IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received April 12, 2019, accepted April 28, 2019, date of publication May 10, 2019, date of current version May 24, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2916082

A Special Points-Based Hybrid Prediction Strategy for Dynamic Multi-Objective Optimization

JIANXIA LI, RUOCHEN LIU, (Member, IEEE), RUINAN WANG, JIN LIU, AND CAIHONG MU, (Member, IEEE)
Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an 710071, China

Corresponding author: Ruochen Liu (ruochenliu@xidian.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61876141, Grant 61373111, Grant 61672405, and Grant 61871306, and in part by the Provincial Natural Science Foundation of Shaanxi of China under Grant 2019JZ-26.

ABSTRACT Dynamic multi-objective optimization problem (DMOP) is such a type of optimization problems that multiple contradictory objectives change over time. This paper designs a special point-based hybrid prediction strategy (SHPS) integrated into the decomposition-based multi-objective optimization algorithm with differential evolution (MOEA/D-DE) to handle DMOPs, which is denoted as MOEA/D-DE-SHPS. In the SHPS, when historical information is insufficient to establish prediction model of population prediction strategy (PPS), the prediction (PRE) and variation (VAR) method are adapted to generate the initial population of the new environment. Meanwhile, the PPS predicts the whole population of new environment according to the history information collected from past environments; therefore, once collected historical information is inaccurate, the predicted population may be located in the wrong search region. To overcome the shortcoming, we propose the special point-based strategy in which the initial population of the new environment consists of two parts of individuals: the predicted special points and the predicted population by PPS (except the special points). The empirical results show that MOEA/D-DE-SHPS is promising for handling DMOPs.

INDEX TERMS Multi-objective optimization, decomposition, differential evolution, special points, hybrid prediction strategy.

I. INTRODUCTION

Many optimization problems refer to optimizing multiple contradictory objectives, which are denoted as multi-objective optimization problems (MOPs). For MOPs, there is no optimal solution that satisfies simultaneous optimization of multiple objectives. Therefore, the goal of dealing with MOPs is to obtain Pareto-optimal Set (*PS*).

Evolutionary algorithms (EAs) take a vital role in dealing with MOPs [1]–[7], and a large number of outstanding works have emerged, for instance, non-dominated sorting genetic algorithm II (NSGA-II) [2]; multi-objective evolutionary algorithm based on decomposition (MOEA/D) [3], [4]; regularity model based multi-objective estimation of distribution algorithm (RM-MEDA) [6], [7].

The associate editor coordinating the review of this manuscript and approving it for publication was Orazio Gambino.

However, there exist some MOPs changing over time, called as dynamic multi-objective optimization problems (DMOPs). How to find the *PS* of each environment quickly and effectively is a challenge that must be faced for solving DMOPs. Recently, scholars have done some researches to deal with DMOPs [8]–[10]. For instance, population prediction strategy (PPS) [10] could predict the position of individuals of new environment on the basic of historical information of *PS* of previous environments. However, inaccurate prediction may mislead search and lead to the inability to find the Pareto front (*PF*) quickly.

This paper develops a special points-based hybrid prediction strategy (SHPS) to respond to the environmental changes. Firstly, PPS uses historical information of the previous 23 environments to construct an auto regression (AR) model to predict initial population of new environment, so when historical information is not enough to establish AR model of PPS, prediction (PRE) & variation (VAR) method [8] is adopted to generate the new population. Secondly, when AR model works, PPS predicts the whole population based on the historical information, once the collected information is not accurate, the whole predicted population may be located in wrong search region, and the algorithm can't respond to environmental changes effectively. To avoid invalid prediction mislead the search, we propose a special points-based strategy, in which, the initial population of new environment is generated by combining the predicted special points and the population (except the special points) predicted by PPS. SHPS is integrated into the differential evolutional-based MOEA/D (MOEA/D-DE) to handle DMOPs, referred to as MOEA/D-DE-SHPS. The reason for adopting MOEA/D-DE as the basic optimization algorithm is its lower computational complexity and better performance [4], [11]–[14].

The main structure of this paper consists: Section 2 introduces the related background. Section 3 detailedly describes MOEA/D-DE-SHPS. Section 4 presents test functions, performance metric, empirical results and analysis. Finally, conclusion and the future research are shown in the last section.

II. THE RELATED BACKGROUND

This section mainly introduces the definition of DMOP and some representative works on solving DMOPs, then presents the descriptions of special points, PPS, PRE&VAR.

A. THE DEFINITION OF DMOPs

The mathematic description of a DMOP is presented as follows [15]:

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

where t is the time variable, $x = (x_1, x_2, \ldots, x_n)^T$ is the n-dimension decision vector within the decision space Ω_x, $F(x,t)$ represents the objective function. $g(x,t) \le 0$ and $h(x,t) = 0$ represent inequality and equality constraints, respectively.

B. REPRESENTATIVE WORKS ON SOLVING DMOPs

Yamasaki [16] firstly proposed a dynamic Pareto optimum genetic algorithm. After Farina *et al.* [15] proposed the dynamic test functions, application instances and solving method of DMOPs, dynamic multi-objective optimization algorithms (DMOOAs) gradually got some scholars' attentions. Many researchers have designed a lot of DMOOAs [8]–[10], [17]–[19] based on evolutionary computation [9], immune-based algorithms [20], and particle swarm optimization (PSO) [21].

The goal of a DMOOA is to track the changing *PS* and respond to the environmental changes effectively, so it is important to react to the changes for a DMOOA when environment changes, this paper makes a simple summary of the representative works on solving DMOPs from the perspective of the adopted response mechanisms.

1) DIVERSITY INTRODUCTION STRATEGY

To some extent, it is necessary to introduce diversity [17], [22]–[24] when an environmental change occurs, so that algorithms can respond to the environmental change effectively.

Deb *et al.* [17] proposed two typical diversity introduction strategy. For the first method, part of individuals are replaced by randomly generated new individuals to obtain initial population of new environment, known as DNSGA-II-A. In another method, a certain proportion individuals of population are disturbed by Gaussian noise when a change occurs, called as DNSGA-II-B.

However, the DMOOAs based on the diversity introduction need to analyze how to determine the proportion of diversity introduction for different DMOPs.

2) DIVERSITY MAINTAINING STRATEGY

Diversity maintaining strategy focuses on maintaining diversity when an environmental change occurs. For instance, dynamic orthogonal multi-objective evolutionary algorithm (DOMOEA) [25] directly adopted *PS* of the previous environment to act as initial population of new environment.

This approach may be suitable for solving DMOPs with small changes, but it may performs poorly when the change is severe.

3) MEMORY-BASED STRATEGY

Memory-based strategy stores the old optimal solutions found in previous environments and reuses the stored solutions in the new environment.

Dynamic constrained multi-objective optimization artificial immune system (DCMOAIS) [26] performs well in dealing with DMOPs. In DCMOAIS, T-module works to detect the environmental changes and initialize the population based on historical information. B-module is designed to search for *PS*. M-model stores all non-dominated solutions, and when a change occurs, M-model assists T-model to initialize the population.

Memory-based strategy may be effective when environmental changes are periodical and recurrent. However, the stored information may cause the redundant and effect the performance of algorithm.

4) PREDICTION STRATEGY

In certain cases, environment changes may follow a certain pattern that can be predicted, therefore it is vital for us to find the rule of the pattern to predict next environmental change.

Feed-forward prediction strategy (FPS) [27] was combined with queuing multi-objective optimizer (QMOO) to deal with DMOPs. A sequence of optimum solutions found in the previous environments is used to build forecasting model. When the environmental change occurs, the forecasting model is triggered to generate the individuals for next environment.

