Manuscript Details

Manuscript number SWEVO_2018_931

Title Improved NSGA-III with selection-and-elimination operator

Abstract

NSGA-III is a well-known many-objective optimization algorithm in which the reference-point strategy is incorporated to maintain population diversity. However, the convergence capability of NSGA-III is poor in many cases. In this paper, a new selection-and-elimination operator is designed to balance convergence and diversity. First, a selection operator is employed to identify the reference point with the minimum niche count, and then one individual with the shortest penalty-based boundary intersection distance is chosen. Second, a reference point with the maximum niche count is identified, and one individual with the longest penalty-based boundary intersection distance is removed by the elimination operator. To test performance, this modification is verified on benchmark problems with up to 15 objectives, and compared with five other state-of-the-art algorithms. Simulation results demonstrate that our modification achieves the best performance.

Submission Files Included in this PDF

File Name [File Type]

paper-NSGA-III-SE_V2.pdf [Manuscript File]

Improved NSGA-III with selection-and-elimination operator

Zhihua Cui[1], Yu Chang[1], Jiangjiang Zhang[1], Xingjuan Cai[1*], Wensheng Zhang[2]
[1]Complex System and Computational Intelligence Laboratory,
TaiYuan University of Science and Technology, Taiyuan 003024, China;
[2]State Key Laboratory of Intelligent Con-trol and Management of Complex Systems,
Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China.

Abstract: NSGA-III is a well-known many-objective optimization algorithm in which the reference-point strategy is incorporated to maintain population diversity. However, the convergence capability of NSGA-III is poor in many cases. In this paper, a new selection-and-elimination operator is designed to balance convergence and diversity. First, a selection operator is employed to identify the reference point with the minimum niche count, and then one individual with the shortest penalty-based boundary intersection distance is chosen. Second, a reference point with the maximum niche count is identified, and one individual with the longest penalty-based boundary intersection distance is removed by the elimination operator. To test performance, this modification is verified on benchmark problems with up to 15 objectives, and compared with five other state-of-the-art algorithms. Simulation results demonstrate that our modification achieves the best performance.

Keywords: Many-objective optimization, NSGA-III, Reference-point strategy, selection operator, elimination operator;

1. Introduction

Over the past few decades, multi-objective evolutionary algorithms (MOEAs) emerged as a popular project that has since developed maturely to solve the problems with two or three objectives based on their capabilities related to convergence and diversity. Famous MOEAs algorithms such as NSGA-II [1], SPEA2 [2], and IBEA [3] have performed well in multi-objective optimization problems (MOPs) because the individuals can approximate of the Pareto set after running. Usually, selection strategies in MOEAs can be divided into two criteria: a Pareto-based non-dominated sorting strategy, where solutions with a better Pareto rank are selected to provide selection pressure and increase convergence; and a diversity-related approach, used to determine the final choice following the former criteria.

The number of objectives is often more than three, known as many-objective optimization problems (MaOPs). MaOPs are widely seen in industrial and engineering applications, such as robust decision making [4], job-shop scheduling problem [5], water resource engineering [6], engineering optimization problems [7], and so on. Thus, it is not surprising that MaOPs have become a common MOEA-related research activity. However, popular Pareto dominance-based MOEAs that can handle two or three objectives optimization problems well face several challenges in solving MaOPs. First, along with an increasing number of objectives, many non-dominated solutions emerge exponentially and degrade selection pressure to guide solutions toward the Pareto front [8]. Second, because of the inefficiency of Pareto-based non-dominated MaOPs, use of a diversity-preservation operation (the second criterion noted above) plays a leading role in environmental selection. Third, when the dimensionality of the objective space is more than three, visualization becomes highly challenging, thereby complicating identification and evaluation of superior solutions.

To address the above difficulties, scholars have sought to improve the capability of Pareto-based MOEAs for MaOPs. One strategy is to modify the classical Pareto dominance relation to enhance selection pressure toward the Pareto front. This type of idea has been widely adopted in handling MaOPs, such as L-optimality [9], ε-dominance [10][11], preference order ranking [12], and fuzzy dominance [13]. Through modifying the Pareto dominance relationship, these strategies have been shown to be effective for solving MaOPs.

The second approach is to modify the diversity protection mechanism. Deb and Jain [14] proposed an improved MSGA-II algorithm, called NSGA-III, by modifying the diversity operator based on a reference-point strategy. Zhang et al. [15] presented a knee-point-driven evolutionary algorithm (KnEA) that adopts a neighbor-punishment density estimation scheme based on the knee point. Cheng et al. [16] introduced a many-objective environmental selection algorithm (MaOEA-DDFC), which employs projection points on the hyperplane to measure directional diversity and a favorable convergence function to measure solution convergence.

In addition, a decomposition-based approach has recently been proposed for dealing with MaOPs. As the most popular algorithm, Zhang and Li [17] proposed a multi-objective evolutionary algorithm based on decomposition (MOEA/D) and decomposed a MOP into a group of single-objective problems. Asafuddoula et al. [18] proposed an improved decomposition-based evolutionary algorithm (I-DBEA). Yuan et al. [19] presented a function balancing convergence and diversity in decomposition-based many-objective optimizers (MOEA/DDU) to achieve better balance between convergence and diversity in many-objective optimization. Li et al. [20] proposed an evolutionary many-objective optimization algorithm based on dominance and decomposition (MOEA/DD).

NSGA-III (Deb and Jain [14]) employs a reference-point-based strategy to decompose the objective space, and every reference point can be associated with multiple solutions. This approach can effectively maintain the capability of diversity but ignores convergence toward the Pareto front. Recently, Yuan et al. [21] introduced an improved NSGA-III procedure (θ-NSGA-III) that uses a new dominated relation to achieve a balance between convergence and diversity. Bi and Wang [22] added the idea of objective space decomposition into NSGA-III to enhance convergence. However, the framework of these two strategies is largely different from the classical NSGA-III. Our aim in this paper is to modify the general work of NSGA-III and maintain niche preservation to uphold the advantageous preponderant diversity strategy of NSGA-III. Moreover, to improve the convergence capability, we adopt penalty-based boundary intersection (PBI) distance in MOEA/D.

For this purpose, an improved NSGA-III with selection-and-elimination operator

(NSGA-III-SE) is proposed for many-objective optimization. We propose a selection-and-elimination operator in environmental selection that maintain the niche-preservation approach in NSGA-III. However, the selection process chooses the individual with the minimum niche count and shortest PBI distance, whereas the elimination process removes the individual with the maximum niche count amd longest PBI distance. In every round, the selection and elimination process is carried out simultaneously. The PBI distance replaces the perpendicular distance to evaluate the rank of solutions in the same subspace, thus improving convergence.

The rest of this paper is organized as follows. Section 2 outlines the related background of this paper. The proposed NSGA-III-SE strategy is introduced in detail in section 3. Experimental studies involving popular test problems and contrastive experiments are presented in Section 4. Finally, conclusions and future work are discussed in Section 5.

2. Background

2.1 Definitions

A multi-objective problem (MOP) can be usually viewed as a minimization problem, defined mathematically as follows [23] [24]:

$$\begin{cases} \min\ y = F(x) = (f_1(x), f_2(x), \ldots, f_m(x),)^T \\ s.t.\ \ g_i(x) \le 0,\ i = 1, 2, \ldots, q \\ \quad\ \ h_j(x) = 0,\ j = 1, 2, \ldots, p \\ \quad\ \ x \in [x_{\min}, x_{\max}] \end{cases} \tag{1}$$

where $\Omega = \Pi_{i=1}^{n}[a_i, b_i] \subseteq R^n$ is the decision space, and $x = (x_1, x_2, \ldots, x_n) \in \Omega$ is an n-dimensional decision variable. The function of $\Omega \rightarrow R^M$ is to map from the decision space to objective space, and R^M is an M-dimensional objective space. $x \in [x_{\min}, x_{\max}]$ denotes the limit of the decision variable. $g_i(x) \le 0 (i = 1, 2, \ldots, q)$ represents q inequality constraints, and $h_j(x) = 0 (j = 1, 2, \ldots, p)$ represents p equality constraints.

Definition 1 (Pareto dominance): A vector solution x_A dominates another solution x_B (expressed as $x_A \preceq x_B$) if and only if $f_i(x_A) \le f_i(x_B)$ for every $i = 1, 2, \ldots, m$, and $f_j(x_A) < f_j(x_B)$ for at least one $j = 1, 2, \ldots, m$.

Definition 2 (Pareto optimal solution): A solution $x^* \in \Omega$ is deemed the Pareto optimal solution if there is no solution $x \in \Omega$ to satisfy $x \preceq x*$.

Definition 3 (Pareto optimal set): The set of all Pareto optimal solutions comprises the Pareto optimal set $\mathrm{P}^* = \{x \in \Omega \mid \nexists x' \in \Omega, x' \preceq x\}$.

Definition 4 (Pareto optimal front): The curved surface that constitutes the objective vector of all Pareto optimal solutions in Pareto optimal set P^* is expressed as $\mathrm{PF}^* = \{\mathrm{F}(x) \in R^M \mid x \in \mathrm{P}^*\}$.

2.2 NSGA-III

The algorithm of NSGA-III was proposed based on NSGA-II [1] by K. Deb to solve MaOPs and its basic framework is similar to NSGA-II aside from the selection mechanism. The major procedure of NSGA-III is introduced below.

First, H reference points are generated on a hyperplane. Then, an initialization procedure is used and N members are randomly generated, where N is the population size. If the termination criteria are not satisfied, the next steps will be run repeatedly. At the *t*-th generation, *N* members of parent population P_t through mating selection have a simulated binary crossover (SBX) operator and polynomial mutation [25] to produce an offspring population Q_t with size *N*. Next, populations P_t and Q_t are combined to form a new population $R_t = P_t \cup Q_t$ with size 2N. To selecte *N* solutions from R_t, the first step is Pareto-based non-dominated sorting that partitions R_t into several levels(F_1, F_2, and so on). Then, an empty population S_t is produced and members of non-domination levels are added into S_t one level at a time, starting from F_1, until the size of S_t equals or exceeds *N* for the first time. Suppose that the last level added into S_t is F_l, the solutions from $l+1$-th level onwards are rejected. Next, the individuals in $S_t \setminus F_l$ are added into P_{t+1}, and the remaining individuals are chosen from F_l using the environment selection mechanism.

