



A regularity model-based multi-objective estimation of distribution memetic algorithm with auto-controllable population diversity

Qiaoyong Jiang^{1,2} · Jianan Cui¹ · Lei Wang¹ · Yanyan Lin³ · Yali Wu⁴ · Xinhong Hei¹

Received: 30 December 2021 / Accepted: 31 January 2023 / Published online: 23 February 2023
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2023

Abstract

The regularity model-based multi-objective estimation of distribution algorithm (RM-MEDA) employs the local principal component analysis to split the population into several clusters, and each cluster is used to construct an affine subspace by combing the cluster center, principal components and additional Gaussian noise. However, such affine subspace greatly limits the sampling range of trail solutions, which will lead to the rapid loss of population diversity. To address this issue, an improved RM-MEDA with auto-controllable population diversity (RM-MEDA-AcPD) is suggested in this paper. In RM-MEDA-AcPD, the simplex crossover method is employed to extend the representation range of the affine subspace, the main purpose of which is to push solutions forward along the orthogonal direction of the affine subspace. In addition, a random noise model related to the evolution process is designed to replace the original Gaussian noise model, which reduces the risk of rapid loss of population diversity. In experimental studies, we have compared eight regularity property-based multi-objective evolutionary algorithms with the RM-MEDA-AcPD on benchmark problems with disconnected Pareto fronts. The experimental results demonstrate that the performance of RM-MEDA-AcPD significantly outperforms the other nine comparison algorithms in solving these test instances.

Keywords Multi-objective optimization · Affine subspace · Population diversity · Simplex crossover method · Random noise model

1 Introduction

In the realization of life, we often encounter multiple conflicting objectives that need to be optimized at the same time. These problems are usually known as multi-objective optimization problems (MOPs) and can be modeled as different mathematical models. This paper only considers the following MOPs with boundary constraints:

$$\min \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))$$

$$\text{s.t. } \mathbf{x} = (x_1, x_2, \dots, x_D) \in \Omega, \quad (1)$$

where $\Omega = \prod_{i=1}^D [L_i, U_i]$ represents the search space, $\mathbf{x} = (x_1, x_2, \dots, x_D)$ is a candidate solution of search space, each component of which is limited to the interval $[L_i, U_i]$, $i = 1, 2, \dots, D$, and $\mathbf{F}(\mathbf{x})$ is the objective vector.

Because of the conflict between the objectives, decision makers usually seek Pareto optimal solutions for the Eq. (1). Suppose that \mathbf{x}_1 and \mathbf{x}_2 are two solutions of the problem, \mathbf{x}_1 is said to be Pareto better than \mathbf{x}_2 ($\mathbf{x}_1 \prec \mathbf{x}_2$), if and only if $f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2)$ for each $i \in \{1, 2, \dots, m\}$ and $\mathbf{F}(\mathbf{x}_1) \neq \mathbf{F}(\mathbf{x}_2)$. $\mathbf{x}^* \in \Omega$ is called a Pareto optimal solution if no other solution $\mathbf{x} \prec \mathbf{x}^*$ exists. All the Pareto optimal solutions constitute the *Pareto set* (PS), and their images in the objective space constitute the *Pareto front* (PF).

To provide decision-makers with as many good decision-making programs as possible, various multi-objective optimization techniques have been developed over the past few decades. Among them, multi-objective evolutionary algorithms (MOEAs) have been proved to be an effective optimization method in solving MOPs since they have advan-

✉ Qiaoyong Jiang
jiangqiaoyong@126.com

¹ The School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China

² Shaanxi Key Laboratory of Network Computing and Security Technology, Xi'an 710048, China

³ The School of Information Engineering, Xi'an University, Xi'an 710065, China

⁴ The School of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China

tages over other methods in generating a set of Pareto optimal solutions to fit the PS of Eq. (1) in a single run. According to different environmental selection criteria, the current popular MOEAs generally include the Pareto domination-based [1,2], decomposition-based [3,4] and performance indicator-based approaches [5,6], in which the first kind of MOEAs employs the Pareto domination and some density methods to sort candidate solutions, the second kind of MOEAs uses the aggregate function methods to choose a better one, and the third kind of MOEAs utilizes the performance indicators to select the next generation population.

Among these MOEAs, estimation of distribution algorithms (EDAs) are considered as one of the most promising EAs [7]. They use an explicit probability distribution model instead of a genetic operator to extract statistical population information, and trial solutions are directly sampled from the built model. Obviously, the built probability distribution model directly determines the performance of EDAs. Based on different probability distribution models, many MOEDAs have been proposed [8]. For example, Mario and Edmondo proposed a Parzen-based continuous MOEDA (MOPED). In MOPED, the Parzen estimator was used to evaluate the probability density of Pareto optimal solutions in the objective space. Tatsuya et al. [10] developed a Voronoi-based EDA (VEDA) which employed a Voronoi diagram to construct stochastic models. Martíet al. [11] designed a growing neural gas network-based MOEDA (MB-GNG). Karshenas et al. [12] built a joint probabilistic modeling of variables and objectives using a multidimensional Bayesian network. Cheng et al. [13] put forward a group of Gaussian process-based inverse models to describe the non-dominated solutions in the object space (IM-MOE). Subsequently, two adaptive IM-MOEAs were developed [14,15]. Lin et al. [16] proposed an adaptive MOEDA (AMEDA). In AMEDA, a global or local multivariate Gaussian model was built for each solution sampling. Recently, Sun et al. [17] developed a reference line-based many-objective EDA (MaOEDA/RL), in which the estimation model was built using the reference lines to sample solutions with favorable proximity. In addition to the above, some MOEDAs based on other models have been proposed, such as Bayesian multi-objective optimization algorithm (BMOA) [18], mixture-based multi-objective iterated density estimation evolutionary algorithm (MIDEA) [19], multi-objective Bayesian optimization algorithm (mBOA) [20], multi-objective covariance matrix adaptation (MO-CMA) [21]. For more related work on EDAs, please refer to literature [22].

The mathematical programming theory [23] has proved that under mild smoothness conditions, the PS of a continuous MOP is a piecewise continuous $(m - 1)$ -D manifold. However, the regularity property of MOPs has not been exploited explicitly in the above mentioned MOEDAs. In view of this, Zhang et al. put forward a regularity model-

based MOEDA (RM-MEDA) [24] which employed the $(m - 1)$ -D local principal component analysis (LPCA) to construct the probability learning model in the decision space. Many simulation results have verified that the RM-MEDA can quickly approximate the PSs in addressing certain MOPs with variable linkages, but it also converges very slowly in some other MOPs. This contrast phenomenon exposes some limitations of RM-MEDA model, especially the rapid loss of population diversity. In RM-MEDA, the population is generated by sampling the affine subspace model determined by the base vector and the linear combination of the first $m - 1$ principal components and adding Gaussian noise. However, each base vector is pre-determined by the cluster center in each generation, which will greatly limit the sampling range of affine subspace. In addition, the Gaussian noise model used in RM-MEDA has nothing to do with the state of evolution, which makes it difficult for RM-MEDA to gain profit from the Gaussian noise model in maintaining population diversity. To solve the problem of rapid loss of diversity caused by the above two factors, we propose an improved RM-MEDA with auto-controllable population diversity (RM-MEDA-AcPD). This proposed method uses a simplex crossover (SPX) method [25] and a new noise model to control the population diversity, in which the former is used to increase the sampling range of affine subspace, while the latter is used to control the noise intensity.

The primary contributions of this paper are as follows.

- The SPX method is employed to generate a random point in each cluster instead of the cluster center. In addition, the scaling factor of SPX is dynamically adjusted in a linear decreasing manner. The purpose of these two modifications is to extend the representation range of the affine subspace and push solutions forward along the orthogonal direction of affine subspace.
- A new noise model related to the evolution process is designed to replace the original Gaussian noise model, in which the noise intensity is relatively large at the early stage of the evolution, and with the development of the evolution, the intensity of noise becomes smaller and smaller. The purpose of this design is to effectively balance the exploration and exploitation during the evolutionary process.
- Systematical experiments are conducted to study the performance and behavior of the proposed RM-MEDA-AcPD on a set of test problems. The experimental results demonstrate that the proposed algorithm has obvious advantages over eight state-of-the-art regularity property-based MOEAs. In addition, the sensitivity of algorithmic parameters and the CPU-time costs of the proposed algorithm are also experimentally investigated.

The rest of the paper is organized as follows. In Sect. 2, the basic idea of RM-MEDA and related works are briefly introduced. The details of the proposed modified SPX method and random noise model are presented in Sect. 3. Section 4 is devoted to experimental setup. In Sect. 5, the performance of RM-MEDA-AcPD is verified by comparing it with eight state-of-the-art regularity property-based MOEAs, and some analysis on the effectiveness of the RM-MEDA-AcPD is given. An additional discussion about RM-MEDA-AcPD is provided in Sect. 6. Finally, Sect. 7 concludes the paper and suggests possible future research.

2 RM-MEDA and related works

Algorithm 1: RM-MEDA Framework

```

1 Initialization:  $\text{Pop}(0)$  (initialized population),  $K$  (the number of clusters)
  and  $t = 0$ ;
2 while not terminate do
3   Cluster: Use the LPCA to partition the points of  $\text{Pop}(t)$  into  $K$  clusters;
4   Modeling: Build the probability distribution model  $\xi$  to model the
    distribution of the solutions in each cluster;
5   Reproduction: Generate a trail solution set  $Q$  by sampling the
    probability distribution model in each cluster;
6   Selection: Select the best first half solutions from  $Q \cup \text{Pop}(t)$  as the
    next population  $\text{Pop}(t+1)$ ;
7    $t = t + 1$ ;
8 end
9 Return  $\text{Pop}(t)$ .
```

2.1 RM-MEDA

According to the Karush–Kuhn–Tucker condition, it can be induced that under mild conditions, the PS of a continuous MOP is a piecewise continuous $(m-1)$ -D manifold. Based on the regularity property, RM-MEDA uses the LPCA to extract regularity patterns of the population in the decision space for building the following probability distribution model.

$$\xi = \zeta + \epsilon \quad (2)$$

where ζ is uniformly distributed over a $(m-1)$ -D piecewise continuous manifold, and ϵ is a zero-mean noise vector.

After that, new trail solutions are produced by sampling the model of Eq. (2) in the decision space. Finally, the non-dominated sorting-based mechanism is employed to select some elitist solutions for the next generation. The RM-MEDA framework is described in Algorithm 1, and the basic idea is shown in Fig. 1. Next, the details of RM-MEDA are presented.

- **Cluster** During the evolution, RM-MEDA maintains a population of N points $\text{Pop}(t) = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$,

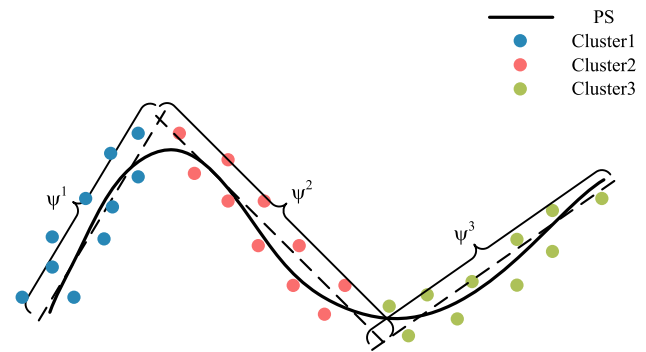


Fig. 1 Illustration of the basic idea of RM-MEDA. Individual solutions are scattered around the PS in the decision space

Algorithm 2: $C^1, C^2, \dots, C^K \leftarrow \text{Cluster}(\text{Pop}(t), K)$

```

1 Input:  $\text{Pop}(t)$  (the current population),  $K$  (the number of clusters);
2 Randomly initialize  $L_j^{m-1}, j = 1, 2, \dots, K$ , to be an affine  $(m-1)$ -D
  principle subspace containing a point randomly chosen from the  $\text{Pop}(t)$ ;
3 while not terminate do
4   Partition the points of  $\text{Pop}(t)$  into  $K$  clusters  $C^1, C^2, \dots, C^K : C^j =$ 
     $\{\mathbf{x} | \mathbf{x} \in \text{Pop}(t) \wedge \text{dist}(\mathbf{x}, L_j^{m-1}) \leq \text{dist}(\mathbf{x}, L_i^{m-1}), \forall i \neq j\}$ ;
5   Update the  $L_j^{m-1}, j = 1, 2, \dots, K$ , by the mean and the covariance
    matrix of the points in  $C^j$ ;
6 end
7 Output:  $C^1, C^2, \dots, C^K$ .
```

where t is the current generation. The main aim of modeling is to better depict the ζ . For the sake of simplicity, assume that the ζ is composed of K manifolds: $\psi^1, \psi^2, \dots, \psi^K$, each of which is a $(m-1)$ -D hyperrectangle. An illustration is shown in Fig. 1. To achieve this purpose, RM-MEDA partitions the points in $\text{Pop}(t)$ into K disjoint clusters C^1, C^2, \dots, C^K by minimizing the following error function

$$\begin{aligned}
 \min \quad & \sum_{j=1}^K \sum_{\mathbf{x} \in C^j} \text{dist}(\mathbf{x}, L_j^{m-1})^2 \\
 \text{s.t.} \quad & L_j^{m-1} = \{\mathbf{p} \in \mathbb{R}^D | \mathbf{p} = \bar{\mathbf{x}}_j \\
 & + \sum_{i=1}^{m-1} \theta_i U_i^j, \theta_i \in \mathbb{R}, i = 1, 2, \dots, m-1\}. \quad (3)
 \end{aligned}$$

where L_j^{m-1} is the affine $(m-1)$ -D principle subspace of the points in C^j , $\text{dist}(\mathbf{x}, L_j^{m-1})$ is the Euclidean distance from \mathbf{x} to its project in L_j^{m-1} , $\bar{\mathbf{x}}_j$ is the mean of the points in C^j , and the i th principal component U_i^j is a unity eigenvector associated with the i th largest eigenvalue of the covariance matrix of the points in C^j . The details can be found in Algorithm 2.

- **Modeling** Based on the above clustering results, RM-MEDA firstly calculates the range of the projections of

the points of each cluster C^j in the i th principal component U_i^j :

$$a_i^j = \min_{\mathbf{x} \in C^j} (\mathbf{x} - \bar{\mathbf{x}}_j)^T U_i^j \quad (4)$$

and

$$b_i^j = \max_{\mathbf{x} \in C^j} (\mathbf{x} - \bar{\mathbf{x}}_j)^T U_i^j \quad (5)$$

for $i = 1, 2, \dots, m-1$. Then, the j th affine subspace ψ^j can be represented as

$$\begin{aligned} \psi^j &= \left\{ \mathbf{x} \in \mathbb{R}^D \mid \mathbf{x} = \bar{\mathbf{x}}_j + \sum_{i=1}^{m-1} \alpha_i U_i^j, \alpha_i^j - 0.25(b_i^j - a_i^j) \right. \\ &\quad \left. \leq \alpha_i \leq a_i^j + 0.25(b_i^j - a_i^j) \right\} \end{aligned} \quad (6)$$

and

$$\sigma_j = \frac{1}{D - m + 1} \sum_{i=m}^D \lambda_i^j \quad (7)$$

where λ_i^j is the i th largest eigenvalue of the covariance matrix of the points in C^j .

- **Reproduction** To sample a trail solution from the built model, an integer $\tau \in \{1, 2, \dots, K\}$ is firstly generated with

$$Prob(\tau = k) = \frac{vol(\psi^k)}{\sum_{j=1}^K vol(\psi^j)} \quad (8)$$

where $vol(\psi^k)$ is the $(m-1)$ -D volume of ψ^k . Then, a point \mathbf{x}' is uniformly randomly generated from the ψ^τ , and a noise vector ε' is generated from $N(0, \sigma_\tau \mathbf{I})$. Finally, a new trail solution \mathbf{x} is obtained by $\mathbf{x}' + \varepsilon'$.

