Information Sciences 679 (2024) 121032

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Two-stage particle swarm optimization with dual-indicator fusion ranking for multi-objective problems

Qing Xu [a,1], Yuhao Chen [a,1], Cisong Shi [a,1], Junhong Huang [b,1], Wei Li [a,*]

[a] School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou 341000, China
[b] Department of Computer Science and Technology, Jilin University, Changchun 130012, China

ARTICLE INFO

Keywords:
External archive maintenance
Two-stage hybrid mutation
Valuable information fusion
Global leader selection

ABSTRACT

Elite solutions guiding population evolution are often used as one of main ideas to improve the performance of multi-objective particle swarm optimization (MOPSO). However, in most research work, sole Pareto dominance criterion is often used to evaluate solutions. This sole criterion may easily cause some problems, such as the premature convergence. In this study, we propose an MOPSO variant with dual-indicator fusion ranking (TPSO-DF), to evaluate elite solutions and to guide search without sacrificing diversity. In TPSO-DF, two indicators are introduced by using the convergence and diversity information, respectively. Both indicators are then fusioned in a ranking measure to focus on valuable information and to filter out solutions with these valuable information. Meanwhile, an adaptive global leader selection strategy is introduced to take full advantage of valuable information and to guide population evolution toward the optimal direction. As another contribution of this study, a two-stage hybrid mutation strategy is designed by utilizing the valuable information differently in different evolutionary states of the algorithm to enhance performance. Compared to eight representative multi-objective evolutionary algorithms, the performance of TPSO-DF is validated by extensive experiments on ZDT and DTLZ test suites, as well as one practical problem. Experimental results show that TPSO-DF can achieve competitive performance on most of the test functions.

1. Introduction

Most problems in our real-world are with multi objectives conflicting with each other. These problems are often named as multi-objective optimization problems (MOPs) [1,2] which are often much harder to be solved than single-objective problems. Recently, as an efficient technique to solve MOPs, evolutionary algorithms (EAs) have been widely adopted to solve real-world optimization problems in various fields, such as academic and industry [3,4]. Derived from EAs, multi-objective evolutionary algorithms (MOEAs), e.g., particle swarm optimization (PSO) [5], genetic algorithm (GA) [6] and differential evolution (DE) [7], have been popular in solving various MOPs, including real-world problems, for example robotics, logistics and networks in industrial areas.

Among these MOEAs, MOPSO has been widely applied due to its simple structure and efficient computation [8]. However, when solving MOPs, being similar with other evolutionary algorithms, MOPSO also suffers from the issues of external elitist archive main-

* Corresponding author.
E-mail addresses: xuqing@jxust.edu.cn (Q. Xu), vergil.n.jose@jxust.edu.cn (Y. Chen), 1032375864@qq.com (C. Shi), huangjh23@mails.jlu.edu.cn (J. Huang), liwei@jxust.edu.cn (W. Li).

[1] Contributing authors.

https://doi.org/10.1016/j.ins.2024.121032
Received 1 February 2024; Received in revised form 10 May 2024; Accepted 11 June 2024

Available online 21 June 2024
0020-0255/

tenance or global leader selection mechanism. The main reasons lie in the external archive maintenance to store optimal solutions and the global leader selection to direct solutions to approximate the true Pareto font (PF). During the evolutionary process facing multiple objectives, it is difficult to adaptively evaluate valuable information and the quality of solutions with sole Pareto dominance criterion [5,9]. In recent years, many MOPSO variants [10–13] have been designed with enhanced strategies, such as elitism, niching and archives [5,9], to achieve an optimal trade-off between convergence and diversity. Most of these variants share the similar idea of utilizing multiple external archives [14,15] or separating convergence global leaders and diversity ones [16].

It is obvious that experimental results in related research verified that the performance of MOPSO can be efficiently improved by diverse archives and global leaders. Nevertheless, it should be aware that valuable information is still unclear in most work. In addition, during the evolution process, the number of nondominated solutions can increase sharply and diverse archives may cause extra expense [5]. Besides, when facing with multiple objectives, it can be difficult to measure the solution crowdedness by simple crowding distance criterion [17]. Moreover, the above improved strategies search for solutions in sparse regions, which is likely to guide the final solutions far from the true PF [18]. Since without clear valuable information definition, most current archives are difficult to constantly maintain solutions while well balancing between convergence and diversity, as well as to obtain uniformly distributed optimal Pareto solution set.

Based on the above observations, a new MOPSO variant is constructed based on dual-indicator fusion ranking, which is abbreviated as TPSO-DF. In the TPSO-DF, two indicators are introduced by using the convergence and diversity information, respectively. Both indicators are then fusioned in a ranking measure to focus on valuable information and to filter out solutions with these valuable information. It is worth noting that an adaptive global leader selection strategy is also introduced aiming to properly use valuable information and to guide population evolution toward the optimal direction. Moreover, a two-stage hybrid mutation strategy is designed by utilizing the valuable information differently in different evolutionary states of the algorithm in order to achieve an optimal trade-off between convergence and diversity, as well as to further enhance the algorithm robustness. Compared to other eight well-established MOEA variants, the performance of the TPSO-DF is validated by extensive experiments on ZDT and DTLZ test suites, as well as one practical problem. The experimental results confirm the competitive performance of our TPSO-DF.

The rest of this paper is mainly divided into 4 sections. Basic MOPSO, as well as some related work, are introduced briefly in Section 2. The proposed TPSO-DF is illustrated in detail in Section 3. Next, in Section 4, the experiments and corresponding discussions are analyzed. Finally, this work is concluded in Section 5.

2. Basic MOPSO and related work

2.1. Basic MOPSO

Multi-Objective particle swarm optimization (MOPSO) algorithm is built from particle swarm optimization (PSO) to solve multi-objective optimization problems (MOPs) with several minimization or maximization objective functions conflicted with each other. Take a minimization problem as an example, it can be mathematically presented as:

$$\min_{x\in\Omega} F(x) = \left\{f_1(x), f_2(x), \cdots, f_m(x)\right\} \tag{1}$$

$$s.t. \quad g_j(x) \le 0 \quad 0 \le j \le k \tag{2}$$

where $x = (x_1, x_2, \cdots, x_D)^T$ represents a D dimensional decision vector defined in objective space Ω. $g_j(x) \le 0$ denotes k constraints, including equality and inequality ones. $f_i(x)$ represents the ith objective function which is one of m total functions. Pareto set (PS) collects a set of trade-off solutions found by the MOP. These solutions are mapped to objective space, called Pareto Front (PF).

It is difficult to give the total order relation between solutions due to conflicted objective functions. Thus, specific order relationship based on partial ranking should be defined. Some Pareto-related notions are clarified.

Definition 1 *(Pareto dominance)*. For solutions X and Y, X is considered to Pareto dominate Y (expressed by X $\prec$Y), if the following condition is true:

$$\forall i \in (1, 2, \cdots, m) : f_i(X) \le f_i(Y) \tag{3}$$

$$\exists i \in (1, 2, \cdots, m) : f_i(X) < f_i(Y) \tag{4}$$

Definition 2 *(Pareto optimal)*. Solution X is seen as Pareto optimal in the objective space, if

$$\neg\exists Y \in \Omega : Y \prec X \tag{5}$$

Definition 3 *(Pareto set)*. Trade-off solutions found by the MOP are collected in PS.

$$PS = \{X | X \in S,\ S\ is\ the\ set\ of\ all\ the\ Pareto\ optimal\ solutions\} \tag{6}$$

Definition 4 *(Pareto front)*. PF stores objective vectors mapped by solutions in PS.

$$PF = \{F(X) | X \in PS\} \tag{7}$$

In basic PSO, a set of solutions is firstly initiated. The swarm is then evolved through iteration to find optimal solutions. During the evolutionary process, every solution has its own velocity and position updated by personal best experience based on its own knowledge and global best experience to approximate the optimum position of the population in history [9]. Suppose that $x_i(t) = (x_{i,1}(t), x_{i,2}(t), ..., x_{i,D}(t)), v_i(t) = (v_{i,1}(t), v_{i,2}(t), ..., v_{i,D}(t))$ are respectively the position and velocity of ith solution at tth generation with always D dimensions. They are then updated as Eq. (8) and Eq. (9):

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1r_1(pb_{i,j}(t) - x_{i,j}(t)) + c_2r_2(gb_{i,j}(t) - x_{i,j}(t)) \tag{8}$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \tag{9}$$

where $Pb_i(t) = (pb_{i,1}(t), pb_{i,2}(t), ...pb_{i,D}(t))$ and $Gb_i(t) = (gb_{i,1}(t), gb_{i,2}(t), ...gb_{i,D}(t))$ denote respectively the position of personal best and global best experience for ith solution. r_1 and r_2 are two numbers randomly generated between [0,1]. In addition, learning factors c_1 and c_2, as well as the inertial weight ω are the key flight parameters to control balancing between exploration and exploitation [19]. ω is used to determine the preservation degree of the previous velocity of a particle. c_1 and c_2 are used to control the influence of self-cognitive $(Pb_i(t))$ and social component of a particle $(Gb_i(t))$. When c_1 and ω are large, c_2 is small, the swarm is more likely enhance global exploration by enlarging the search scope. Inversely, if c_1 and ω are small, c_2 is large, the population tends to improve local exploitation.

MOPSO, which is built from PSO for MOPs, differs from PSO in two points. Firstly, MOPSO contains the process of external archive construction and maintenance to store the positions of Pareto nondominated solutions. Secondly, the selection of $Gbest$ and $Pbest$ is completely different in MOPSO. Compared with other paradigms of MOEAs, MOPSO has a great advantage due to its simple structure but yet efficient computation.

2.2. MOPSO variants

Although the basic MOPSO has shown the priority to solve MOPs, there is still much space to solve complex optimization problems. The main issues lie in the external archive maintenance and the global leader selection which aim to obtain optimized results that approximate the true PF while distributing them as evenly as possible in the PF. These problems are caused by the difficulty in defining valuable information and solutions carrying valuable information with sole Pareto dominance criterion, as well as in selecting right global leader to guide population evolution toward the optimal direction. Accordingly, most of MOPSO variants focus on (1) maintaining multiple external archives and (2) selecting global leader by utilizing some elite solutions.

(1) Introduction of multiple external archives

Stimulated by balancing convergence and diversity, as well as the performance degradation of MOEAs with the increase of the problem complexity, Praditwong and Yao [20] proposed using two archives to separate convergence (CA) and diversity (DA). CA collects new nondominated solutions dominating other solutions in archives. DA stores other nondominated solutions. A dual-archive algorithm (ITAA) is proposed with a truncation strategy [21]. Cai [22] proposed a ranking method to sort the solutions in a population for MOEAs. Dai [23] incorporated a multi-search strategy based on dual-archive scheme to increase diversity. Besides, other strategies, including the update of members and the delete of useless solutions, are designed to make dual-archive algorithms more robust [15,24–27].

(2) Global leader selection

In addition to the external archive maintenance, the global leader selection is another popular academic subject. Sierra et al. [28] proposed a new optimizer with the list of available leaders by applying Pareto dominance and a crowding factor. Wu et al. [5] proposed the MOPSO (AMOPSO/ESE) with an evolutionary state estimation strategy to adaptively differentiate exploitation state and exploration one. Recently, based on adaptive MOPSOs with candidate estimation, Han et al. [29] proposed an ACE-MOPSO to balance convergence and diversity in fine, and finally to make MOPSO more robust.