To prepare for environment selection, a normalization operator is first employed, where objective points and reference points are handled in a unit range. Then, individuals are associated with reference points by calculating the perpendicular distance. Finally, the niche-preservation operation is used to choose solutions from F_l. For the *j*-th reference point, the niche count ρ_j is defined as the number of solutions in $S_t \setminus F_l$ associated with the *j*-th reference point. First, the reference points set $J_{\min}$

with the minimum ρ_j value is identified. If $|J_{\min}|>1$, then one $\bar{j} \in J_{\min}$ is randomly chosen. The below two scenarios are then employed:

(1) If some individuals in F_l are associated with the $\bar{j}th$ reference point, then we consider two cases:

Case 1: If $\rho_j = 0$, the individual with the shortest perpendicular distance to the $\bar{j}th$ reference line in F_l associated with the $\bar{j}th$ reference point is selected and added into P_{t+1}. Then, the count of ρ_j is increased by 1.

Case 2: If $\rho_j > 0$, the individual chosen randomly in F_l that is associated with the $\bar{j}th$ reference point is selected and added into P_{t+1}. Then, the count of ρ_j is increased by 1.

(2) If no individual in F_l is associated with the $\bar{j}th$ reference point, the preference point is excluded from further consideration for the current generation and reselection of $\bar{j}$.

This process is repeated until the P_{t+1} equals *N*.

3. The proposed algorithm NSGA-III-SE

NSGA-III-SE is an elitist Pareto-based MOEA in principle. The main difference between NSGA-III-SE and NSGA-III is the environment selection operator and uses PBI distance to replace the perpendicular distance from solutions to reference lines. In this section, we introduce the proposed NSGA-III-SE algorithm in detail. There are six components, and the framework of the proposed algorithm is described first. Then, reference-point generation, mating selection, and adaptive normalization are respectively introduced in the following subsections. Finally, we describe the adaptive penalty distance and selection-and-elimination operator, which are the primary ideas of NSGA-III-SE.

3.1 Framework of the proposed algorithm

The framework of the proposed NSGA-III-SE is described by Algorithm 1, is similar to NSGA-III, but the mating selection mechanism and environment selection mechanism differ. First, a set of K reference points are generated, which can be described as $Z = \{Z_1, Z_2, ..., Z_K\}$. For an m-objective problem, $Z_j (j \in \{1, 2, ..., K\})$ is an m-dimensional vector represented by $Z_j = (Z_{j,1}, Z_{j,2}, ..., Z_{j,m})^T$, where

$Z_{j,k} \geq 0, k = 1, 2, ..., m$ and $\sum_{k=1}^{m} Z_{j,k} = 1$. Next, the initial population is randomly produced. Steps 4–23 are run until the termination criterion is satisfied. In Step 5, the offspring population Q_t are produced using the genetic operators as in NSGA-III. After combining population P_t and Q_t into a new population R_t with size 2N, the members of R_t are subject to non-dominated sorting to classify different Pareto-based non-dominated levels (F_1, F_2, and so on). Next, members of different levels are added from F_1 to P_{t+1} one level at a time, starting from F_1 until the size of S_t equals or exceeds N for the first time. If it equals *N*, then we break; otherwise, we employ the following procedures. Steps 16–20 introduce the primary idea behind NSGA-III-SE. First, calculate the number selected and eliminate members from F_l, then employ adaptive normalization for S_t such that members in S_t are allocated to reference point Z. Finally, use the selection-and-elimination operator to choose the better individuals and remove poor solutions.

Algorithm 1: General framework of NSGA-III-SE

Input: N (population size), p (number of divisions)

Output: population P

1. Z ←Generate_reference_points();
2. P_0 ←Population_initialization();
3. $t \leftarrow 0$
4. **while** termination criterion is not met **do**
5. Q_t ←Genetic_operator(P_t)
6. $R_t \leftarrow P_t \cup Q_t$
7. ($F_1, F_2, \ldots$)=Non_dominated_sort(R_t)
8. $S_t \leftarrow \phi, i = 1$
9. **Repeat**
10. $S_t \leftarrow S_t \cup F_i$, and $i \leftarrow i+1$
11. Until $|S_t| \geq$ N
12. Last front to be included: F_l - F_i
13. **If** $|S_t| =$ N then
14. $P_{t+1} = S_t$, **break**
15. Else
16. Individuals to be selected from F_l: $K = N - (|S_t| - |F_l|)$
17. Or individuals to be eliminated from F_l: $T = |F_l| - K$
18. Adaptive normalization (S_t)

19. $[[\pi(s), d(s)]] \leftarrow$ Allocation_operator (S_t, Z)
20. $P_{t+1} \leftarrow$ Selection_and_elimination_operator (K, π, d, S_t)
21. **End if**
22. $t \leftarrow t+1$
23. **end while**

In this paper, we employ the RP-dominance strategy [26] in the mating selection mechanism to replace the tournament selection operator. In environment selection, we use the selection-and-elimination operator to handle the individuals in F_l, which choose solutions with better convergence and diversity and remove poor solutions by considering the PBI distance and niche-preservation approach. By contrast, NSGA-III employs the niche-preservation approach and considers the perpendicular distance from solutions to reference lines.

3.2 Reference points generation

For reference-point generation, Das and Dennis's systematic approach [27] in NSGA-III is employed to generate a set of reference points. If p divisions are considered along each objective, the total number of reference points (H) in an M-objective problem is given by

$$H = \binom{M+p-1}{p} \tag{2}$$

According to Formula (2), the numbers of p and M determine the number of reference points. But when p is too small, the intermediate points will not be created; when p is bigger than M, it leads to an excessive number of points in the high-dimensional objective space. Therefore, we adopt a two-layer reference-point generation method as suggested in [14]. First, we suppose that $B = \{b^1, b^2, ..., b^{N_1}\}$ and $I = \{i^1, i^2, ..., i^{N_2}\}$ represent the set of reference points in the boundary and inside layers (where $N_1 + N_2 = N$), respectively, which are initialized with different p settings. Then, the coordinates of the weight vector in the inside layer I must be shrunken by the following formula:

$$A = A + \tau(C - A) = (1-\tau)A + \tau C \tag{3}$$

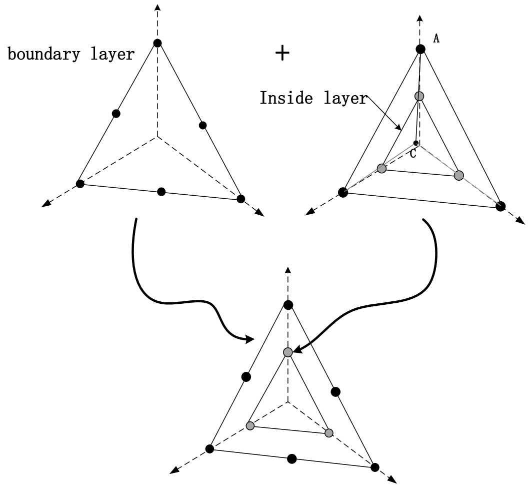


Fig. 1. Module of two-layer weight vector generation method.

where A is the weight vector in the inside layer I, C is the center weight vector, and τ is a scaling factor, generally set to 0.5. B and I comprise the final weight vector set W, as shown in Fig. 1.

3.3 Mating selection

The mating selection of NSGA-III-SE employs the RP-dominance strategy in which the first population P and set of reference points Z are produced, and every member in P is associated with the nearest reference point. The solution u is considered RP-dominance strategy solution v ($u \prec_{RP} v$) if it satisfies the below conditions:

(1) u Pareto dominates v.

(2) u and v are Pareto-equivalent.

a) $RP(u) = RP(v)$ and $d_1(u) < d_1(v)$ or

b) $RP(u) \neq RP(v), d_1(u) < d_1(v),$ *and* $RPDensity(u) < RPDensity(v)$.

$d_1(u)$ is the d_1 distance of u; the value of d_1 will be introduced in the Section 3.5. $RP(u)$ is the reference point assigned to u, and $RPDensity(u)$ represents the number of solutions associated with the reference point of solution u. Due to space limitations, a detailed introduction about RP-dominance can be found in [26].

3.4 Adaptive normalization

The normalization mechanism is used to solve problems whose objective values are scaled differently. First, the ideal point $\overline{z} = (z_1^{\min}, z_2^{\min}, ..., z_M^{\min})$ should be found, after which we can employ the following achievement scalarizing function:

$$ASF(x,z,w^j) = \max_{i=1}^{M}\left(\frac{f_i(\mathrm{x}) - \mathrm{z}_i^{\min}}{w_i^j}\right) \tag{4}$$

where w^j is the axis direction of the objective axis f_j. For $w_i^j = 0$, it is set to a small number 10^{-6}. Finally, the objective function can be normalized as follows:

$$f_i^n(x) = \frac{f_i(\mathrm{x}) - \mathrm{z}_i^{\min}}{a_i - \mathrm{z}_i^{\min}}, \quad for\ i = 1, 2, ..., M \tag{5}$$

where $\sum_{i=1}^{M} f_i^n = 1$.

3.5 Adaptive penalty distance

When considering convergence and diversity, we use the penalty-based boundary intersection distance (PBI distance) to replace the perpendicular distance from solutions to reference lines. The PBI distance can be represented by $d(x) = d_{j,1}(x) + \theta d_{j,2}(x)$, where $d_{j,1}(x)$ is the projection distance from $f^n(x)$ to the $j-th$ reference line L, and $d_{j,2}(x)$ is the perpendicular distance from $f^n(x)$ to L, as indicated in Fig. 2. $d_{j,1}(x)$ and $d_{j,2}(x)$ can be formulated as follows:

$$d_{j,1}(x) = \| (f^n(x))^T w^j \| / \| w^j \| \tag{6}$$

$$d_{j,2}(x) = \| f^n(x) - d_{j,1}(x)(w^j / \| w^j \|) \| \tag{7}$$

where θ is a predefined penalty parameter and a key factor in balancing convergence and diversity in PBI. A smaller value of θ emphasizes convergence, whereas a large θ value focuses on diversity. In [22], experimental results reveal a better effect when $\theta = 5$.