2.2 Related works

Since the RM-MEDA was proposed by Zhang et al., many improved RM-MEDA versions (RM-MEDAs) have been designed. Wang et al. observed that the number of clusters K was problem-dependent and had a significant effect on the performance of RM-MEDA. Motivated by the observation above, they presented a reducing redundant cluster operator (RRCO) to build more precise model during the evolution [26]. Based on the research route of the problem, Shi et al. [27] suggested a full variate Gaussian-model to maintain the population diversity of RM-MEDA. Dong et al. [28] found that the setting of the extension scale in RM-MEDA was also problem-dependent. To solve this problem, differential evolution was used to mutate the projections of

the parent solutions in the latent space to produce promising candidate offspring solutions. Li et al. [29] pointed out that the RM-MEDA was easy to generate poor solutions when the distribution of population had no obvious regularity. To overcome this limit, a local learning method was proposed to generate partial solutions in the neighborhood of elitist solutions. To enhance the exploration ability of RM-MEDA, Zhou et al. [30] put forward a global RM-MEDA, in which the initial population was inserted into several globally Pareto optimal solutions, and the biased crossover combined globally statistical information in the current population and the location information of some best solutions. Subsequently, Zhou et al. [31] further generalized the idea of RM-MEDA. Luo et al. [32] designed a RM-MEDA with two steps training method, in which at the first step the K -means method was employed to divide the points in population into primary K disjoint clusters, and at the second step the LPCA method was further employed for each cluster. Recently, Lin et al. [33] put forward a mixed learning model for RM-MEDA and IM-MOEA. In addition, the RM-MEDA is applied to solve MOPs with noise [34], many-objective optimization problems (MaOPs) [35] and dynamic MOPs [36].

Based on the above introduction, we notice that the improvements of population diversity of RM-MEDA mainly focus on the adaptive setting of the number of clusters K , the mixture of clustering models and genetic operators. However, the population diversity of RM-MEDA depends largely on the representation range of the constructed affine subspace which combines the cluster center, principal components and additional Gaussian noise. Therefore, how to effectively expand the representation range of affine subspace and control noise intensity is the key to improve the search efficiency of RM-MEDA. Driven by this, we propose a new representation model of affine subspace for RM-MEDA.

3 Proposed algorithm

3.1 Advantages and disadvantages of RM-MEDA

For many continuous m -D MOPs with the regularity property, their PS is a piecewise continuous $(m-1)$ -D manifold. Based on this characteristic, RM-MEDA employed the LPCA to reduce the search dimensionality. For example, the data set of Fig. 2b is epistasis between the variables, they can be mapped onto a principal axis by minimizing the linear correlations (see Fig. 2a). Specifically, the RM-MEDA method makes full use of the fact that most of the information in the data is concentrated in a few principal components in many cases, and selecting fewer principal components to represent data can not only be used as dimension reduction of features, but also can capture the linear correlation of data. Therefore, compared with other existing methods, the RM-

Fig. 2 Sample data sets: **a** data without epistasis, **b** data with epistasis

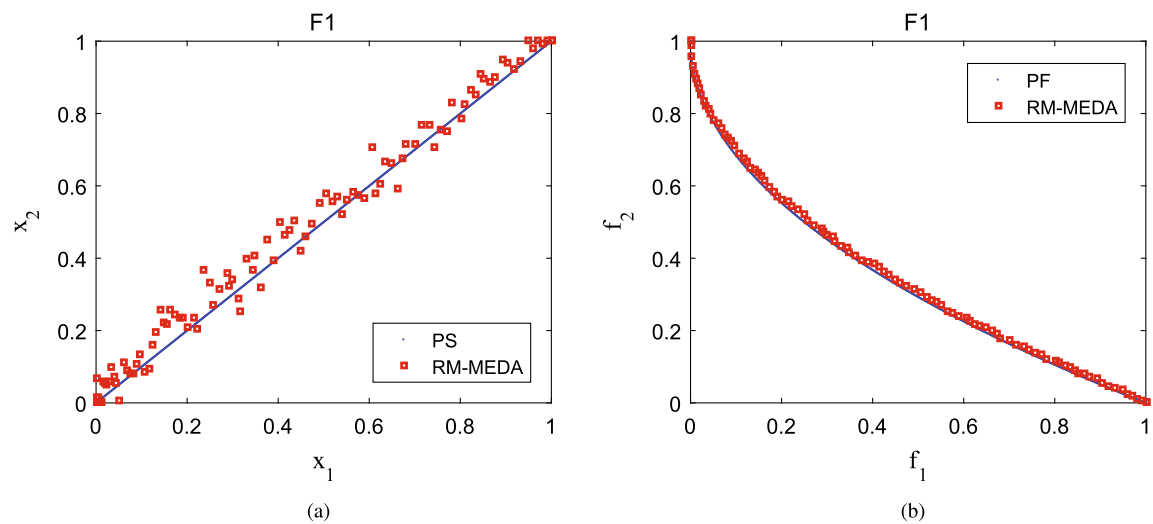
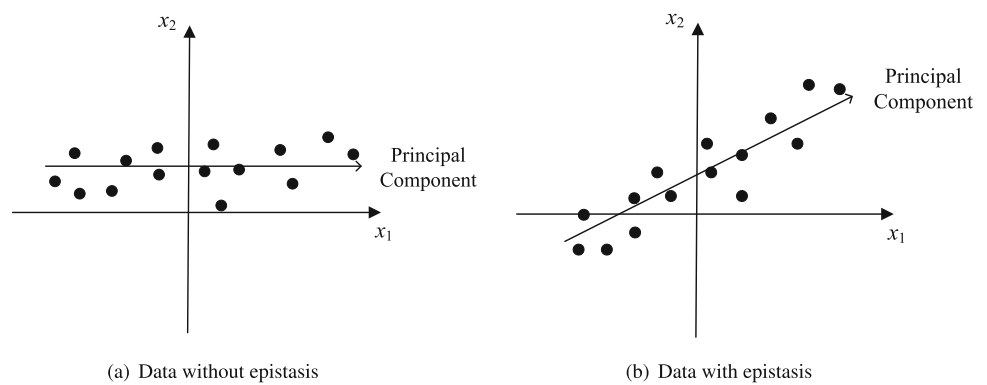


Fig. 3 The F1 simulation result obtained by RM-MEDA with 5000 function evaluations, where the red points denote the obtained final non-dominated solutions, while the blue points denote the PF (colour figure online)

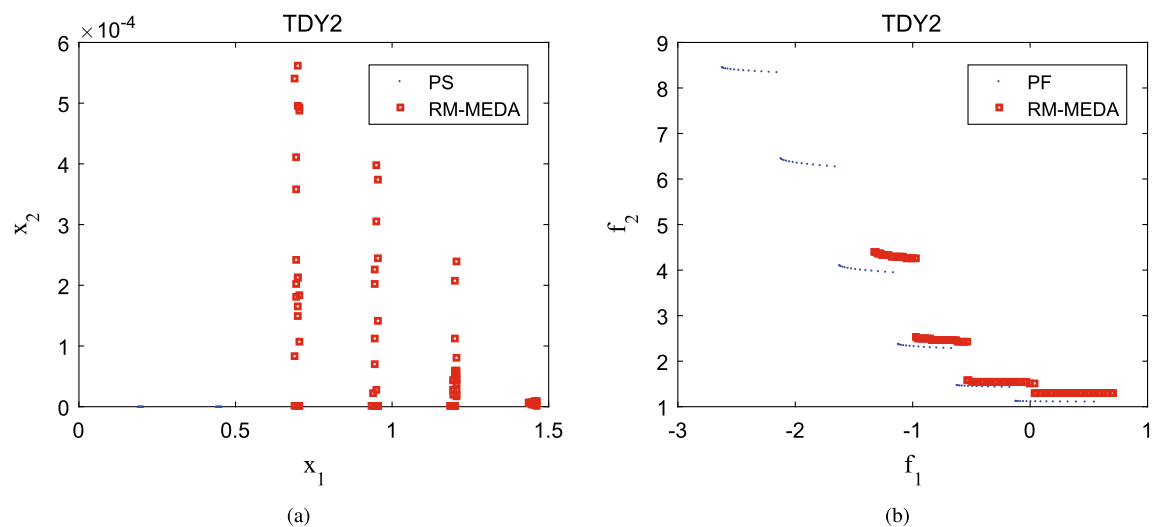


Fig. 4 The TDY2 simulation result obtained by RM-MEDA with 300,000 function evaluations, where the red points denote the obtained final non-dominated solutions, while the blue points denote the PF (colour figure online)

Table 1 The description of six test functions TDY1-TDY6

Name	Search space	Objectives	Characteristics of PF
TDY1	$[0, 1.5]^D$	$f_1(\mathbf{x}) = -h - \cos(\frac{h\pi}{0.6}) + \frac{2}{J_1} \sum_{j \in J_1} (x_j - \sin(6\pi x_1 + \frac{j\pi}{D}))^2$ $f_2(\mathbf{x}) = -1 + h - \cos(\frac{h\pi}{0.6}) + \frac{2}{J_2} \sum_{j \in J_2} (x_j - \sin(6\pi x_1 + \frac{j\pi}{D}))^2$ where $h = \frac{x_1 + x_2}{2}$, $J_1 = \{j j \text{ is odd and } 3 \leq j \leq D \text{ and } J_2 = \{j j \text{ is even and } 4 \leq j \leq D\}$.	Three disconnected curves
TDY2	$[0, 1.46]^D$	$f_1(\mathbf{x}) = 2x_1 + 4 \sin(8\pi x_1) + g^2(\mathbf{x})$ $f_2(\mathbf{x}) = 8 \exp(-2x_1^2) + g^2(\mathbf{x})$ where $g(\mathbf{x}) = 1 + \frac{9}{D-1} \sum_{i=2}^D x_i$.	Six disconnected curves
TDY3	$[0, 2]^D$	$f_1(\mathbf{x}) = x_1 + 2 \sin(8\pi x_1) + \sin(4\pi x_1) + g^2(\mathbf{x})$ $f_2(\mathbf{x}) = 8 \exp(-2x_1^2) + 4 \exp(-2(x_1 - 0.2)^2) + g^2(\mathbf{x})$ where $g(\mathbf{x}) = 1 + \frac{9}{D-1} \sum_{i=2}^D x_i$.	Four disconnected curves
TDY4	$[0, 2]^D$	$f_1(\mathbf{x}) = -2x_1 + g^2(\mathbf{x})$ $f_2(\mathbf{x}) = 2x_1 + 32 \sin^2(2\pi x_1) + g^2(\mathbf{x})$ where $g(\mathbf{x}) = 1 + \frac{9}{D-1} \sum_{i=2}^D x_i$.	One isolated point and four disconnected curves
TDY5	$[0.6, 4.6] \times [0, 3]^{D-1}$	$f_1(\mathbf{x}) = x_1 + g^2(\mathbf{x})$ $f_2(\mathbf{x}) = \exp(-y) + \sin(2\pi y) + g^2(\mathbf{x})$ where $y = x_1^{0.7}$ and $g(\mathbf{x}) = 1 + \frac{9}{D-1} \sum_{i=2}^D x_i$.	Three disconnected curves
TDY6	$[0.7, 4.6] \times [0, 3]^{D-1}$	$f_1(\mathbf{x}) = x_1 + g^2(\mathbf{x})$ $f_2(\mathbf{x}) = -0.5y + \sin(4\pi y) + g^2(\mathbf{x})$ where $y = \log x_1$ and $g(\mathbf{x}) = 1 + \frac{9}{D-1} \sum_{i=2}^D x_i$.	Four disconnected curves

MEDA is better at solving the single-mode continuous MOPs with linear PS or piecewise approximate linear PS, since the population can approach the PS quickly and extend the population to the whole PS using LPCA technology. However, the RM-MEDA loses its advantages for the discontinuous MOPs. This view is supported by the following MOPs [24]:

$$\mathbf{F1}(\mathbf{x}) = \begin{cases} f_1(\mathbf{x}) = x_1 \\ f_2(\mathbf{x}) = g(\mathbf{x})[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})}] \\ g(\mathbf{x}) = 1 + 9 \left(\sum_{i=2}^D (x_i - x_1)^2 \right) / (D - 1), \mathbf{x} \in [0, 1]^{30} \end{cases} \quad (9)$$

From Fig. 3a, b, we observe that the F1 problem can be solved very well by RM-MEDA, although the computational cost is very little (5000 FES). However, Fig. 4a, b show a negative example TDY2 which is defined in Table 1. From them, we can see that the TDY2 problem cannot be solved well by RM-MEDA, even if the computation cost is large enough (300000FES). This phenomenon exposes some limitations of RM-MEDA:

(1) The RM-MEDA is prone to lose population diversity, which may lead to the loss of some PFs during the search process. As shown in Fig. 4, the RM-MEDA found only 4 of the 6 disconnected curves;

(2) If all solutions are close to one line segment in the decision space (see Fig. 5), the RM-MEDA will lose its search ability in the orthogonal direction. As an example, in Fig. 4a, the ordinate values of all solutions are close to zero, which means that these solutions are distributed near a straight line which is not the true PS. However, as shown in Fig. 4b, these solutions still cannot approach the true PF perfectly, which implies that the RM-MEDA loses no search power to approach the true PS.

To overcome these problems, we propose a new representation model of the affine subspace to expand its representation range by modifying two components: the cluster center and Gaussian noise. These two components will be detailed in the following subsection.

3.2 Modification of the affine subspace model

According to Eq. (6), for each cluster the affine subspace ψ is determined by the cluster center $\bar{\mathbf{x}}$ and the linear combination of the first $m - 1$ principal components U_1, U_2, \dots, U_{m-1} . Since the cluster center $\bar{\mathbf{x}}$ is pre-determined in each generation, the sampling range of affine subspace ψ will be greatly limited. However, this limitation will be eased by increasing the flexibility of the affine subspace model representation. To achieve this goal, the SPX method [25] is used to generate a random point in each cluster instead of the cluster center $\bar{\mathbf{x}}$.

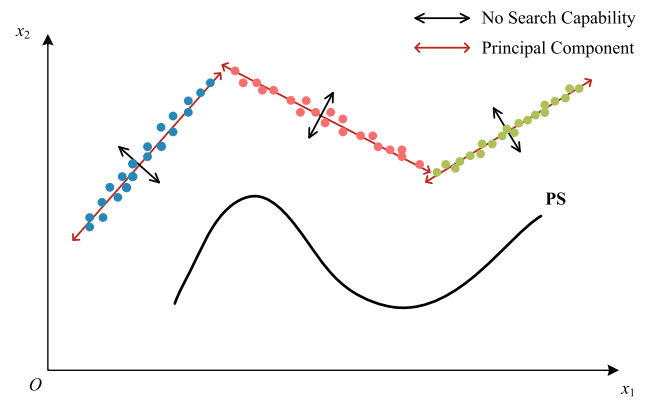


Fig. 5 The RM-MEDA loses its search power in the orthogonal direction. The red double arrows denote the obtained principal components, while the blue double arrows denote their orthogonal direction (colour figure online)

In \mathbb{R}^D , $D + 1$ points $\mathbf{p}_i, i = 1, 2, \dots, D + 1$ form a simplex. For convenience, in a two-dimensional search space three points $\mathbf{p}_1, \mathbf{p}_2$, and \mathbf{p}_3 form a simplex. We expand or shrink this simplex in each direction by ρ ($\rho \geq 0$) times, where ρ is a scaling factor. Let $\mathbf{o} = \frac{1}{3} \sum_{i=1}^3 \mathbf{p}_i$ and $\tilde{\mathbf{p}}_i = \rho(\mathbf{p}_i - \mathbf{o})$. Therefore, $\tilde{\mathbf{p}}_1, \tilde{\mathbf{p}}_2$, and $\tilde{\mathbf{p}}_3$ constitute a new simplex. Then, a point \mathbf{s} is randomly sampled from the simplex, i.e., $\mathbf{s} = \mathbf{o} + \sum_{i=1}^3 r_i \tilde{\mathbf{p}}_i$, where $\sum_{i=1}^3 r_i = 1$ and $0 < r_i < 1, i = 1, 2, 3$. This process for generating offspring can be generalized into the D -dimensional search space. Therefore, the affine subspace model can be modified as

$$\tilde{\psi}^j = \left\{ \mathbf{x} \in \mathbb{R}^D \mid \mathbf{x} = \mathbf{s}_j + \sum_{i=1}^{m-1} \alpha_i U_i^j, a_i^j - 0.25(b_i^j - a_i^j) \leq \alpha_i \leq a_i^j + 0.25(b_i^j - a_i^j) \right\} \quad (10)$$

where \mathbf{s}_j is sampled from the simplex constructed by $C^j, j = 1, 2, \dots, K$.