In [10], PPS utilized historical information gathered from the previous environments to predict initial population for new environment. In PPS, the *PS* consists of a center point and a manifold, which are predicted respectively. For the center prediction, PPS uses the center of the previous 23 environments to conduct a univariate AR model to predict the center point of new environment. PPS also uses the manifolds of the previous two environments to predict the manifold for new environment. Then the predicted center and manifold are combined together to create the initial population of new environment. Steady-state and generational evolutionary algorithm (SGEA) [19] reused some outdated solutions with better distribution and relocated other solutions according to the information gathered from previous and new environments. Directed search strategy (DSS) [18] reinitialized the new population based on the moving direction of *PS* and orthogonal direction when the environmental change occurs.

Prediction strategy may be effective when the prediction is correct. However, the prediction is based on historical information, if the algorithms fail to track the optimal solutions in the previous environments, the historical information may be not helpful. Prediction approaches just suitable for dealing with DMOPs that are easily to be predicted. If the environmental changes are unpredicted, prediction strategy may not perform well.

5) SELF-ADAPTIVE STRATEGY

To better handle DMOPs, some researchers proposed self-adaptive approaches to respond to environmental changes.

Liu *et al.* [28] proposed a self-adaptive diversity introduction (SADI) to solve DMOPs, in which, the ratio of introduction diversity is adaptively determined by the extent of environmental change. However, experimental results have shown that SADI performed poorly on solving DMOPs.

So far, research on self-adaptive strategy is still in its infancy, and an effective self-adaptive response strategy has not yet been proposed.

C. SPECIAL POINTS

This section introduces the concepts of special points, including boundary points, close-to-ideal (CTI) point and knee point.

1) BOUNDARY POINTS

For the minimization problems, boundary points are those individuals with the smallest objective function value in each dimension in the objective space. Shown as Fig. 1, A and G are the boundary points.

2) CTI POINT

CTI point [29] is the point closest to the ideal point. Supposing point H is the ideal point, whose description is shown as

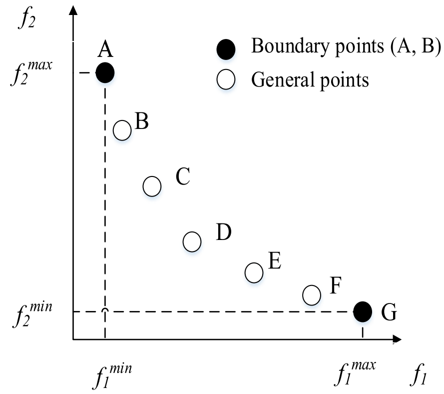

FIGURE 1. Boundary points.

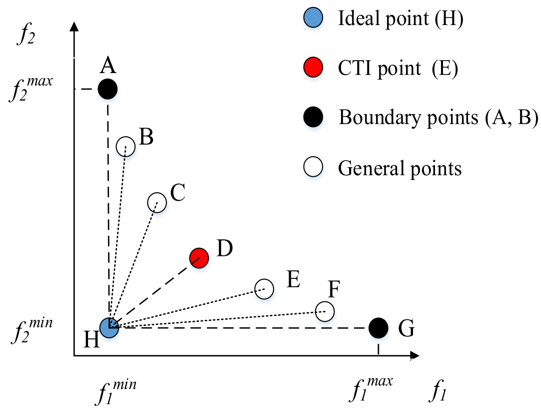

FIGURE 2. CTI point.

Eq. (2).

$$H_i = min\left\{f(P_i^1), \ldots, f(P_i^{num})\right\} \tag{2}$$

where $i = 1, 2, \ldots, m$, m represents the dimension of objective space. *num* represents the number of non-domination solutions. $f(P_i^{num})$ represents the i-th dimension objective function value of *num*-th individual. As shown in Fig. 2, H represents the ideal point, D is CTI point, which is the closest point to the ideal point.

3) KNEE POINT

The knee point [30]–[33] is refer to the point with the maximum marginal rates of return. As shown in Fig. 3, A and G represent the boundary points. The connection of A and G could obtain a line L. The knee point is the point with the largest vertical distance to L. It can be seen from Fig. 3 that the distance from D to L is the largest, therefore D is the knee point.

The definition of the L is shown as Eq. (3).

$$ax + by + c = 0 \tag{3}$$

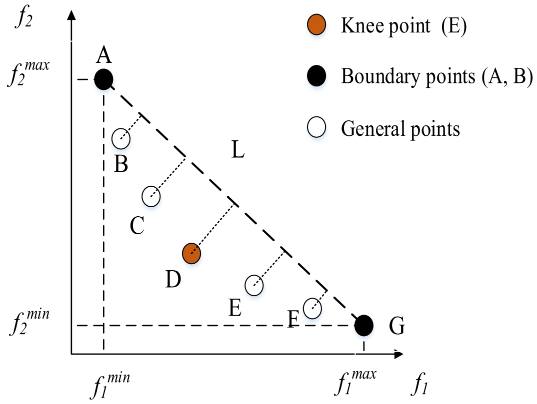

FIGURE 3. Knee point.

Assuming that the coordinate of point K is (x_k, y_k), the distance from K to L is defined as:

$$d(K, L) = \frac{|ax_k + by_k + c|}{\sqrt{a^2 + b^2}} \tag{4}$$

D. PPS

PPS predicts the initial population of new environment on the basic of historical information gathered from previous environments when a change is detected. The PS consists of a center and a manifold which are predicted respectively.

In the $t - th$ environment, the PS is divided into a center C^t and a manifold M^t, so PS^t can be formulated as follows:

$$PS^t = C^t + M^t \tag{5}$$

If the PS of the $t - th$ environment is $PS^t = x^t$. Then the center of PS^t can be estimated as shown in Eq. (6).

$$C^t = \frac{1}{|PS^t|} \sum_{x^t \in PS^t} x^t \tag{6}$$

where $|PS^t|$ is the cardinality of PS^t. Then every point x^t in PS^t can be defined as:

$$x^t = C^t + \tilde{x}^t \tag{7}$$

With these in mind, we can derive that the manifold of PS^t is shown as Eq. (8).

$$M^t = \tilde{x}^t \tag{8}$$

Followed are the detailed introduction of the prediction of center and manifold.

1) CENTER PREDICTION

Firstly, an AR model is used to predict the center of population of the $(t + 1) - th$ environment, i.e. C^{t+1}, and the order of the model is set as p=3, the model is denoted as AR (3) model which is described as:

$$C^{t+1} = \sum_{j=1}^{p} \theta^j C^{t-j+1} + \varepsilon^t \tag{9}$$

where θ is the parameter of AR(3) model, $\varepsilon^t \sim N(0, \sigma 1^t)$ is a white noise, and $\sigma 1^t = (\sigma 1_1^t, \sigma 1_2^t, \ldots, \sigma 1_n^t)^T$.