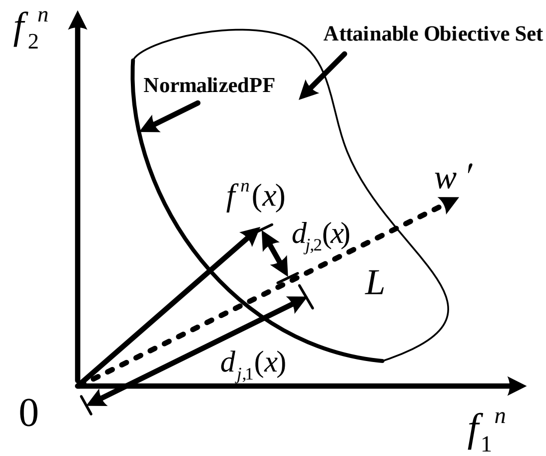


Fig. 2. Illustration of distance $d_{j,1}(x)$ and $d_{j,2}(x)$.

3.6 Selection-and-elimination operator

In environment selection, we employ the selection-and-elimination operator that inherits the core idea of the niche-preservation strategy in NSGA-III with some different operations. First, the values of K and T are calculated in Algorithm 1. K represents the number of solutions selected from F_l , and T represents the number of individuals eliminated from F_l, which satisfy the relation $K+T=|F_l|$. To begin, $k=0$, $t=0$. When $k<K$ and $t<T$, we first use the selection operator to identify the reference points set $J_{\min}$ with the minimum ρ_j value . If $|J_{\min}|>1$, then one $\bar{j}_1 \in J_{\min}$ is randomly chosen. Next, we find the solutions associated with the reference point $\bar{j}_1$ in F_l and choose one with the minimum d value into P_{t+1}. Meanwhile, the niche count $\rho_{\bar{j}_1}$ of the $\bar{j}_1-th$ reference point is increased by 1 such that $k=k+1$. Then, the conditions of $k<K$ and $t<T$ are judged again; if conditions are satisfied, then we employ the elimination operator to identify the reference-point set $J_{\max}$ with the maximum ρ_j value. Similarly, if $|J_{\min}|>1$, then one $\bar{j}_2 \in J_{\max}$ is chosen randomly. Next, we find the solutions associated with reference point $\bar{j}_2$ in F_l and remove the solution with a maximum d value. Next, the value of $\rho_{\overline{j2}}=\rho_{\overline{j2}}-1$ and $t=t+1$. After a cycle, we choose a better solution in P_{t+1} and remove a worse solution from F_l, and the evaluation criteria consider convergence as well as diversity. If the cycle continues to be satisfied, then we select a better solution and eliminate a worse solution again and so on. Once the cycle is not satisfied, we judge whether $k=K$. If $k=K$, then K better solutions have been selected; otherwise, T worse solutions have been eliminated, and the remaining solutions are added into P_{t+1}.

Algorithm 2: Selection-and-elimination operator

Input: K, T, $\pi(\mathrm{s} \in \mathrm{S_t})$, $\mathrm{d}(\mathrm{s} \in \mathrm{S_t})$, $\mathrm{S_t}$

Output: P_{t+1}

Compute niche count of refence point $j \in Z$: $\rho_{\mathrm{j}} = \sum_{s \in St}((\pi(\mathrm{s})=\mathrm{j})?1:0)$

1. $k=0$, $t=0$
2. **while** $k<K$ and $t<T$
3. $\quad J_{\min}=\{\mathrm{j}:\mathrm{argmin}_{\mathrm{j}\in Z}\ \rho_{\mathrm{j}}\}$

$\overline{j_1} = random(J_{\min})$

$I_{\overline{j_1}} = \{s : \pi(s) = \overline{j}_1, s \in S_t\}$

$P_{t+1} = P_{t+1} \cup (s : \mathrm{argmin}_{s \in I_{\overline{j_1}}} d(s))$

$\rho_{\overline{j_1}} = \rho_{\overline{j_1}} - 1$

$k = k + 1$

while $k < K$ and $t < T$

$J_{\max} = \{j : \mathrm{argmax}_{j \in Z}\ \rho_j\}$

$\overline{j_2} = random(J_{\max})$

$I_{\overline{j_2}} = \{s : \pi(s) = \overline{j}_2, s \in S_t\}$

$S_t = S_t \backslash (s: \mathrm{argmax}_{s \in I_{\overline{j_2}}} d(s))$

$\rho_{\overline{j2}} = \rho_{\overline{j2}} - 1$

$t = t + 1$

end while

end while

if $k = K$

continue

else

Choose the remaining individual to P_{t+1}

end if

4. Experimental evaluation

In this section, we verify the performance of NSGA-III-SE via several experimental studies. First, we introduce related background about the standard benchmark problems, performance metrics, and corresponding parameter settings. Then, the performance of NSGA-III-SE is verified by comparing it empirically with five popular MOEAs for MaOPs, namely NSGA-III [14], MOEA/DD [20], MOEA-D-DE [28], KnEA [15], and GrEA[29]. Finally, we discuss the experimental results in detail.

4.1 Experimental Setting

To ensure a fair comparison, we adopted the recommended parameter values for the compared MOEAs that have achieved the best performance. Specifically, the parameter setting of six MOEAs in this paper as following:

(1) Population size

The population size setting of all algorithms is similar. As listed in Table 1, we show different population sizes according to different numbers of objective functions. In NSGA-III, we used a two-layer reference-point strategy to set the reference points. Parameters $p1$ and $p2$ are parameters in NSGA-III to control the number of reference points.

Table 1 Population parameter setting.

No. of objectives (*M*)	Setting ($p1$, $p2$)	Population size (*N*)
3	$p = 12$	91
5	$p = 6$	210
8	3, 2	156
10	3, 2	276
15	2, 1	136

(2) Termination conditions and run time

For each test function, experiments are run 20 times using each algorithm, representing the standard maximum number of iterations in this paper. For DTLZ test function sets, the maximum number of iterations for DTLZ1 is 700; the maximum number for DTLZ3 is 1000; and for DTZ2 and DTLZ4–DTLZ7, the maximum number is 250. For WFG test function sets, the maximum number of iterations for WFG1 is 1000; the maximum number for WFG2 is 700; and the maximum number for WFG3–WFG9 is 250.

(3) Settings for genetic operator

The crossover probability is $p_c = 1.0$ and its distribution index is $\eta_c = 30$. The mutation probability is $p_m = 1/V$ and its distribution index is $\eta_m = 20$.

(4) Penalty parameter θ

Because MOEA/DD, MOEA-D-DE, and the proposed NSGA-III-SE each employ the PBI function, the penalty parameter θ most be set to 5 in accordance with the original study.

4.2 Benchmark and Performance indicators

In our empirical studies, the standard benchmark problems without inequality or equality constraints are used to test the performance of NSGA-III-SE. We chose two well-known test suites for MaOPs: the DTLZ test suite [30] and WFG test suite [31], including DTLZ1–DTLZ7 and WFG1–WFG9. For each DTLZ instance and WFG instance, the number of objectives was considered from three to fifteen, such that M={3,5,8,10,15}. For DTLZ, the Pareto optimal front of the corresponding DTLZ interval is $f_i \in [0, 0.5]$, and that of the remaining DTLZ functions is $f_i \in [0,1]$. According to [30], the number of decision variables is set as $D = M + k - 1$, where $k = 5$ for DTLZ1, $k = 10$ for DTLZ2–DTLZ6, and $k = 20$ for DTLZ7. As described in [31], for WFG test instances, the number of decision variables is set as $D = K + L$, where the position-related variable parameter $K = 2*(M-1)$ and distance-related parameter $L = 10$ [32].

In many-objective evolutionary algorithms, the two main performance metrics, convergence and diversity, can primarily evaluate algorithm performance. We hope the solutions converge well, indicating that every individual approach the true PF as soon as possible; however, we also hope the whole population will be uniformly distributed and cover the entire true PF. If convergence is focused on excessively, it is easy to fall into a local optimal solution. If too much attention is paid to diversity, some individuals will not converge. The widely used quality indicator, inverse generation distance (IGD) [33], was used to measure the convergence and diversity of solutions simultaneously as follows.

For any algorithm, assume that P is the set of non-dominated points in the objective space, and $P*$ is a set of point uniformly spread over the true PF. Then, IGD is expressed as

$$IGD(P, P*) = \frac{\sum_{i=1}^{|P|} d(P_i, P*)}{|P*|} \tag{8}$$

where $|P|$ represents the number of solutions in set P, and $d(P_i, P*)$ is the minimum Euclidean distance from the solution P_i to $P*$ in the objective space. The set P with a smaller IGD value is better.

4.3 Comparisons of NSGA-III-SE with five other MOEAs

To illustrate the performance of NSGA-III-SE, we compared it with five state-of-the-art algorithms in DTLZ and WFG test suites. The performance metric IGD generated by 20 independent simulations performed on the DTLZ and WFG test function sets are summarized in Tables 2 and 3, where the best results are highlighted. In the tables, the symbols ‘+’, ‘-’, and ‘=’ indicate that the results obtained by other algorithms were better, worse, or similar to those obtained by NSGA-III-SE. Comparison of the experimental results indicate that the proposed NSGA-III-SE is competitive in solving MaOPs.