In general hybrid evolutionary algorithms, the SPX method is usually used as an independent local search technology to accelerate the convergence of the algorithm. In contrast, here the SPX method is used to generate basis vectors to expand the representation range of affine subspace, which is closely related to the scaling factor ρ . If $\rho > 1$, the simplex $\mathbf{o} - \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3$ is expanded as the simplex $\mathbf{o} - \tilde{\mathbf{p}}_1 \tilde{\mathbf{p}}_2 \tilde{\mathbf{p}}_3$, which will make the sampling range of affine subspace in a relatively large band region (see Fig. 6a). When the basis vector is located near the lower left corner of the simplex $\mathbf{o} - \tilde{\mathbf{p}}_1 \tilde{\mathbf{p}}_2 \tilde{\mathbf{p}}_3$, the sampled points are far away from the PS. When the basis vector is located near the lower right corner of the simplex $\mathbf{o} - \tilde{\mathbf{p}}_1 \tilde{\mathbf{p}}_2 \tilde{\mathbf{p}}_3$, the sampled points are close to the PS. Therefore, the SPX method can provide search power in the orthogonal direction for RM-MEDA, especially in the middle and early stages of evolution. If $\rho < 1$, the sim-

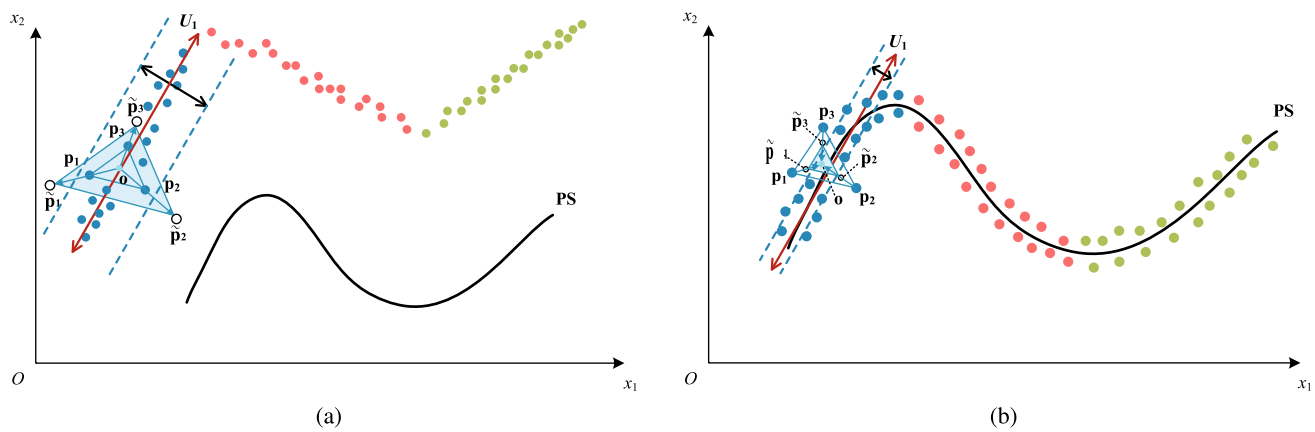


Fig. 6 Adaptive sampling from the simplex during evolution: **a** sampling from the expanded simplex $o - \tilde{p}_1\tilde{p}_2\tilde{p}_3$ ($\rho > 1$), **b** sampling from the shrunken simplex $o - \tilde{p}_1\tilde{p}_2\tilde{p}_3$ ($\rho < 1$)

plex $o - p_1p_2p_3$ is shrunk as the simplex $o - \tilde{p}_1\tilde{p}_2\tilde{p}_3$, and the sampling range of affine subspace is located in a relatively small band region (see Fig. 6b). When the basis vector is located at the shrunken simplex $o - \tilde{p}_1\tilde{p}_2\tilde{p}_3$ in the late stage of evolution, the sampled points may be closer to the PS. In particular, when the parameter ρ is equal to 0, this sampling process is the same as that of RM-MEDA. To effectively control the representation range of affine subspace, we dynamically adjust the ρ decreasing linearly from 5 to 0, which is formulated as follows:

$$\rho(t) = 5(1 - t/T) \quad (11)$$

where t is the current generation, and T is the maximum generation. Therefore, as the evolution goes on, the representation range of affine subspace changes from large to small, which is conducive to controlling the diversity of population.

3.3 Random noise model

In RM-MEDA, Gaussian noise is further added to each affine subspace model, which can not only better approximate part of the PS, but also provide diversity for further searching. Therefore, the noise is an essential component and plays a very important role for RM-MEDA. We note that the Gaussian noise model used in RM-MEDA has nothing to do with the state of evolution. This makes it difficult for RM-MEDA to gain profit from the Gaussian noise model in solving some complex MOPs. The main reason is that the Gaussian noise model is not very useful for exploring a wider area. Therefore, a more reasonable noise model should be able to effectively balance exploration and exploitation during the evolutionary process. That is, at the early stage of the evolution, the intensity of noise is relatively large, which causes a large-scale disturbance on the evolutionary system. With the development of the evolution, the intensity of noise becomes smaller

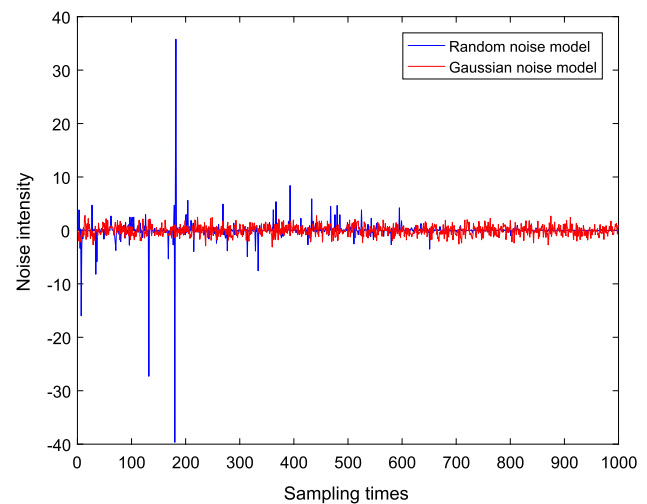


Fig. 7 The simulation results of two noise models, where the red points denote the Gaussian noise, while the blue points denote the random noise (colour figure online)

and smaller, which plays a fine-tuning role on the evolutionary system.

The MOEA/D-M2M [37] is a very popular and efficient evolutionary multi-objective algorithm framework, which was originally proposed to solve unbalanced MOPs [38]. Such problems require evolutionary multi-objective algorithm to maintain good population diversity in decision space. In MOEA/D-M2M, the division technology based on reference vectors is mainly used to maintain the diversity of the population. However, the mutation strategy also plays an important role in maintaining population diversity. In MOEA/D-M2M, each individual is mutated by adding noise whose intensity is changed from large to small with the progress of iteration. Inspired by this variation, we use a similar method of increasing noise to enhance the diversity of the population. The noise model is formulated in Eq.(12).

$$\gamma(t) = (2 * rand_1 - 1) * (1 - (2 * rand_2 - 1)^{-(1-\frac{t}{T})^{0.7}}) \quad (12)$$

where $rand_1$ and $rand_2$ are random numbers between 0 and 1, t is the current generation, and T is the maximum generation.

The defined $\gamma(t)$ can gradually approach zero with increasing evolutionary generation, which is similar to simulated annealing. Figure 7 shows the simulation comparison of the two models by sampling 1000 times. From it, we can clearly see that in the early stage of the evolution, in many cases, the intensity of random noise is obviously higher than that of the Gaussian noise, and in the later stage of the evolution, the intensity is reversed. Therefore, the random noise model is more adaptive to the evolution process of RM-MEDA than the Gaussian noise model.

3.4 Main framework

The main process of RM-MEDA-AcPD is presented in Algorithm 3. At the beginning, the population **Pop** is initialized with random sampling in the decision space. In the main loop, we first apply LPCA to partition the **Pop** into K clusters: C^1, C^2, \dots, C^K in Line 11. Then, some cluster C^j is

selected by the roulette method in Line 14, the selection probability of which is proportional to the volume of the affine subspace. In Line 15, a point s_j is sampled from the simplex constructed by C^j using the MSPX method. Afterward, a new point \mathbf{x} is generated by sampling the modified affine subspace model and adding random noise in Lines 16–17. Finally, Lines 22–31 do the environmental selection using the non-dominated sorting method.

From the main framework given above, we can see that compared with the RM-MEDA, the computational cost of the proposed RM-MEDA-AcPD lies in the SPX method. The computational complexity of the SPX method for producing an offspring is only $O(D)$. Therefore, the RM-MEDA-AcPD does not increase the computational complexity. This is another important reason why we use the SPX method, in addition to its simple operation.

4 Experimental setup

4.1 Test problems

To test the effectiveness of RM-MEDA-AcPD, six benchmark problems with disconnected PFs [39] are tested. Specifically, TDY1–TDY6 with 30 decision variables are employed for empirical studies. The characteristics of six test problems are list in Table 1.

4.2 Performance metrics

In our experimental studies, the following two widely used performance metrics are considered. Both of them can simultaneously measure the convergence and diversity of the obtained solutions in the objective space.

- Inverted generational distance (IGD) [24] is a distance-based metric, which is defined as

$$IGD(P^*, P) = \frac{1}{|P^*|} \sum_{i=1}^{|P^*|} d_i(\mathbf{v}_i, P) \quad (13)$$

where P is an approximation set of the true PF and P^* is a subset of the true PF. The $d_i(\mathbf{v}_i, P)$ is the minimum Euclidean distance between the i th solution \mathbf{v}_i in P^* and all solution set members in P . The smaller is the IGD value, the better is the quality of P for approximating P^* .

- Hypervolume (HV) metric [40] measures the volume of solutions that is dominated by the approximation set P . Given a reference point $\mathbf{z}^r = (z_1^r, z_1^r, \dots, z_m^r)$, $HV(\mathbf{z}^r, P)$ can be computed as

Algorithm 3: The procedure of RM-MEDA-AcPD

```

1 Input:  $N$ : the population size.
2  $K$ : the number of clusters.
3  $T$ : the maximum generation.
4  $MaxFES$ : the maximum number of function evaluations.
5 %Initialization
6 Create  $N$  points  $\mathbf{Pop}_i(0)$  ( $i = 1, 2, \dots, N$ ) randomly;
7 Evaluate the objective vectors  $\mathbf{F}(\mathbf{Pop}_1(0)), \mathbf{F}(\mathbf{Pop}_2(0)), \dots, \mathbf{F}(\mathbf{Pop}_N(0))$ ;
8 Set  $FES = N$ ,  $t = 0$ ,  $T = \lfloor \frac{MaxFES}{N} \rfloor$ ;
9 while  $FES \leq MaxFES$  do
10   %Cluster
11   Use the LPCA to divide the points of  $\mathbf{Pop}(t)$  into  $K$ 
     clusters:  $C^1, C^2, \dots, C^K$  by Algorithm 2;
12   %Modeling and Reproduction
13   for  $i = 1, 2, \dots, N$  do
14     Select a cluster  $C^j$  by roulette method based on Eq. (8);
15     Generate a point  $s_j$  from the simplex constructed by  $C^j$  using the
       MSPX method of Sect. 3.2;  $\leftarrow$ 
16     Sample a point  $\mathbf{x}$  from the modified affine subspace model  $\tilde{\psi}^j$  by
       Eq. (10) and Eq. (11);  $\leftarrow$ 
17     Revise  $\mathbf{x}$  by adding the random noise of Eq. (12);  $\leftarrow$ 
18      $\mathbf{Pop}_{N+i}(t) = \mathbf{x}$ ;
19      $FES = FES + 1$ ;
20   end
21   %Environment selection
22    $A = \emptyset$ ,  $i = 1$ ;
23    $\mathcal{F}$  = non-dominated-sort ( $\mathbf{Pop}(t)$ );
24   while  $|A| + |\mathcal{F}_i| < N$  do
25      $A = A \cup \mathcal{F}_i$ ,  $i = i + 1$ ;
26   end
27   while  $|\mathcal{F}_i| > N - |A|$  do
28     Distance-assignment ( $\mathcal{F}_i$ );
29     Remove the element in  $\mathcal{F}_i$  with the smallest crowding distance;
30   end
31    $\mathbf{Pop}(t+1) = A \cup \mathcal{F}_i$ ;
32    $t = t + 1$ ;
33 end
34 Output: PS and PF.

```

Table 2 The statistical results (Mean (Std)) of the nine algorithms over 51 independent runs on the TDY1-TDY6 with 30D in terms of IGD and HV metrics

Algorithms	Results (IGD)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA-AcPD	Mean	9.34E−04	1.44E−02	1.42E−02	1.10E−02	3.15E−03	9.24E−03
	Std	(9.49E−05)	(2.19E−03)	(1.27E−03)	(1.37E−03)	(2.33E−04)	(4.61E−04)
	<i>p</i>	/	/	/	/	/	/
	Significance	/	/	/	/	/	/
RM-MEDA	Mean	1.04E−03	1.46E+00	1.98E+00	8.33E−01	7.85E−01	3.34E−01
	Std	(3.01E−04)	(1.14E+00)	(3.78E−01)	(9.54E−01)	(2.98E−01)	(2.28E−01)
	<i>p</i>	5.64E−02	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18
	Significance	≈	+	+	+	+	+
IRM-MEDA	Mean	2.35E−03	2.06E+00	2.03E+00	1.75E+00	6.80E−01	2.28E−01
	Std	(3.88E−03)	(9.07E−01)	(2.92E−01)	(5.34E−01)	(1.84E−01)	(2.61E−01)
	<i>p</i>	1.44E−07	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.78E−17
	Significance	+	+	+	+	+	+
RM-MEDA+LL	Mean	1.19E−03	2.30E−01	1.72E+00	1.74E−01	6.68E−01	2.69E−01
	Std	(3.22E−04)	(3.67E−01)	(7.66E−01)	(9.56E−02)	(3.02E−01)	(1.46E−01)
	<i>p</i>	1.08E−09	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18
	Significance	+	+	+	+	+	+
FRM-MEDA	Mean	7.76E−03	4.52E+00	4.15E+00	5.65E+00	5.79E+00	5.96E+00
	Std	(4.68E−03)	(1.91E+00)	(2.31E+00)	(2.97E+00)	(3.65E+00)	(3.39E+00)
	<i>p</i>	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18
	Significance	+	+	+	+	+	+
IM-MOEA	Mean	6.37E−02	1.60E−01	9.98E−02	1.58E−01	2.52E−02	5.48E−02
	Std	(5.08E−02)	(6.72E−02)	(2.08E−01)	(1.93E−02)	(9.48E−02)	(4.21E−03)
	<i>p</i>	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18
	Significance	+	+	+	+	+	+
A-IM-MOEA	Mean	4.81E−02	1.41E−01	6.07E−02	2.12E−01	2.58E−01	5.44E−02
	Std	(1.76E−02)	(1.89E−02)	(5.75E−03)	(4.35E−01)	(6.28E−01)	(7.97E−03)
	<i>p</i>	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18
	Significance	+	+	+	+	+	+
MOEA/D	Mean	1.01E−01	7.61E−01	2.35E+00	6.09E−01	1.87E+00	7.50E−01
	Std	(1.46E−01)	(4.13E−01)	(2.38E−01)	(5.89E−01)	(2.94E−01)	(1.58E−01)
	<i>p</i>	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18	3.30E−18
	Significance	+	+	+	+	+	+
MOEA/D-M2M	Mean	2.72E−01	5.34E−01	8.42E−01	3.63E−01	5.24E−01	8.18E−02
	Std	(9.78E−03)	(4.15E−02)	(6.83E−03)	(2.31E−02)	(9.26E−04)	(8.03E−03)
	<i>p</i>	3.30E−18	3.30E−18	4.99E−18	3.30E−18	1.07E−17	3.30E−18
	Significance	+	+	+	+	+	+
Algorithms	Results (HV)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA-AcPD	Mean	3.05E−01	6.32E−01	8.48E−01	3.83E−01	5.27E−01	8.32E−01
	Std	(1.34E−05)	(1.02E−03)	(6.11E−04)	(8.53E−04)	(5.60E−04)	(3.43E−04)
	<i>p</i>	/	/	/	/	/	/
	Significance	/	/	/	/	/	/