(3) Other relevant MOPSO variants

In addition to multiple external archives and global leader selection, there are some other relevant MOPSO variants. For example, Deb et al. [30] pointed out that the problem of poor-distributed approximation set is due to the neglect of crowding distance, so they introduced a crowding distance technique, called NSGA-II. To reduce nondominated solutions in archive, Laumanns et al. [31] proposed a ϵ-dominance method with a relax domination degree. Recently, based on reference-based method, Deb et al. [32–34] incorporated some reference vectors in their proposed variant to achieve an even-distributed Pareto optimal solutions. Nevertheless, these MOPSO variants perform well largely because the even-distributed shapes of predefined reference vectors are quite similar to the PF of benchmark functions (i.e., DTLZ1-4 and WFG4-9).

3. Our approach

In this section, firstly, the motivation of TPSO-DF is analyzed. Then, the main framework and detailed procedure are described. Last, the computational complexity of TPSO-DF is briefly illustrated.

3.1. Motivation

Recently, MOPSOs have been widely applied to solve MOPs for the simple structure yet efficient computation. But the performance of most MOPSOs has still much space, including difficult global leader selection and poor balance of convergence and diversity,

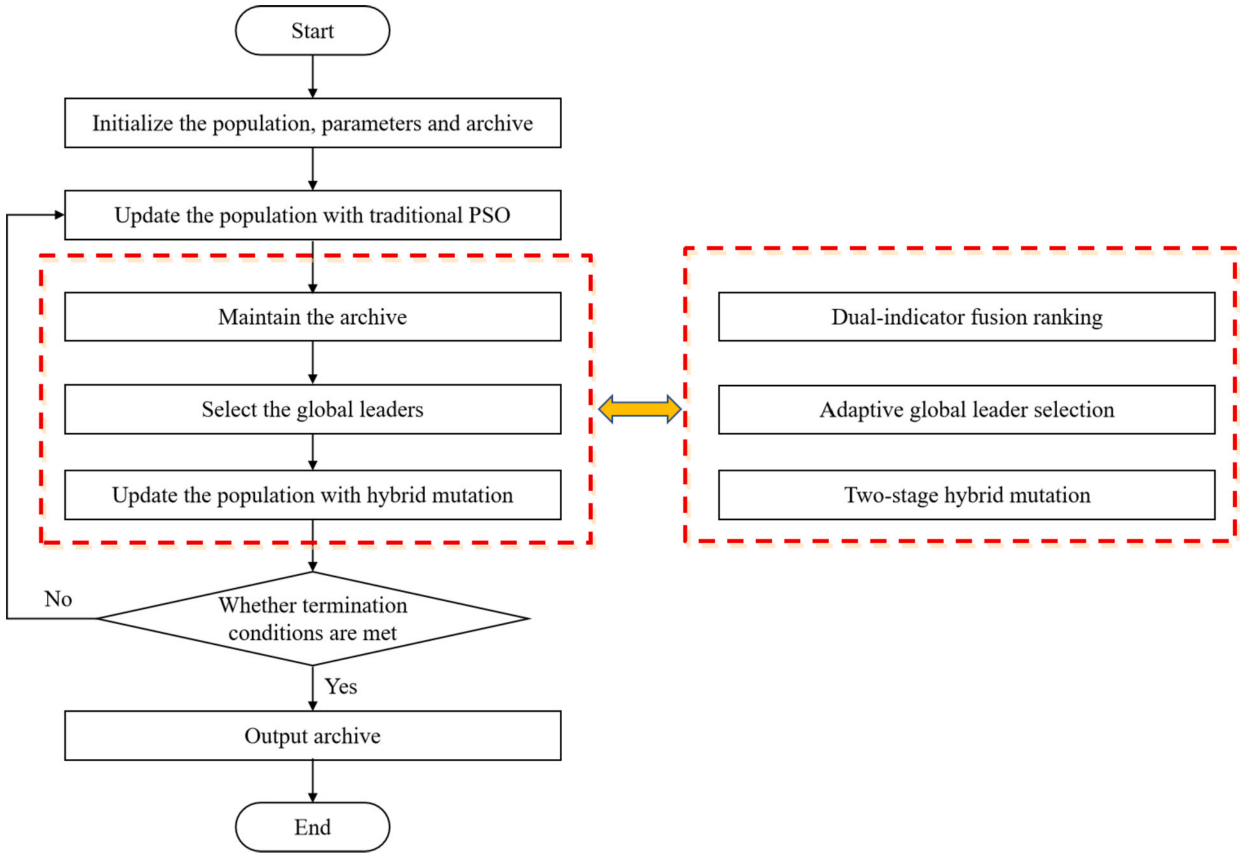

Fig. 1. The flowchart of proposed TPSO-DF.

especially for complex optimization problems like multimodal problems. This drawback is mainly due to the difficulty in diversifying solutions in the external archive and in evaluating solution quality with sole Pareto dominance criterion. From the above MOPSO description, the performance of MOPSO is seriously relied on the selection of an appropriate leader in the external archive, since the selected leaders determine the swarm evolution direction.

Many MOPSO variants have been proposed so far to enhance MOPSO performance. Most of these variants focus on selecting the global leader with Pareto dominance or some elite solutions from which the valuable information is beneficial to improve exploitation and pick up convergence speed. However, if the information is not taken full advantage of use, the population evolution may be guided to improper directions or the algorithm would become greedy for too strong exploitation. Based on these observations, we are inspired to consider: (1) how to define valuable information and thus efficiently find solutions with valuable information to well utilize the convergence information without sacrificing the diversity, (2) how to select an optimal global leader to properly use valuable information and to guide population evolution toward the optimal direction, and (3) how to better dynamically balance between exploration and exploitation and to finally further improve the algorithm performance.

3.2. Main framework of TPSO-DF

Fig. 1 illustrates the framework of our proposed TPSO-DF. Compared to fundamental MOPSO, TPSO-DF is innovated mainly due to the incorporation of the dual-indicator fusion ranking measure in the external archive maintenance, the adaptive global leader selection and the two-stage hybrid mutation strategy. According to the dual-indicator fusion ranking measure, two cooperative indicators, named CR aiming to approach the true PF and DR aiming at achieving uniform solution distribution, as well as a fusioned comprehensive indicator FR balancing between convergence and diversity by the ratio of CR and FR, are developed in TPSO-DF. Between the population and the external archive, there exists also a bi-directional communication. Moreover, the adaptive global leader selection strategy and the two-stage hybrid mutation strategy can help the algorithm to better balance between exploration and exploitation.

Algorithm 1 shows the pseudocode of TPSO-DF which begins by initiating the population P and external archive A (line 1-2). Global and personal optimal solutions are also randomly distributed in the first iteration (line 3-4). Then, the original MOPSO operations, as well as improved techniques, e.g., population updating (line 7), external archive maintenance (line 8), adaptive selection of global leader (line 10) and two-stage hybrid mutation (line 11) are applied in each generation. When the terminal condition is reached, solutions in the external archive are stored as final results (line 13).

Algorithm 1 Overall framework of TPSO-DF.

Input: Population size N_p, population P, fixed external archive size N_e, external archive A.
Output: External archive A.
P = Init(N_p);
A = Init(A, P, N_e);
$Pbest$ = Init(P);
$Gbest$ = RandInit_Gbest(N_p, P);
$[\omega, c_1, c_2]$ = RandInit_Para(N_p);
while Termination condition is not reached **do**
  P = UpdatePopulationPSO($P, Gbest, Pbest, \omega, c_1, c_2$)(Eq. (8)-(9));
  A = UpdateA(P, non-dominated solutions) (Algorithm 3);
  $Pbest$ = UpdatePbest($P, Pbest$);
  $Gbest$ = Gbest_selection(N_p, N_e, A) (Algorithm 4);
  P = UpdatePopulationTwo-stage(P) (Algorithm 5);
end while
return A.

To better explain TPSO-DF, each component as well as the computational complexity are described in the following parts.

3.3. Details of TPSO-DF

This section details the major components of TPSO-DF, including dual-indicator fusion ranking for external archive maintenance, adaptive global leader selection and two-stage hybrid mutation strategy.

3.3.1. Dual-indicator fusion ranking for external archive maintenance

External archive A, where stores optimal solutions, is an essential element of MOPSO. An improper external archive maintenance, such as sole criterion based on Pareto dominance, can mislead global leader selection, and finally badly direct population evolution due to too fast convergence speed as well as lose of diversity. Hence, in order to select high quality leaders and to direct solutions to approximate the true Pareto font (PF), it is essential to evaluate solutions by simultaneously taking convergence and diversity into account. Based on the above motivation, two indicators using the convergence and diversity information, respectively, as well as a ranking measure fusionning this information are proposed in our modified external archive maintenance. Both indicators and the measure are introduced in detail as follows.

To shorten the distance between solutions and the true PF of MOP, a convergence-related ranking indicator CR, calculating as in equation (10)-(11), is applied to nondominated solutions based on the similarity (measured by Euclidean distance) of the solution to the ideal point z^* of the MOP.

$$CR_{x\in A}(x) = \sum_{y\in A} ED(x,y) \tag{10}$$

$$ED_{x,y\in A}(x,y) = \begin{cases} 0, & \text{if } \sum_{j=1}^{M}(z_j^* - f_j(x))^2 < \sum_{j=1}^{M}(z_j^* - f_j(y))^2 \\ 1, & \text{otherwise} \end{cases} \tag{11}$$

where $ED(x,y)$ implies whether the euclidean distance of solution y or that of x to the ideal point z^* is closer. If solution x to z^* is closer than that of y, it means that x is more similar to the ideal point, then $ED(x,y)$ is assigned 0. Else, $ED(x,y)$ is assigned 1. $CR(x)$ gives the convergence ranking of solution x, that is to say among all other solutions in the archive, the number of solutions who are better than x.

The ideal point $z^* = (z_1^*, z_2^*, ..., z_M^*)$, obtained by calculating the minimum value z_j^{min} of each objective function f_j in the external archive A, is calculated by $z_j^* = min_{x\in A} f_j(x), \quad j = 1, 2, ..., M$, where x denotes the nondominated solution in A.

In order to maintain maximum high quality solutions without sacrificing diversity, DR is applied to compare the diversity of nondominated solutions by using crowding distance. DR is calculated as in equation (12)-(13).

$$DR_{x\in A}(x) = \sum_{y\in A} CCD(x,y) \tag{12}$$

$$CCD_{x,y\in A}(x,y) = \begin{cases} 0, & \text{if } \quad CD(x) < CD(y) \\ 1, & \text{otherwise} \end{cases} \tag{13}$$

where $CCD(x,y)$ implies whether the crowding distance of solution y is smaller or that of x is smaller. $DR(x)$ gives the diversity ranking indicator of solution x, that is to say among all solutions in the archive, the number of solutions who have less diversity than x.

The pseudocode for calculating crowding distance (CD) which measures whether nondominated solutions are in crowding areas or not [30], is shown in Algorithm 2. $CD(x)$ accumulates the crowding distance of solution x on all objectives (see lines 1-10). The larger is this $CD(x)$, the better is the diversity of the solution x. Specifically, for each objective, an infinite number is given to the CD of boundary solutions who are the smallest or largest among all solutions. In this way, boundary solutions have always the most diversity to guide solutions to extend the territory of the approximate PF.

Algorithm 2 Crowding distance (CD) calculation.