Considering the $i - th$ dimension, and setting the length of time series is $M = 23$. The parameters of AR(3) model are calculated according to the information accumulated from the previous 23 environments, which is described as follows.

$$\begin{aligned} C_i^t &= \theta_i^1 C_i^{t-1} + \theta_i^2 C_i^{t-2} + \theta_i^3 C_i^{t-3} \\ C_i^{t-1} &= \theta_i^1 C_i^{t-2} + \theta_i^2 C_i^{t-3} + \theta_i^3 C_i^{t-4} \\ &\cdots \\ C_i^{t-20} &= \theta_i^1 C_i^{t-21} + \theta_i^2 C_i^{t-22} + \theta_i^3 C_i^{t-23} \end{aligned} \tag{10}$$

Let $\Psi_i = (C_i^t, C_i^{t-1}, \ldots, C_i^{t-20})^T$, and

$$\Phi_i = \begin{bmatrix} C_i^{t-1} & C_i^{t-2} & C_i^{t-3} \\ C_i^{t-2} & C_i^{t-3} & C_i^{t-4} \\ \vdots & \vdots & \vdots \\ C_i^{t-21} & C_i^{t-22} & C_i^{t-23} \end{bmatrix} \tag{11}$$

Eq. (10) can be abbreviated as:

$$\Psi_i = \Phi_i(\theta_i^1, \theta_i^2, \theta_i^3)^T \tag{12}$$

The least squares regression method is used to calculate θ, as described in Eq. (13).

$$(\theta_i^1, \theta_i^2, \theta_i^3)^T = (\Phi_i^T \Phi_i)^{-1} \Phi_i^T \Phi_i \tag{13}$$

Finally, the parameter $\sigma 1_i^t$ is the average squared error, shown as Eq. (14).

$$\sigma 1_i^t = \frac{1}{M - p} \sum_{k=t-M+p}^{t} [C_i^k - \theta_i^1 C_i^{k-1} - \theta_i^2 C_i^{k-2} - \theta_i^3 C_i^{k-3}] \tag{14}$$

2) MANIFOLD PREDICTION

PPS uses the manifold of two previous environments, M^t and M^{t-1}, to predict the manifold of population of the $(t + 1) - th$ environment, i.e. M^{t+1}. Considering the i-th dimension, M_i^{t+1} is predicted by Eq. (15).

$$M_i^{t+1} = M_i^t + \varepsilon_i^t \tag{15}$$

where $\varepsilon_i^t \sim N(0, \sigma 2_i^t)$, $\sigma 2_i^t$ is given as follows:

$$\sigma 2_i^t = \frac{1}{n} D(M^t, M^{t-1})^2 \tag{16}$$

where $D(M^t, M^{t-1})$ is the distance between manifolds M^t and M^{t-1}, defined as Eq. (17).

$$\sigma 2_i^t = \frac{1}{|M^t|} \sum_{x \in M^t} \min_{y \in M^{t-1}} \|x - y\| \tag{17}$$

Finally, the combination of C^{t+1} and M^{t+1} form the initial population of the $(t + 1) - th$ environment, i.e. POP^{t+1}.

$$POP^{t+1} = C^{t+1} + M^{t+1} \tag{18}$$

The framework of PPS is shown in Algorithm 1.

Algorithm 1 PPS

Input: M (maximum length of historical center points sequence), p (the order of AR(p) model), t (the time step);

Output: POP^{t+1} (initial population of the $(t+1)-th$ environment)

Step 1: if $t \leq M$, turn to Step 2, otherwise, turn to Step 3;

Step 2: POP^{t+1} consists of half randomly generated individuals and half of the individuals inherited from the PS^t;

Step 3: Generate POP_{t+1} according to Eq. (18); For each individual x^{t+1} in POP^{t+1}, if x^{t+1} is outside the boundary, then it will be repaired according to the following equation;

$$x_i^{t+1} = \begin{cases} x_i^{t+1}, & \text{if } a_i \leq x_i^{t+1} \leq b_i \\ 0.5(a_i + x_i^t), & \text{if } x_i^{t+1} < a_i \\ 0.5(b_i + x_i^t), & \text{if } x_i^{t+1} > b_i \end{cases}$$

where $i = 1, 2, \ldots, n$.

Step 4: Output POP^{t+1};

E. PRE&VAR METHOD

PRE&VAR method, firstly, generates a random number r, if $r < 0.5$, PRE is activated to generate the initial population of new environment, otherwise, the VAR is triggered to respond to react to the environmental changes. The mathematic expression is shown as Eq. (19).

$$x^{t+1} = \begin{cases} PRE, & \text{if } r < 0.5 \\ VAR, & \text{otherwise} \end{cases} \tag{19}$$

where r represents the random number in [0, 1].

1) PRE

Supposing that the PS of the t previous environments are $PS^t, PS^{t-1}, \ldots, PS^1$, and $x^t, x^{t-1}, \ldots x^1$, $x^i \in PS^i$, $i = 1, \ldots, t$, then the individuals of initial population of the $(t+1)-th$ environment can be predicted as shown in Eq. (20).

$$x^{t+1} = F(x^t, x^{t-1}) = x^t + (x^t - x^{t-1}) \tag{20}$$

In PRE, for each individual $x^t \in PS^t$, its parent is the nearest point x^{t-1} in PS^{t-1}, as shown in Eq. (21).

$$x^{t-1} = arg \min_{x \in PS^{t-1}} \left\| x - x^t \right\|_2 \tag{21}$$

where $||x - x^t||_2$ denotes the Euclidean Distance between x and x^t.

2) VAR

VAR is introduced to enhance the diversity, in which, a "predicted" noise shown as Eq. (22) is added to the initial population.

$$\varepsilon \sim N(0, \delta I) \tag{22}$$

Algorithm 2 PRE&VAR

Input: t (the time step), PS^t (the PS of $t-th$ environment), PS^{t-1} (the PS of $(t+1)-th$ environment);

Output: initial population of the $(t+1)-th$ environment)

Step 1: if $t = 1$, turn to Step 2, otherwise, turn to Step 3;

Step 2: Adding a Gauss noise on PS^t to obtain the initial population of the $(t+1)-th$ environment;

Step 3: Generate new individuals x^{t+1} according to PRE&VAR method, defined as Eq. (19);

Step 4: Output the new population consisting of the individuals of x^{t+1};

where δ is the standard deviation, which is estimated as Eq. (23).

$$\delta^2 = \frac{1}{4n} \left\| x^t - x^{t-1} \right\|_2^2 \tag{23}$$

where n is the number of decision vector.

Then, initial individuals in the $(t+1)-th$ environment could be generated using the following Eq. (24).

$$x^{t+1} = x^t + \varepsilon \tag{24}$$

The PRE&VAR method is shown as Algorithm 2.

III. THE PROPOSED MOEA/D-DE-SHPS

This paper proposes a special points-based hybrid prediction strategy (SHPS) to respond to the environmental changes.

PPS could predict the population after an environmental change occurs. However, PPS relies on the historical information collected in previous environments, the inaccurate historical information may lead to the predicted population to be located in incorrect search region. And AR model works after the historical information is enough to establish AR model, i.e. after the 23-rd environmental change in this paper. When the historical information is not enough, the PPS chooses half individuals randomly from the previous environment and randomly generates another half individuals, which is not a valid method to predict new population, so the information collected from the previous environments may not accurate, which could influence the predicted accuracy of the AR model. To solve this problem, we employ PRE&VAR method when the information is not enough to build AR model. PRE&VAR only relies on historical information of the previous two environments and the experimental results in [8] have indicated that PRE&VAR is of better performance.

Meanwhile, PPS predicts the whole population based on the historical information when the environmental change occurs, once the collected information is not accurate, the whole predicted population may be located in wrong search region, and the algorithm can't react to environmental changed effectively. To solve this problem, we introduce the special points to reduce the influence of incorrect prediction.

This article focuses on special points to avoid the fact that population prediction depends completely on historical information of central points and manifolds. At the same time, when the population's center point and manifold are not accurate, or when the population predicted by PPS is not accurate enough, to some extent, the attention to special points can weaken the impact of wrong prediction when the environmental change occurs.

The main idea is that the predicted population is combined with two parts of individuals, one part is the population predicted by PPS, which is predicted according to historical information of the center of the population and the manifold, and the other part is the special points predicted by using the historical information of the special points.

The flow of MOEA/D-DE-SHPS is given in Algorithm 3.

A. CHANGE DETECTION

Change detection mechanism aims to judge whether changes have occurred in the environment, that is, whether the problem at the current environment is different from the problem at the previous environment.

There are mainly two mechanisms to detect change, including re-evaluating solutions [9], [10], [17], [18] and checking population statistical information [34]. The former mechanism selects a part of the individuals from the population and re-evaluates them. If there is a difference between two generations, then it is identified that the environment changes. The main idea of the latter mechanism is that if the objective solution set of two generations belongs to the different statistical distribution, then it is considered that the environment changes.