Table 2 continued

Algorithms	Results (HV)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA	Mean	3.05E-01	4.29E-01	7.06E-01	2.72E-01	3.65E-01	7.43E-01
	Std	(7.73E-05)	(1.65E-01)	(4.93E-02)	(1.06E-01)	(8.89E-02)	(4.60E-02)
	<i>p</i>	2.06E-07	3.30E-18	3.30E-18	3.29E-18	3.30E-18	3.30E-18
	Significance	+	+	+	+	+	+
IRM-MEDA	Mean	3.05E-01	3.94E-01	7.25E-01	1.95E-01	3.99E-01	7.88E-01
	Std	(1.20E-01)	(4.49E-02)	(5.79E-02)	(6.94E-02)	(6.94E-02)	(6.54E-02)
	<i>p</i>	4.43E-18	3.30E-18	3.30E-18	3.30E-18	3.30E-18	4.14E-15
	Significance	+	+	+	+	+	+
RM-MEDA+LL	Mean	3.05E-01	5.86E-01	7.34E-01	3.43E-01	3.70E-01	7.36E-01
	Std	(1.43E-04)	(4.66E-02)	(6.68E-02)	(1.98E-02)	(8.08E-02)	(4.30E-02)
	<i>p</i>	3.50E-18	3.30E-18	3.30E-18	3.30E-18	3.30E-18	3.30E-18
	Significance	+	+	+	+	+	+
FRM-MEDA	Mean	3.01E-01	3.61E-02	2.10E-01	2.11E-02	3.55E-03	2.66E-02
	Std	(2.55E-03)	(8.87E-02)	(2.59E-01)	(5.70E-02)	(2.53E-02)	(1.15E-01)
	<i>p</i>	3.30E-18	1.86E-19	2.24E-18	2.31E-19	1.99E-20	9.05E-20
	Significance	+	+	+	+	+	+
IM-MOEA	Mean	2.92E-01	6.22E-01	8.45E-01	3.65E-01	5.25E-01	8.23E-01
	Std	(7.22E-03)	(4.76E-03)	(5.07E-04)	(1.25E-02)	(1.12E-03)	(3.25E-03)
	<i>p</i>	3.30E-18	3.30E-18	3.30E-18	3.30E-18	5.10E-14	3.30E-18
	Significance	+	+	+	+	+	+
A-IM-MOEA	Mean	2.94E-01	6.21E-01	8.45E-01	3.54E-01	4.77E-01	8.23E-01
	Std	(4.50E-03)	(6.88E-03)	(1.58E-03)	(5.14E-02)	(1.19E-01)	(5.33E-03)
	<i>p</i>	3.30E-18	3.30E-18	4.70E-18	3.30E-18	1.81E-14	3.30E-18
	Significance	+	+	+	+	+	+
MOEA/D	Mean	2.77E-01	6.08E-01	7.69E-01	3.40E-01	1.79E-01	8.17E-01
	Std	(5.71E-02)	(2.24E-02)	(2.20E-02)	(6.23E-02)	(8.18E-02)	(1.32E-02)
	<i>p</i>	3.30E-18	1.44E-04	3.30E-18	5.19E-12	3.30E-18	3.30E-18
	Significance	+	+	+	+	+	+
MOEA/D-M2M	Mean	2.72E-01	5.34E-01	8.42E-01	3.63E-01	5.24E-01	8.18E-01
	Std	(9.78E-03)	(4.15E-02)	(6.83E-03)	(2.31E-02)	(9.26E-04)	(8.03E-03)
	<i>p</i>	3.30E-18	3.30E-18	3.30E-18	3.30E-18	3.30E-18	3.30E-18
	Significance	+	+	+	+	+	+

The best result for each problem is highlighted in bold

$$HV(\mathbf{z}^r, P) = L\left(\cup_{\mathbf{x} \in P} [f_1(\mathbf{x}), z_1^r] \times [f_2(\mathbf{x}), z_2^r] \times \cdots \times [f_m(\mathbf{x}), z_m^r]\right) \quad (14)$$

where L is the Lebesgue measure. The larger is the HV value, the better is the quality of P for approximating P^* . In our experiments, the obtained solutions is normalized, and the \mathbf{z}^r is set to $(1, 1, \dots, 1)$.

- Maximum spread (MS) metric [41] measures the coverage of P to P^* . It is defined as

$$MS = \sqrt{\frac{1}{m} \sum_{i=1}^m \left\{ \frac{\min(f_i^{max}, F_i^{max}) - \min(f_i^{min}, F_i^{min})}{F_i^{max} - F_i^{min}} \right\}^2} \quad (15)$$

where f_i^{max} and f_i^{min} are the maximum and minimum values of P in the i th objective, respectively. F_i^{max} and F_i^{min} are the maximum and minimum values of P^* in the i th objective function, respectively. The larger is the MS value, the better is the coverage rate of P to P^* .

- Uniform distribution (UD) metric [41] measures the distribution of non-dominated solutions and is formulated as

$$UD(S) = \frac{1}{1 + D_{nc}} \quad (16)$$

where S is the non-dominated solution set, $D_{nc} = \sqrt{\sum_{\mathbf{s}_i \in S} \frac{(nc(\mathbf{s}_i) - \bar{nc}(S))^2}{|S|-1}}$, $nc(\mathbf{s}_i) = |\{\mathbf{s}_j \in S \mid \|\mathbf{s}_i - \mathbf{s}_j\| < \sigma\}| - 1$, and $nc(\mathbf{s}_i)$ is the niche count of \mathbf{s}_i . \bar{nc} is the

Table 3 The statistical results (Mean (Std)) of the nine algorithms over 51 independent runs on the TDY1-TDY6 with 30D in terms of MS and UD metrics

Algorithms	Results (MS)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA-AcPD	Mean	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00
	Std	(9.33E−07)	(0.00E+00)	(0.00E+00)	(0.00E+00)	(1.13E−04)	(3.28E−05)
	<i>p</i>	/	/	/	/	/	/
	Significance	/	/	/	/	/	/
RM-MEDA	Mean	1.00E+00	8.18E−01	6.30E−01	9.17E−01	7.99E−01	9.46E−01
	Std	(4.97E−07)	(1.43E−01)	(1.51E−01)	(1.15E−01)	(6.87E−02)	(8.64E−02)
	<i>p</i>	4.02E−03	1.26E−11	5.29E−20	8.68E−07	2.57E−13	4.81E−03
	Significance	+	+	+	+	+	+
IRM-MEDA	Mean	9.78E−01	7.55E−01	5.83E−01	7.62E−01	7.99E−01	9.50E−01
	Std	(6.21E−02)	(1.17E−01)	(1.74E−01)	(6.18E−02)	(6.68E−02)	(9.18E−02)
	<i>p</i>	4.16E−06	3.14E−16	1.39E−20	1.39E−20	2.57E−13	1.01E−02
	Significance	+	+	+	+	+	+
RM-MEDA+LL	Mean	1.00E+00	9.73E−01	5.72E−01	1.00E+00	8.53E−01	9.76E−01
	Std	(0.00E+00)	(6.21E−02)	(2.24E−01)	(0.00E+00)	(1.06E−01)	(6.90E−02)
	<i>p</i>	6.65E−05	9.63E−04	1.96E−19	0.00E+00	8.20E−05	6.19E−01
	Significance	+	+	+	≈	+	≈
FRM-MEDA	Mean	1.00E+00	8.35E−01	7.69E−01	9.86E−01	9.91E−01	9.98E−01
	Std	(3.95E−04)	(1.04E−01)	(1.11E−01)	(4.07E−02)	(3.68E−02)	(1.43E−02)
	<i>p</i>	3.17E−05	8.04E−14	2.94E−17	1.84E−03	5.55E−07	1.04E−01
	Significance	+	+	+	+	+	≈
IM-MOEA	Mean	1.00E+00	1.00E+00	9.79E−01	1.00E+00	9.95E−01	1.00E+00
	Std	(7.82E−04)	(2.70E−04)	(7.07E−02)	(1.74E−05)	(3.14E−02)	(6.00E−04)
	<i>p</i>	4.62E−05	7.63E−14	1.39E−20	1.39E−20	1.12E−03	5.08E−17
	Significance	+	+	+	+	+	+
A-IM-MOEA	Mean	9.97E−01	1.00E+00	1.00E+00	9.92E−01	9.88E−01	1.00E+00
	Std	(2.09E−02)	(1.73E−04)	(4.40E−04)	(4.38E−02)	(5.73E−02)	(7.31E−04)
	<i>p</i>	3.17E−05	1.78E−09	1.39E−20	1.39E−20	2.50E−04	1.06E−15
	Significance	+	+	+	+	+	+
MOEA/D	Mean	7.35E−01	7.16E−01	2.61E−01	8.82E−01	7.12E−01	4.31E−01
	Std	(1.21E−01)	(1.05E−01)	(3.05E−02)	(8.80E−02)	(1.69E−02)	(6.10E−02)
	<i>p</i>	4.84E−19	1.39E−20	1.39E−20	8.04E−14	2.77E−18	6.91E−20
	Significance	+	+	+	+	+	+
MOEA/D-M2M	Mean	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00
	Std	(0.00E+00)	(1.31E−06)	(8.14E−09)	(0.00E+00)	(1.29E−06)	(5.21E−07)
	<i>p</i>	6.65E−05	1.39E−20	1.39E−20	0.00E+00	1.93E−05	2.08E−13
	Significance	−	+	+	≈	+	+
Algorithms	Results (UD)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA-AcPD	Mean	4.50E−01	9.71E−01	8.77E−01	7.60E−01	6.07E−01	9.95E−01
	Std	(1.27E−02)	(5.90E−02)	(1.12E−16)	(4.31E−02)	(2.49E−02)	(2.42E−02)
	<i>p</i>	/	/	/	/	/	/
	Significance	/	/	/	/	/	/

Table 3 continued

Algorithms	Results (UD)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA	Mean	4.45E-01	7.34E-01	7.75E-01	8.56E-01	6.01E-01	9.56E-01
	Std	(1.98E-02)	(2.42E-01)	(1.63E-01)	(8.59E-02)	(7.56E-02)	(8.88E-02)
	<i>p</i>	2.17E-02	1.98E-09	1.38E-07	3.78E-10	9.25E-01	1.11E-03
	Significance	+	+	+	—	≈	+
IRM-MEDA	Mean	4.42E-01	6.50E-01	7.19E-01	6.19E-01	6.40E-01	9.68E-01
	Std	(2.53E-02)	(2.20E-01)	(2.05E-01)	(1.43E-01)	(5.71E-02)	(5.93E-02)
	<i>p</i>	5.07E-02	1.25E-11	1.36E-10	3.54E-09	4.20E-04	3.90E-03
	Significance	≈	+	+	+	—	+
RM-MEDA+LL	Mean	4.45E-01	9.32E-01	6.64E-01	8.34E-01	6.01E-01	9.65E-01
	Std	(1.47E-02)	(8.56E-02)	(2.81E-01)	(5.75E-02)	(8.03E-02)	(6.58E-02)
	<i>p</i>	1.48E-01	1.05E-02	8.74E-10	5.56E-10	7.68E-01	3.78E-03
	Significance	≈	+	+	—	≈	+
FRM-MEDA	Mean	3.32E-01	3.81E-01	5.27E-01	7.59E-01	6.04E-01	7.08E-01
	Std	(8.30E-02)	(2.50E-01)	(2.38E-01)	(2.30E-01)	(2.26E-01)	(1.81E-01)
	<i>p</i>	2.33E-13	4.62E-17	1.02E-17	7.46E-01	7.81E-01	1.15E-16
	Significance	+	+	+	≈	≈	+
IM-MOEA	Mean	7.91E-01	3.69E-01	6.80E-01	7.00E-01	3.07E-01	7.26E-01
	Std	(1.53E-01)	(3.16E-02)	(2.51E-02)	(2.95E-02)	(2.34E-02)	(4.03E-02)
	<i>p</i>	2.66E-18	2.71E-19	1.38E-20	3.31E-16	3.11E-18	2.70E-20
	Significance	—	+	+	+	+	+
A-IM-MOEA	Mean	7.70E-01	6.77E-01	7.18E-01	5.30E-01	4.23E-01	6.99E-01
	Std	(1.58E-01)	(1.52E-01)	(6.50E-02)	(1.47E-01)	(1.12E-01)	(4.43E-02)
	<i>p</i>	2.85E-18	6.98E-15	1.39E-20	8.78E-16	6.00E-15	2.79E-20
	Significance	—	+	+	+	+	+
MOEA/D	Mean	4.03E-01	2.00E-01	1.37E-01	6.03E-01	1.53E-01	2.48E-01
	Std	(2.30E-01)	(9.30E-02)	(1.12E-01)	(3.57E-01)	(5.65E-02)	(9.16E-02)
	<i>p</i>	1.25E-01	2.50E-19	8.84E-21	7.27E-01	2.57E-18	2.40E-20
	Significance	≈	+	+	≈	+	+
MOEA/D-M2M	Mean	4.35E-01	2.41E-01	2.05E-01	2.02E-01	2.01E-01	1.66E-01
	Std	(6.82E-02)	(4.59E-02)	(3.10E-02)	(5.57E-02)	(4.95E-02)	(3.33E-02)
	<i>p</i>	1.64E-01	2.72E-19	1.39E-20	2.60E-18	3.30E-18	2.79E-20
	Significance	≈	+	+	+	+	+

The best result for each problem is highlighted in bold

average of $nc(s_i)$ and σ is set to 0.01. The larger is the UD value, the better is the distribution of P .

4.3 MOEAs for comparisons

We consider eight state-of-the-art MOEAs, including RM-MEDA [24], IRM-MEDA [26], RM-MEDA+LL [29], FRM-MEDA [27], IM-MOEA [13], A-IM-MOEA [14], MOEA/D [42] and MOEA/D-M2M [37] for comparisons. All of them are regularity property-based MOEAs, in which the first four MOEAs are based on the regularity property of decision space, and the rest are based on the regularity property of objective space. Since the general ideas of the first six

MOEAs have been introduced in Sect. 2.2, respectively, here we only present the working principles of MOEA/D and MOEA/D-M2M.