Input: External archive A.
Output: CD of each solution x in the external archive A.

```
for i = 1 to |A| do
    CD(x_i) = 0;
end for
for j = 1 to M do
    Sort A by jth objective;
    CD(x_1) = CD(x_|A|) = ∞;
    for i = 2 to |A| − 1 do
        CD(x_i) = CD(x_i) + (f_j(x_{i+1}) − f_j(x_{i−1})) / (f_j(x_|A|) − f_j(x_1));
    end for
end for
return CD of each solution x in the external archive A.
```

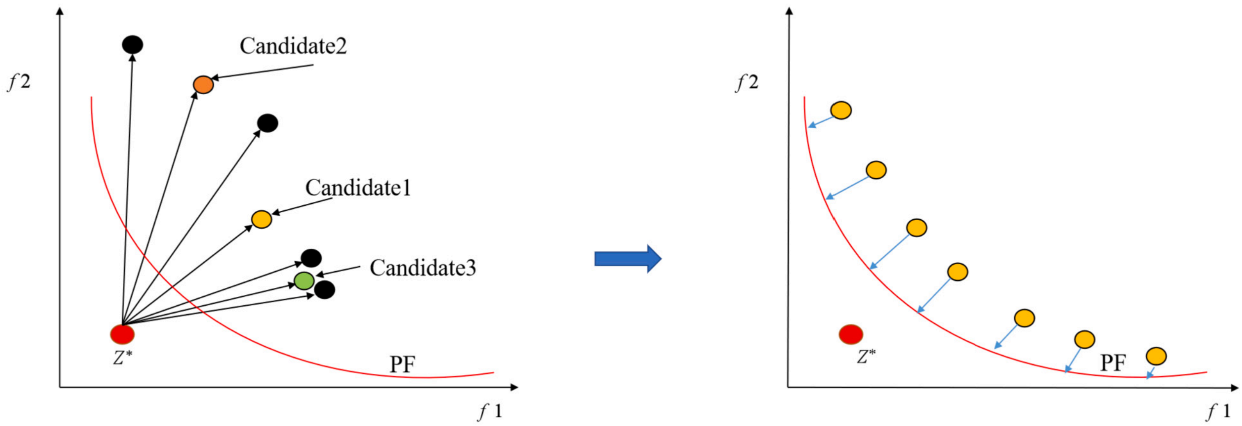

Fig. 2. Selecting solutions based on dual-indicator fusion ranking.

After obtaining CR and DR of all solutions in the archive, the dual-indicator fusion ranking measure FR, identifying comprehensively the dominant relationship between two solutions, is calculated by equation (14).

$$FR_{x\in A}(x) = \frac{CR(x)}{DR(x)} \tag{14}$$

If a solution is poorly diverse (with small DR) and poorly convergent (with large CR), it is assigned a large FR ranking and has a higher probability of being removed from the archive A. The selection of elite solutions based on dual-indicator fusion ranking measure is shown in Fig. 2. Candidate2 has good diversity performance but not enough convergence. Candidate3 has strong convergence, but the attachment aggregates other solutions and thus has poor diversity. Candidate1 has both good convergence and diversity, so it should be kept as the optimal solution. This allows for more even distribution of retained individuals while approaching PF.

The process of dual-indicator fusion ranking mechanism for external archive maintenance is as follows. First of all, a fixed size N is preset for the archive, adding non-dominated solutions from the population to A. When the archive overflows in the evolutionary process, sort the solutions in the archive according to the FR ranking and randomly remove one solution from the bottom 10% of solutions with largest FR (a slight fluctuation should be beneficial for the diversity of the swarm. Specifically, bottom 10% of solutions with largest FR is set in this work). As a consequence, a predefined number of elite solutions with high quality are stored in A. The specific implementation process is shown in Algorithm 3.

Algorithm 3 Dual-indicator fusion ranking for external archive maintenance.

Input: Initial external archive A, population P.
Output: External archive A after maintenance.

```
A = nondominated solutions in P;
while length(A) > N do
    Calculate Z*;
    Calculate CR of each solution in A (Eq. (10)-(11));
    Calculate CD of each solution in A (Algorithm 2);
    Calculate DR of each solution in A (Eq. (12)-(13));
    Calculate FR of each solution in A (Eq. (14));
    Randomly remove one solution from the bottom 10% solutions in A with largest FR;
end while
return A.
```

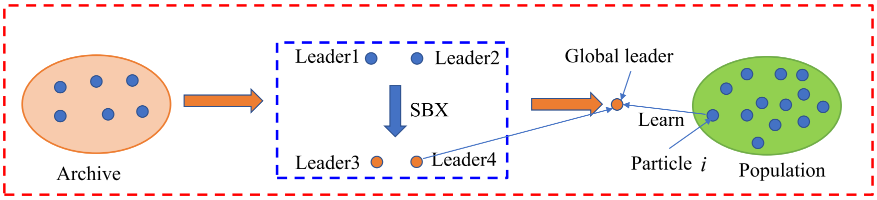

Fig. 3. Global leader selection when fewer solutions on the archive.

3.3.2. Adaptive global leader selection strategy

In MOPSO, in order to guide population to evolve toward the true PF, it is vital to propose a proper global leader selection strategy. Traditional global leader selections have some drawbacks. For example, sole Pareto-dominance-defined global leader can lead to severe lose of diversity while global leader based on random selection in external archive can lead to convergence problem [35,36]. Therefore, in order to properly use valuable information and to take convergence as well as diversity into account simultaneously, we propose an additional adaptive global leader selection based on valuable information fusion in which each solution has its own global leader. Furthermore, genetic algorithm-based operations are incorporated in order to improve performance.

According to the A size, global leader is selected either from generated parental solutions or directly from the updated A. If the number of solutions in A does not attain the fixed size N_e yet, to properly make full use of selected elite solutions, the operator simulated binary crossover (SBX) [37] operator of genetic algorithm is adopted to generate parental solutions. The details of SBX are briefly introduced below.

Suppose that $x_{i_1} = (x_{i_1 1}, x_{i_1 2}, \dots, x_{i_1 D})$ and $x_{i_2} = (x_{i_2 1}, x_{i_2 2}, \dots, x_{i_2 D})$ are respectively two parents randomly selected in the archive A, then the SBX operator is presented as

$$\begin{aligned} x^o_{p_1} &= 1/2(x_{i_1} + x_{i_2}) - 1/2\beta(x_{i_1} - x_{i_2}) \\ x^o_{p_2} &= 1/2(x_{i_1} + x_{i_2}) + 1/2\beta(x_{i_1} - x_{i_2}) \end{aligned} \tag{15}$$

where $x^o_{p_1}$ and $x^o_{p_2}$ represent two offsprings generated after SBX operation. Parameter β involved is calculated as

$$\beta = \begin{cases} (2\mu)^{\frac{1}{1+\eta_c}}, & \text{if} \quad \mu \le 0.5 \\ (2-2\mu)^{\frac{-1}{1+\eta_c}}, & \text{otherwise} \end{cases} \tag{16}$$

where μ denotes a number randomly generated between [0,1], and η_c denotes the crossover index which is artificially defined. A larger η_c indicates a higher probability that an offspring solution is closer to the parent. Here η_c is set to the common 20. After the above operation, each solution includes four candidate leaders in the selection of the global leader, two parent solutions selected from the archive, and two offspring solutions generated through the SBX. To ensure the diversity of global leaders, a solution randomly selects one of its four candidate leaders to be the global leader for that solution. This process is shown in Fig. 3.

When the number of solutions in A attains N_e, offsprings are likely to be guided to regions far from the PF by some poorer candidates selected from the whole archive. Therefore, solutions with good FR ranking (Specifically, solutions with 50% best FR is set in this work) are selected from the archive as candidates for global leaders.

The pseudocode to select global leaders is shown in Algorithm 4 to incorporate the above operations. For a population with N_p solutions, two situations are identified. When the size of A is still small than its fixed size N_e, it needs more diversity. Here two solutions are randomly selected from A as parents (line 3). Two candidates are then generated by the SBX operation (line 4) which is derived from the generated parents according to Eq. (15). The global leader is randomly generated from two selected solutions and two generated candidates (line 5). If the size of A attains its fixed size N_e, it needs more convergence. Here A is firstly sorted by the fusion ranking mechanism FR (line 8). Then the global leader is randomly selected from the first half of sorted A (line 10) for each solution. Hence, each solution has its own global leader.

3.3.3. Two-stage hybrid mutation strategy

MOPSOs use the PSO operator in updating the population, each dimension of the velocity and the position of a solution can be updated with Eq. (8) and Eq. (9). As shown in the above analysis, PSO operator has some drawbacks [38]. On the one hand, the update of velocity is largely influenced by the previous population experience, which will lead to the lack of exploration ability in the early evolution process. On the other hand, the local exploitation of the PSO operator is poor, which affects the convergence efficiency in the late evolutionary stage. To alleviate these drawbacks and to properly use selected elite solutions in different stages, a two-stage hybrid mutation strategy is proposed by adopting polynomial-based mutation (PM) [5] and differential mutation (DM) operators.

In order to keep solution convergence without sacrificing diversity in the early exploration stage (here is set to first 80% of total iterations), offsprings are generated based on the PM operation $x^o_{p_i} (i = 1, 2, \dots, N_p)$ with a mutation probability p_m. For the solution $x^o_{p_i} = (x^o_{p_i 1}, x^o_{p_i 2}, \dots, x^o_{p_i D})$, the PM operation is presented as Eq. (17).

Algorithm 4 Gbest_selection(N_p, N_e, A).

Input: Population size N_p, fixed external archive size N_e, external archive A.
Output: Global leader $Gbest$ for each solution.

```
if length(A) < N_e then
    for i = 1 to N_p do
        leader1, leader2 ← Randomly select two solutions from A;
        leader3, leader4 ← SBX(leader1, leader2)
        Gbest_i ← Select one solution randomly from leader1, leader2, leader3, leader4;
    end for
else
    A ← Sort A according to FR;
    for i = 1 to N_p do
        Gbest_i ← Select one solution randomly from the first half of sorted A;
    end for
end if
return Gbest_i of each solution in the external archive A.
```

$$x^o_{p_i j} = \begin{cases} x^o_{p_i j} + \xi_{ij} \times (u_j - l_j), & \text{if} \quad rand(0,1) \le p_m \\ x^o_{p_i j}, & \text{otherwise} \end{cases} \tag{17}$$

where u_j and l_j represent respectively the highest and lowest value on the jth dimension. ξ_{ij} is defined by Eq. (18)

$$\xi_{ij} = \begin{cases} (2\gamma + (1-2\gamma)(1 - \frac{x^o_{p_i j} - l_j}{u_j - l_j})^{(1+\eta_m)})^{\frac{1}{1+\eta_m}} - 1 & \text{if} \quad \gamma \le 0.5 \\ 1 - (2\gamma + (1-2\gamma)(1 - \frac{u_j - x^o_{p_i j}}{u_j - l_j})^{(1+\eta_m)})^{\frac{1}{1+\eta_m}}, & \text{otherwise} \end{cases} \tag{18}$$

where γ is a parameter randomly distributed in [0,1] with uniform, η_m denotes the mutation distribution index.

In the late exploitation stage of the evolution, it is necessary to enhance the local exploitation and thus pick up the speed of convergence. In our work, a differential mutation (DM) operation based on valuable information of leaders is applied to further improve the solution quality and accelerate convergence in the late evolutionary process. For the solution $x^o_{p_i} = (x^o_{p_i 1}, x^o_{p_i 2}, \dots, x^o_{p_i D})$, the DM operator is presented as Eq. (19).

$$x^o_{p_i j} = \begin{cases} x^o_{p_i j} + r \times (gb_{p_i j} - x^o_{p_i j}) + (1-r) \times (pb_{p_i j} - x^o_{p_i j}), & \text{if} \quad rand(0,1) \le 0.5 \\ x^o_{p_i j}, & \text{otherwise} \end{cases} \tag{19}$$

where r denotes a random number between [0,1], $gb_{p_i j}, pb_{p_i j}$ are the global leader and the historical personal optimal position for solution $x^o_{p_i}$ on the jth dimension, respectively.

The main processes of the two-stage hybrid mutation mechanism are as follows. In the first stage (specifically, 0.8*maximum generations is set in this work), after updating the population using the PSO operator, the PM mechanism is introduced according to Eq. (17) to enhance the global exploration ability, and the DM mechanism is introduced in the second stage to enhance the local exploitation ability. The implementation details are shown in Algorithm 5, where FEs represents the used number of fitness evaluations. $MaxFEs$, denoting the highest value of FE, is used to terminate the algorithm.