In this paper, we employ the first change detection mechanism [15] to determine whether the environment changes, which is described as Eq. (25).

$$\zeta(k) = \frac{\sum_{i=1}^{n_s} \left\| \frac{F(x^i,k)-F(x^i,k-1)}{R(k)-U(k)} \right\|}{n_s} \tag{25}$$

where $R(k) = (r_1, r_2, \ldots, r_m)^T$ and $U(k) = (u_1, u_2, \ldots, u_m)^T$ represent the maximum and minimum objective function values of those individuals used to detect changes of $k-th$ generation, n_s represents the number of individuals selected to judge environmental changes. when $\zeta(k)$ exceeds a certain threshold, it means that an environmental change occurs, and the response mechanism, i.e. SHPS, needs to work to respond to environmental change. In this paper, n_s is set as the 10% of the population size.

B. SHPS

In SHPS, when the historical information collected from the previous environments is not enough to build AR model, the PRE&VAR method is used to generate initial population of the new environment. If the information is enough to build AR model, in order to avoid the predicted initial population is located in wrong search region, SHPS introduces special points-based strategy, the predicted population includes two parts, one part of individual is predicted by PPS utilizing the history information of center and manifold, the other part of individuals is the predicted special points. As same as PPS, each special point is predicted using AR model introduced in Section II-D.1, based on the historical information of all the special points. SHPS is shown as Algorithm 4.

Algorithm 3 MOEA/D-DE-SHPS

Input: N (population size), T (the number of neighbor vectors of each weight vector), T_{max} (the number of environmental changes);

Output: $PS^1, \ldots, PS^{T_{max}}$ (the PS of each environment)

Step 1: **Initialization:**

Step 1.1: Set time step $t = 0$;

Step 1.2: Initialize a population POP^t with the individuals $x^1, \ldots, x^N$, and compute the objective value of each individual, $FV^i = F(x^i)$;

Step 1.3: Generate a group of evenly distributed weight vectors: $\lambda^1, \ldots, \lambda^N$;

Step 1.4: Compute Euclidean distance between λ^i and other weight vectors, and find the T closest weight vectors of λ^i. The T weight vectors form the neighbor vectors of λ^i. For λ^i, the index of each neighbor weight vector stores in $B(i)$;

Step 1.5: Initialize reference point $z = (z_1, \ldots, z_m)^T$, where $z_j = \min_{1\le i\le N} f_j(x^i)$;

Step 2: **Change detection**: detect weather the environment changes. If an environmental change occurs, output PS^t, and set $t = t + 1$, continue; otherwise, turn to Step 4;

Step 3: **React to environmental changes**: generate the initial population of new environment by SHPS;

Step 4: **MOEA/D-DE**:

For i=1:N

Step 4.1: Randomly choose three indexes from $B(i)$, then find the three individuals corresponded by the above three indexes from pop^t as the parent individuals and generate new offspring individual y by genetic operators, including differential crossover and polynomial mutation;

Step 4.2: If y exceeds the feasible region, it should be fixed within the feasible region;

Step 4.3: For each $j = 1, .., m$, if $z_j > f_j(y)$, then set $z_j = f_j(y)$;

Step 4.4: For each k in $B(i)$, if $g^{te}(y|\lambda^k, z) \le g^{te}(x^k|\lambda^k, z)$, then set $x^k = y$ and $FV^k = F(y)$;

end

Step 5: **Termination criteria**: if termination criteria is met, end; Otherwise, turn to Step 2;

IV. EMPIRICAL STUDIES

This section discusses the performance of our proposed algorithm by conducting two empirical studies, including a comparison between MOEA/D-DE-SHPS with other five DMOOAs, i.e., DNSGA-II-A [17], DNSGA-II-B [17], RM-MEDA based on PRE&VAR (RM-MEDA-PRE&VAR)

Algorithm 4 SHPS

Input: t (the time step);
Output: initial population of the $(t+1)-th$ environment);
Step 1: If $t \leq M$, turn to Step 2, otherwise turn to Step 3;
Step 2: Generate the initial population of the $(t+1)-th$ environment using PRE&VAR;
Step 3: Use the historical information of special points to predict the special points of the $(t+1)-th$ environment through AR model;
Step 4: Use PPS to predict the individuals other than special points of the $(t+1)-th$ environment;
Step 5: Combine the predicted special population and the predicted population by PPS to obtain the initial population of new environment;

[8], RM-MEDA based on PPS (RM-MEDA-PPS) [10], and SGEA [19], and a comparison between SHPS and other two response strategies, i.e., PRE&VAR and PPS.

A. TEST FUNCTIONS

Eight test functions including five FDA functions [15], i.e. FDA1, FDA2, FDA3, FDA4, FDA5, and three dMOP functions [9], i.e. dMOP1, dMOP2, dMOP3, are used to examine the performance of different DMOOAs. The dynamics of all test functions are realized by $t = \lfloor \tau/\tau_T \rfloor / n_T$, where τ_T and n_T are the frequency and intensity of environmental changes, respectively.

B. PERFORMANCE METRIC

Inverted generational distance (IGD) [10], [35] can measure both convergence and diversity.

Supposing that P^{t^*} is true PF of the $t-th$ environment, P^t is the PF of the $t-th$ environment obtained by a DMOOA, then IGD could be defined as Eq. (26).

$$IGD_t(P^{t^*}, P^t) \frac{\sum_{v \in P^{t^*}} d(v, P^t)}{|P^{t^*}|} \tag{26}$$

where $d(v, P^t) = \min_{u \in P^t} \|F(v) - F(u)\|$ means Euclidean distance between individual v and its closest neighbor in P^t. The smaller IGD, the better the performance of the algorithm.

If the environment changes T_{max} times, then the average of IGD_t of the T_{max} environments, i.e. $MIGD$ is denoted as follows:

$$MIGD = \frac{1}{T_{max}} \sum_{t=1}^{T_{max}} IGD_t(P^{t^*}, P^t) \tag{27}$$

C. COMPARISON WITH OTHER FIVE WELL-KNOWN DMOOAs

1) PARAMETER SETTINGS

This section carries out a comparison between MOEA/D-DE-SHPS and DNSGA-II-A [17], DNSGA-II-B [17], RM-MEDA-PRE&VAR [8], RM-MEDA-PPS [10], and SGEA [19]. Most parameters are set based on the original references.

TABLE 1. Parameter settings of the six algorithms.

Algorithms	Parameter Settings
DNSGA-II-A DNSGA-II-B	SBX Crossover probability: $p_c = 1$; Cross distribution index: $\eta_c = 20$; Polynomial mutation probability: $p_m = 1/n$; Mutation distribution index: $\eta_m = 20$;
RMMEDA-PRE&VAR	The number of cluster: $K = 5$;
RMMEDA-PPS	The number of cluster: $K = 5$; The AR(p) model: $p = 3$; The length of time series: $M = 23$;
SGEA	SBX Crossover probability: $p_c = 1$; Crossover distribution index: $\eta_c = 20$; Polynomial mutation probability: $p_m = 1/n$; Mutation distribution index: $\eta_m = 20$;
MOEA/D-DE-SHPS	The number of the neighbor weight vectors of each weight vector: $T = 20$; The AR(p) model: $p = 3$; The length of time series: $M = 23$; Differential crossover probability: $CR = 0.8$; Differential scale factor: $F = 0.5$; Gaussian mutation probability: $p_m = 1/n$; Mutation distribution index: $\eta_m = 20$;

All the six algorithms share some general parameters: (1) population size: $N = 100$ (two-objective) or $N = 210$ (three-objective); (2) the number of individuals chosen to detect environmental changes n_s is set as $n_s = 0.1 \times N$; (3) the number of environment changes T_{max} is set as $T_{max} = 100$; (4) the max iteration generations $IterMax$ is set as $IterMax = 100 \times \tau_T$.