(1) Results on the DTLZ Suite

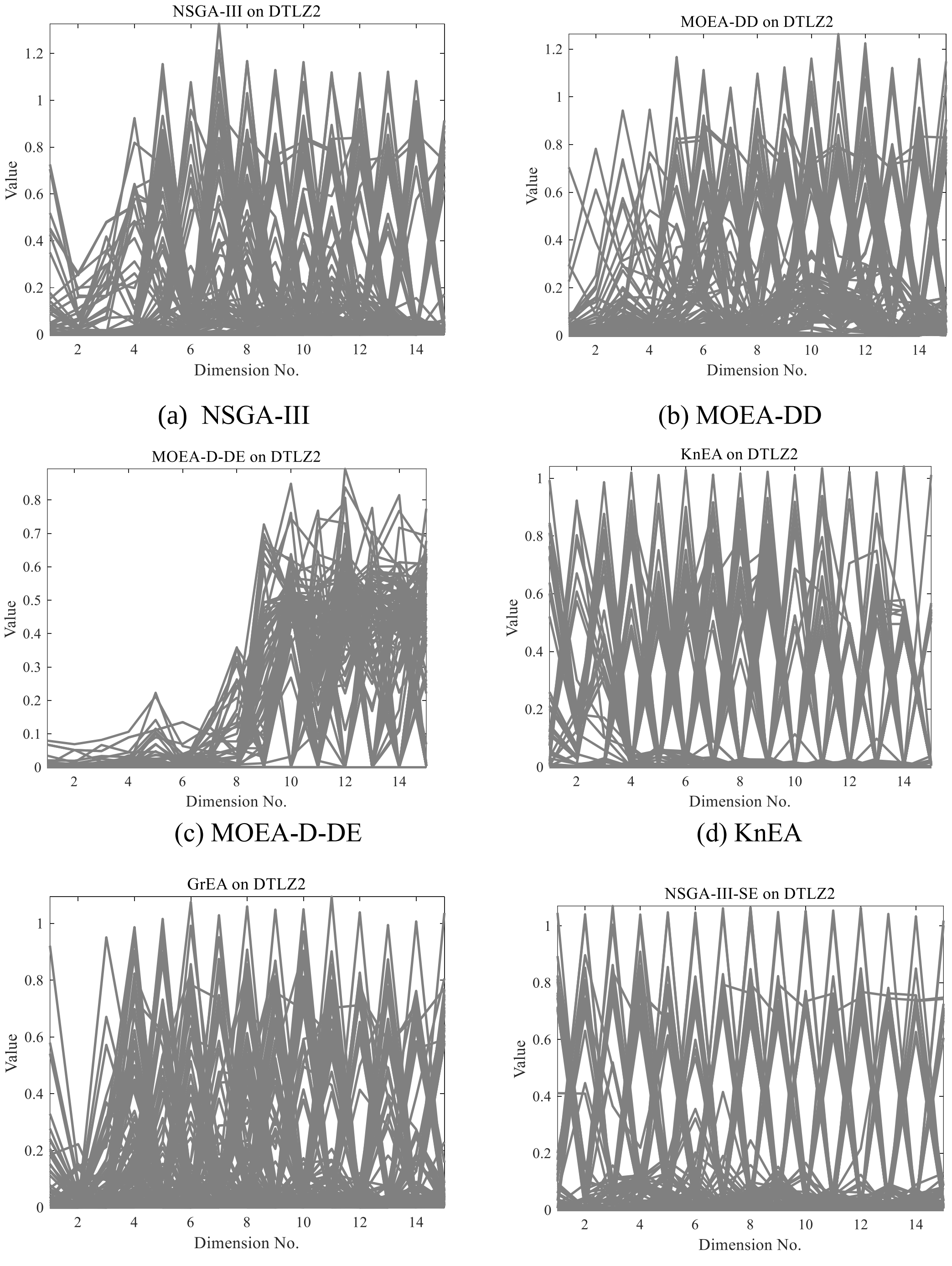
Table 2 lists the comparison results of NSGA-III-SE with five other popular

MOEAs in terms of IGD values in the DTLZ1–DTLZ7 test suite with 3–15 objectives. It presents the average and standard deviation of IGD values for the six algorithms, where the best values in all test suites are highlighted and the last two rows list the comparison evaluation results.

Table 2 Comparisons of results of NSGA-III-SE and five competitive algorithms on DTLZ1-DTLZ7 using IGD.

Problem	Obj	NSGA-III	MOEA-DD	MOEA-D-DE	KnEA	GrEA	NSGA-III-SE
DTLZ1	3	2.1543e+1 (4.67e+0) =	1.7803e+1 (6.98e+0) =	**1.0424e+1 (7.95e+0) +**	1.8833e+1 (5.10e+0) =	2.3719e+1 (5.63e+0) -	1.9187e+1 (4.99e+0)
	5	2.6312e+1 (6.57e+0) -	2.3257e+1 (4.92e+0) =	2.5918e+1 (5.34e+0) -	2.5721e+1 (5.95e+0) =	2.3124e+1 (8.34e+0) =	**2.1716e+1 (6.13e+0)**
	8	2.4304e+1 (5.77e+0) -	2.0249e+1 (4.42e+0) =	**1.8194e+1 (7.41e+0) =**	2.1878e+1 (6.21e+0) =	2.3863e+1 (5.54e+0) -	1.8820e+1 (5.94e+0)
	10	2.9006e+1 (7.82e+0) =	2.5058e+1 (4.91e+0) +	2.4701e+1 (6.27e+0) +	**2.3693e+1 (6.52e+0) +**	2.7752e+1 (1.01e+1) =	3.0205e+1 (8.26e+0)
	15	2.4547e+1 (7.66e+0) -	1.5843e+1 (4.63e+0) =	**1.4876e+1 (7.51e+0) =**	2.0291e+1 (6.43e+0) =	2.3229e+1 (9.01e+0) -	1.7638e+1 (4.46e+0)
DTLZ2	3	4.3432e-1 (4.06e-2) -	4.2914e-1 (4.77e-2) =	**3.8111e-1 (5.73e-2) +**	4.0920e-1 (3.96e-2) =	4.2560e-1 (3.66e-2) -	4.0077e-1 (2.86e-2)
	5	6.0037e-1 (3.09e-2) -	5.8965e-1 (2.03e-2) -	6.4804e-1 (3.41e-2) -	5.7741e-1 (3.12e-2) -	5.9319e-1 (2.82e-2) -	**5.2985e-1 (1.91e-2)**
	8	9.1187e-1 (3.93e-2) -	8.5127e-1 (3.18e-2) =	9.5008e-1 (3.98e-2) -	8.7175e-1 (3.17e-2) -	8.6157e-1 (3.10e-2) -	**8.3533e-1 (2.77e-2)**
	10	**9.8256e-1 (3.48e-2) =**	1.0061e+0 (2.72e-2) =	9.9760e-1 (3.13e-2) =	1.0049e+0 (2.33e-2) =	9.8972e-1 (3.33e-2) =	9.9529e-1 (4.42e-2)
	15	1.2019e+0 (3.87e-2) -	1.1921e+0 (3.08e-2) -	1.2176e+0 (6.83e-2) -	1.1923e+0 (3.62e-2) -	1.1616e+0 (3.03e-2) =	**1.1513e+0 (3.72e-2)**
DTLZ3	3	2.0569e+2 (3.64e+1) =	1.9772e+2 (4.02e+1) =	**6.4988e+1 (4.87e+1) +**	1.7367e+2 (4.10e+1) +	1.8318e+2 (3.98e+1) =	1.9376e+2 (2.64e+1)
	5	3.4033e+2 (5.74e+1) -	3.0411e+2 (6.90e+1) =	**2.1185e+2 (1.93e+1) +**	2.9686e+2 (5.64e+1) =	3.2692e+2 (5.56e+1) =	2.9775e+2 (7.07e+1)
	8	3.2470e+2 (6.85e+1) =	2.4212e+2 (5.80e+1) +	**1.7128e+2 (5.64e+1) +**	2.6721e+2 (5.07e+1) =	3.2882e+2 (7.23e+1) =	2.9860e+2 (5.60e+1)
	10	3.7768e+2 (8.08e+1) -	3.4042e+2 (5.79e+1) =	**2.2002e+2 (2.96e+1) +**	3.3535e+2 (6.78e+1) =	3.6137e+2 (5.73e+1) -	3.0847e+2 (6.13e+1)
	15	3.2894e+2 (7.91e+1) -	2.3946e+2 (4.45e+1) =	**1.6759e+2 (6.31e+1) +**	2.7064e+2 (5.86e+1) =	3.5774e+2 (7.01e+1) -	2.5449e+2 (4.78e+1)
DTLZ4	3	8.4491e-1 (1.19e-1) -	7.6320e-1 (1.65e-1) =	**6.7183e-1 (9.25e-2) =**	8.4819e-1 (1.36e-1) -	7.0019e-1 (1.21e-1) =	6.9453e-1 (1.27e-1)
	5	9.6079e-1 (6.94e-2) -	8.9075e-1 (6.53e-2) -	8.5645e-1 (5.48e-2) -	9.2034e-1 (5.50e-2) -	8.9392e-1 (8.12e-2) -	**7.9939e-1 (6.57e-2)**
	8	1.1226e+0 (7.84e-2) -	1.0391e+0 (5.87e-2) -	9.7959e-1 (3.39e-2) =	1.2571e+0 (8.41e-2) -	1.0609e+0 (7.32e-2) -	**9.7084e-1 (6.61e-2)**
	10	1.1593e+0 (5.32e-2) =	1.1603e+0 (4.74e-2) =	1.1660e+0 (5.34e-2) =	**1.1331e+0 (3.94e-2) +**	1.1437e+0 (6.21e-2) =	1.1592e+0 (3.56e-2)
	15	1.2384e+0 (4.29e-2) -	1.2146e+0 (4.89e-2) -	**1.1232e+0 (3.28e-2) +**	1.3730e+0 (5.10e-2) -	1.1637e+0 (4.16e-2) =	1.1551e+0 (3.79e-2)
DTLZ5	3	3.6181e-1 (4.90e-2) =	3.2825e-1 (3.73e-2) =	**2.5872e-1 (3.71e-2) +**	3.3363e-1 (3.63e-2) =	3.5578e-1 (4.77e-2) =	3.3710e-1 (5.26e-2)
	5	3.4778e-1 (4.22e-2) -	3.5519e-1 (3.07e-2) -	3.3150e-1 (4.51e-2) -	3.4909e-1 (3.58e-2) -	3.4223e-1 (4.54e-2) -	**2.9665e-1 (3.32e-2)**
	8	3.6714e-1 (3.80e-2) -	3.6344e-1 (4.11e-2) -	3.4793e-1 (6.34e-2) =	3.5783e-1 (5.37e-2) -	3.7891e-1 (4.74e-2) -	**3.2290e-1 (4.18e-2)**
	10	**3.6450e-1 (3.72e-2) =**	3.6506e-1 (4.31e-2) =	3.6521e-1 (3.09e-2) =	3.7434e-1 (3.90e-2) =	3.7114e-1 (4.08e-2) =	3.7208e-1 (2.60e-2)
	15	3.7727e-1 (4.29e-2) +	3.8742e-1 (4.47e-2) =	**3.6569e-1 (5.63e-2) +**	3.7037e-1 (3.88e-2) +	3.8757e-1 (3.31e-2) =	4.0919e-1 (4.54e-2)
DTLZ6	3	8.0084e+0 (2.07e-1) =	7.9490e+0 (3.20e-1) =	**4.3777e+0 (1.08e+0) +**	7.6605e+0 (3.67e-1) +	7.9925e+0 (2.74e-1) =	7.9360e+0 (3.11e-1)
	5	8.0007e+0 (2.48e-1) =	8.0973e+0 (2.05e-1) -	**5.0994e+0 (7.19e-1) +**	7.9611e+0 (2.72e-1) =	7.9319e+0 (3.21e-1) =	7.7925e+0 (3.36e-1)
	8	8.2069e+0 (2.50e-1) =	8.1480e+0 (3.50e-1) =	**5.2619e+0 (7.27e-1) +**	7.9717e+0 (2.06e-1) =	7.9423e+0 (3.15e-1) =	8.0917e+0 (1.58e-1)
	10	**8.3456e+0 (1.38e-1) =**	8.3497e+0 (1.33e-1) =	8.3801e+0 (1.07e-1) =	8.3879e+0 (1.48e-1) =	8.3977e+0 (1.20e-1) =	8.3594e+0 (1.34e-1)
	15	8.1727e+0 (1.97e-1) =	8.0795e+0 (4.27e-1) =	**5.3117e+0 (6.84e-1) +**	8.2572e+0 (1.82e-1) =	8.2552e+0 (2.95e-1) =	8.0715e+0 (3.73e-1)
DTLZ7	3	8.3896e+0 (6.17e-1) -	**7.2014e+0 (6.41e-1) +**	7.6075e+0 (1.18e+0) =	7.7306e+0 (7.71e-1) =	7.7845e+0 (8.33e-1) =	7.7330e+0 (9.74e-1)
	5	1.4307e+1 (8.05e-1) -	1.3510e+1 (9.39e-1) -	1.3171e+1 (1.18e+0) =	1.3677e+1 (1.12e+0) -	1.3385e+1 (1.28e+0) =	**1.2670e+1 (1.38e+0)**
	8	2.4789e+1 (1.93e+0) =	2.2487e+1 (1.66e+0) +	**2.1681e+1 (2.51e+0) +**	2.3788e+1 (1.44e+0) =	2.4160e+1 (1.62e+0) =	2.4160e+1 (1.87e+0)
	10	2.9125e+1 (2.65e+0) =	**2.8717e+1 (2.20e+0) =**	2.9255e+1 (2.10e+0) =	2.9742e+1 (1.89e+0) =	2.9082e+1 (1.83e+0) =	2.8854e+1 (1.78e+0)
	15	4.9863e+1 (2.97e+0) -	**4.4447e+1 (4.01e+0) +**	4.5317e+1 (5.28e+0) =	4.8244e+1 (3.13e+0) =	4.8545e+1 (3.05e+0) =	4.7036e+1 (4.05e+0)
Best/All		3/35	3/35	18/35	2/35	0/35	9/35
est/Worse/Simil		1/19/15	5/9/21	16/6/13	5/10/20	0/12/23	---