Algorithm 5 UpdatePopulationTwo-stage(P).

Input: Population P, population size N_p.
Output: Updated population P.

```
if FEs < 0.8 * MaxFEs then
    for i = 1 to N_p do
        Calculate x^o_{p_i j} by Eq. (17);
    end for
else
    for i = 1 to N_p do
        Calculate x^o_{p_i j} by Eq. (19);
    end for
end if
return P.
```

3.4. Complexity analysis

As shown in Algorithm 1, to calculate the computational complexity, in one generation, apart from the update of velocity and position, the main components including the external archive updating, the selection of global leader and the two-stage hybrid mutation, are considered. Suppose that N represents the population size as well as the external archive A size, m is the number of objectives, D denotes the number of decision variables, then the complexity of archive A maintenance includes two cases. If there is still space in the archive A, the complexity is ($O(1)$) since solutions are directly added. If A is full, the worst-case complexity of

Table 1
Features of the tested functions.

Problem	Dimension of variables	Number of objectives	PF characteristics	Size of PF
ZDT1	30	2	Convex, Continues PF	1000
ZDT2	30	2	Concave, Continues PF	1000
ZDT3	30	2	Discontinuous, Disconnected PF	1000
ZDT4	10	2	Multi-modal, Convex, Continues PF	1000
ZDT6	10	2	Multi-modal, Concave, Non-uniform	1000
DTLZ1	7	3	Multi-modal, Continues PF	10000
DTLZ2	12	3	Concave, Continues PF	10000
DTLZ3	12	3	Multi-modal, Concave	10000
DTLZ4	12	3	Concave, Highly non-uniform PF	10000
DTLZ5	12	3	Mostly degenerated, Concave	10000
DTLZ6	12	3	Mostly degenerated, Concave, Biased	10000
DTLZ7	22	3	Multi-modal, Disconnected PF, Mixed	10000

Table 2
Detail parameter settings.

Algorithm	Parameter settings
NSGAII	$p_c = 1.0, p_m = \frac{1}{D}, \eta_c = \eta_m = 20$
MOEA/D	$T = \frac{N}{10}, p_c = 1.0, p_m = \frac{1}{D}, \eta_c = \eta_m = 20$
MOPSO	$\omega \in [0.1, 0.5], c_1, c_2 \in [1.5, 2.5], div = 10$
MPSO/D	$\omega \in [0.1, 0.9], c_1, c_2 \in [1.5, 2.0], F = 0.5, p_c = 1.0, \eta_c = \eta_m = 20$
SMPSO	$\omega \in [0.1, 0.5], c_1, c_2 \in [1.5, 2.5], p_m = \frac{1}{D}, \eta_m = 20$
dMOPSO	$\omega \in [0.1, 0.5], c_1, c_2 \in [1.5, 2.0], T_a = 2$
NMPSO	$\omega \in [0.1, 0.5], c_1, c_2, c_3 \in [1.5, 2.5], p_c = 1.0, p_m = \frac{1}{D}, \eta_c = \eta_m = 20$
MMOPSO	$\omega \in [0.1, 0.5], c_1, c_2 \in [1.5, 2.0], \delta = 0.7, p_c = 1.0, p_m = \frac{1}{D}, \eta_c = \eta_m = 20$
TPSO-DF	$\omega \in [0.1, 0.5], c_1, c_2 \in [1.5, 2.5], p_c = 1.0, p_m = \frac{1}{D}, \eta_c = \eta_m = 20$

solution evaluation depends on fast non-dominated sorting($O(mN^2)$), calculation of Euclidean distance($O(N)$) from solutions to ideal points, and calculation of crowding distance($O(mN\,log\,N)$). The complexity of global leader selection involving SBX is ($O(ND)$). The complexity of two-stage hybrid mutation strategy mainly involves PM($O(mN)$) and DM($O(mN)$). Therefore, the final complexity of TPSO-DF is ($O(mN^2)$).

4. Experimental verifications

To validate the performance of the proposed TPSO-DF, a comprehensive set of experiments is conducted on ZDT, DTLZ, as well as one practical problem. This section is divided into three parts. (1) Effectiveness verifications of the TPSO-DF, including the dual-indicator fusion ranking measure, the adaptive global leader selection strategy and the two-stage hybrid mutation strategy. (2) Performance comparison with other EAs. (3) Performance comparison on a practical problem.

4.1. Experimental setup

4.1.1. Test problems

In the following experiments, two widely used test suites, ZDT [39] (including five bi-objective cases under different dimensions) and DTLZ [32] (including seven 3-objective cases under different dimensions), are applied to compare TPSO-DF with contrast algorithms. To be specific, ZDT5, being discrete, as well as DTLZ8 and DTLZ9, with constraints, are not included. Obviously, with different complicated characteristics (such as nonlinearity, nonconvexity, nondifferentiability, and multimodality) and PF shapes involved in test benchmarks, it is difficult for algorithms to achieve approximated PFs with an optimal trade-off between convergence and diversity. The detailed features and parameter settings for each test function are given in Table 1.

4.1.2. Parameter settings

As for the parameter settings, TPSO-DF is compared with 8 representative MOEAs which perform competitively for solving MOPs (i.e., NSGAII [30], MOEA/D [40], MOPSO [41], MPSO/D [42], SMPSO [10], dMOPSO [43], NMPSO [36], MMOPSO [44]). As shown in Table 2, all the relative parameter settings are set to the same with their original paper for all compared algorithms. Specifically, the population size is set to 100 and the maximal number of fitness evaluations, i.e., MaxFEs, is set to 5000 for all problems. All experiments independently run 30 times implementing on PlatEMO [45], which uses MATLAB R2021b on a server with Intel(R) Core(TM) i7-7700HQ CPU @ 2.80 GHz.

For MMOPSO, NMPSO and TPSO-DF, in which SBX and PM operations are used for generating offsprings, the crossover probability p_c is set to 1, the mutation probability p_m is set to $\frac{1}{D}$ (D denotes the dimensionality of decision variables) and the distribution index

η_c is set to 20 for a fair comparison. The reference vectors incorporated in dMOPSO, MMOPSO and MOEA/D are generated by normal boundary intersection method (NBI). In terms of learning factors, c_1 and c_2 are randomly distributed between [1.5, 2.5]. Parameters c_1, c_2, c_3 (only for NMPSO) are random numbers distributed in [1.5, 2.5] for MOPSO, NMPSO, SMPSO and [1.5, 2.5] for dMOPSO, MPSO/D. To be fair, the inertial weight ω is generated uniformly between [0.1,0.5] for all algorithms. The threshold T_a in dMOPSO is set to 2. δ in MMOPSO is applied to update velocity and control search patterns. To generate hypercubes in MOPSO, the division number is set to 10. In addition, T denotes the size of neighborhood in MOEA/D.

4.1.3. Performance metrics

Inverse generational distance (IGD) [46] and hypervolume (HV) [47], are applied to comprehensively quantify the convergence and diversity of a PF obtained by test algorithms. IGD metric calculates the distance of solutions on the true PF to their closest solutions on the approximated PF in average as shown in Eq. (20):

$$IGD(\Theta,\Omega) = \frac{\Sigma_{x\in\Omega} dist(x,\Theta)}{|\Omega|} \tag{20}$$

where Ω represents a set of solutions uniformly distributed along the true PF, Θ is a set of optimal solutions on the approximated PF. Moreover, $|\Omega|$ represents the number of solutions, $dist(x,\Theta)$ denotes the minimum Euclidean distance of x to its nearest solution in Θ.

As shown by Eq. (21), HV metric illustrates the union volume of all hypercubes which are formed with solutions in the approximated PF and a reference point on diagonal.

$$HV = \delta(U_{i=1}^{|Q|}\Theta_i) \tag{21}$$

where δ denotes the Lebesgue measure to calculate the hypervolume. Θ_i represents the hypercube formed with ith solution and the reference point.

Besides, to make the experimental results more statistically significant, Wilcoxon rank sum test is executed at a significance level of 0.05. Symbols '+', '−' and '=' show that the results of other MOEAs are significantly better than, worse than and similar with the results obtained by TPSO-DF, respectively.

4.2. Effectiveness test for the three modifications in TPSO-DF

In TPSO-DF, to enhance the performance of MOPSO, three modifications are incorporated. The first one is to introduce the dual-indicator fusion ranking measure. The second is to design the adaptive global leader selection strategy. The third is to construct the two-stage hybrid mutation strategy. To validate the effectiveness of these modifications, related MOPSO variants based on TPSO-DF are constructed as follows.

- TPSO-DF1: TPSO-DF1 is constructed by applying the commonly used congestion distance-based archive mechanism to replace the dual-indicator fusion ranking measure.
- TPSO-DF2: TPSO-DF2 is designed through applying random global leader in the external archive in stead of the adaptive global leader selection strategy.
- TPSO-DF3: TPSO-DF3 is created by removing the two-stage hybrid mutation strategy directly.

Consequently, in this subsection, there are four algorithms in this experiment: TPSO-DF1, TPSO-DF2, TPSO-DF3 and TPSO-DF. The experiment is conducted on the ZDT and DTLZ test suites. To be fair, related parameters of all compared algorithms are set as in the previous part, such as the inertial weight. The number of independent run is set to 30 for each benchmark function. The average and standard deviation statistics of IGD and HV metrics are recorded as the final result. Table 3 and Table 4 give respectively the IGD and the HV index values for the four algorithms in the 12 test cases. Specifically, the best results are marked with boldface, and the Wilcoxon rank test is used to compare the significant differences between algorithms.

From Table 3 and Table 4, TPSO-DF obtains the best IGD index values 7 times and HV index values 8 times in the 12 test cases, respectively. In addition, TPSO-DF significantly outperforms its three variants in 9, 7 and 8 test cases for IGD index values and 8, 7 and 5 test cases for HV index values, respectively.

To be specific, TPSO-DF1 gives just better results for multi-modal, concave, non-uniform test case on IGD metric, while TPSO-DF2 performs better than TPSO-DF1 and TPSO-DF3, especially TPSO-DF1. The possible reason is that the adaptive global leader selection strategy can improve the quality of solutions as population leaders and focus more exact on elite solutions, while the dual-indicator fusion ranking measure and two-stage hybrid mutation strategy can better balance between exploration and exploitation by applying information fusion and diverse genetic operator, thus TPSO-DF2 can achieve better results on discontinuous function and 3-objective test cases which are harder to be solved than bi-objective test cases. After combining these three modifications, TPSO-DF can perform the best which confirms that these three modifications can effectively cooperate together to enhance the performance as much as possible.

Table 3
Comparison of TPSO-DF1, TPSO-DF2, TPSO-DF3 and TPSO-DF algorithms on IGD metric.