Table 1 presents the other key parameters of the six algorithms.

2) EMPIRICAL RESULTS

This section measures the performance of the DMOOAs under different types of environmental changes, i.e., (τ_T, n_T) is set as (5,10), (10,10), and (15,10). All the six algorithms perform 20 independent runs on each test function.

Table 2 presents the statistical results of $MIGD$ over 20 runs for each algorithm, in which, the black represents the best result of all six algorithms.

We could find that, for most test functions, MOEA/D-DE-SHPS performs best, including FDA1, FDA3, FDA5, dMOP3 under all different types of (τ_T, n_T); FDA2 and FDA4 when (τ_T, n_T) is set to (5,10), (10,10); dMOP1 when (τ_T, n_T) is set to (15,10); and dMOP2 when (τ_T, n_T) is set to (10,10), (15,10).

For dMOP1, when (τ_T, n_T) is set to (5,10) and (10,10), SGEA is of the best performance, MOEA/D-DE-SHPS takes the second place. For dMOP2, when τ_T is set as 5, SGEA outperforms MOEA/D-DE-SHPS. The reason of this phenomenon may be that SHPS rely on the degree of accuracy of history information. When τ_T is set to 5 or 10, the environment changes quickly, the static algorithm have not found

TABLE 2. The statistical results of *MIGD* of all six algorithms.

functions	(τ_t, n_t)	Statistic	DNSGA-II-A	DNSGA-II-B	RM-MEA -PRE&VAR	RM-MEA -PPS	SGEA	MOEA/D -DE-SHPS
FDA1	(5,10)	Mean	0.1730	0.1696	0.0210	0.9549	0.0191	**0.0167**
		Std	0.0212	0.0285	*0.0015*	0.3425	0.0016	0.0023
	(10,10)	Mean	0.0267	0.0285	0.0111	0.0864	0.0123	**0.0082**
		Std	0.0018	0.0038	5.0880e-4	0.0479	**2.6007e-4**	8.9158e-4
	(15,10)	Mean	0.0137	0.0141	0.0074	0.0123	0.0090	**0.0063**
		Std	5.4101e-4	2.2915e-4	**1.6038e-4**	0.0014	7.5327e-4	4.1175e-4
FDA2	(5,10)	Mean	0.0479	0.0488	0.0489	0.1238	0.0229	**0.0186**
		Std	0.0022	0.0030	0.0068	0.0436	**0.0014**	0.0030
	(10,10)	Mean	0.0357	0.0329	0.0261	0.0340	0.0093	**0.0093**
		Std	0.0025	0.0030	0.0012	0.0116	0.0011	**5.8960e-4**
	(15,10)	Mean	0.0125	0.0133	0.0231	0.0139	**0.0074**	0.0076
		Std	0.0017	0.0012	3.4477e-4	0.0024	**8.2773e-4**	0.0012
FDA3	(5,10)	Mean	0.2114	0.2088	0.0449	0.8502	0.0327	**0.0302**
		Std	0.0333	0.0304	0.0038	0.4238	0.0039	**0.0028**
	(10,10)	Mean	0.0308	0.0250	0.0378	0.2539	0.0358	**0.0194**
		Std	0.0027	0.0026	**9.3585e-4**	0.1277	0.0048	0.0012
	(15,10)	Mean	0.0227	0.0230	0.0403	0.0628	0.0265	**0.0156**
		Std	0.0012	0.0017	**9.2772e-4**	0.0251	0.0037	0.0014
FDA4	(5,10)	Mean	0.4480	0.4492	0.0931	0.2120	0.0849	**0.0623**
		Std	0.0185	0.0192	0.0028	0.0246	0.0019	**5.8707e-4**
	(10,10)	Mean	0.1708	0.3343	0.0686	0.0919	0.0669	**0.0528**
		Std	0.0117	0.0218	0.0016	0.0056	9.9599e-4	**1.9084e-4**
	(15,10)	Mean	0.0701	0.0682	0.0623	0.0696	**0.0493**	0.0499
		Std	0.0103	0.0025	7.0931e-4	0.0017	3.7789e-4	**9.9890e-5**
FDA5	(5,10)	Mean	0.6675	0.6717	0.1026	0.2498	0.1213	**0.0362**
		Std	0.0172	0.0228	0.0024	0.0570	0.0021	**3.9025e-4**
	(10,10)	Mean	0.4535	0.1186	0.0988	0.0730	0.0682	**0.0291**
		Std	0.0107	0.0208	0.0021	0.0127	0.0020	**1.9203e-4**
	(15,10)	Mean	0.3664	0.3695	0.1062	0.0419	0.0467	**0.0272**
		Std	0.0056	0.0048	0.0023	0.0025	0.0014	**1.8989e-4**
DMOP1	(5,10)	Mean	0.1196	0.0977	0.1168	0.5621	**0.0182**	0.0241
		Std	0.0512	0.0327	0.0170	0.8370	**0.0059**	0.0086
	(10,10)	Mean	0.0375	0.0291	0.0474	0.0578	**0.0072**	0.0087
		Std	0.0164	0.0107	0.0125	0.0956	0.0025	**0.0021**
	(15,10)	Mean	0.0238	0.0194	0.0295	0.0215	0.0074	**0.0057**
		Std	0.0088	0.0078	0.0067	0.0047	0.0015	**0.0014**
DMOP2	(5,10)	Mean	0.2263	0.2236	0.4501	1.2615	**0.0170**	0.0221
		Std	0.0279	0.0266	0.0387	0.2897	**5.7780e-4**	0.0045
	(10,10)	Mean	0.0361	0.0169	0.0120	0.0778	0.0118	**0.0095**
		Std	0.0020	0.0027	2.85e-4	0.0443	**2.70e-4**	0.0013
	(15,10)	Mean	0.0168	0.0169	0.0388	0.0158	0.0081	**0.0069**
		Std	5.6923e-4	5.7845e-4	0.0123	0.0014	**2.3540e-4**	7.9911e-4
DMOP3	(5,10)	Mean	0.1414	0.1474	0.1053	0.5473	0.1066	**0.0481**
		Std	0.0089	0.0169	0.0050	0.1049	0.0143	**0.0041**
	(10,10)	Mean	0.0937	0.0932	0.0579	0.1861	0.1196	**0.0137**
		Std	0.0015	0.0016	0.0024	0.0251	0.0050	**0.0013**
	(15,10)	Mean	0.0701	0.0698	0.0329	0.0911	0.1051	**0.0081**
		Std	0.0115	0.0102	0.0037	0.0091	0.0045	**5.4320e-4**

the optimal solutions, but the environment changes, so the history information used to predict new population is not accurate. So the predicted population may diverge the promising search region. SGEA introduces the external population, which improves the performance of algorithm in some extent. So for the problems that environment changes quickly, SGEA is better than MOEA/D-DE-SHPS.

However, for FDA2 and FDA4, when (τ_T, n_T) is set to (15,10), SGEA is the best and MOEA/D-DE-SHP is the second best, but the performance between the two algorithms is similar.

Meanwhile, MOEA/D-DE-SHPS is of poor stability, perhaps because SHPS predicts population using the collected information, so the degree of accurate of history information may influence the power of MOEA/D-DE-SHPS.