For the normalized test problems DTLZ1–DTLZ7, NSGA-III-SE demonstrated great performance on overall test problems but was not better than MOEA-D-DE. NSGA-III-SE performed best on the DTLZ2 problem and showed many advantages in the DTLZ4 and DTLZ5 tests. NSGA-III was superior only on the DTLZ2, DTLZ5, and DTLZ6 problems with 10 objectives; it performed poorly on other test problems. MOEA-DD only worked well on the DTLZ7 test problem and was worse or equal to NSGA-III-SE on most other problems. MOEA-D-DE performed very well on DTLZ1, DTLZ3, and DTLZ6 problems because normalization was not always necessary for normalized problems and exerted negative effects on the performance of NSGA-III-SE. By contrast, MOEA-D-DE (without normalization) had clear advantages in dealing with normalized problems but exhibited similar performance to NSGA-III-SE on DTLZ4, DTLZ5, and DTLZ7 test problems. MOEA-D-DE performed worse than NSGA-III-SE on DTLZ2. Finally, KnEA and GrEA showed no superiority on DTLZ test problems. In brief, we found that NSGA-III-SE performed better than the NSGA-III, MOEA-DD, KnEA, and GrEA algorithms and only worse than MOEA-D-DE with the DTLZ test suite.



(a) NSGA-III (b) MOEA-DD

(c) MOEA-D-DE (d) KnEA

(e) GrEA (f) NSGA-III-SE

Fig. 3. Parallel coordinates of non-dominated fronts obtained by six algorithms in 15-objectives DTLZ2 instance.

To verify the diversity of the proposed algorithm and visually depict the solutions' distribution, Fig. 3 shows the parallel coordinates of non-dominated fronts obtained by NSGA-III-SE and the other five MOEAs, respectively, for a 15-objective DTLZ2 instance. This run was associated with the result closest to the average IGD value. From the six plots, it is obvious that the non-dominated front of the proposed NSGA-III-SE algorithm was best in convergence and diversity and slightly better than NSGA-III, MOEA-DD, KnEA, and GrEA. The non-dominated front of MOEA-D-DE was far from PF because the maximum values of some objectives were much smaller than 0.5.

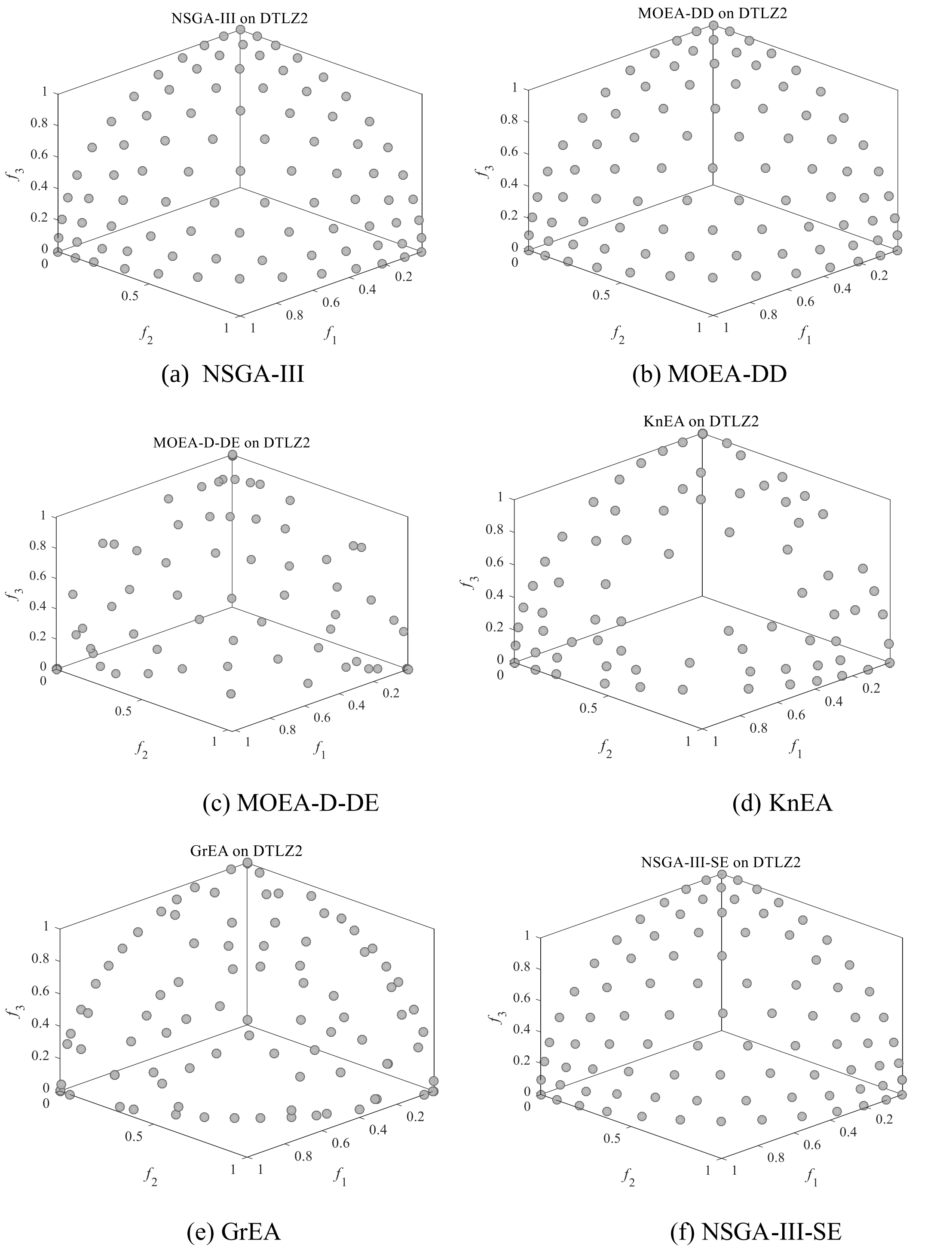


Fig. 4. Plots of non-dominated fronts obtained by six algorithms in the 3-objective DTLZ2 instance.

Figure 4 shows the non-dominated fronts obtained by six algorithms in the 3-objective DTLZ2 instance, where the points represent the set of final non-dominated points when achieving the maximum number of iterations for every algorithm. The solutions of NSGA-III, MOEA-DD, and the proposed NSGA-III-SE exhibited good diversity, presumably because they each had a normalized mechanism. However, MOEA-D-DE lacked good diversity because the individuals were not distributed in an orderly manner along the true Pareto front. The central solutions of KnEA were sparse and the ambient ones were dense, resulting in poor diversity. For GrEA, several central regions were sparse because the steady-state algorithm was incompatible with the normalized mechanism.