Problem	TPSO-DF1	TPSO-DF2	TPSO-DF3	TPSO-DF
ZDT1	4.8194e-3(2.09e-4)-	4.2777e-3(1.10e-4)-	4.2643e-3(1.02e-4)-	**4.1907e-3(1.12e-4)**
ZDT2	5.3785e-3(8.59e-4)-	4.4150e-3(1.32e-4) =	3.0182e-1(4.10e-1)-	**4.3958e-3(1.50e-4)**
ZDT3	5.4058e-3(2.35e-4)-	**5.1391e-3(1.78e-4)** =	5.3063e-3(1.69e-4)-	5.2020e-3(1.99e-4)
ZDT4	1.0987e+1(6.40e+0)-	5.3842e+0(2.62e+0)-	6.2649e+0(4.18e+0)-	**9.8836e-3(2.49e-3)**
ZDT6	**3.6708e-3(1.41e-4)** =	4.2637e-3(3.77e-4)-	3.6720e-3(1.69e-4) =	4.1424e-3(1.17e-3)
DTLZ1	1.6013e+1(4.37e+0)-	1.3200e+1(3.86e+0)-	1.7609e+1(4.70e+0)-	**6.4728e-1(2.82e-1)**
DTLZ2	6.8264e-2(1.85e-3)-	6.4300e-2(1.79e-3) =	**6.3411e-2(1.88e-3)** =	6.3489e-2(1.59e-3)
DTLZ3	1.4622e+2(3.76e+1)-	1.1976e+2(2.53e+1)-	1.6618e+2(1.50e+1)-	**1.9722e+0(8.04e-1)**
DTLZ4	1.8676e-1(2.29e-2)+	**1.4733e-1(1.67e-2)**+	1.8249e-1(1.76e-2)+	2.0687e-1(2.22e-2)
DTLZ5	6.1221e-3(4.05e-4)-	4.7568e-3(1.42e-4)-	4.7186e-3(9.12e-5)-	**4.6274e-3(1.19e-4)**
DTLZ6	6.1825e-3(4.25e-4)-	4.9776e-3(1.47e-4)-	4.6180e-3(1.26e-4) =	**4.5825e-3(1.16e-4)**
DTLZ7	8.3782e-2(5.39e-3)+	**8.3725e-2(4.94e-3)**+	1.3842e-1(2.64e-1)-	8.7052e-2(5.75e-3)
(+/-/=)	(2/9/1)	(2/7/3)	(1/8/3)	
Best/All	(1/12)	(3/12)	(1/12)	**(7/12)**

Table 4
Comparison of TPSO-DF1, TPSO-DF2, TPSO-DF3 and TPSO-DF algorithms on HV metric.

Problem	TPSO-DF1	TPSO-DF2	TPSO-DF3	TPSO-DF
ZDT1	7.1883e-1(3.49e-4)-	7.1968e-1(1.51e-4)-	7.1975e-1(1.73e-4) =	**7.1983e-1(1.67e-4)**
ZDT2	4.4340e-1(4.90e-4)-	4.4444e-1(1.20e-4) =	2.9083e-1(1.90e-1)-	**4.4447e-1(1.93e-4)**
ZDT3	5.9934e-1(5.24e-4) =	**5.9947e-1(2.91e-4)** =	5.9938e-1(2.09e-4) =	5.9940e-1(6.92e-4)
ZDT4	0.0000e+0(0.00e+0)-	0.0000e+0(0.00e+0)-	0.0000e+0(0.00e+0)-	**7.1232e-1(3.04e-3)**
ZDT6	3.8816e-1(1.70e-4)-	3.8759e-1(3.06e-4)-	3.8815e-1(1.46e-4)-	**5.0040e-1(1.50e-1)**
DTLZ1	0.0000e+0(0.00e+0)-	0.0000e+0(0.00e+0)-	0.0000e+0(0.00e+0)-	**3.7652e-2(8.99e-2)**
DTLZ2	5.2496e-1(3.81e-3)-	5.3751e-1(3.17e-3)-	5.4069e-1(4.01e-3) =	**5.4109e-1(4.56e-3)**
DTLZ3	0.0000e+0(0.00e+0) =	0.0000e+0(0.00e+0) =	0.0000e+0(0.00e+0) =	**2.9034e-3(1.54e-2)**
DTLZ4	4.6676e-1(1.76e-2)+	**5.1429e-1(7.49e-3)**+	4.8339e-1(1.75e-2)+	4.4776e-1(2.36e-2)
DTLZ5	1.9841e-1(1.96e-4)-	1.9979e-1(8.37e-5)-	2.0008e-1(4.54e-5)-	**2.0013e-1(4.97e-5)**
DTLZ6	1.9885e-1(2.78e-4)-	1.9983e-1(9.26e-5)-	**2.0010e-1(4.57e-5)**+	2.0000e-1(7.12e-5)
DTLZ7	2.6684e-1(2.16e-3)+	**2.6730e-1(2.61e-3)**+	2.6157e-1(3.23e-2)+	2.6019e-1(3.59e-3)
(+/-/=)	(2/8/2)	(2/7/3)	(3/5/4)	
Best/All	(0/12)	(3/12)	(1/12)	**(8/12)**

4.3. Comparison with other MOEA variants

To further validate the performance of our approach, ZDT and DTLZ test suites are applied to compare TPSO-DF with eight other paradigms of MOEAs, including NSGAII, MOEA/D, MOPSO, MOPSO/D, SMPSO, dMOPSO, MMOPSO and NMPSO.

4.3.1. IGD metric and HV metric

As observed from Table 5 on IGD metric, TPSO-DF generally outperforms among test algorithms. TPSO-DF performs significantly better than MOPSO, MPSO/D, dMOPSO, SMPSO, NMPSO, MMOPSO, MOEA/D and NSGAII on 12, 10, 10, 9, 9, 8, 8, 8 out of 12 comparisons. However, TPSO-DF performs worse than MPSO/D, dMOPSO, NMPSO, MMOPSO, MOEA/D and NSGAII on 2, 2, 2, 3, 3, 4 comparisons. Besides, TPSO-DF obtains similar results to MMOPSO, NMPSO, MOEA/D and SMPSO on 1, 1, 1, 3 test cases, respectively. Overall, the comparisons of the experimental results confirm the high effectiveness of TPSO-DF to solve the tested problems.

In addition, as shown in the last row of Table 5 on IGD metric, TPSO-DF outperforms on 5 of the 12 benchmark functions, following by MOEA/D which performs best on 2 benchmark cases. NSGAII, MPSO/D, SMPSO and MMOPSO outperforms on DTLZ3, DTLZ4, ZDT6 and ZDT4 separately. Other compared MOEAs, MOPSO, dMOPSO and NMPSO, perform best on none of tested functions, which is mainly due to the sole decomposition method in dMOPSO and the SBX and PM operations of NMPSO cannot efficiently enhance the effectiveness of the algorithm.

Compared with MMOPSO, TPSO-DF shows outstanding performance due to the incorporation of the dual-indicator fusion ranking measure. This measure ensures that TPSO-DF can focus on exact key information and can finally constantly well balance between convergence and diversity. Furthermore, the selection of global leader based on the dual-indicator fusion ranking, as well as the SBX and PM operations can allow the evolution towards solutions carrying valuable information and also improves the search ability significantly. Thus finally achieve good performance when solving various MOPs.

It can be clearly seen that similar conclusions can be derived from the comparison results regarding HV metric shown in Table 6. TPSO-DF also achieves best performance on most problems and outperforms other compared algorithms.

4.3.2. Friedman rank test

To further determine whether there is a difference between the algorithms in general, the Friedman test [48] is performed on the above experimental results. It is first assumed that there is no difference between all algorithms. The ranking of each algorithm in

Table 5
Comparison of eight MOEAs and TPSO-DF on IGD metric.

Problem	IGD	MOEA/D	NSGAII	MOPSO	MPSO/D	SMPSO	dMOPSO	MMOPSO	NMPSO	TPSO-DF
ZDT1	mean	3.88E-02	5.04E-03	7.93E-01	1.31E-02	4.59E-02	1.09E-02	4.95E-03	2.74E-02	**4.19E-03**
	std	3.17E-02	2.38E-04	1.86E-01	2.57E-03	1.04E-02	1.90E-03	1.86E-04	8.57E-03	**1.12E-04**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
ZDT2	mean	1.36E-01	5.11E-03	1.54E+00	1.42E-02	4.98E-03	3.39E-02	5.18E-03	1.99E-02	**4.40E-03**
	std	9.77E-02	2.16E-04	5.02E-01	3.96E-03	2.30E-04	1.09E-01	2.96E-04	4.80E-03	**1.50E-04**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
ZDT3	mean	4.33E-02	5.52E-03	6.57E-01	3.07E-02	1.97E-02	1.36E-02	5.57E-03	1.01E-01	**5.20E-03**
	std	2.16E-02	2.22E-04	1.63E-01	9.82E-03	3.73E-02	9.34E-04	3.06E-04	7.87E-04	**2.49E-03**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
ZDT4	mean	9.27E-02	8.76E-03	1.65E+01	7.02E+00	9.87E-01	5.85E-03	**4.73E-03**	3.25E-02	9.88E-03
	std	5.06E-02	4.57E-03	6.27E+00	2.88E+00	8.97E-01	1.41E-03	**3.07E-04**	2.39E-02	2.49E-03
	sig	(-)	(+)	(-)	(-)	(-)	(+)	(+)	(-)	
ZDT6	mean	1.42E-02	4.47E-03	5.59E-01	1.31E-01	**3.82E-03**	3.86E-03	4.27E-03	4.40E-03	4.14E-03
	std	3.31E-03	6.43E-04	1.68E+00	5.17E-02	**1.96E-04**	1.68E-03	2.06E-04	4.78E-04	1.17E-03
	sig	(-)	(-)	(-)	(-)	(=)	(+)	(-)	(-)	
DTLZ1	mean	**2.14E-02**	3.58E-02	5.57E+00	1.81E+00	1.75E+00	1.83E+00	1.30E-01	2.28E-02	6.47E-01
	std	**1.49E-03**	4.51E-02	2.73E+00	6.49E-01	2.47E+00	1.24E+00	2.48E-01	6.40E-04	2.82E-01
	sig	(+)	(+)	(-)	(-)	(=)	(-)	(+)	(+)	
DTLZ2	mean	**5.45E-02**	6.98E-02	1.75E-01	5.45E-02	7.16E-02	1.35E-01	7.14E-02	7.69E-02	6.35E-02
	std	**1.69E-06**	2.86E-03	2.83E-02	8.91E-05	3.94E-03	9.60E-03	2.47E-03	2.74E-03	1.59E-03
	sig	(+)	(-)	(-)	(+)	(-)	(-)	(-)	(-)	
DTLZ3	mean	9.86E-01	**6.35E-01**	6.34E+01	3.91E+01	1.26E+01	3.00E+01	5.07E+00	2.57E+00	1.97E+00
	std	2.42E+00	**8.89E-01**	5.25E+01	9.78E+00	1.67E+01	4.95E+01	4.07E+00	2.02E+00	8.04E-01
	sig	(+)	(+)	(-)	(-)	(=)	(-)	(-)	(=)	
DTLZ4	mean	3.87E-01	6.73E-02	2.77E-01	**5.51E-02**	2.97E-01	2.50E-01	1.00E-01	9.24E-02	2.07E-01
	std	3.23E-01	2.35E-03	1.39E-01	**4.60E-04**	2.21E-01	3.40E-02	1.60E-01	8.49E-02	2.22E-02
	sig	(=)	(+)	(-)	(+)	(-)	(-)	(+)	(+)	
DTLZ5	mean	3.38E-02	5.82E-03	1.18E-02	3.24E-02	5.41E-03	4.18E-02	6.26E-03	1.43E-02	**4.63E-03**
	std	5.62E-05	2.57E-04	2.32E-03	1.40E-03	2.58E-04	7.94E-03	4.84E-04	2.49E-03	**1.19E-04**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
DTLZ6	mean	3.39E-02	5.92E-03	2.51E+00	3.07E+00	5.50E-03	3.37E-02	6.62E-03	1.61E-02	**4.58E-03**
	std	5.23E-05	3.27E-04	9.79E-01	3.24E-01	3.39E-04	2.68E-05	5.03E-04	3.67E-03	**1.16E-04**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
DTLZ7	mean	1.99E-01	1.05E-01	3.01E+00	5.01E-01	1.11E-01	2.16E-01	1.69E-01	1.19E-01	**8.71E-02**
	std	1.64E-01	8.26E-02	9.25E-01	1.68E-01	6.67E-02	2.42E-01	1.94E-01	1.40E-01	**5.75E-03**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(=)	(-)	
(+/-/=)		(3/8/1)	(4/8/0)	(0/12/0)	(2/10/0)	(0/9/3)	(2/10/0)	(3/8/1)	(2/9/1)	
Best/All		(2/12)	(1/12)	(0/12)	(1/12)	(1/12)	(0/12)	(1/12)	(0/12)	**(6/12)**

each test function is calculated using the average metrics in Table 5 and 6, which leads to the average ranking as well as the final ranking, and the results are shown in Table 7. The average ranking of TPSO-DF in IGD and HV is 2.42 and 3.08, respectively. The final ranking is first, from which it reflects that TPSO-DF has obtained the most superior performance.