To observe the performance of six algorithms more intuitively, we also show the obtained *PF* of the six comparison algorithms when (τ_T, n_T) is set to (10, 10). In the following

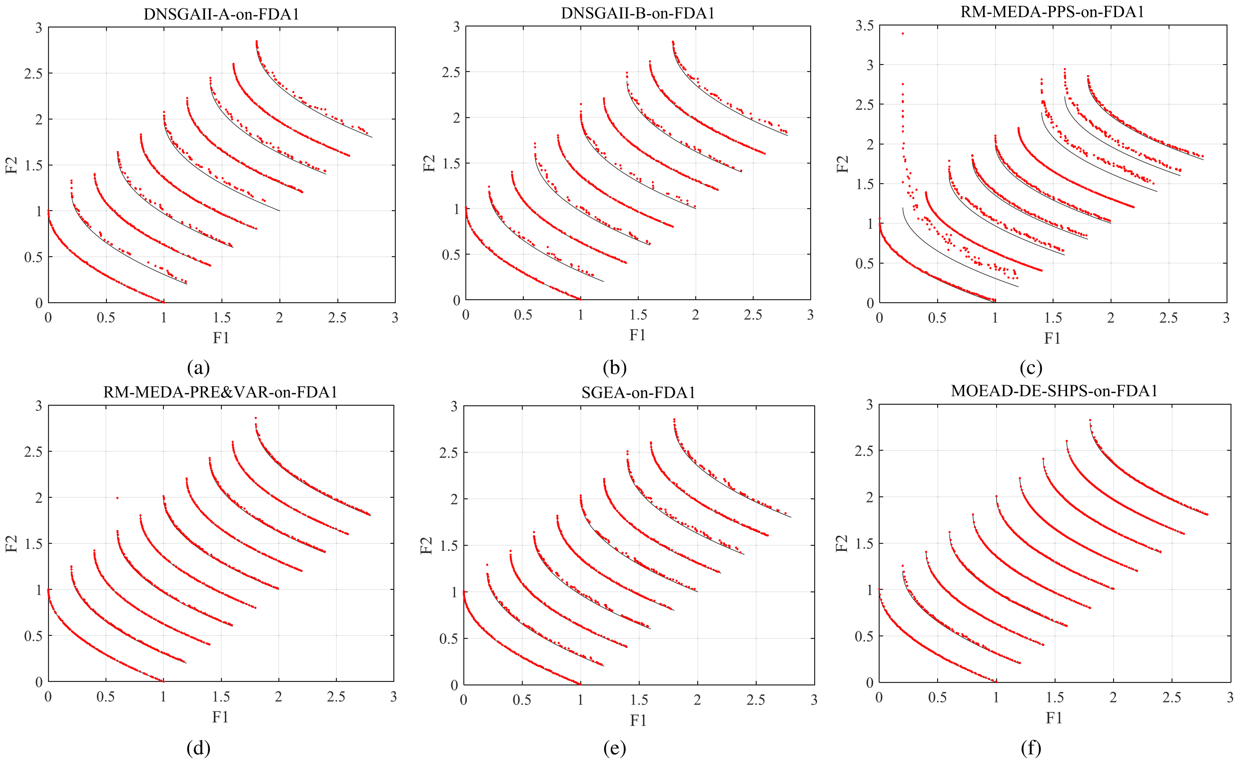

FIGURE 4. The *PF* of FDA1: (a) DNSGA-II; (b) DNSFA-II-B; (c) RM-MEDA-PPS; (d) RM-MEDA-PRE&VAR; (e) SGEA; (f) MOEA/D-DE-SHPS.

Figs. 4-10, the red represents the *PF* obtained by DMOOAs, and the black represents the true *PF*.

Fig. 4 shows the *PF* obtained by the six algorithms on FDA1 when $t = 10, 20, 30, 40, 50, 60, 70, 80, 90, 100$, and (τ_T, n_T) is set to (10,10). Due to the characteristic of FDA1 that *PS* changes with time and *PF* remains fixed, so we move the obtained *PF* and true *PF* simultaneously. We can find that the six algorithms except RMMEDA-PPS can converge to the true *PF*, meanwhile, MOEA/D-DE-SHPS get the best results, followed is RM-MEDA-PRE&VAR. Both MOEA/D-DE-SHPS and RM-MEDA-PRE&VAR have better distribution and convergence, and other four algorithms are poorly distributed.

Fig. 5 is the obtained *PF* of the six algorithms on FDA2 when $t = 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60$, and (τ_T, n_T) is set to (10,10). As shown in Fig. 5, the best algorithm is MOEA/D-DE-SHPS, it could converge to true *PF* and get better distribution. DNSGA-II-A, DNSGA-II-B and SGEA have the similar performance just behind MOEA/D-DE-SHPS. However the RM-MEDA-PPS and RM-MEDA-PRE&VAR are not competitive.

Fig. 6 shows the obtained *PF* of the six algorithms on FDA3 when $t = 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50$, and (τ_T, n_T) is set to (10,10). We can see that MOEA/D-DE-SHPS is of the best performance, SGEA and MOEA/D-SHPS both can converge to true *PF* and have better distribution. RM-MEDA-PRE&VAR can converge to true *PF*, however, there are some isolated points. DNSGA-II-A and DNSGA-II-B can converge to true *PF*, but the distribution of solutions is not good. RM-MEDA-PPS has the poorest performance.

Fig. 7 shows the obtained *PF* on FDA4 when $t = 69$, and (τ_T, n_T) is set to (10,10). MOEA/D-DE-SHPS performs best. DNSGA-II-A and DNSGA-II-B cannot converge to true *PF* at all. Other three algorithms including RMMEDA-PPS, RMMEDA-PRE&VAR and SGEA have the similar performance. The solutions gotten by RM-MEDA-PPS have the good distribution, however the solutions don't converge to true *PF* completely. We can see that solutions obtained by MOEA/D-DE-SHPS converge to true *PF* completely, meanwhile, the solutions have the best distribution. The results match the statistical results presented in Table 2.

Fig. 8 presents the *PF* obtained by the six algorithms on dMOP1 when $t = 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70$, and (τ_T, n_T) is set to (10,10). All the six algorithms performs well on dMOP1, and we can't get more information just according to Fig. 8.

Fig. 9 presents the *PF* got by the six algorithms on dMOP2 when $t = 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74$, and (τ_T, n_T) is set to (10,10). Obviously, only MOEA/D-DE-SHPS could converge to true *PF* and has better distribution. The performance of SGEA just behinds MOEA/D-DE-SHPS, the solutions gotten by SGEA approximately converge to the true *PF*. RM-MEDA-PPS and RM-MEDA-PRE&VAR just converge to the true *PF* at some environments, for other environments, the solutions don't converge. However,

