimal fronts, such as test instances derived from MaF4 and

MaF5. Thus, in future work, we will extend the LSEA to handle MaOPs with a broader range of Pareto-optimal front shapes and research the theoretical basis of multi-objective evolutionary optimization.

References

1. Pham, T.P.; Fahringer, T. Evolutionary multi-objective workflow scheduling for volatile resources in the cloud. *IEEE Trans. Cloud Comput.* **2022**, *10*, 1780–1791. [CrossRef]
2. Chen, H.; Zhu, X.; Liu, G.; Pedrycz, W. Uncertainty-aware online scheduling for real-time workflows in cloud service environment. *IEEE Trans. Serv. Comput.* **2021**, *14*, 1167–1178. [CrossRef]
3. Li, H.; Wang, D.; Zhou, M.; Fan, Y.; Xia, Y. Multi-swarm co-evolution based hybrid intelligent optimization for bi-objective multi-workflow scheduling in the cloud. *IEEE Trans. Parallel Distrib. Syst.* **2022**, *33*, 2183–2197. [CrossRef]
4. Chen, H.; Huang, H.; Zuo, X.; Zhao, X. Robustness enhancement of neural networks via architecture search with multi-objective evolutionary optimization. *Mathematics* **2022**, *10*, 2724. [CrossRef]
5. Wang, Y.; Li, K.; Wang, G.G. Combining key-points-based transfer learning and hybrid prediction strategies for dynamic multi-objective optimization. *Mathematics* **2022**, *10*, 2117. [CrossRef]
6. Yu, G.; Ma, L.; Jin, Y.; Du, W.; Liu, Q.; Zhang, H. A survey on knee-oriented multi-objective evolutionary optimization. *IEEE Trans. Evol. Comput.* **2023**, *in press*.
7. Stewart, R.H.; Palmer, T.S.; DuPont, B. A survey of multi-objective optimization methods and their applications for nuclear scientists and engineers. *Prog. Nucl. Energy* **2021**, *138*, 103830. [CrossRef]
8. Falcón-Cardona, J.G.; Coello, C.A.C. Indicator-based multi-objective evolutionary algorithms: A comprehensive survey. *ACM Comput. Surv.* **2020**, *53*, 1–35. [CrossRef]
9. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
10. He, Z.; Yen, G.G.; Zhang, J. Fuzzy-based Pareto optimality for many-objective evolutionary algorithms. *IEEE Trans. Evol. Comput.* **2013**, *18*, 269–285. [CrossRef]
11. Jiang, S.; Yang, S. A strength Pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization. *IEEE Trans. Evol. Comput.* **2017**, *21*, 329–346. [CrossRef]
12. Zitzler, E.; Thiele, L.; Laumanns, M.; Fonseca, C.M.; Da Fonseca, V.G. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. Evol. Comput.* **2003**, *7*, 117–132. [CrossRef]
13. Sun, Y.; Yen, G.G.; Yi, Z. IGD indicator-based evolutionary algorithm for many-objective optimization problems. *IEEE Trans. Evol. Comput.* **2018**, *23*, 173–187. [CrossRef]
14. Zhang, Q.; Li, H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731. [CrossRef]
15. Trivedi, A.; Srinivasan, D.; Sanyal, K.; Ghosh, A. A survey of multiobjective evolutionary algorithms based on decomposition. *IEEE Trans. Evol. Comput.* **2017**, *21*, 440–462. [CrossRef]
16. Das, I.; Dennis, J.E. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM J. Optim.* **1998**, *8*, 631–657. [CrossRef]
17. Hua, Y.; Liu, Q.; Hao, K.; Jin, Y. A survey of evolutionary algorithms for multi-objective optimization problems with irregular Pareto fronts. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 303–318. [CrossRef]
18. Liu, Q.; Jin, Y.; Heiderich, M.; Rodemann, T. Surrogate-assisted evolutionary optimization of expensive many-objective irregular problems. *Knowl.-Based Syst.* **2022**, *240*, 108197. [CrossRef]
19. Tian, Y.; He, C.; Cheng, R.; Zhang, X. A multistage evolutionary algorithm for better diversity preservation in multiobjective optimization. *IEEE Trans. Syst. Man, Cybern. Syst.* **2021**, *51*, 5880–5894. [CrossRef]