The Friedman test values calculated from the mean ranking of IGD and HV of the algorithms are 10.69 and 9.60, respectively. Both of them are greater than the critical value of 2.045 obtained at the significance level of 0.05. Therefore, the original suppose is rejected, which confirms a significant difference between the algorithms. The Friedman test can reflect that TPSO-DF can obtain the most superior convergence and diversity in general, which further proves the effectiveness of the proposed strategies in TPSO-DF.

4.3.3. Approximated PFs obtained

To compare the final optimization results of TPSO-DF and other MOEAs more intuitively, the distribution of the approximate PFs of TPSO-DF and other MOEAs for typical test samples of ZDT1, ZDT3, DTLZ6, and DTLZ7 are plotted in Figs. 4–7. It is obvious that the proposed TPSO-DF can obtain a set of solutions uniformly distributed on the true PF of test functions. While the distributions of the approximated PFs obtained by other compared algorithms are less uniform.

For example, regarding ZDT1, the results show that MOPSO and MPSO/D have the worst optimization results with almost no optimal solutions in their true PFs, MOEA/D, dMOPSO, and NMPSO also have only a few optimal solutions in their true PFs. Moreover, MOPSO can hardly converge to the true PF for ZDT3 problem with disconnected PF, which is also concluded by Table 5 on IGD metric. In terms of DTLZ6 with degenerated and biased PF, MOPSO and MPSO/D have almost no optimal solutions in their true PFs while

Table 6
Comparison of eight MOEAs and TPSO-DF on HV metric.

Problem	HV	MOEA/D	NSGAII	MOPSO	MPSO/D	SMPSO	dMOPSO	MMOPSO	NMPSO	TPSO-DF
ZDT1	mean	6.89E-01	7.18E-01	5.80E-02	7.05E-01	6.60E-01	7.09E-01	7.18E-01	6.89E-01	**7.20E-01**
	std	1.91E-02	4.52E-04	7.62E-02	3.43E-03	1.44E-02	2.75E-03	3.26E-04	9.56E-03	**1.67E-04**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
ZDT2	mean	3.07E-01	4.43E-01	4.54E-03	4.26E-01	4.44E-01	4.13E-01	4.44E-01	4.35E-01	**4.44E-01**
	std	7.47E-02	4.41E-04	2.37E-02	6.07E-03	3.22E-04	6.15E-02	1.76E-04	2.84E-03	**1.93E-04**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
ZDT3	mean	**6.26E-01**	5.99E-01	1.54E-01	5.80E-01	6.01E-01	5.97E-01	5.99E-01	5.67E-01	5.99E-01
	std	**5.66E-02**	1.82E-04	1.13E-01	7.79E-03	1.80E-02	1.31E-03	1.13E-04	3.20E-04	6.92E-04
	sig	(=)	(-)	(-)	(-)	(+)	(-)	(=)	(-)	
ZDT4	mean	6.10E-01	7.12E-01	0.00E+00	0.00E+00	1.43E-01	7.16E-01	**7.19E-01**	6.83E-01	7.12E-01
	std	5.77E-02	4.77E-03	0.00E+00	0.00E+00	1.63E-01	2.10E-03	**5.34E-04**	3.28E-02	3.04E-03
	sig	(-)	(=)	(-)	(-)	(-)	(+)	(+)	(-)	
ZDT6	mean	3.70E-01	3.86E-01	3.42E-01	2.39E-01	3.88E-01	3.88E-01	3.88E-01	3.88E-01	**5.00E-01**
	std	4.70E-03	1.31E-03	1.16E-01	4.72E-02	1.46E-04	1.95E-03	2.03E-04	4.22E-04	**1.50E-01**
	sig	(-)	(-)	(-)	(-)	(-)	(=)	(-)	(-)	
DTLZ1	mean	**8.37E-01**	7.99E-01	0.00E+00	0.00E+00	2.31E-01	2.51E-02	6.87E-01	8.33E-01	3.77E-02
	std	**5.62E-03**	1.17E-01	0.00E+00	0.00E+00	2.64E-01	7.65E-02	3.00E-01	2.28E-03	8.99E-02
	sig	(+)	(+)	(-)	(-)	(+)	(=)	(+)	(+)	
DTLZ2	mean	5.59E-01	5.30E-01	3.45E-01	5.58E-01	5.14E-01	3.84E-01	5.31E-01	**5.61E-01**	5.41E-01
	std	3.75E-05	3.32E-03	3.41E-02	3.60E-04	7.10E-03	1.54E-02	4.24E-03	**1.06E-03**	4.56E-03
	sig	(+)	(-)	(-)	(+)	(-)	(-)	(-)	(+)	
DTLZ3	mean	**3.10E-01**	3.08E-01	0.00E+00	0.00E+00	4.65E-02	7.64E-03	2.57E-02	5.52E-02	2.90E-03
	std	**2.28E-01**	2.36E-01	0.00E+00	0.00E+00	9.59E-02	2.07E-02	1.04E-01	1.61E-01	1.54E-02
	sig	(+)	(+)	(=)	(=)	(+)	(=)	(=)	(=)	
DTLZ4	mean	3.98E-01	5.35E-01	3.82E-01	**5.57E-01**	4.32E-01	4.33E-01	5.18E-01	5.54E-01	4.48E-01
	std	1.65E-01	4.06E-03	9.16E-02	**7.10E-04**	1.14E-01	1.99E-02	8.08E-02	3.96E-02	2.36E-02
	sig	(=)	(+)	(-)	(+)	(=)	(-)	(+)	(+)	
DTLZ5	mean	1.82E-01	1.99E-01	1.89E-01	1.80E-01	1.99E-01	1.56E-01	1.99E-01	1.96E-01	**2.00E-01**
	std	3.56E-05	2.10E-04	3.80E-03	1.14E-03	2.16E-04	1.19E-02	1.72E-04	6.16E-04	**4.97E-05**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
DTLZ6	mean	1.82E-01	1.99E-01	6.61E-03	0.00E+00	2.00E-01	1.82E-01	1.99E-01	1.95E-01	**2.00E-01**
	std	3.56E-05	2.26E-04	3.62E-02	0.00E+00	1.81E-04	2.42E-05	1.71E-04	1.40E-03	**7.12E-05**
	sig	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	
DTLZ7	mean	2.51E-01	2.65E-01	1.11E-03	8.01E-02	2.58E-01	2.38E-01	2.59E-01	**2.68E-01**	2.60E-01
	std	1.33E-02	8.28E-03	2.84E-03	3.83E-02	7.48E-03	2.10E-02	1.86E-02	**2.23E-02**	3.59E-03
	sig	(-)	(+)	(-)	(-)	(=)	(-)	(=)	(+)	
(+/-/=)		(3/7/2)	(4/7/1)	(0/11/1)	(2/9/1)	(3/7/2)	(1/8/3)	(3/7/2)	(4/7/1)	
Best/All		(3/12)	(0/12)	(0/12)	(1/12)	(0/12)	(0/12)	(1/12)	(2/12)	**(5/12)**

Table 7
Comparison results of IGD and HV on Friedman test.

	IGD		HV	
Algorithm	Average Rank	Final Rank	Average Rank	Final Rank
MOEA/D	5.75	6	5.08	6
NSGAII	2.92	2	3.67	3
MOPSO	8.42	9	8.33	9
MPSO/D	6.17	8	6.67	8
SMPSO	4.67	4	4.58	5
dMOPSO	5.83	7	5.92	7
MMOPSO	3.75	3	3.42	2
NMPSO	5.08	5	3.92	4
TPSO-DF	**2.42**	**1**	**3.08**	**1**

MOEA/D and dMOPSO obtain only a few nondominated solutions on the true PF. For DTLZ7, the optimal solutions obtained by MOPSO and MPSO/D are not on their true PFs. The optimal solutions obtained by NSGAII and SMPSO are mainly distributed on two regions of their true PFs, while the other two regions have no optimal solutions.

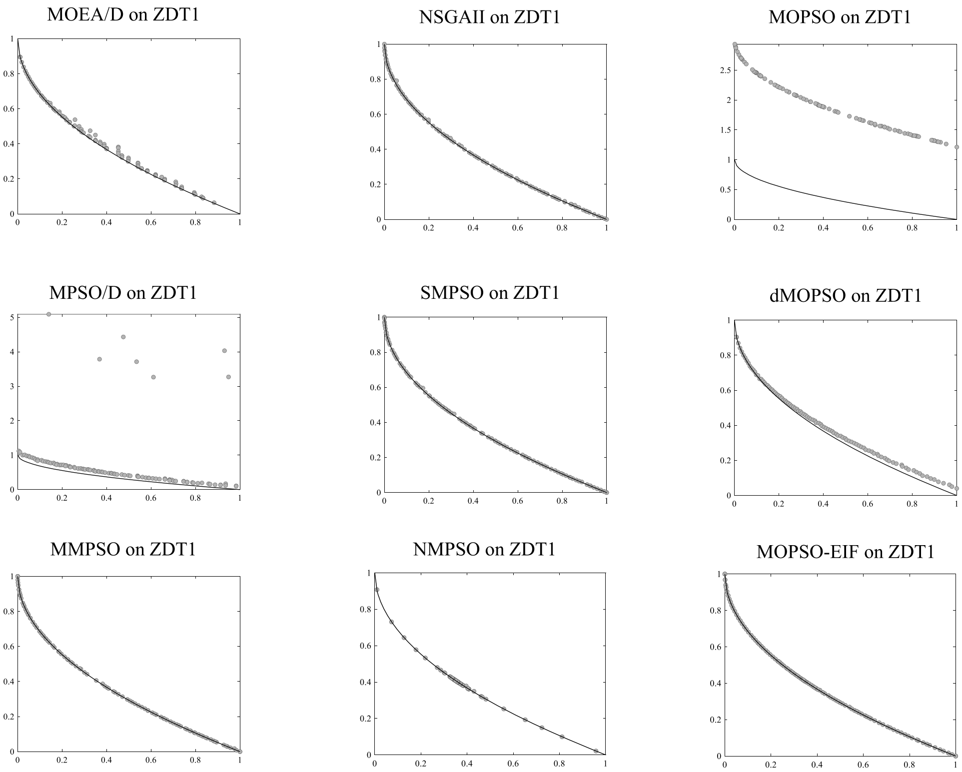

Fig. 4. Comparison of eight MOEAs and TPSO-DF on approximated PFs for ZDT1.