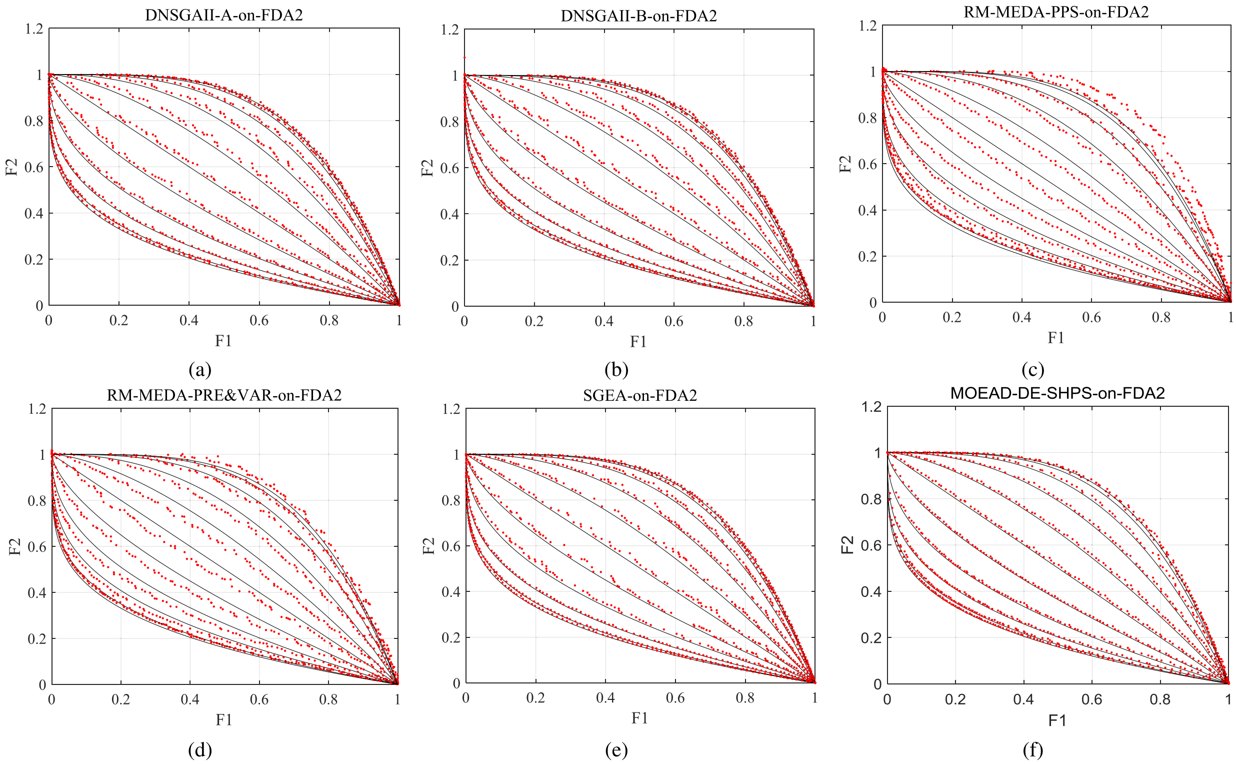

FIGURE 5. The *PF* of FDA2: (a) DNSGA-II; (b) DNSFA-II-B; (c) RM-MEDA-PPS; (d) RM-MEDA-PRE&VAR; (e) SGEA; (f) MOEA/D-DE-SHPS.

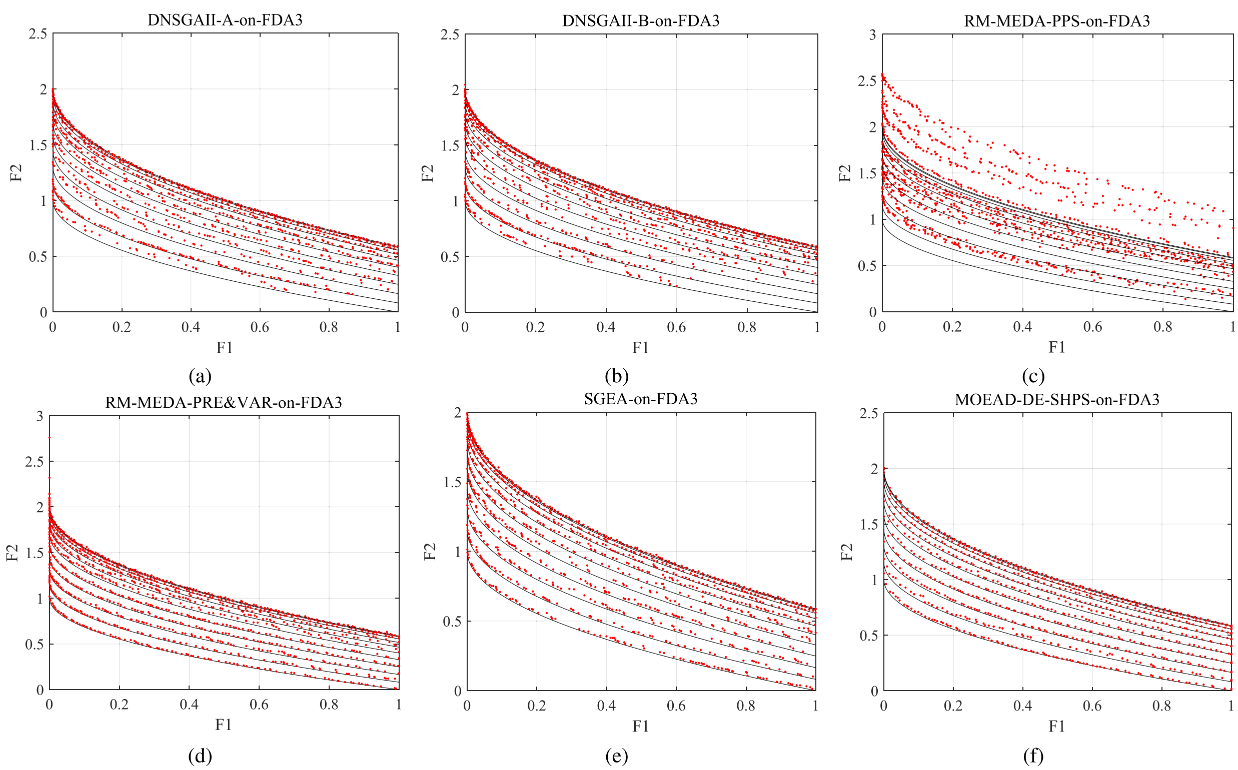

FIGURE 6. The *PF* of FDA3: (a) DNSGA-II; (b) DNSFA-II-B; (c) RM-MEDA-PPS; (d) RM-MEDA-PRE&VAR; (e) SGEA; (f) MOEA/D-DE-SHPS.

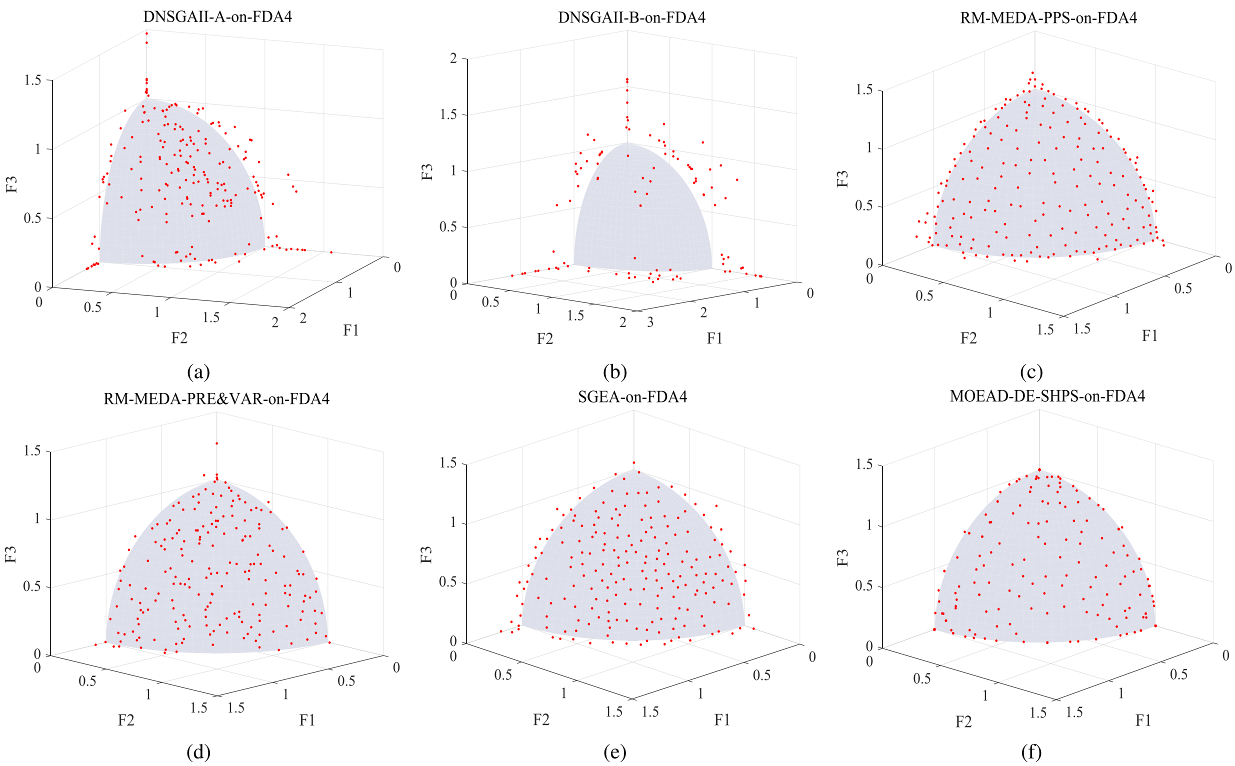

FIGURE 7. The *PF* of FDA4: (a) DNSGA-II; (b) DNSFA-II-B; (c) RM-MEDA-PPS; (d) RM-MEDA-PRE&VAR; (e) SGEA; (f) MOEA/D-DE-SHPS.