20. Cheng, R.; Jin, Y.; Olhofer, M.; Sendhoff, B. A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Trans. Evol. Comput.* **2016**, *20*, 773–791. [CrossRef]
21. Kinoshita, T.; Masuyama, N.; Liu, Y.; Ishibuchi, H. Reference vector adaptation and mating selection strategy via adaptive resonance theory-based clustering for many-objective optimization. *arXiv* **2022**, arXiv:2204.10756.
22. Chen, H.; Cheng, R.; Pedrycz, W.; Jin, Y. Solving many-objective optimization problems via multistage evolutionary search. *IEEE Trans. Syst. Man, Cybern. Syst.* **2021**, *51*, 3552–3564. [CrossRef]
23. Feng, W.; Gong, D.; Yu, Z. Multi-objective evolutionary optimization based on online perceiving Pareto front characteristics. *Inf. Sci.* **2021**, *581*, 912–931. [CrossRef]
24. Ma, X.; Yu, Y.; Li, X.; Qi, Y.; Zhu, Z. A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms. *IEEE Trans. Evol. Comput.* **2020**, *24*, 634–649. [CrossRef]
25. Li, M.; Yao, X. What weights work for you? Adapting weights for any Pareto front shape in decomposition-based evolutionary multiobjective optimisation. *Evol. Comput.* **2020**, *28*, 227–253. [CrossRef]
26. Liu, Q.; Jin, Y.; Heiderich, M.; Rodemann, T. Coordinated Adaptation of Reference Vectors and Scalarizing Functions in Evolutionary Many-Objective Optimization. *IEEE Trans. Syst. Man, Cybern. Syst.* **2023**, *in press*.
27. Zhao, H.; Zhang, C. An online-learning-based evolutionary many-objective algorithm. *Inf. Sci.* **2020**, *509*, 1–21. [CrossRef]
28. Zheng, W.; Sun, J. Two-stage hybrid learning-based multi-objective evolutionary algorithm based on objective space decomposition. *Inf. Sci.* **2022**, *610*, 1163–1186. [CrossRef]
29. Gu, F.; Cheung, Y.M. Self-organizing map-based weight design for decomposition-based many-objective evolutionary algorithm. *IEEE Trans. Evol. Comput.* **2018**, *22*, 211–225. [CrossRef]
30. Liu, Y.; Ishibuchi, H.; Masuyama, N.; Nojima, Y. Adapting reference vectors and scalarizing functions by growing neural gas to handle irregular Pareto fronts. *IEEE Trans. Evol. Comput.* **2020**, *24*, 439–453. [CrossRef]
31. Liu, Q.; Jin, Y.; Heiderich, M.; Rodemann, T.; Yu, G. An adaptive reference vector-guided evolutionary algorithm using growing neural gas for many-objective optimization of irregular problems. *IEEE Trans. Cybern.* **2022**, *52*, 2698–2711. [CrossRef]
32. Jain, H.; Deb, K. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach. *IEEE Trans. Evol. Comput.* **2014**, *18*, 602–622. [CrossRef]
33. Elarbi, M.; Bechikh, S.; Coello, C.A.C.; Makhlouf, M.; Said, L.B. Approximating complex Pareto fronts with predefined normal-boundary intersection directions. *IEEE Trans. Evol. Comput.* **2020**, *24*, 809–823. [CrossRef]
34. Das, S.; Mullick, S.S.; Suganthan, P.N. Recent advances in differential evolution–an updated survey. *Swarm Evol. Comput.* **2016**, *27*, 1–30. [CrossRef]
35. Liu, Q.; Cui, C.; Fan, Q. Self-adaptive constrained multi-objective differential evolution algorithm based on the state–action–reward–state–action method. *Mathematics* **2022**, *10*, 813. [CrossRef]
36. Abed-alguni, B.H.; Alawad, N.A.; Barhoush, M.; Hammad, R. Exploratory cuckoo search for solving single-objective optimization problems. *Soft Comput.* **2021**, *25*, 10167–10180. [CrossRef]