(2) Results on the WFG Suite

We next compared the quality of the solution sets obtained by the six algorithms on WFG1–WFG9 test problems in terms of hypervolume. Table 3 presents the comparison results of NSGA-III-SE with the other five popular MOEAs in terms of IGD values on the WFG1–WFG9 test suite with 3–15 objectives, where the best values in all test suites are highlighted and the last two rows list the comparison evaluation results.

Table 3 Comparisons of results of NSGA-III-SE and five competitive algorithms on WFG1-WFG9 using IGD

Problem	Obj	NSGA-III	MOEA-DD	MOEA-D-DE	KnEA	GrEA	NSGA-III-SE
WFG1	3	2.1221e+0 (1.01e-1) -	2.1450e+0 (2.26e-1) -	2.1253e+0 (7.98e-2) -	2.0433e+0 (8.95e-2) =	**1.8834e+0 (8.72e-2) +**	1.9984e+0 (5.54e-2)
	5	2.6571e+0 (8.79e-2) =	2.6488e+0 (1.04e-1) =	2.9302e+0 (1.13e-1) -	2.5916e+0 (6.20e-2) +	**2.5263e+0 (5.52e-2) +**	2.6349e+0 (8.28e-2)
	8	3.1847e+0 (1.17e-1) =	3.2367e+0 (1.11e-1) =	3.5084e+0 (1.10e-1) -	3.2232e+0 (1.83e-1) =	**3.0554e+0 (8.53e-2) +**	3.2781e+0 (2.01e-1)
	10	3.5415e+0 (1.01e-1) =	3.7086e+0 (6.40e-2) -	3.7506e+0 (1.18e-1) -	3.5018e+0 (1.11e-1) =	**3.3864e+0 (7.37e-2) +**	3.5444e+0 (1.64e-1)
	15	4.5363e+0 (1.72e-1) -	4.4921e+0 (9.28e-2) =	4.7021e+0 (8.53e-2) -	4.4516e+0 (1.83e-1) =	4.4170e+0 (1.65e-1) =	**4.3933e+0 (1.70e-1)**
WFG2	3	5.7020e-1 (4.16e-2) -	7.5366e-1 (8.22e-2) -	6.3200e-1 (6.27e-2) -	**5.0971e-1 (3.40e-2) =**	5.2131e-1 (3.42e-2) =	5.2780e-1 (2.85e-2)
	5	9.2601e-1 (9.45e-2) -	1.1155e+0 (2.52e-1) -	1.0274e+0 (7.36e-2) -	9.0092e-1 (7.65e-2) =	8.7627e-1 (5.59e-2) =	**8.7481e-1 (5.93e-2)**
	8	1.6970e+0 (2.45e-1) =	2.0404e+0 (2.85e-1) -	1.7799e+0 (1.20e-1) -	1.9313e+0 (4.29e-1) -	**1.5272e+0 (1.44e-1) =**	1.6381e+0 (2.07e-1)
	10	2.9107e+0 (3.23e-1) =	3.5598e+0 (4.76e-1) -	**2.5417e+0 (2.67e-1) +**	3.1299e+0 (2.69e-1) -	2.7055e+0 (3.52e-1) =	2.8200e+0 (3.15e-1)
	15	5.3449e+0 (1.46e+0) =	7.7718e+0 (2.26e+0) -	**4.2728e+0 (8.27e-1) +**	7.0552e+0 (2.13e+0) =	4.5841e+0 (1.10e+0) +	6.0531e+0 (1.94e+0)
WFG3	3	6.5022e-1 (3.83e-2) =	7.5467e-1 (4.85e-2) -	6.7449e-1 (6.82e-2) -	**6.2702e-1 (2.50e-2) =**	6.4168e-1 (2.07e-2) =	6.2787e-1 (3.51e-2)
	5	8.8900e-1 (2.98e-2) -	9.2532e-1 (4.99e-2) -	9.6976e-1 (5.12e-2) -	8.7838e-1 (4.11e-2) -	8.8624e-1 (4.33e-2) -	**8.3737e-1 (3.15e-2)**
	8	1.3084e+0 (4.82e-2) =	1.4971e+0 (7.54e-2) -	1.4818e+0 (8.18e-2) -	1.3295e+0 (6.03e-2) =	1.3131e+0 (5.13e-2) =	**1.2911e+0 (9.92e-2)**
	10	1.5417e+0 (6.11e-2) =	**1.5111e+0 (5.07e-2) =**	1.5190e+0 (7.24e-2) =	1.5130e+0 (5.81e-2) =	1.5183e+0 (6.92e-2) =	1.5248e+0 (4.42e-2)
	15	2.3091e+0 (8.33e-2) -	2.5340e+0 (1.09e-1) -	2.2895e+0 (1.49e-1) -	2.3234e+0 (1.41e-1) -	**2.1406e+0 (1.27e-1) =**	2.1690e+0 (1.07e-1)
WFG4	3	6.4156e-1 (6.80e-2) =	6.4309e-1 (6.35e-2) =	7.2616e-1 (4.73e-2) -	6.0979e-1 (4.02e-2) =	**5.8759e-1 (5.26e-2) +**	6.2829e-1 (5.08e-2)
	5	2.1746e+0 (1.86e-1) -	1.9949e+0 (1.87e-1) =	**1.7647e+0 (1.01e-1) +**	2.1914e+0 (1.78e-1) -	2.0635e+0 (1.49e-1) -	1.9154e+0 (1.54e-1)
	8	6.5844e+0 (5.30e-1) -	6.0764e+0 (3.39e-1) =	**5.2141e+0 (5.67e-1) +**	6.5155e+0 (2.48e-1) -	6.3844e+0 (3.56e-1) -	6.1520e+0 (3.00e-1)
	10	9.5043e+0 (5.17e-1) =	9.5544e+0 (4.83e-1) =	9.6412e+0 (3.94e-1) -	9.5489e+0 (3.33e-1) =	9.5152e+0 (3.40e-1) =	**9.2522e+0 (5.49e-1)**
	15	1.8525e+1 (7.63e-1) -	1.8192e+1 (9.48e-1) =	**1.6405e+1 (1.08e+0) +**	1.8612e+1 (9.90e-1) -	1.8030e+1 (9.19e-1) =	1.8091e+1 (7.05e-1)
WFG5	3	7.4535e-1 (3.24e-2) =	7.8767e-1 (3.50e-2) -	**6.5833e-1 (5.31e-2) +**	7.2087e-1 (2.90e-2) =	7.2898e-1 (2.04e-2) =	7.2625e-1 (3.45e-2)
	5	1.6987e+0 (6.72e-2) -	1.6755e+0 (7.43e-2) -	1.7097e+0 (5.89e-2) -	1.7212e+0 (7.49e-2) -	1.6170e+0 (9.23e-2) -	**1.5347e+0 (5.42e-2)**
	8	4.9782e+0 (2.73e-1) -	4.8351e+0 (3.35e-1) =	5.2490e+0 (3.22e-1) -	4.9273e+0 (1.67e-1) -	4.7147e+0 (2.28e-1) =	**4.6933e+0 (2.88e-1)**
	10	**7.1298e+0 (3.05e-1) =**	7.1614e+0 (2.53e-1) =	7.2169e+0 (2.81e-1) =	7.1584e+0 (3.36e-1) =	7.2275e+0 (2.55e-1) =	7.2655e+0 (2.96e-1)
	15	1.5150e+1 (5.60e-1) -	1.4825e+1 (5.23e-1) -	1.5840e+1 (8.95e-1) -	1.5697e+1 (1.19e+0) -	1.5466e+1 (5.04e-1) -	**1.4312e+1 (6.62e-1)**
WFG6	3	8.4790e-1 (2.56e-2) =	9.2959e-1 (5.20e-2) -	9.2525e-1 (6.18e-2) -	**8.3107e-1 (3.70e-2) =**	8.3340e-1 (2.73e-2) =	8.4850e-1 (4.90e-2)
	5	1.8754e+0 (9.05e-2) -	1.8135e+0 (6.66e-2) -	1.9065e+0 (1.38e-1) -	1.8329e+0 (6.94e-2) -	1.8192e+0 (6.65e-2) -	**1.6816e+0 (6.25e-2)**
	8	5.2568e+0 (2.61e-1) -	5.1621e+0 (2.28e-1) -	5.3422e+0 (4.87e-1) -	5.2608e+0 (3.30e-1) -	5.0159e+0 (2.33e-1) =	**4.9255e+0 (2.81e-1)**
	10	7.6159e+0 (3.51e-1) =	7.6962e+0 (3.41e-1) =	**7.4233e+0 (2.93e-1) =**	7.7040e+0 (2.98e-1) =	7.4359e+0 (3.31e-1) =	7.5052e+0 (3.67e-1)
	15	1.5557e+1 (6.74e-1) -	1.5108e+1 (7.15e-1) =	1.6397e+1 (9.24e-1) -	1.6615e+1 (1.27e+0) -	1.5668e+1 (7.18e-1) -	**1.4885e+1 (6.26e-1)**
WFG7	3	6.7593e-1 (2.24e-2) =	7.1634e-1 (4.87e-2) -	7.8312e-1 (4.98e-2) -	**6.5806e-1 (2.91e-2) =**	6.6154e-1 (2.13e-2) =	6.6877e-1 (3.21e-2)
	5	1.8110e+0 (1.23e-1) -	1.7132e+0 (1.28e-1) -	1.9083e+0 (1.46e-1) -	1.7928e+0 (1.00e-1) -	1.7378e+0 (1.11e-1) -	**1.6317e+0 (9.66e-2)**
	8	5.4334e+0 (3.14e-1) -	5.3437e+0 (4.10e-1) =	5.9210e+0 (4.31e-1) -	5.3889e+0 (3.65e-1) =	5.2404e+0 (2.53e-1) =	**5.2186e+0 (2.63e-1)**
	10	8.1682e+0 (3.94e-1) =	8.0083e+0 (4.13e-1) =	7.9440e+0 (4.12e-1) =	8.1816e+0 (2.82e-1) =	8.1274e+0 (3.30e-1) =	**7.9196e+0 (2.88e-1)**
	15	1.6289e+1 (6.11e-1) -	1.5914e+1 (7.75e-1) =	1.7456e+1 (1.10e+0) -	1.6544e+1 (1.06e+0) -	1.6140e+1 (7.72e-1) -	**1.5627e+1 (8.65e-1)**
WFG8	3	8.9589e-1 (4.77e-2) =	9.7437e-1 (7.90e-2) -	9.5584e-1 (5.03e-2) -	8.7844e-1 (3.63e-2) =	**8.7122e-1 (3.19e-2) =**	8.7251e-1 (3.67e-2)
	5	2.0576e+0 (7.25e-2) -	2.0106e+0 (7.13e-2) -	2.0516e+0 (1.00e-1) -	2.0319e+0 (9.45e-2) -	1.9678e+0 (6.12e-2) -	**1.8944e+0 (8.32e-2)**
	8	5.4680e+0 (2.55e-1) -	5.4525e+0 (2.07e-1) -	5.7389e+0 (3.91e-1) -	5.5948e+0 (2.63e-1) -	5.3811e+0 (2.71e-1) -	**5.1770e+0 (2.27e-1)**
	10	7.7553e+0 (3.55e-1) =	**7.7392e+0 (2.79e-1) =**	7.7602e+0 (3.94e-1) =	7.8521e+0 (3.01e-1) =	7.7720e+0 (4.16e-1) =	7.8088e+0 (3.47e-1)
	15	1.5647e+1 (4.98e-1) =	**1.5178e+1 (6.85e-1) =**	1.6717e+1 (8.73e-1) -	1.6773e+1 (1.26e+0) -	1.6144e+1 (6.03e-1) -	1.5303e+1 (6.98e-1)
WFG9	3	8.4923e-1 (5.06e-2) =	8.8929e-1 (6.44e-2) -	9.0470e-1 (7.51e-2) -	8.2181e-1 (5.59e-2) =	**8.2175e-1 (4.73e-2) =**	8.2744e-1 (3.75e-2)
	5	2.0032e+0 (1.09e-1) -	1.9074e+0 (1.07e-1) -	2.2163e+0 (1.24e-1) -	1.9670e+0 (8.22e-2) -	1.9564e+0 (9.43e-2) -	**1.7835e+0 (6.87e-2)**
	8	5.6724e+0 (4.50e-1) -	5.3549e+0 (3.06e-1) =	6.0079e+0 (4.61e-1) -	5.5065e+0 (3.11e-1) -	5.4366e+0 (2.41e-1) -	**5.2367e+0 (2.65e-1)**
	10	8.0398e+0 (3.60e-1) =	7.9411e+0 (3.76e-1) =	7.9926e+0 (3.78e-1) =	7.9228e+0 (3.48e-1) =	**7.9093e+0 (3.47e-1) =**	8.1287e+0 (4.59e-1)
	15	1.5903e+1 (6.30e-1) -	1.5568e+1 (1.02e+0) =	1.7329e+1 (8.12e-1) -	1.7060e+1 (1.53e+0) -	1.5949e+1 (9.16e-1) -	**1.5392e+1 (8.30e-1)**
Best/All		1/45	3/45	7/45	4/45	11/45	20/45
Best/Worse/Similar		0/23/22	0/24/21	6/33/6	1/22/22	6/15/24	---