- MOEA/D and MOEA/D-M2M are two popular decompo-

sition-based MOEAs. They employ a group of reference vectors to decompose an MOP into a set of subproblems and solve them cooperatively. In MOEA/D, the construction of individual neighborhood is based on the distance of reference vector, while the MOEA/D-M2M uses the cosine similarity of reference vector to divided the population into several sub-populations. Generally, MOEA/D

Table 4 Summary of the Wilcoxon test in terms of IGD, HV, MS and UD metrics between different algorithms on TDY problems

IGD	RM-MEDA	IRM-MEDA	RM-MEDA+LL	FRM-MEDA	IM-MOEA	A-IM-MOEA	MOEA/D	MOEA/D-M2M
RM-MEDA-AcPD	0-1-5	0-0-6	0-0-6	0-0-6	0-0-6	0-0-6	0-0-6	0-0-6
RM-MEDA		3-0-3	3-2-1	0-0-6	5-0-1	5-0-1	2-2-2	5-0-1
IRM-MEDA			3-2-1	0-0-6	5-0-1	5-0-1	3-0-3	5-0-1
RM-MEDA+LL				0-0-6	3-2-1	3-2-1	0-0-6	3-1-2
FRM-MEDA					5-0-1	5-0-1	5-0-1	5-0-1
IM-MOEA						2-3-1	0-0-6	0-0-6
A-IM-MOEA							0-0-6	0-0-6
MOEA/D								5-0-1
HV	RM-MEDA	IRM-MEDA	RM-MEDA+LL	FRM-MEDA	IM-MOEA	A-IM-MOEA	MOEA/D	MOEA/D-M2M
RM-MEDA-AcPD	0-0-6	0-0-6	0-0-6	0-0-6	0-0-6	0-0-6	0-0-6	0-0-6
RM-MEDA		3-0-3	3-2-1	0-0-6	5-0-1	5-0-1	4-0-2	5-0-1
IRM-MEDA			3-1-2	0-0-6	5-0-1	5-0-1	4-0-2	5-0-1
RM-MEDA+LL				0-0-6	5-0-1	5-0-1	3-0-3	4-0-2
FRM-MEDA					5-0-1	5-0-1	5-0-1	5-0-1
IM-MOEA						0-4-2	0-2-4	0-2-4
A-IM-MOEA							0-3-3	0-2-4
MOEA/D								2-2-2
MS	RM-MEDA	IRM-MEDA	RM-MEDA+LL	FRM-MEDA	IM-MOEA	A-IM-MOEA	MOEA/D	MOEA/D-M2M
RM-MEDA-AcPD	0-0-6	0-0-6	0-2-4	0-1-5	0-0-6	0-0-6	0-0-6	0-1-5
RM-MEDA		0-2-4	4-2-0	4-1-1	4-1-1	5-0-1	0-0-6	5-1-0
IRM-MEDA			5-1-0	5-1-0	6-0-0	5-1-0	1-1-4	6-0-0
RM-MEDA+LL				3-1-2	4-1-1	4-1-1	0-0-6	4-2-0
FRM-MEDA					5-1-0	4-1-1	0-0-6	5-1-0
IM-MOEA						0-6-0	0-0-6	1-2-3
A-IM-MOEA							0-0-6	1-2-3
MOEA/D								6-0-0
UD	RM-MEDA	IRM-MEDA	RM-MEDA+LL	FRM-MEDA	IM-MOEA	A-IM-MOEA	MOEA/D	MOEA/D-M2M
RM-MEDA-AcPD	1-1-4	1-1-4	1-2-3	0-2-4	1-0-5	1-0-5	0-2-4	0-1-5
RM-MEDA		1-4-1	1-4-1	0-2-4	1-0-5	1-1-4	0-1-5	0-1-5
IRM-MEDA			2-3-1	1-1-4	2-0-4	1-2-3	0-2-4	0-1-5
RM-MEDA+LL				0-2-4	2-0-4	1-1-4	0-2-4	0-1-5
FRM-MEDA					2-2-2	3-1-2	0-1-5	1-0-5
IM-MOEA						3-1-2	0-1-5	0-0-6
A-IM-MOEA							0-1-5	0-0-6
MOEA/D								3-1-2

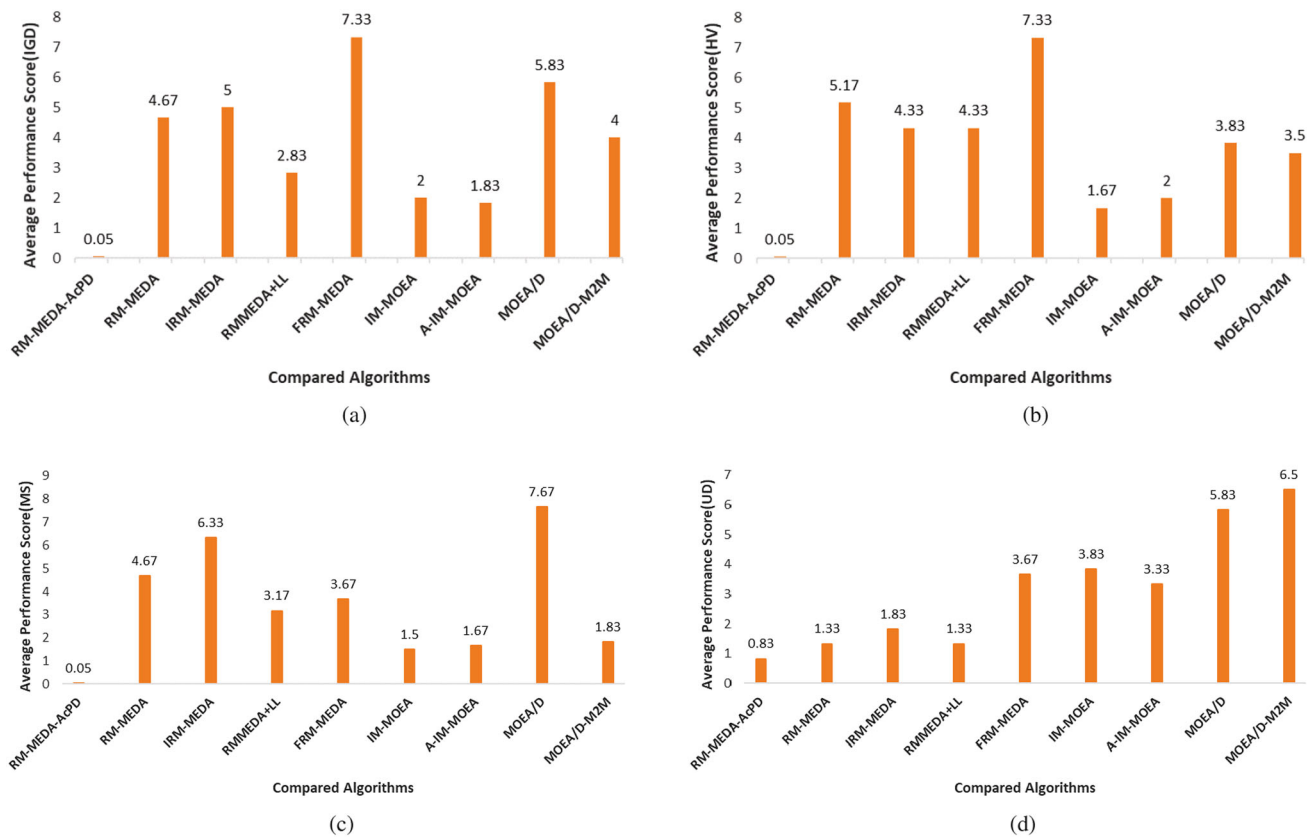


Fig. 8 The comparison of APS values of IGD, HV, MS, UD metrics for RM-MEDA-AcPD, RM-MEDA, IRM-MEDA, RM-MEDA+LL, FRM-MEDA, IM-MOEA, A-IM-MOEA, MOEA/D, and MOEA/D-M2M on the TDY1-TDY6 problems with 30 decision variables, respectively

Table 5 The average ranking of nine algorithms based on the Friedman's test

Algorithm-IGD	Ranking	Algorithm-HV	Ranking	Algorithm-MS	Ranking	Algorithm-UD	Ranking
RM-MEDA-AcPD	1.00	RM-MEDA-AcPD	1.25	RM-MEDA-AcPD	2.42	RM-MEDA-AcPD	1.83
RM-MEDA	6.00	RM-MEDA	6.42	RM-MEDA	6.50	RM-MEDA	3.17
IRM-MEDA	6.33	IRM-MEDA	6.08	IRM-MEDA	7.75	IRM-MEDA	3.83
RM-MEDA+LL	4.33	RM-MEDA+LL	5.42	RM-MEDA+LL	5.17	RM-MEDA+LL	3.67
FRM-MEDA	8.33	FRM-MEDA	8.33	FRM-MEDA	4.92	FRM-MEDA	5.83
IM-MOEA	3.33	IM-MOEA	3.00	IM-MOEA	3.00	IM-MOEA	5.00
A-IM-MOEA	3.17	A-IM-MOEA	3.67	A-IM-MOEA	4.00	A-IM-MOEA	5.17
MOEA/D	7.33	MOEA/D	6.00	MOEA/D	8.83	MOEA/D	8.33
MOEA/D-M2M	5.17	MOEA/D-M2M	4.83	MOEA/D-M2M	2.42	MOEA/D-M2M	8.17

The best result for each problem is highlighted in bold

has stronger convergence ability than that of MOEA/D-M2M, while the MOEA/D-M2M has better ability to maintain diversity than that of MOEA/D.

4.4 Parameter settings

As suggested by the original literatures, the public and private parameters are set as follows.

- Population size is set to 100 for all the algorithms.
- For MOEA/D and MOEA/D-M2M, the polynomial mutation (PM) and simulated binary crossover (SBX) are employed for reproduction. The mutation distribution index η_m of PM is set to 20 and the mutation probability is set to $1/D$. The crossover distribution index η_c of SBX is set to 20 and the crossover probability is set to 1.

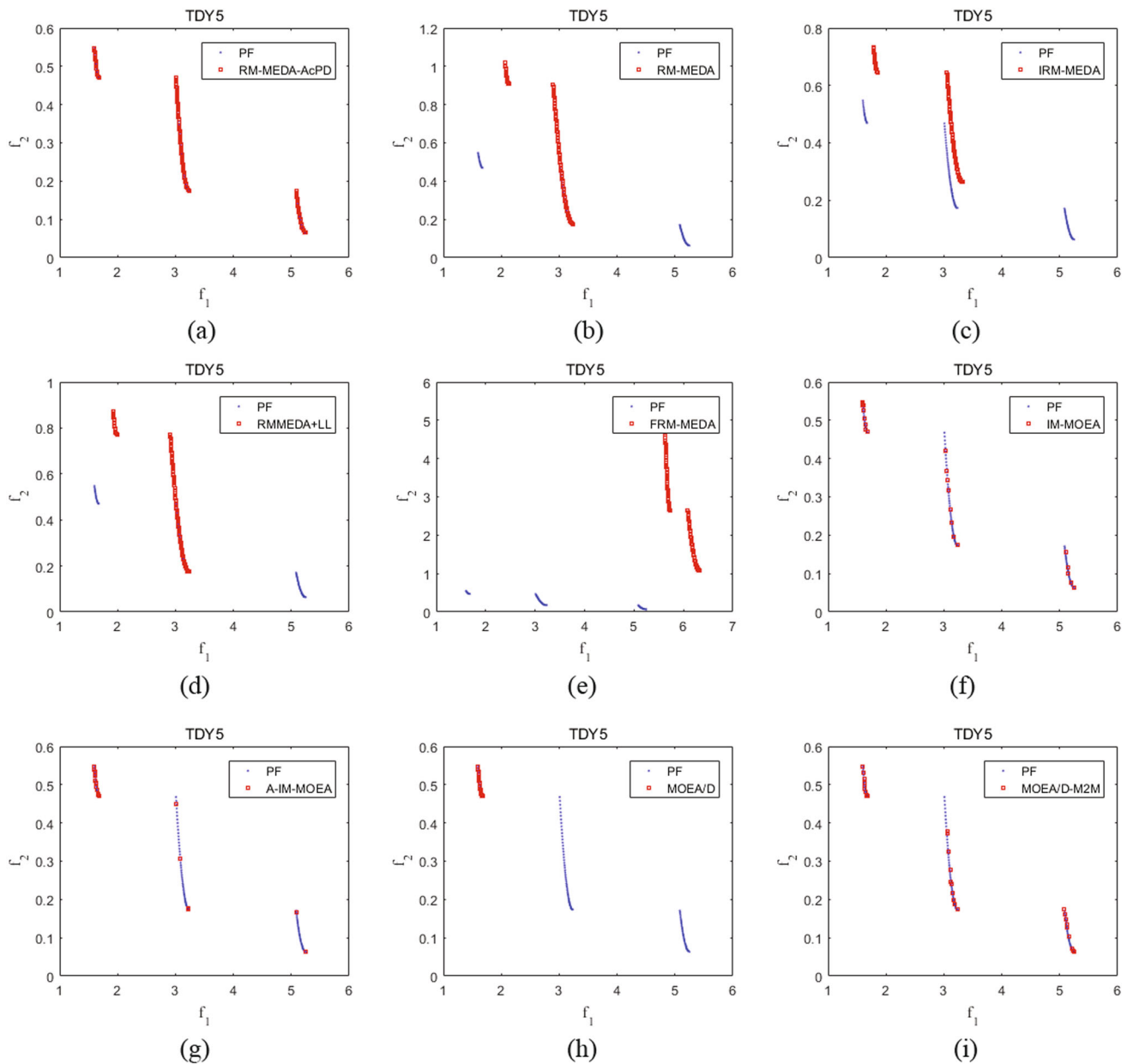


Fig. 9 Final non-dominated solutions obtained by RM-MEDA-AcPD, RM-MEDA, IRM-MEDA, RM-MEDA+LL, FRM-MEDA, IM-MOEa, A-IM-MOEa, MOEA/D, and MOEA/D-M2M on TDY5

problem in the run associated with the median IGD value. The red points denote the obtained final non-dominated solutions, while the blue points denote the PF

- **Termination criterion** For each test problems, all the algorithms are allowed for a maximum number of function evaluations $10000 * D$.
- **Number of runs** Each algorithm runs 51 times independently for each test problem.
- **Statistical test** To test the difference of algorithms, the Wilcoxon's rank sum test [41] at 0.05 significance level is applied to analysis, in which “+ / - / \approx ” denote that the RM-MEDA-AcPD performs better than, worse than and similar to competitive algorithms, respectively.
- The number of clusters is set to 5 for RM-MEDA, RM-MEDA+LL and RM-MEDA-AcPD, and 2 for FRM-MEDA.
- The number of reference vectors is set to 15 for IM-MOEa and A-IM-MOEa.
- The θ is set to 0.4 and 0.2 for RM-MEDA+LL and A-IM-MOEa, respectively.

5 Experimental results

The performance comparisons of the peer algorithms on TDY problems in terms of IGD, HV, MS and UD are shown in Tables 2 and 3, and their pair-wise comparison results are described in Table 4 by the Wilcoxon's rank sum test. To evaluate the overall performance of each algorithm, the average performance score (APS) [44] is computed, in which the APS of an algorithm for a given MOP is defined as the number of times the algorithm is defeated by other algorithms significantly. Here, the average APS values of all algorithms are summarized in Fig. 8, and the average ranking of nine algorithms based on the Friedman's test is summarized in Table 5.

From the results of Wilcoxon's rank sum test, the average APS value and the average ranking, it can be observed that the RM-MEDA-AcPD achieves the best performance in terms of the overall IGD and HV. In fact, RM-MEDA-AcPD beats other algorithms on all test problems in terms of IGD and HV. In addition, the RM-MEDA-AcPD achieves the best performance in terms of the overall MS and UD. The statistically superior performance of RM-MEDA-AcPD is due to the new affine subspace model assisted with random noise, which effectively controls the diversity of the population throughout the entire search process.

To visualize the evolutionary process, Fig. 9 displays the final non-dominated solution sets in the objective space for the TDY5 problem, and Fig. 10 plots the evolution of averaged IGD values and HV values at 20 specified checkpoints for all algorithms in terms of the TDY1-TDY6 problems. According to the simulation results, the final non-dominated solution sets of TDY1-TDY6 problems achieved by the RM-MEDA-AcPD are closer and more uniformly spread along the PFs than other algorithms in the objective space. Finally, the corresponding evolution curves further reveal that the RM-MEDA-AcPD has smaller IGD and larger HV values, and achieves faster convergence on most test problems. The evolution of averaged MS value shown in Fig. 11a–f further confirms that the proposed RM-MEDA-AcPD algorithm has the maximum spread as well as the MOEA/D-M2M.

However, we also note that the average UD values of RM-MEDA-AcPD are not all the best. For example, the UD result of RM-MEDA-AcPD is worse than that of the IRM-MEDA on TDY5 problem, although the other three indicators of RM-MEDA-AcPD are better than this algorithm. The simulation results of Fig. 11 show that the RM-MEDA-AcPD can obtain three PF curves, while the IRM-MEDA has only two. This phenomenon will make the population of IRM-MEDA more clustered than that of RM-MEDA-AcPD, which may make the population distribution of IRM-MEDA more uniform. In view of this, when using the proposed algorithm to solve real-world problems, decision makers should give priority to HV index rather than this uniformity index.

As can be further observed from Fig. 9, most algorithms cannot cover the entire PFs of TDY5. The deeper reasons are as follows. RM-MEDA, IRM-MEDA, RM-MEDA+LL and FRM-MEDA make use of the regularity property of the decision space to build the probabilistic sampling models. As previously analyzed in Sect. 3.1, such built models may cause the rapid loss of population diversity. To further verify this, we use the following formula [45] to track the change of population diversity.

$$DI = \sqrt{\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^D (x_{ij} - \bar{x}_j)^2} \quad (17)$$

where \bar{x}_j is the average value of j th coordinate of the solutions in the current population. The larger is the DI value, the better is the population diversity.