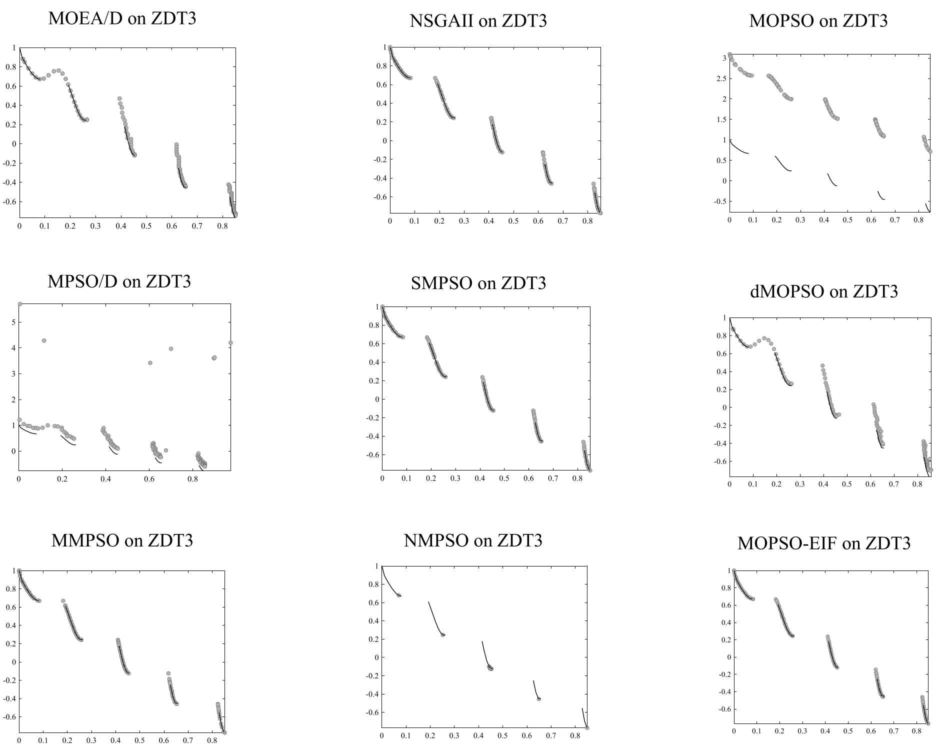

Fig. 5. Comparison of eight MOEAs and TPSO-DF on approximated PFs for ZDT3.

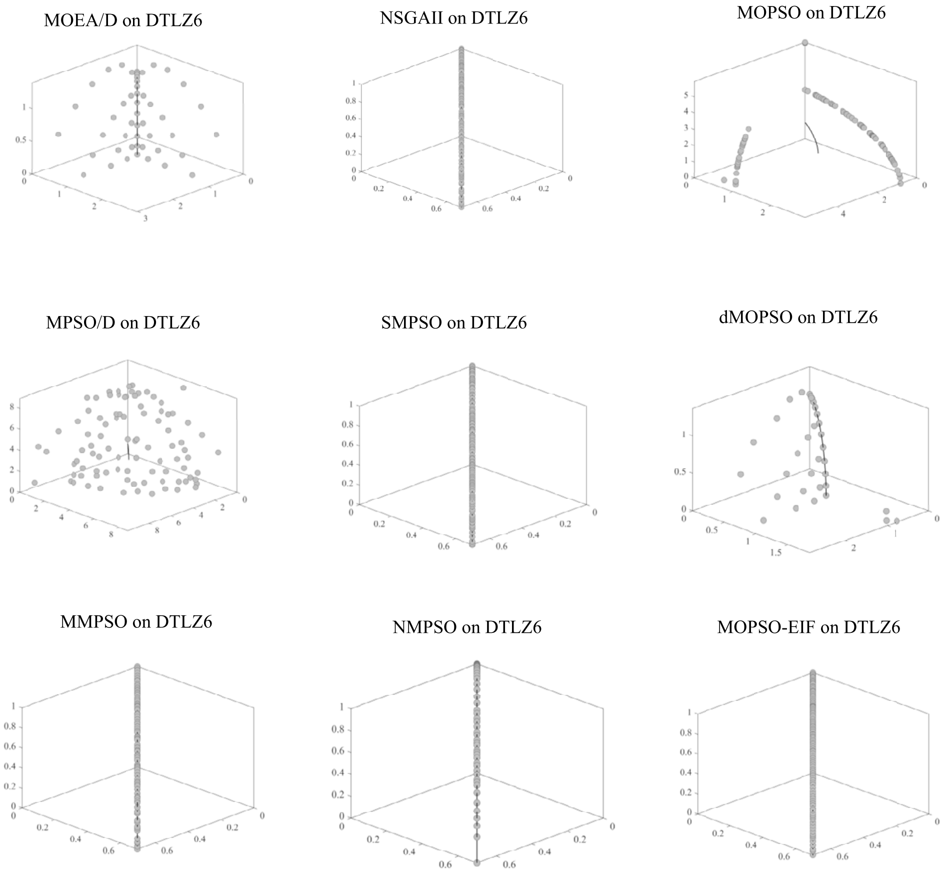

Fig. 6. Comparison of eight MOEAs and TPSO-DF on approximated PFs for DTLZ6.

To sum up, the experimental results obviously confirm that for solving MOPs, the proposed TPSO-DF is very competitive compared to the selected competing MOEAs regarding convergence and diversity.

4.4. TPSO-DF performance on a practical problem

The proposed TPSO-DF is applied to test catalyst designs for methane to hydrogen industrial processes in order to further evaluate its performance in addressing real-world applications. Methane is an abundant natural gas that can be converted to hydrogen and carbon dioxide by catalytic reactions. The TPSO-DF is applied to test catalyst designs for methane to hydrogen industrial processes in practical problems. Methane is an abundant natural gas that can be converted to hydrogen and carbon dioxide by catalytic reactions. This process is of great importance for hydrogen production and energy transition. The design of catalysts has an essential impact on the efficiency, economy and environmental friendliness of methane hydrogen production. The common spherical shape of the catalyst is chosen, and the decision variables include catalyst particle diameter and pore diameter. Corresponding mathematical model is established with two metrics of pressure drop and hydrogen production rate. Eq. (22) shows the mathematical expressions [49].

$$Minimize(f_1, f_2) = \begin{cases} f_1 = 1.92 + 3.52\times 10^5 d_pore - 205.5d_p^7 + 3.3\times 10^1 1d_pore^2 \\ \quad +6376d_p^2 - 1.11\times 10^7 d_{pore}d_p \\ f_2 = -0.17 - 9.42\times 10^5 d_pore + 97d_p + 3.95\times 10^1 1d_pore^2 \\ \quad -2770d_p^2 + 3.8\times 10^7 d_{pore}d_p \end{cases} \tag{22}$$

where d_pore and d_p denote the pore diameter and particle diameter of the catalyst, respectively. f_1 and f_2 denote the normalized pressure drop and hydrogen production rate, respectively.

TPSO-DF is applied to solve the above mathematical model and the obtained optimal results are shown in Fig. 8. The approximate PF of TPSO-DF is complete and evenly distributed in the objective space, which indicates that TPSO-DF provides a set of optimal solutions for the design of catalysts. The decision maker can choose from the optimal solutions based on preferences or actual production design needs.

To further demonstrate the superiority of TPSO-DF, the optimization results of TPSO-DF are compared with the eight algorithms NSGAII, MOEA/D, MOPSO, MPSO/D, SMPSO, dMOPSO, MMOPSO and NMPSO. IGD metrics cannot be adopted to practical problems

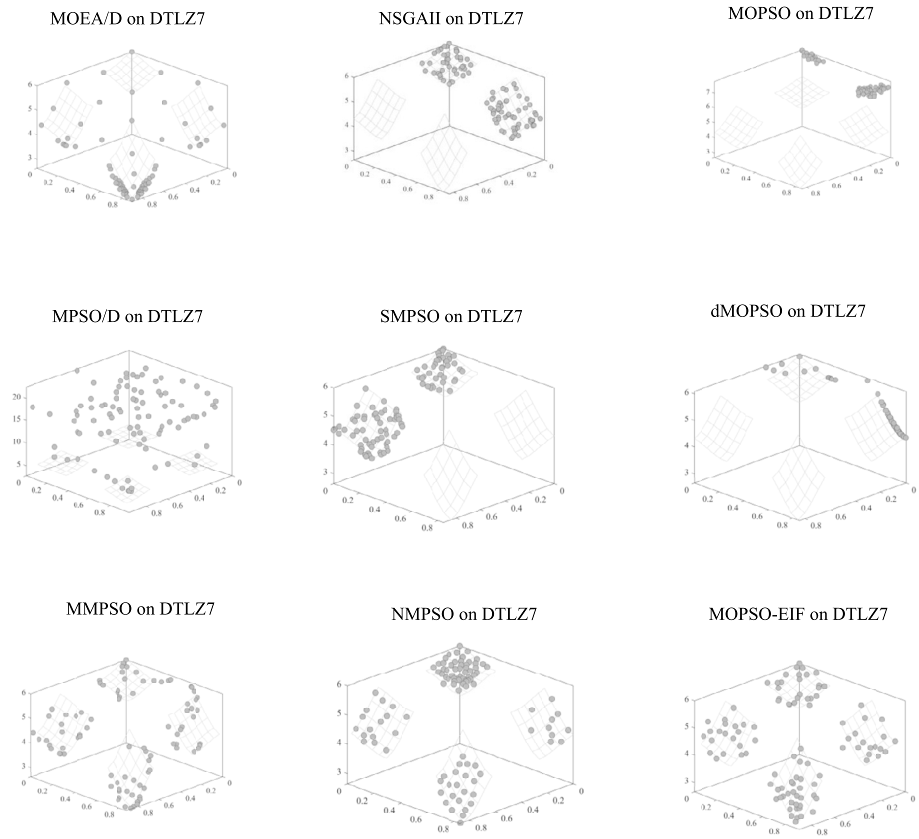

Fig. 7. Comparison of eight MOEAs and TPSO-DF on approximated PFs for DTLZ7.

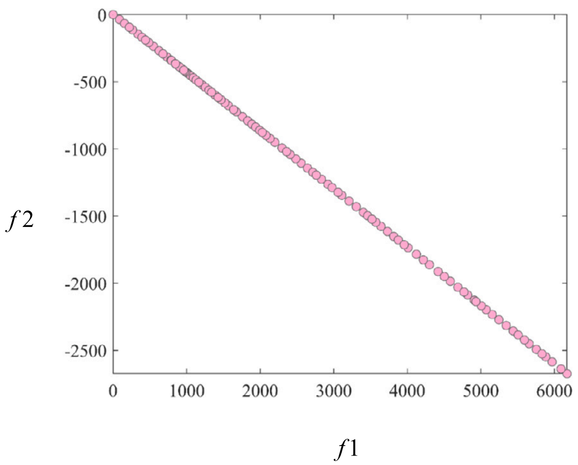

Fig. 8. Optimization results of TPSO-DF for the practical problem.

where the true PF is unknown. HV metrics are applied to quantify the performance. The statistical results of the algorithm run 30 times are shown in Table 8. Compared to other MOEAs, TPSO-DF has the highest accuracy in HV metrics, ranking first among the nine representative MOEAs, which indicates that TPSO-DF performs very well in terms of stability of the algorithm while having good convergence and diversity in solving the above practical problem. Therefore, TPSO-DF can effectively solve the practical MOPs.

5. Conclusions

This paper proposes a two stage multi-objective particle swarm optimization algorithm with dual-indicator fusion ranking (TPSO-DF). In TPSO-DF, an external archive maintenance mechanism based on two indicators using the convergence and diversity informa-

Table 8
Comparison of eight MOEAs and TPSO-DF on HV metric for the practical problem.

Algorithm	Mean	Std	Rank
MOEA/D	6.8889E-02	1.1806E-04	6
NSGAII	6.9158E-02	1.2100E-05	2
MOPSO	2.2985E-03	1.2600E-02	7
MPSO/D	0.0000E+00	0.0000E+00	8
SMPSO	6.9038E-02	1.5500E-04	4
dMOPSO	6.8936E-02	1.2700E-05	5
MMOPSO	6.9157E-02	1.3300E-05	3
NMPSO	0.0000E+00	0.0000E+00	8
TPSO-DF	**6.9159E-02**	**3.8000E-06**	**1**

tion, respectively, as well as a dual-indicator fusion ranking measure is proposed to filter out solutions with outstanding performance in both convergence and diversity. By using an adaptive global leader selection strategy with valuable information fusion, the aggregation of many solutions in one region early in the algorithm is avoided due to the low number of solutions in the archive. In addition, a two-stage hybrid mutation strategy is introduced to further improve the picking quality by exploiting different evolutionary states. The experiments demonstrate the competitiveness and validity of TPSO-DF.