For test problems WFG1–WFG9 with 3 to 15 objectives, as shown in the penultimate row of Table 3, NSGA-III-SE obtained the best results on 20 out of 45 comparisons whereas NSGA-III, MOEA-DD, MOEA-D-DE, KnEA and GrEA performed best on 1, 3, 7, 4, and 11, respectively. Although these compared MOEAs were all redesigned to tackle MaOPs, NSGA-III-SE still achieved the best performance by far. NSGA-III demonstrated superiority only on WFG5 with 10 objectives, it problemed poorly on other test problems. MOEA-DD worked well only on WFG3 with 10 objectives and WFG8 with 10 and 15 objectives. MOEA-D-DE, performed better on WFG4, WFG2 with 8 and 10 objectives, and WFG6 with 10 objectives, demonstrating superiority with the latter. KnEA performed well only on WFG2, WFG3, WFG6, and WFG7 with three objectives. GrEA indicated a clear advantage on WFG1. However, NSGA-III-SE performed well on WFG3 and WFG5–WFG9 problems; it also performed fairly well on WF1, WFG2, and WFG4. In brief, we proposed NSGA-III-SE, which performed better than the other five state-of-the-art algorithms suit MaOPs on the DTLZ test suite.

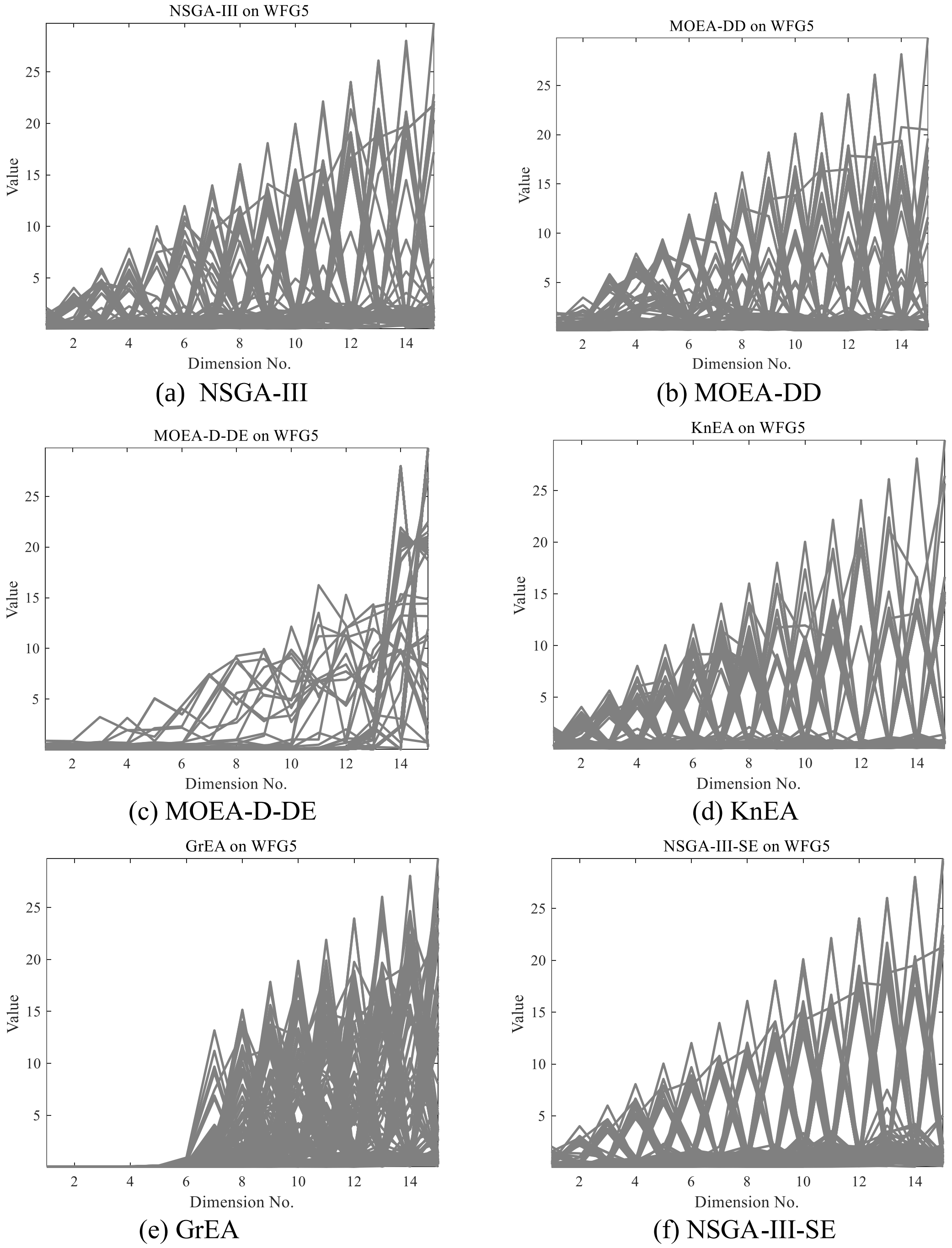
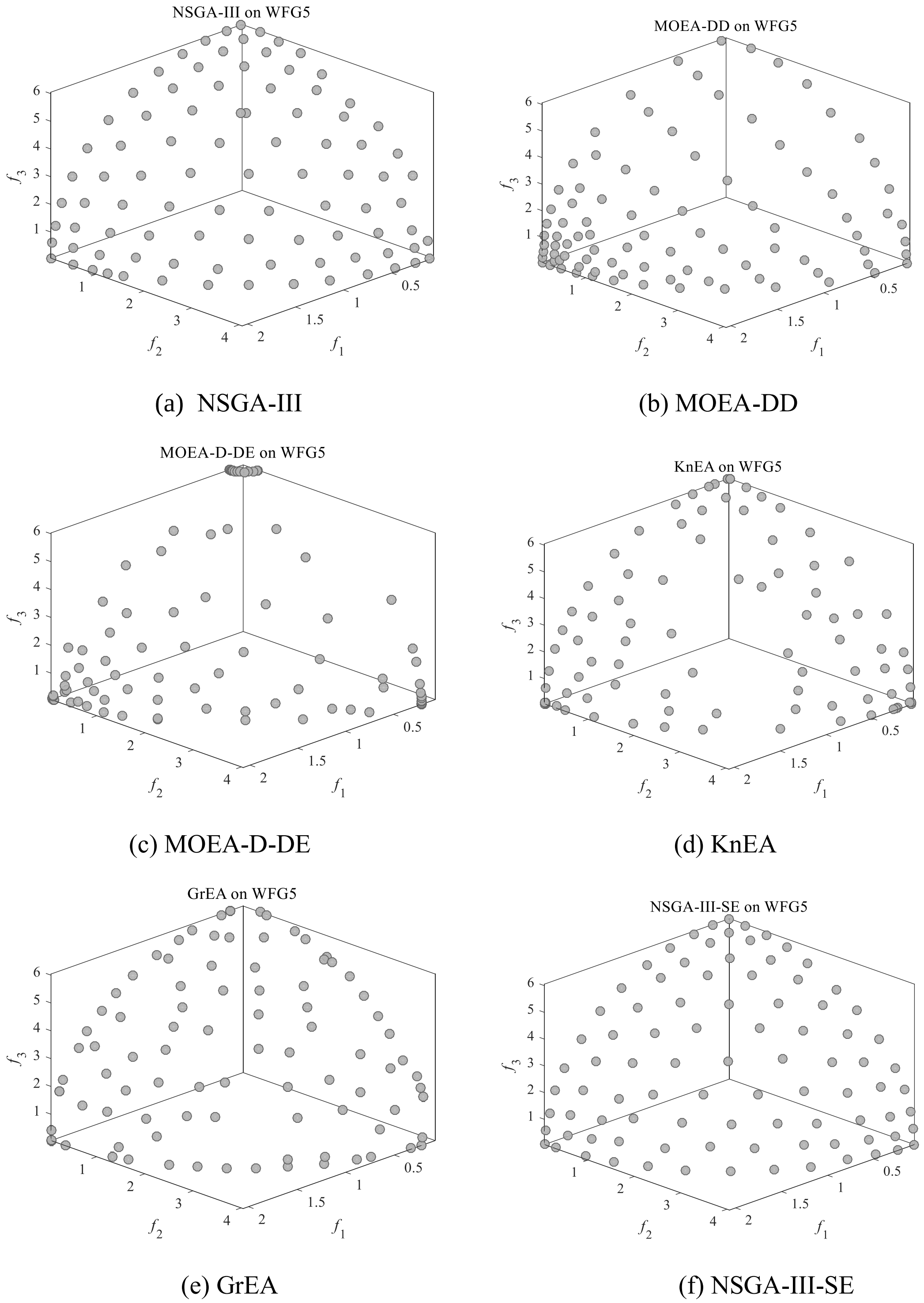