From Fig. 12, it is clear that the DI values of RM-MEDA-AcPD are larger than those of other four RM-MEDAs, which explains why the proposed algorithm outperforms other RM-MEDAs. The remaining four compared algorithms use a group of evenly distributed reference vectors to divide the population. The effectiveness of the division method is based on the assumption that the PF of the solved MOP is evenly distributed. Obviously, such assumption is impossible for TDY problems and will lead to a serious waste of computing resources. To demonstrate this, Fig. 13 shows the position distribution of reference vectors along the true PFs of TDY problems.¹ From it, we observe that the TDY1 only has three short PF curves distributed in the middle part of the objective space, and most areas in the objective space have no PF. For the TDY1, only three of fifteen reference vectors are related to the true PF of the TDY1 problem, and the other twelve reference vectors are not used. The cases of TDY2-TDY6 are similar to that of TDY1.

Finally, the computational efficiency of RM-MEDA-AcPD is assessed in terms of the running time ration. From the comparison results given in Fig. 14, we observe that the computational efficiency of RM-MEDA-AcPD is similar to that of other RM-MEDA algorithms, higher than that of IM-MOEA and A-IM-MOEA, and lower than that of MOEA/D and MOEA/D-M2M. It reveals that the new affine subspace model used in RM-MEDA-AcPD does not significantly improve the complexity. In addition, this also shows that the built models in RM-MEDAs are cheaper than those of IM-MOEA and A-IM-MOEA, and model-based MOEAs are more expensive than genetic operation-based MOEAs.

¹ If $f_i \leq 0$, $f_i + M$ is employed to replace it by adding a constant M such that $f_i + M > 0$.

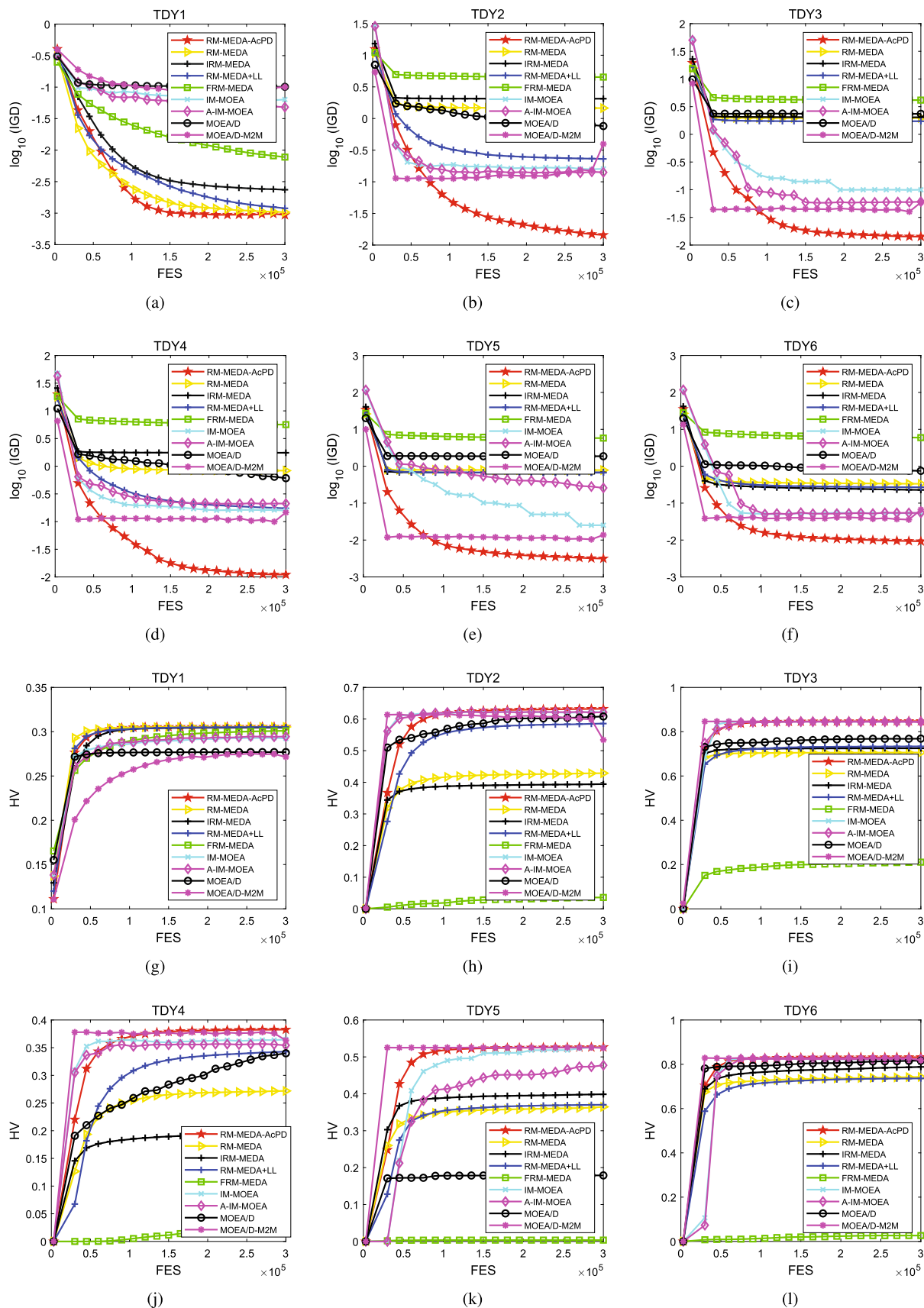


Fig. 10 The evolution of averaged IGD and HV values of RM-MEDA-AcPD, RM-MEDA, IRM-MEDA, RM-MEDA+LL, FRM-MEDA, IM-MOEA, A-IM-MOEA, MOEA/D, and MOEA/D-M2M over 51 runs for six different problems with 30 decision variables, respectively

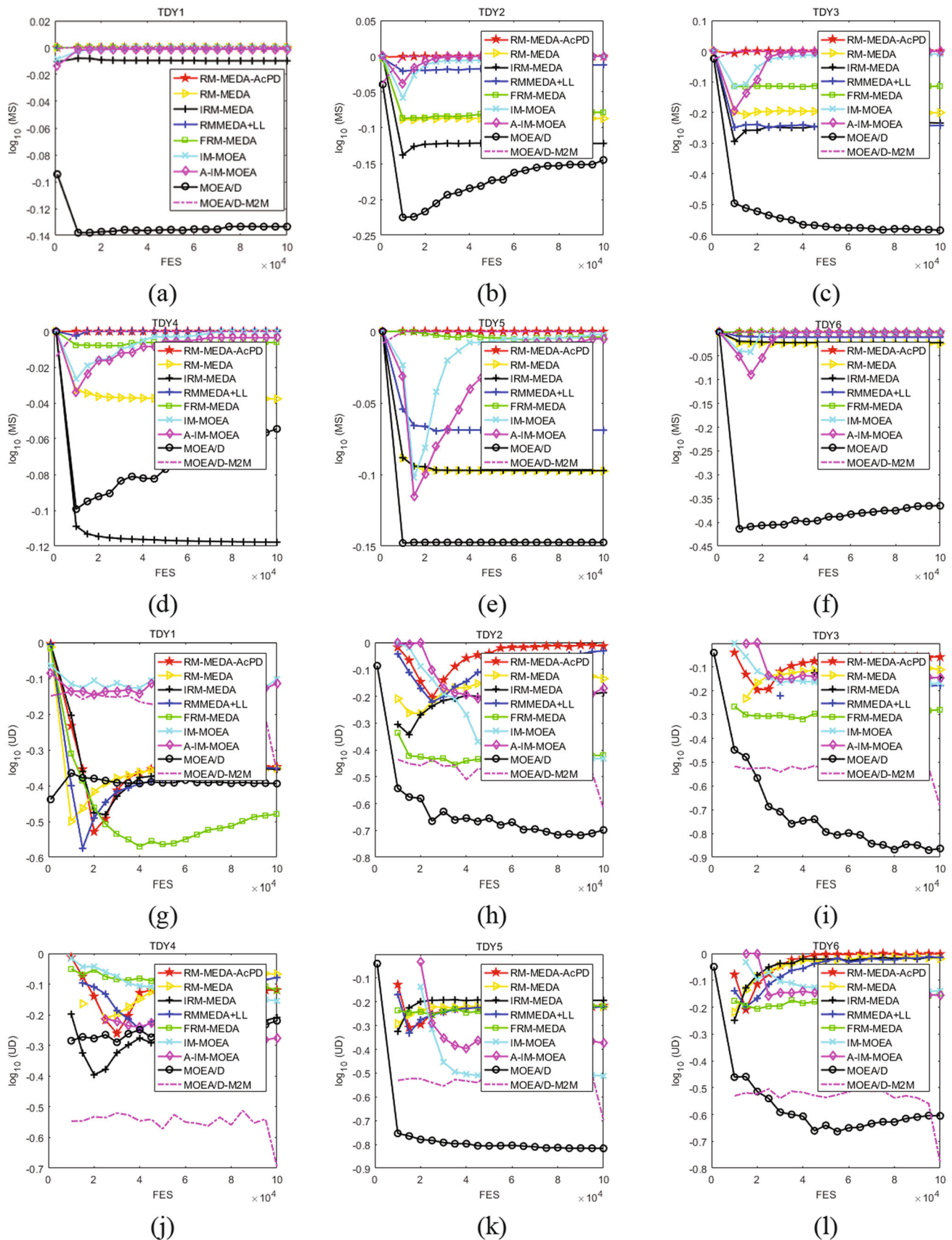


Fig. 11 The evolution of averaged MS and UD values of RM-MEDA-AcPD, RM-MEDA, IRM-MEDA, RM-MEDA+LL, FRM-MEDA, IM-MOEA, A-IM-MOEA, MOEA/D, and MOEA/D-M2M over 51 runs for six different problems with 30 decision variables, respectively

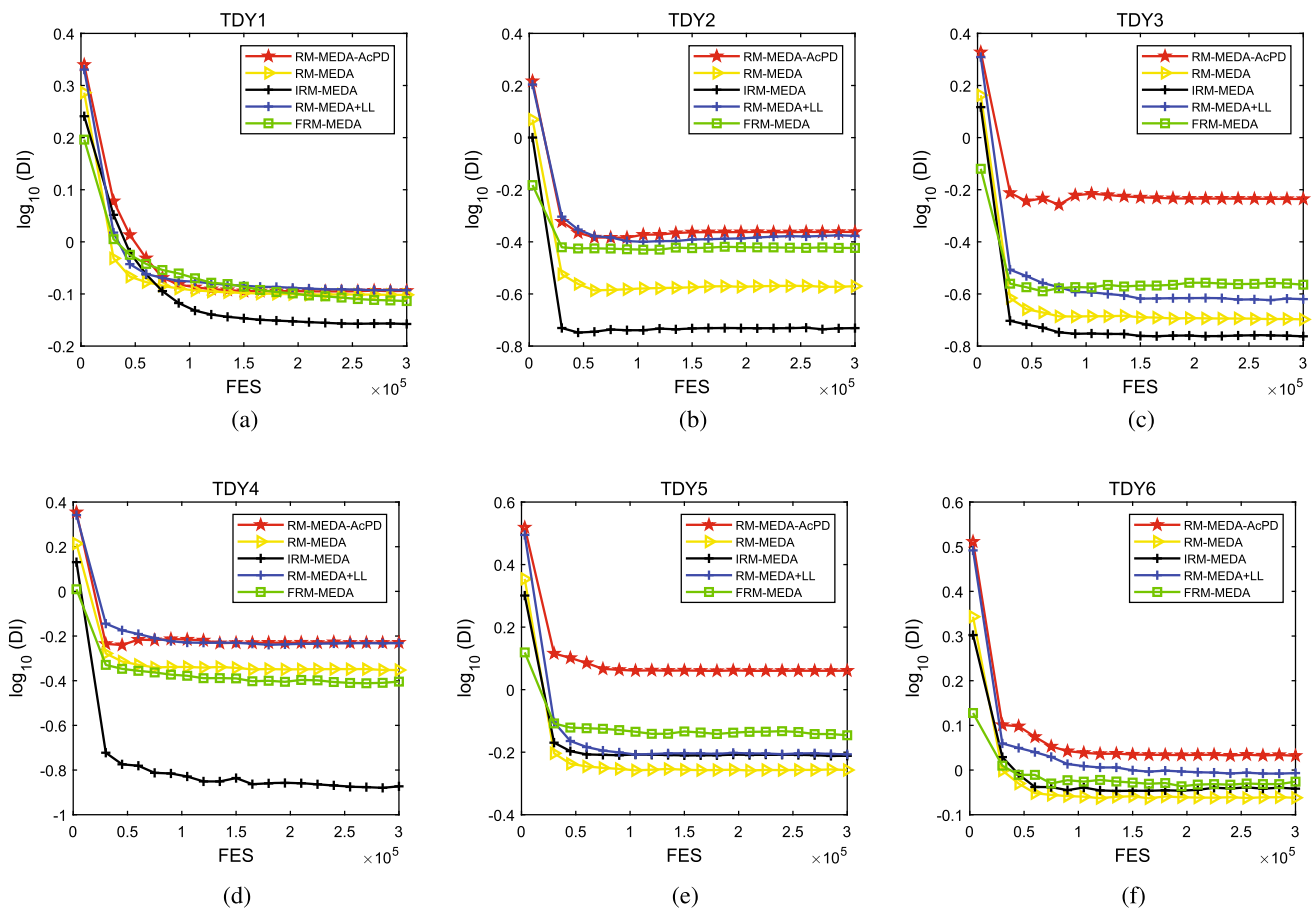


Fig. 12 The evolution of averaged DI values of RM-MEDA-AcPD, RM-MEDA, IRM-MEDA, RM-MEDA+LL, and FRM-MEDA over 51 runs for six different problems with 30 decision variables, respectively

6 Discussion

6.1 Sensitivity to the number of clusters K in RM-MEDA-AcPD

Although the sensitivity of the number of clusters K has been analyzed in RM-MEDA, it is still necessary to further discuss it since the RM-MEDA focuses on the MOPs with regular PFs, while the RM-MEDA-AcPD focuses on the MOPs with irregular PFs. For this purpose, we have tried different value of K : 3, 5, 10 and 15 to study how the performance of RM-MEDA-AcPD is sensitive to the K . The statistical results of RM-MEDA-AcPD with different K values on the TDY1-TDY6 with 30D in terms of IGD and HV are shown in Table 6.

From the results of Wilcoxon's rank sum test, it can be observed that the performance of RM-MEDA-AcPD is sensitive to the K . Specifically, when the K is set to 5, the performance of RM-MEDA-AcPD is significantly better than that of $K = 3$, but similar to that of $K = 10$ or $K = 15$. Therefore, $K = 5$ seems to be appropriate for most test problems. However, too small K value is not recommended

since the PSs of these test problems are made up of several unconnected curves.

6.2 Sensitivity to the scaling factor ρ in RM-MEDA-AcPD

To study how the performance of RM-MEDA-AcPD is sensitive to the scaling factor ρ , we set different maximum value ρ_{max} of the scaling factor ρ : 3, 5, 10, 15. The statistical results of RM-MEDA-AcPD with different ρ_{max} values on the TDY1-TDY6 with 30D in terms of IGD and HV are shown in Table 7.

From the results of Wilcoxon's rank sum test, it can be observed that the performance of RM-MEDA-AcPD is sensitive to the scaling factor ρ . Specifically, when the ρ_{max} is set to 5, the performance of RM-MEDA-AcPD is slightly better than that of $\rho_{max} = 3$, but significantly better than that of $\rho_{max} = 10$ or $\rho_{max} = 15$. Therefore, $\rho_{max} = 5$ seems to be appropriate for most test problems, and too large ρ_{max} value (more than 10) is not recommended.