37. Alkhateeb, F.; Abed-alguni, B.H.; Al-rousan, M.H. Discrete hybrid cuckoo search and simulated annealing algorithm for solving the job shop scheduling problem. *J. Supercomput.* **2022**, *78*, 4799–4826. [CrossRef]
38. Wang, G.; Guo, L.; Wang, H.; Duan, H.; Liu, L.; Li, J. Incorporating mutation scheme into krill herd algorithm for global numerical optimization. *Neural Comput. Appl.* **2014**, *24*, 853–871. [CrossRef]
39. Meghwani, S.S.; Thakur, M. Adaptively weighted decomposition based multi-objective evolutionary algorithm. *Appl. Intell.* **2020**, *51*, 1–23. [CrossRef]
40. Li, K. Decomposition Multi-Objective Evolutionary Optimization: From State-of-the-Art to Future Opportunities. *arXiv* **2021**, arXiv:2108.09588.
41. Yang, M.; Li, C.; Cai, Z.; Guan, J. Differential evolution with auto-enhanced population diversity. *IEEE Trans. Cybern.* **2014**, *45*, 302–315. [CrossRef]
42. Tian, Y.; Cheng, R.; Zhang, X.; Li, M.; Jin, Y. Diversity assessment of multi-objective evolutionary algorithms: Performance metric and benchmark problems. *IEEE Comput. Intell. Mag.* **2019**, *14*, 61–74. [CrossRef]
43. Li, L.; Yen, G.G.; Sahoo, A.; Chang, L.; Gu, T. On the estimation of Pareto front and dimensional similarity in many-objective evolutionary algorithm. *Inf. Sci.* **2021**, *563*, 375–400. [CrossRef]
44. Takagi, T.; Takadama, K.; Sato, H. A distribution control of weight vector set for multi-objective evolutionary algorithms. In Proceedings of the International Conference on Bio-inspired Information and Communication, Pittsburgh, PA, USA, 13–14 March 2019; pp. 70–80.
45. Deb, K.; Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Trans. Evol. Comput.* **2014**, *18*, 577–601. [CrossRef]
46. Wang, R.; Zhang, Q.; Zhang, T. Decomposition-based algorithms using Pareto adaptive scalarizing methods. *IEEE Trans. Evol. Comput.* **2016**, *20*, 821–837. [CrossRef]
47. Tian, Y.; Cheng, R.; Zhang, X.; Jin, Y. PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization. *IEEE Comput. Intell. Mag.* **2017**, *12*, 73–87. [CrossRef]
48. Cheng, R.; Li, M.; Tian, Y.; Zhang, X.; Yang, S.; Jin, Y.; Yao, X. A benchmark test suite for evolutionary many-objective optimization. *Complex Intell. Syst.* **2017**, *3*, 67–81. [CrossRef]

49. Hua, Y.; Jin, Y.; Hao, K. A clustering-based adaptive evolutionary algorithm for multiobjective optimization with irregular Pareto fronts. *IEEE Trans. Cybern.* **2019**, *49*, 2758–2770. [CrossRef]

50. Chen, H.; Cheng, R.; Wen, J.; Li, H.; Weng, J. Solving large-scale many-objective optimization problems by covariance matrix adaptation evolution strategy with scalable small subpopulations. *Inf. Sci.* **2020**, *509*, 457–469. [CrossRef]

51. Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* **1999**, *3*, 257–271. [CrossRef]

52. Bosman, P.A.; Thierens, D. The balance between proximity and diversity in multiobjective evolutionary algorithms. *IEEE Trans. Evol. Comput.* **2003**, *7*, 174–188. [CrossRef]

53. Li, M.; Zhen, L.; Yao, X. How to read many-objective solution sets in parallel coordinates. *IEEE Comput. Intell. Mag.* **2017**, *12*, 88–100. [CrossRef]