Fig. 5. Parallel coordinates of non-dominated fronts obtained by six algorithms in 15-objectives WFG5 instance

To testify the diversity of the proposed algorithm and visually understand the distribution of solutions, Fig. 5 shows the parallel coordinates of non-dominated fronts obtained by NSGA-III-SE and the other five MOEAs, respectively, for the 15-objective WFG5 instance. This particular run is associated with the result closest to the average

IGD value. The six plots indicate that the non-dominated front of the proposed NSGA-III-SE algorithm is best in convergence and diversity and slightly better than NSGA-III, MOEA-DD, and KnEA. Fig. 5 (c) illustrates that the diversity of MOEA-D-DE is quite poor, as is that of GrEA with 2–6 objectives.



(a) NSGA-III (b) MOEA-DD

(c) MOEA-D-DE (d) KnEA

(e) GrEA (f) NSGA-III-SE

Fig. 6. Plots of non-dominated fronts obtained by six algorithms in the three-objective WFG5 instance

Fig. 6 shows that the NSGA-III and NSGA-III-SE solutions had acceptable diversity because the niche-preservation approach can obtain ideal diversity for algorithms. Moreover, the solutions of MOEA-DD were inclined to one side and sparse on the other.

For MOEA-D-DE, KnEA, and GrEA, the results were disappointing given poor diversity.

5. Conclusion

In this paper, we have presented an improved NSGA-III with a selection-and-elimination operator, called NSGA-III-SE, whose environment selection mechanism is based on simultaneously choosing better solutions and removing worse individuals. The presented PBI distance could consider the convergence and diversity and bring individuals closer to the true PF, thus balancing the convergence and diversity of solutions. To verify validity, we conducted several experiments to compare the proposed NSGA-III-SE with five other popular algorithms using different problems from DTLZ and WFG test suites. Experimental results show that the proposed NSGA-III-SE algorithm performed well in most instances, and the obtained solutions exhibited good convergence and diversity.

In future work, we plan to further enhance the performance of NSGA-III-SE. As characteristics of real-world application problems are often complex, adaptive selection metrics should be tailored to different problems accordingly. In addition, constrained many-objective problems appear to pose a new challenge for MOPs; we will attempt to extend the NSGA-III-SE algorithm to handle these types of problems.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grant No.61806138, No.U1636220 and No.61663028, Natural Science Foundation of Shanxi Province under Grant No.201601D011045, the Open Research Fund of Jiangxi Province Key Laboratory of Water Information Cooperative Sensing and Intelligent Processing No.2016WICSIP005, PhD Research Startup Foundation of Taiyuan University of Science and Technology under Grant No.20182002.

References

[1] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, 6(2):182−197, 2002.

[2] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength pareto evolutionary algorithm. In Proceedings of Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems, pages 95−100, 2001.

[3] E.Zitzler and S.Künzli, Indicator-based selection in multi-objective search, in 8th International Conference on Parallel Problem Solving from Nature, 2004, pp. 832–842.

[4] D. Hadla, J. Herman, et al, An open source framework for many-objective robust decision

making, Environmental Modelling & Software, 2015,74: 114-129.
[5] A. Masood, V. Mei, et al, Many-objective genetic programming for job-shop scheduling, 2016 IEEE Congress on Evolutionary Computation (CEC), 2016, 209 - 216
[6] J. R. Kasprzyk, P. M. Reed, B. R. Kirsch, and G. W. Characklis, Managing population and drought risks using many-objective water portfolio planning under uncertainty," Water Resour. Res., vol. 45, no. W12401, doi:10.1029/2009WR008121, Dec. 2009.
[7] A. López-Jsimes and C.A.CC, Including preferences into a multiobjective evolutionary algorithm to deal with many-objective engineering optimization problems, Information Sciences, 2014, 277(1):1-20.
[8] Ishibuchi H, Tsukamoto N, Nojima Y (2008) Evolutionary many-objective optimization: a short review, IEEE World Congress on Computational Intelligence, 2008, pp 2419-2426.
[9] X. Zou, Y. Chen, M. Liu, and L. Kang, A new evolutionary algorithm for solving many-objective optimization problems, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2008, 38(5), pp. 1402–1412.
[10] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, Combining convergence and diversity in evolutionary multiobjective optimization, Evolutionary Computation, 2002, 10(3), pp. 263–282.
[11] D. Hadka and P. Reed, Borg: An auto-adaptive many-objective evolutionary computing framework, Evolutionary Computation, 2013, 21(2), pp. 231–259.
[12] F. di Pierro, S.-T. Khu, and D. A. Savic, An investigation on preference order ranking scheme for multiobjective evolutionary optimization, IEEE Transactions on Evolutionary Computation, 2007, 11(1), pp. 17–45.
[13] G. Wang and H. Jiang, Fuzzy-dominance and its application in evolutionary many objective optimization, 2007 International Conference on Computational Intelligence and Security Workshops, 2007, pp. 195–198.
[14] K. Deb and J. Himanshu. An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part I: Solving problems with box constraints. IEEE Transactions on Evolutionary Computation, in press.2014, 18(4): 577-601.
[15] Zhang X, Tian Y, Jin Y, A knee point-driven evolutionary algorithm for many-objective optimization. IEEE Trans Evol Comput, 2015, 19: 761-776.
[16] Cheng J, Yen GG, Zhang G A many-objective evolutionary algorithm with enhanced mating and environmental selections. IEEE Trans Evol Comput, 2015, 19:592–605.
[17] Zhang Q, Li H MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput, 2007, 11:712–731.
[18] Asafuddoula M, Ray T, Sarker R, A decomposition-based evolutionary algorithm for many objective optimization evolutionary computation. IEEE Trans, 2015, 19:445–460.
[19] Yuan Y, Xu H, Wang B, Zhang B, Yao X, Balancing convergence and diversity in decomposition-based manyobjective optimizers. IEEE Trans Evol Comput,2016, 20:180–198.
[20] Li K, Deb K, Zhang Q, Kwong S, An evolutionary many-objective optimization algorithm based on dominance and decomposition. IEEE Trans Evol Comput 2015,19:694–716.
[21] Yuan, Xu and Wang, An improved NSGA-III Procedure for Evolutionary Many-Objective Optimization, Proceeding of the 2014 conference on genetic and evolutionary computation. ACM, 2014, pp 661-668, dio:10.1145/2576768.2598342.
[22] Bi and Wang, An improved NSGA-III algorithm based on objective space decomposition for

many-objective optimization, Soft Compute, 2017,21:4269-4296, dio:10.1007/s00500-016-2192-0.

[23] N. Srinivas and K. Deb, Multiobjective optimization using nondominated sorting in genetic algorithms, Evol. Comput., 1994, 2(3):221-248.

[24] K. Miettinen, Nonlinear Multiobjective Optimization. Norwell, A:Kluwer, 1999.

[25] K. Deb and R. B, Agrawal. Simulated binary crossover for continuous search space. Complex Systems, 9:1-34,1994.

[26] Maha Elarbi and Slim Bechikh, A New Decomposition-Based NSGA-II for Many-Objective Optimization, IEEE Transactions on Systems, Man and Cybernetics: Systems, 48(7):1191 – 1210, 2017.

[27] I. Das and J. E.Dennis. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. SIAM Journal on Optimization, 8(3):631-657,1998.

[28] Li H, Zhang Q, Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II, IEEE Transactions on Evolutionary Computation, 2009, 13(2), pp 284-302.

[29] Yang et al. A Grid-Based Evolutionary Algorithm for Many-Objective Optimization, IEEE Transactions on Evolutionary Computation, 2013, 17(5), 721-736.

[30] Deb K, Thiele L, Laumanns M, Zitzler E, Scalable test problems for evolutionary multiobjective optimization. In: Abraham A, Jain L, Goldberg R (eds) Evolutionary multiobjective optimization: advanced information and knowledge processing. Springer, London, 2005, pp 105–145.

[31] Huband S, Hingston P, Barone L, While L, A review of multiobjective test problems and a scalable test problem toolkit. IEEE Trans Evolut Comput , 2006, 10:477–506.

[32] Li B, Tang K, Li J, et al. Stochastic Ranking Algorithm for Many-Objective Optimization Based on Multiple Indicators[J]. IEEE Transaction on Cybernetics, 2014, 44(12):2391-2404.

[33] Zitzler E, Thiele L, Laumanns M, Fonseca CM, Da Fonseca VG (2003) Performance assessment of multiobjective optimizers: An analysis and review. IEEE Trans Evol Comput 7:117–132.