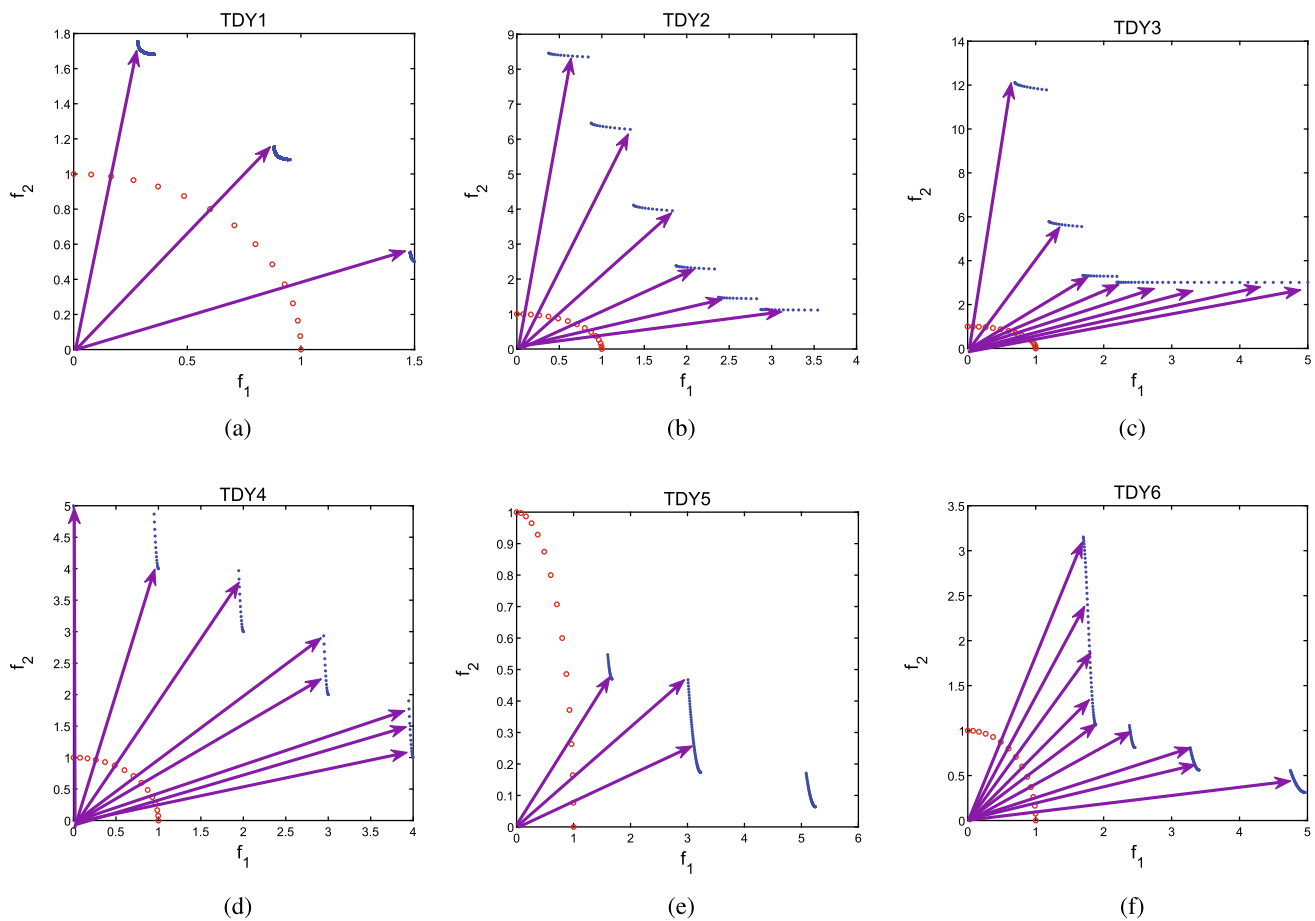


Fig. 13 The validity of reference vector distribution for TDY1-TDY6: **a** only 3 out of 15 reference vectors are covered by the PF of TDY1; **b** only 6 out of 15 reference vectors are covered by the PF of TDY2; **c** only 8 out of 15 reference vectors are covered by the PF of TDY3;

d only 8 out of 15 reference vectors are covered by the PF of TDY4; **e** only 3 out of 15 reference vectors are covered by the PF of TDY5; **f** only 9 out of 15 reference vectors are covered by the PF of TDY6

6.3 Effectiveness of the RM-MEDA-AcPD's components

As previously discussed, RM-MEDA-AcPD include two crucial components: modified SPX method and random noise model. To verify the effectiveness of the two components, numerical experiments were carried out on TDY5 at 30D under the same experimental conditions as Sect. 4. The Fig. 15a is the simulation result of RM-MEDA-AcPD*, in which the random noise model was eliminated in the RM-MEDA-AcPD. The Fig. 15b is the simulation result of RM-MEDA-AcPD**, in which the modified SPX method was eliminated in the RM-MEDA-AcPD. The Fig. 16 shows that the RM-MEDA-AcPD performs better than the RM-MEDA-AcPD* and RM-MEDA-AcPD**, which makes it clear that the random noise model can effectively help the RM-MEDA-AcPD jump out of the Pareto local optimum, and the modified SPX method further provides the power of convergence.

6.4 The performance of RM-MEDA-AcPD on MOPs with continuous PF

The above experiments have verified the effectiveness of the proposed RM-MEDA-AcPD in solving MOPs with disconnected PF. However, it is a question whether the proposed algorithms are still effective for MOPs with continuous PF. To clarify this doubt, we further verify the performance of RM-MEDA-AcPD by optimizing F1 and F5 problems with 30D, in which the F1 has been defined in Eq. (9), and the F5 is defined as follows:

$$\mathbf{F5}(\mathbf{x}) = \begin{cases} f_1(\mathbf{x}) = x_1 \\ f_2(\mathbf{x}) = g(\mathbf{x})[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})}] \\ g(\mathbf{x}) = 1 + 9 \left(\sum_{i=2}^D (x_i^2 - x_1) \right)^2 / (D - 1), \mathbf{x} \in [0, 1]^{30} \end{cases} \quad (18)$$

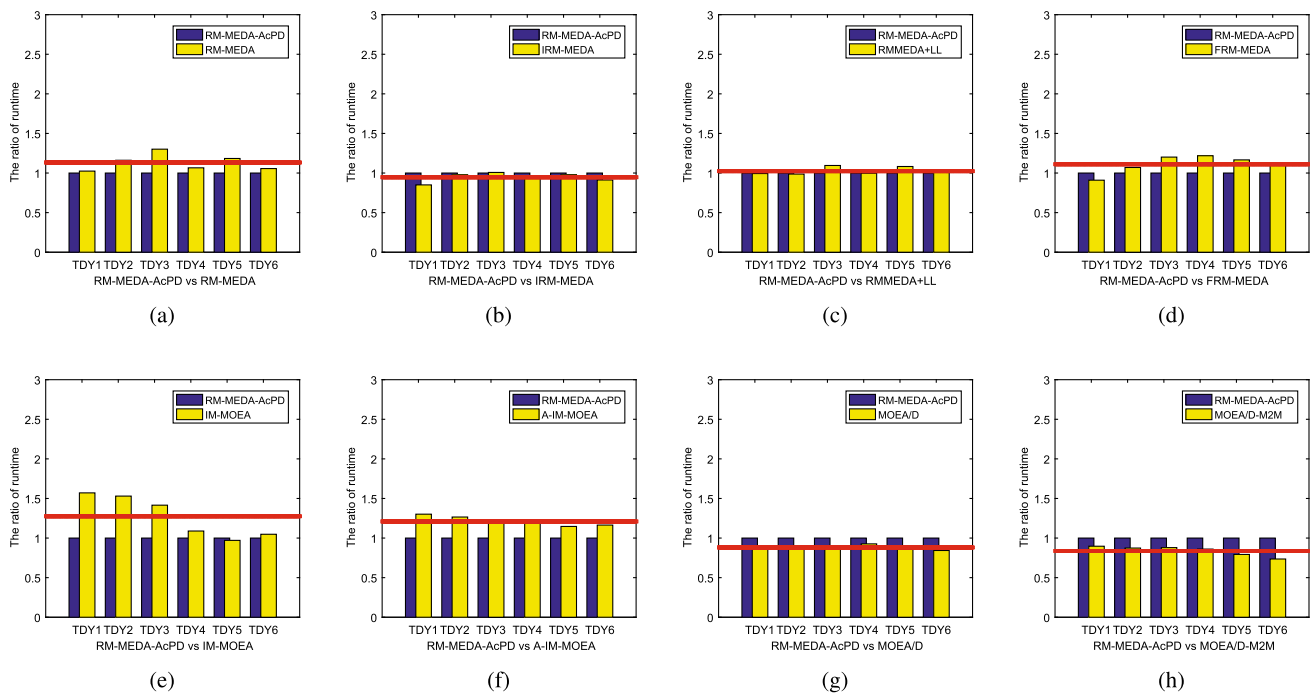


Fig. 14 The comparison of running time ratio for RM-MEDA-AcPD with RM-MEDA, IRM-MEDA, RM-MEDA+LL, FRM-MEDA, IM-MOEA, A-IM-MOEA, MOEA/D, and MOEA/D-M2M on the TDY1-TDY6 problems with 30 decision variables, respectively

The experimental platform and parameter settings for all compared algorithms are the same as Sect. 4. From the Fig. 16, we can see that the performance of RM-MEDA-AcPD on the two MOPs with continuous PF is still satisfactory, although the convergence speed of RM-MEDA-AcPD is slightly lower than that of other RM-MEDAs. The reason is that the improvement of population diversity of RM-MEDA-AcPD is at the expense of convergence speed to a certain extent.

7 Conclusion

In this paper, we have suggested a new representation model of the affine subspace for RM-MEDA, which is developed by the modified simplex crossover and random noise model. The main purpose of sampling is to push solutions forward along the orthogonal direction of affine subspace, and the random noise model is to reduce the risk of rapid loss of population diversity. The combination of these two aspects expands the representation range of the affine subspace.

The experimental comparison results show that the newly proposed representation model of the affine subspace can generate better search efficiency than the original model. A deeper analysis demonstrates that the effectiveness of the new

model is mainly attributed to the improvement of population diversity. In addition, we find that the MOEAs based on the regularity properties of different spaces have different performance in solving benchmark problems with disconnected PFs.

In addition to focusing on the performance of RM-MEDA-AcPD in solving MOPs with disconnected PF, we also discuss the performance of RM-MEDA-AcPD on MOPs with continuous PF. The experimental results show that the effect of RM-MEDA-AcPD on MOPs with continuous PF is still satisfactory, which shows that the proposed algorithm is more robust to different types of problems.

Despite the competitive experimental results of the research, more effective mechanisms of the diversity-driven RM-MEDA are still unknown and need to be explored in the future. As we have seen from the preceding discussion, the population partition based on the objective space seems to be more effective than clustering based on the decision space in maintaining population diversity. Therefore, the performance of RM-MEDA by the population partition based on the objective space needs to be deeply investigated in the future. In addition, as pointed out in [46], providing all the PSs is of great significance for the decision maker. Therefore, it is worthwhile to investigate the ability of RM-MEDA-AcPD in finding different global or local PSs with the same objective values.

Table 6 The statistical results (Mean (Std)) of RM-MEDA-AcPD with different K values over 51 independent runs on the TDY1-TDY6 with 30D in terms of IGD and HV metrics

Algorithms	Results (IGD)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA-AcPD ($K = 5$)	Mean	9.34E-04	1.44E-02	1.42E-02	1.10E-02	3.15E-03	9.24E-03
	Std	(9.49E-05)	(2.19E-03)	(1.27E-03)	(1.37E-03)	(2.33E-04)	(4.61E-04)
	p	/	/	/	/	/	/
	Significance	/	/	/	/	/	/
	Mean	1.15E-03	1.84E-02	1.75E-02	1.20E-02	5.57E-03	1.05E-02
RM-MEDA-AcPD ($K = 3$)	Std	(3.96E-04)	(3.76E-03)	(2.59E-03)	(2.14E-03)	(7.07E-04)	(5.61E-04)
	p	1.63E-05	2.65E-07	7.92E-12	1.69E-02	3.30E-18	5.70E-15
	Significance	+	+	+	+	+	+
	Mean	9.57E-04	1.46E-02	1.44E-02	9.96E-03	1.60E-02	9.38E-03
	Std	(8.55E-05)	(8.83E-04)	(6.24E-04)	(8.25E-04)	(9.54E-02)	(4.29E-04)
RM-MEDA-AcPD ($K = 10$)	p	1.66E-01	7.48E-17	6.49E-11	3.88E-06	2.30E-16	7.30E-14
	Significance	\approx	\approx	\approx	-	+	+
	Mean	9.68E-04	8.94E-03	1.47E-02	9.94E-03	1.36E-01	5.81E-02
	Std	(1.07E-04)	(4.31E-04)	(6.04E-04)	(6.55E-04)	(2.73E-01)	(1.53E-01)
	p	9.70E-02	3.30E-18	2.12E-14	6.60E-07	1.39E-07	6.57E-12
	Significance	\approx	-	\approx	-	+	+
RM-MEDA-AcPD ($K = 15$)	Mean	9.68E-04	8.94E-03	1.47E-02	9.94E-03	1.36E-01	5.81E-02
	Std	(1.07E-04)	(4.31E-04)	(6.04E-04)	(6.55E-04)	(2.73E-01)	(1.53E-01)
	p	9.70E-02	3.30E-18	2.12E-14	6.60E-07	1.39E-07	6.57E-12
	Significance	\approx	-	\approx	-	+	+
	Mean	9.68E-04	8.94E-03	1.47E-02	9.94E-03	1.36E-01	5.81E-02
RM-MEDA-AcPD ($K = 5$)	Mean	3.05E-01	6.32E-01	8.48E-01	3.83E-01	5.27E-01	8.32E-01
	Std	(1.34E-05)	(1.02E-03)	(6.11E-04)	(8.53E-04)	(5.60E-04)	(3.43E-04)
	Significance	/	/	/	/	/	/
	Mean	3.05E-01	6.31E-01	8.47E-01	3.82E-01	5.22E-01	8.31E-01
	Std	(2.12E-04)	(1.67E-03)	(1.13E-03)	(1.48E-03)	(1.11E-03)	(4.20E-04)
RM-MEDA-AcPD ($K = 3$)	p	5.00E-16	3.19E-06	5.10E-08	1.72E-02	3.30E-18	1.20E-17
	Significance	+	+	+	+	+	+
	Mean	3.05E-01	6.32E-01	8.49E-01	3.84E-01	5.25E-01	8.30E-01
	Std	(7.41E-06)	(4.00E-04)	(1.73E-04)	(4.44E-04)	(8.89E-04)	(3.24E-04)
	p	2.23E-06	3.37E-17	1.50E-10	1.28E-09	6.31E-17	1.07E-14
RM-MEDA-AcPD ($K = 10$)	Significance	+	\approx	\approx	-	+	+
	Mean	3.05E-01	6.35E-01	8.48E-01	3.84E-01	5.27E-01	8.31E-01
	Std	(8.36E-06)	(1.38E-04)	(8.21E-05)	(7.13E-05)	(2.57E-03)	(9.02E-04)
	p	8.99E-07	3.30E-18	2.43E-16	8.44E-18	1.24E-07	4.73E-12
	Significance	+	-	\approx	-	+	+
RM-MEDA-AcPD ($K = 15$)	Mean	3.05E-01	6.35E-01	8.48E-01	3.84E-01	5.27E-01	8.31E-01
	Std	(8.36E-06)	(1.38E-04)	(8.21E-05)	(7.13E-05)	(2.57E-03)	(9.02E-04)
	p	8.99E-07	3.30E-18	2.43E-16	8.44E-18	1.24E-07	4.73E-12
	Significance	+	-	\approx	-	+	+
	Mean	3.05E-01	6.35E-01	8.48E-01	3.84E-01	5.27E-01	8.31E-01

The best result for each problem is highlighted in bold

Table 7 The statistical results (Mean (Std)) of RM-MEDA-AcPD with different ρ_{max} values over 51 independent runs on the TDY1–TDY6 with 30D in terms of IGD and HV metrics