CRediT authorship contribution statement

Qing Xu: Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Yuhao Chen:** Validation, Software, Methodology, Investigation, Data curation, Conceptualization. **Cisong Shi:** Writing – original draft, Visualization, Software, Formal analysis. **Junhong Huang:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Wei Li:** Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 62066019) and Jiangxi Province Key Laboratory of Multidimensional Intelligent Perception and Control of China (Grant No. 2024SSY03161).

References

[1] Li Wei, Yangtao Chen, Yuehua Dong, Ying Huang, A solution potential-based adaptation reference vector evolutionary algorithm for many-objective optimization, Swarm Evol. Comput. 84 (2024) 101451.
[2] Peng Liang, Yangtao Chen, Yafeng Sun, Ying Huang, Wei Li, An information entropy-driven evolutionary algorithm based on reinforcement learning for many-objective optimization, Expert Syst. Appl. 238 (2024) 122164.
[3] Li Wei, Peng Liang, Bo Sun, Yafeng Sun, Ying Huang, Reinforcement learning-based particle swarm optimization with neighborhood differential mutation strategy, Swarm Evol. Comput. 78 (2023) 101274.
[4] Li Wei, Jianghui Jing, Yangtao Chen, Yishan Chen, A cooperative particle swarm optimization with difference learning, Inf. Sci. 643 (2023) 119238.
[5] Bolin Wu, Wang Hu, Junjie Hu, Gary G. Yen, Adaptive multiobjective particle swarm optimization based on evolutionary state estimation, IEEE Trans. Cybern. 51 (7) (2021) 3738–3751.
[6] Yanan Sun, Bing Xue, Mengjie Zhang, Gary G. Yen, A new two-stage evolutionary algorithm for many-objective optimization, IEEE Trans. Evol. Comput. 23 (5) (2019) 748–761.
[7] Xianpeng Wang, Zhiming Dong, Lixin Tang, Multiobjective differential evolution with personal archive and biased self-adaptive mutation selection, IEEE Trans. Syst. Man Cybern. Syst. 50 (12) (2020) 5338–5350.
[8] Ran Cheng, Yaochu Jin, A competitive swarm optimizer for large scale optimization, IEEE Trans. Cybern. 45 (2) (2015) 191–204.
[9] Shixin Cheng, Hao Zhan, Huiqin Yao, Huayu Fan, Yan Liu, Large-scale many-objective particle swarm optimizer with fast convergence based on alpha-stable mutation and logistic function, Appl. Soft Comput. 99 (2021) 106947.
[10] A.J. Nebro, J.J. Durillo, J. Garcia-Nieto, C.A. Coello Coello, F. Luna, E. Alba, SMPSO: a new PSO-based metaheuristic for multi-objective optimization, in: 2009 IEEE Symposium on Computational Intelligence in Milti-Criteria Decision-Making, IEEE, Nashville, TN, USA, 2009, pp. 66–73.
[11] Yiqiong Yuan, Jun Sun, Dongmei Zhou, Multi-objective random drift particle swarm optimization algorithm with adaptive grids, in: 2016 IEEE Congress on Evolutionary Computation (CEC), IEEE, Vancouver, BC, Canada, 2016, pp. 2064–2070.
[12] Wang Hu, Gary G. Yen, Adaptive multiobjective particle swarm optimization based on parallel cell coordinate system, IEEE Trans. Evol. Comput. 19 (1) (2015) 1–18.

[13] Wang Hu, Gary G. Yen, Guangchun Luo, Many-objective particle swarm optimization using two-stage strategy and parallel cell coordinate system, IEEE Trans. Cybern. 47 (6) (2017) 1446–1459.
[14] Carlo Raquel, Prospero Naval, An effective use of crowding distance in multiobjective particle swarm optimization, in: GECCO 2005 - Genetic and Evolutionary Computation Conference, 2005, pp. 257–264.
[15] Yingying Cui, Xi Meng, Junfei Qiao, A multi-objective particle swarm optimization algorithm based on two-archive mechanism, Appl. Soft Comput. 119 (2022) 108532.
[16] Wang Hu, Gary G. Yen, Density estimation for selecting leaders and maintaining archive in MOPSO, in: 2013 IEEE Congress on Evolutionary Computation, Cancun, Mexico, IEEE, 2013, pp. 181–188.
[17] S. Kukkonen, K. Deb, Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems, in: 2006 IEEE International Conference on Evolutionary Computation, IEEE, Vancouver, BC, Canada, 2006, pp. 1179–1186.
[18] Miqing Li, Shengxiang Yang, Xiaohui Liu, Shift-based density estimation for Pareto-based algorithms in many-objective optimization, IEEE Trans. Evol. Comput. 18 (3) (2014) 348–365.
[19] Honggui Han, Wei Lu, Lu Zhang, Junfei Qiao, Adaptive gradient multiobjective particle swarm optimization, IEEE Trans. Cybern. 48 (11) (2018) 3067–3079.
[20] Kata Praditwong, Xin Yao, A new multi-objective evolutionary optimisation algorithm: the two-archive algorithm, in: 2006 International Conference on Computational Intelligence and Security, IEEE, Guangzhou, China, 2006, pp. 286–291.
[21] Bingdong Li, Jinlong Li, Ke Tang, Xin Yao, An improved two archive algorithm for many-objective optimization, in: 2014 IEEE Congress on Evolutionary Computation (CEC), IEEE, Beijing, China, 2014, pp. 2869–2876.
[22] Lei Cai, Shiru Qu, Yuan Yuan, Xin Yao, A clustering-ranking method for many-objective optimization, Appl. Soft Comput. 35 (2015) 681–694.
[23] Cai Dai, Two-archive evolutionary algorithm based on multi-search strategy for many-objective optimization, IEEE Access 7 (2019) 79277–79286.
[24] Handing Wang, Licheng Jiao, Xin Yao, Two_Arch2: an improved two-archive algorithm for many-objective optimization, IEEE Trans. Evol. Comput. 19 (4) (2015) 524–541.
[25] Xingyin Wang, Yuping Wang, Junhua Liu, Si Guo, Liwen Liu, Two-archive based evolutionary algorithm using adaptive reference direction and decomposition for many-objective optimization, in: 2018 14th International Conference on Computational Intelligence and Security (CIS), Hangzhou, China, IEEE, 2018, pp. 20–24.
[26] Li Ke, Renzhi Chen, Guangtao Fu, Xin Yao, Two-archive evolutionary algorithm for constrained multi-objective optimization, IEEE Trans. Evol. Comput. (2018).
[27] Yiping Liu, Gary G. Yen, Dunwei Gong, A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies, IEEE Trans. Evol. Comput. 23 (4) (2019) 660–674.
[28] Margarita Reyes Sierra, Carlos A. Coello Coello, Improving PSO-based multi-objective optimization using crowding, mutation and ∈-dominance, in: Carlos A. Coello Coello, Arturo Hernández Aguirre, Eckart Zitzler (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, Springer, Berlin Heidelberg, 2005, pp. 505–519.
[29] HongGui Han, LinLin Zhang, Ying Hou, JunFei Qiao, Adaptive candidate estimation-assisted multi-objective particle swarm optimization, Sci. China, Technol. Sci. 65 (8) (2022) 1685–1699.
[30] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[31] Marco Laumanns, Lothar Thiele, Kalyanmoy Deb, Eckart Zitzler, Combining convergence and diversity in evolutionary multiobjective optimization, Evol. Comput. 10 (3) (2002) 263–282.
[32] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable multi-objective optimization test problems, in: Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600), Honolulu, HI, USA, vol. 1, IEEE, 2002, pp. 825–830.
[33] S. Huband, P. Hingston, L. Barone, L. While, A review of multiobjective test problems and a scalable test problem toolkit, IEEE Trans. Evol. Comput. 10 (5) (2006) 477–506.
[34] Hisao Ishibuchi, Yu Setoguchi, Hiroyuki Masuda, Yusuke Nojima, Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes, IEEE Trans. Evol. Comput. 21 (2) (2017) 169–190.
[35] Xuewen Xia, Ling Gui, Fei Yu, Hongrun Wu, Bo Wei, Ying-Long Zhang, Zhi-Hui Zhan, Triple archives particle swarm optimization, IEEE Trans. Cybern. 50 (12) (2020) 4862–4875.
[36] Qiuzhen Lin, Songbai Liu, Qingling Zhu, Chaoyu Tang, Ruizhen Song, Jianyong Chen, Carlos A. Coello Coello, Ka-Chun Wong, Jun Zhang, Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems, IEEE Trans. Evol. Comput. 22 (1) (2018) 32–46.
[37] Qingling Zhu, Qiuzhen Lin, Zhihua Du, Zhengping Liang, Wenjun Wang, Zexuan Zhu, Jianyong Chen, Peizhi Huang, Zhong Ming, A novel adaptive hybrid crossover operator for multiobjective evolutionary algorithm, Inf. Sci. 345 (2016) 177–198.
[38] Li Wei, Yangtao Chen, Qian Cai, Cancan Wang, Ying Huang, Soroosh Mahmoodi, Dual-stage hybrid learning particle swarm optimization algorithm for global optimization problems, Complex Syst. Model. Simul. 2 (4) (2022) 288–306.
[39] Eckart Zitzler, Kalyanmoy Deb, Lothar Thiele, Comparison of multiobjective evolutionary algorithms: empirical results, Evol. Comput. 8 (2) (2000) 173–195.
[40] Qingfu Zhang, Hui Li, MOEA/D: a multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731.
[41] C.A. Coello Coello, M.S. Lechuga, MOPSO: a proposal for multiple objective particle swarm optimization, in: Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600), Honolulu, HI, USA, vol. 2, IEEE, 2002, pp. 1051–1056.
[42] Cai Dai, Yuping Wang, Miao Ye, A new multi-objective particle swarm optimization algorithm based on decomposition, Inf. Sci. 325 (2015) 541–557.
[43] Saúl Zapotecas Martínez, Carlos A. Coello Coello, A multi-objective particle swarm optimizer based on decomposition, in: Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation, Dublin Ireland, ACM, 2011, pp. 69–76.
[44] Qiuzhen Lin, Jianqiang Li, Zhihua Du, Jianyong Chen, Zhong Ming, A novel multi-objective particle swarm optimization with multiple search strategies, Eur. J. Oper. Res. 247 (3) (2015) 732–744.
[45] Ye Tian, Ran Cheng, Xingyi Zhang, Yaochu Jin, PlatEMO: a Matlab platform for evolutionary multi-objective optimization [educational forum], IEEE Comput. Intell. Mag. 12 (4) (2017) 73–87.
[46] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. Da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, IEEE Trans. Evol. Comput. 7 (2) (2003) 117–132.
[47] P.A.N. Bosman, D. Thierens, The balance between proximity and diversity in multiobjective evolutionary algorithms, IEEE Trans. Evol. Comput. 7 (2) (2003) 174–188.
[48] Robert G.D. Steel, A multiple comparison rank sum test: treatments versus control, Biometrics 15 (4) (1959) 560.
[49] Shueh-Hen Cheng, Hsuan Chang, Yih-Hang Chen, Hsi-Jen Chen, Yung-Kang Chao, Yu-Hsiang Liao, Computational fluid dynamics-based multiobjective optimization for catalyst design, Ind. Eng. Chem. Res. 49 (21) (2010) 11079–11086.