Algorithms	Results (IGD)	TDY1	TDY2	TDY3	TDY4	TDY5	TDY6
RM-MEDA-AcPD ($\rho_{max} = 5$)	Mean	9.34E–04	1.44E–02	1.42E–02	1.10E–02	3.15E–03	9.24E–03
	Std	(9.49E–05)	(2.19E–03)	(1.27E–03)	(1.37E–03)	(2.33E–04)	(4.61E–04)
	p	/	/	/	/	/	/
	Significance	/	/	/	/	/	/
	Mean	9.78E–04	1.44E–02	1.42E–02	1.11E–02	3.11E–03	9.41E–03
RM-MEDA-AcPD ($\rho_{max} = 3$)	Std	(9.79E–05)	(2.11E–03)	(1.27E–03)	(1.33E–03)	(2.29E–04)	(4.43E–04)
	p	3.22E–02	2.87E–01	5.69E–01	5.25E–01	3.59E–01	2.61E–01
	Significance	+	\approx	\approx	\approx	\approx	\approx
	Mean	9.19E–04	1.81E–02	1.53E–02	1.15E–02	3.55E–03	9.69E–03
	Std	(8.81E–05)	(4.19E–03)	(1.23E–03)	(1.86E–03)	(5.54E–04)	(5.18E–04)
RM-MEDA-AcPD ($\rho_{max} = 10$)	p	6.16E–01	1.54E–05	3.30E–06	3.95E–01	5.55E–07	1.48E–04
	Significance	\approx	+	+	\approx	+	+
	Mean	9.43E–04	2.74E–02	1.84E–02	1.36E–02	3.14E–02	1.16E–02
	Std	(9.96E–05)	(7.78E–03)	(3.03E–03)	(4.51E–03)	(1.33E–01)	(1.03E–03)
	p	5.38E–01	1.27E–13	4.22E–13	1.08E–02	9.48E–18	7.96E–18
RM-MEDA-AcPD ($\rho_{max} = 15$)	Significance	\approx	+	+	+	+	+
RM-MEDA-AcPD ($\rho_{max} = 5$)	Mean	3.05E–01	6.32E–01	8.48E–01	3.83E–01	5.27E–01	8.32E–01
	Std	(1.34E–05)	(1.02E–03)	(6.11E–04)	(8.53E–04)	(5.60E–04)	(3.43E–04)
	Significance	/	/	/	/	/	/
	Mean	3.05E–01	6.32E–01	8.48E–01	3.82E–01	5.27E–01	8.32E–01
	Std	(1.95E–05)	(1.04E–03)	(5.54E–04)	(8.94E–04)	(5.24E–04)	(3.24E–04)
RM-MEDA-AcPD ($\rho_{max} = 3$)	p	5.23E–02	2.78E–01	2.58E–02	8.41E–01	3.73E–01	1.83E–01
	Significance	\approx	\approx	+	\approx	\approx	\approx
	Mean	3.05E–01	6.31E–01	8.47E–01	3.82E–01	5.26E–01	8.32E–01
	Std	(1.90E–05)	(1.90E–03)	(6.03E–04)	(1.27E–03)	(9.34E–04)	(3.37E–04)
	p	1.46E–01	3.74E–05	4.05E–07	5.34E–01	6.26E–06	1.72E–07
RM-MEDA-AcPD ($\rho_{max} = 10$)	Significance	\approx	+	+	\approx	+	+
	Mean	3.05E–01	6.27E–01	8.46E–01	3.81E–01	5.24E–01	8.30E–01
	Std	(3.34E–05)	(3.20E–03)	(1.25E–03)	(2.37E–03)	(1.90E–03)	(5.93E–04)
	p	2.69E–01	2.45E–13	1.07E–12	2.81E–02	3.01E–17	6.30E–18
	Significance	\approx	+	+	+	+	+

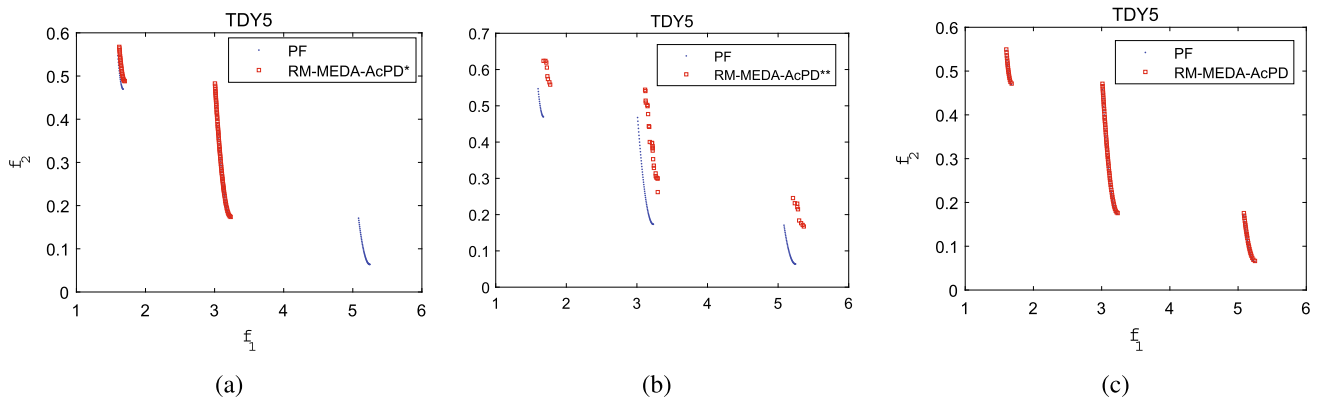
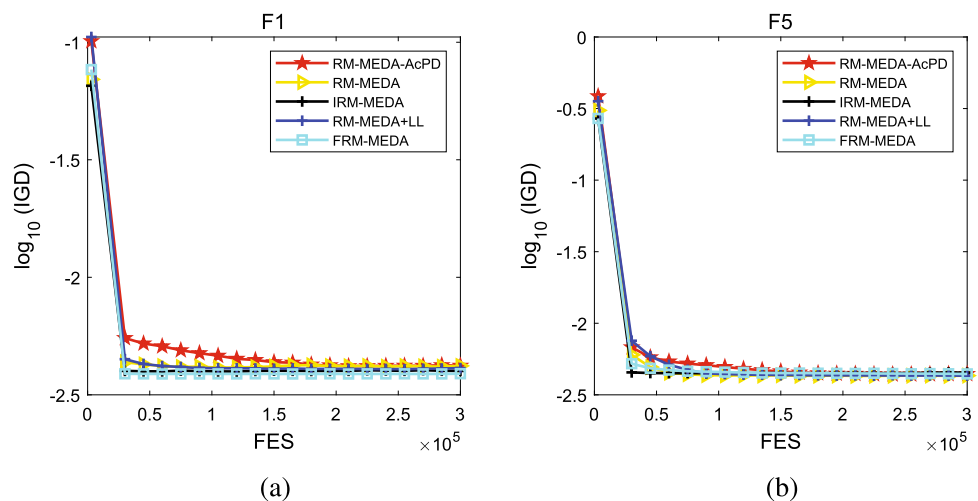


Fig. 15 Final non-dominated solutions obtained by RM-MEDA-AcPD*, RM-MEDA-AcPD**, RM-MEDA-AcPD on TDY5 problem in the run associated with the median IGD value. The red points denote the obtained final non-dominated solutions, while the blue points denote the PF (colour figure online)

Fig. 16 The evolution of averaged IGD values of RM-MEDA-AcPD, RM-MEDA, IRM-MEDA, RM-MEDA+LL, and FRM-MEDA over 51 runs for F1 and F5 problems with 30 decision variables, respectively



Acknowledgements The authors wish to thank the partial support of the National Natural Science Foundation of China (61803301, 62176146, 62272384), the Key Project of Shaanxi Key Research and Development Program (2020ZDLGR07-06), the Natural Science Foundation of Shaanxi (2022JQ-674, 2021JM-343), the Three year action plan project of Xi'an University (2021XDJH20), and the Doctoral Foundation of Xi'an University of Technology (112-256081812). They also thank Prof. Ran Cheng, Prof. Yong Wang, Prof. Aimin Zhou, Prof. Hui Li, Prof. Hanlin Liu and Prof. Yanan Sun for selflessly sharing their codes, which has greatly promoted our research work.

References

- Zhang XY, Tian Y, Cheng R et al (2015) An efficient approach to non-dominated sorting for evolutionary multi-objective optimization. *IEEE Trans Evol Comput* 19(2):201–213
- Li K, Deb K, Zhang QF et al (2017) Efficient nondomination level update method for steady-state evolutionary multiobjective optimization. *IEEE Trans Cybern* 47(9):2838–2849
- Wang R, Zhang Q, Zhang T (2016) Decomposition-based algorithms using Pareto adaptive scalarizing methods. *IEEE Trans Evol Comput* 20(6):821–837
- Ma XL, Yu YN, Li XD et al (2020) A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms. *IEEE Trans Evol Comput* 24(4):634–649
- Tian Y, Cheng R, Zhang XY et al (2018) An indicator based multi-objective evolutionary algorithm with reference point adaptation for better versatility. *IEEE Trans Evol Comput* 22(4):609–622
- Shang K, Ishibuchi H (2020) A new hypervolume-based evolutionary algorithm for many-objective optimization. *IEEE Trans Evol Comput* 24(5):839–852
- Hauschild M, Pelikan M (2011) An introduction and survey of estimation of distribution algorithms. *Swarm Evol Comput* 1(3):111–128
- Cheng R, He C, Jin YC et al (2018) Model-based evolutionary algorithms: a short survey. *Complex Intell Syst* 4:283–292
- Costa M, Minisci E (2003) MOPED: a multi-objective Parzen-based estimation of distribution algorithm for continuous problems, vol 2632. *Lecture notes in computer science*. Springer, Berlin, Heidelberg
- Okabe T, Jin YC, Bernhard S et al (2004) Voronoi-based estimation of distribution algorithm for multi-objective optimization. In: *Proceedings of the 2004 congress on evolutionary computation*, Portland, OR, USA, pp 1594–1601
- Martí L, García J, Berlanga A et al (2011) MB-GNG: addressing drawbacks in multi-objective optimization estimation of distribution algorithms. *Oper Res Lett* 39(2):150–154

12. Karshenas H, Santana R, Bielza C et al (2013) Multi-objective estimation of distribution algorithm based on joint modeling of objectives and variables. *IEEE Trans Evol Comput* 18(4):519–542
13. Cheng R, Jin YC, Narukawa K et al (2015) A multiobjective evolutionary algorithm using Gaussian process based inverse modeling. *IEEE Trans Evol Comput* 19(6):761–856
14. Lin YY, Liu H, Jiang QY (2018) Dynamic reference vectors and biased crossover use for inverse model based evolutionary multi-objective optimization with irregular Pareto fronts. *Appl Intell* 48:3116–3142
15. Cheng R, Jin YC, Narukawa K (2015) Adaptive reference vector generation for inverse model based evolutionary multi-objective optimization with degenerate and disconnected Pareto fronts. *Lect Notes Comput Sci* 9018:127–140
16. Lin T, Zhang H, Zhan K et al (2017) An adaptive multiobjective estimation of distribution algorithm with a novel Gaussian sampling strategy. *Soft Comput* 21:6043–6061
17. Sun YN, Yen GG, Zhang Y (2017) Reference line-based estimation of distribution algorithm for many-objective optimization. *Knowl-Based Syst* 132:129–143
18. Laumanns M, Ocenasek J (2002) Bayesian optimization algorithms for multi-objective optimization. In: *International conference on parallel problem solving from nature*. Springer, Berlin, pp 298–307
19. Bosman P A, Thierens D (2006) Multi-objective optimization with the Naive MIDEA. In: *Towards a new evolutionary computation. Advances in estimation of distribution algorithms*. Springer, Berlin, pp 123–157
20. Ocenasek J, Kern S, Hansen N et al (2004) A mixed Bayesian optimization algorithm with variance adaptation. In: *International conference on parallel problem solving from nature*. Springer, Berlin, pp 352–361
21. Igel C, Hansen N, Roth S (2007) Covariance matrix adaptation for multi-objective optimization. *Evol Comput* 15(1):1–28
22. Cheng R, He C, Jin YC et al (2018) Model-based evolutionary algorithms: a short survey. *Complex Intell Syst* 4:283–292
23. Miettinen K (1999) *Nonlinear multiobjective optimization*. Kluwer Academic, Norwell
24. Zhang QF, Zhou AM, Jin YC (2008) RM-MEDA: a regularity model based multi-objective estimation of distribution algorithm. *IEEE Trans Evol Comput* 12(1):41–63
25. Glaudell R, Garcia RT, Garcia JB (1965) Nelder–Mead simplex method. *Comput J* 7:308–313
26. Wang Y, Xiang J, Cai ZX (2012) A regularity model-based multiobjective estimation of distribution algorithm with reducing redundant cluster operator. *Appl Soft Comput* 12(11):3526–3538
27. Shi MF, He ZS, Chen ZY et al (2018) A full variate Gaussian model-based RM-MEDA without clustering process. *Int J Mach Learn Cybern* 9:1591–1608
28. Dong B, Zhou AM, Zhang GX (2016) Sampling in latent space for a multiobjective estimation of distribution algorithm. In: *Proceedings of the 2016 IEEE congress on evolutionary computation (CEC)*, Vancouver, BC, pp 3027–3034
29. Li YY, Xu X, Li P et al (2014) Improved RM-MEDA with local learning. *Soft Comput* 18(7):1383–1397
30. Zhou AM, Zhang QF, Jin YC et al (2007) Global multiobjective optimization via estimation of distribution algorithm with biased initialization and crossover. In: *Proceedings of the 9th annual conference on Genetic and evolutionary computation*. ACM, pp 617–623
31. Zhou AM, Zhang QF, Jin YC (2009) Approximating the set of Pareto–Optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm. *IEEE Trans Evol Comput* 13(5):1167–1188
32. Luo CY, Lu B, Chen MY (2010) Regularity model-based multiobjective estimation of distribution algorithm with two steps training method. *Control Decis* 25(7):1105–1112
33. Lin YY, Liu H, Jiang QY (2019) A double learning models-based multi-objective estimation of distribution algorithm. *IEEE Access* 7(1):144580–144590
34. Wang HD, Zhang QF et al (2016) Regularity model for noisy multiobjective optimization. *IEEE Trans Cybern* 46(9):1997–2009
35. Sun YN, Yen GG, Zhang Y (2018) Improved regularity model-based EDA for many-objective optimization. *IEEE Trans Evol Comput* 22(5):662–678
36. Zhang QY, Yang SX, Jiang SY et al (2020) Novel prediction strategies for dynamic multiobjective optimization. *IEEE Trans Evol Comput* 24(2):260–274
37. Liu HL, Gu FQ, Zhang QF (2014) Decomposition of a multi-objective optimization problem into a number of simple multi-objective subproblems. *IEEE Trans Evol Comput* 18(3):450–455
38. Liu HL, Chen L, Deb K et al (2017) Investigating the effect of imbalance between convergence and diversity in evolutionary multi-objective algorithms. *IEEE Transactions on Evolutionary Computation* 21(3):408–425
39. Chow CK, Yuen SY (2012) A multi-objective evolutionary algorithm that diversifies population by its density. *IEEE Trans Evol Comput* 16(2):149–172
40. Zhang H, Zhou AM, Song SM et al (2016) A self-organizing multiobjective evolutionary algorithm. *IEEE Trans Evol Comput* 20(5):792–806
41. Jiang SW, Ong YS, Zhang J et al (2014) Consistencies and contradictions of performance metrics in multiobjective optimization. *IEEE Trans Cybern* 44(12):2391–2404
42. Zhang QF, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Comput* 11(6):712–731
43. Alcalá-Fdez J, Sánchez L, García S et al (2009) KEEL: a software tool to assess evolutionary algorithms for data mining problems. *Soft Comput* 13(3):307–318
44. Li BD, Tang K, Li JL et al (2016) Stochastic ranking algorithm for many-objective optimization based on multiple indicators. *IEEE Trans Evol Comput* 20(6):924–938
45. Poláková R, Tvrdík J, Bujok P (2019) Differential evolution with adaptive mechanism of population size according to current population diversity. *Swarm Evol Comput* 50:100519
46. Yue CT, Liang JJ, Suganthan PN et al (2020) MMOGA for solving multimodal multiobjective optimization problems with local Pareto sets. In: *Proceedings of the 2020 IEEE congress on evolutionary computation*, Glasgow, United Kingdom, pp 1–8

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.