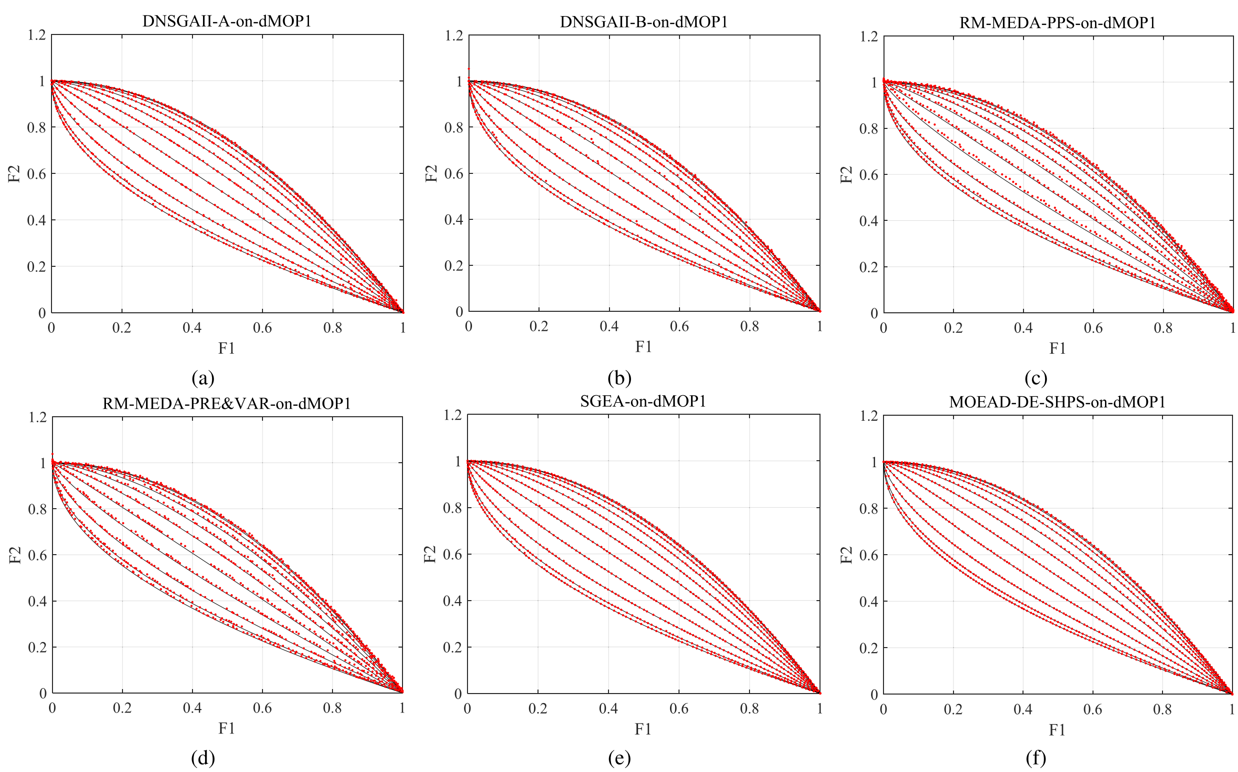

FIGURE 8. The *PF* of DMOP1: (a) DNSGA-II; (b) DNSFA-II-B; (c) RM-MEDA-PPS; (d) RM-MEDA-PRE&VAR; (e) SGEA; (f) MOEA/D-DE-SHPS.

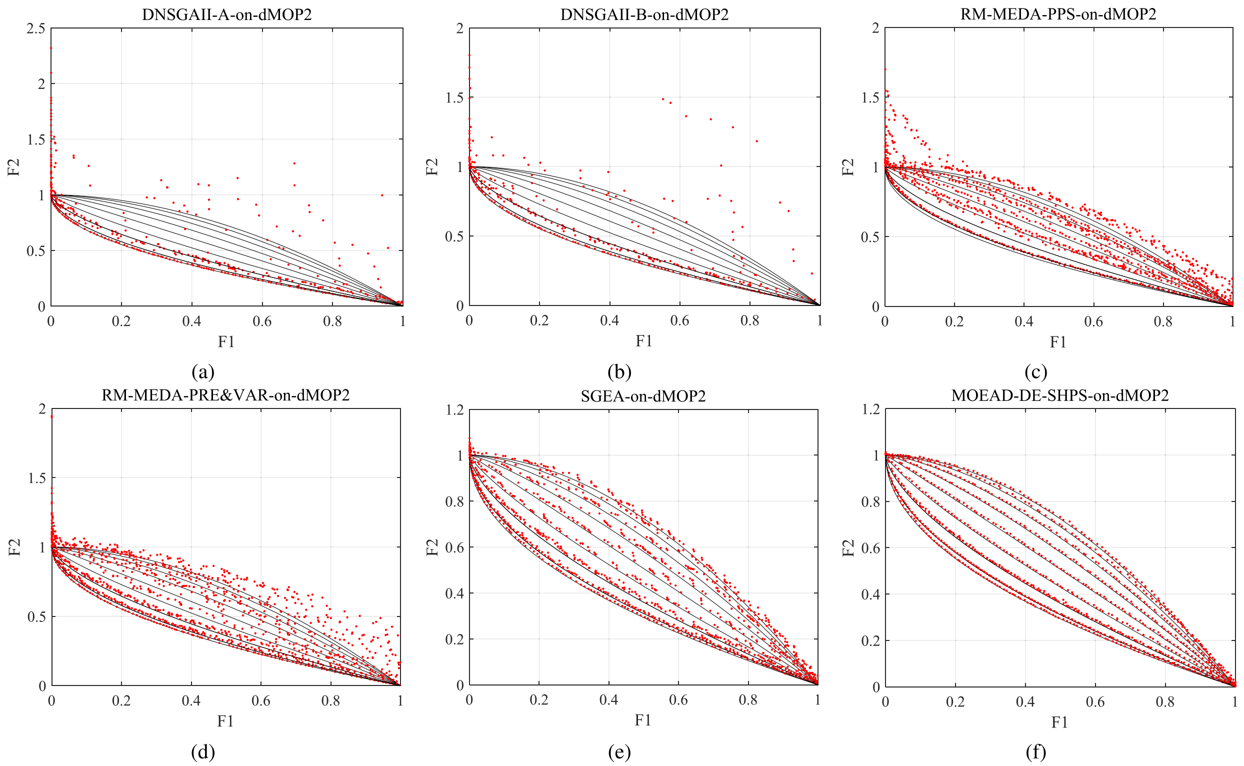

FIGURE 9. The *PF* of DMOP2: (a) DNSGA-II; (b) DNSFA-II-B; (c) RM-MEDA-PPS; (d) RM-MEDA-PRE&VAR; (e) SGEA; (f) MOEA/D-DE-SHPS.

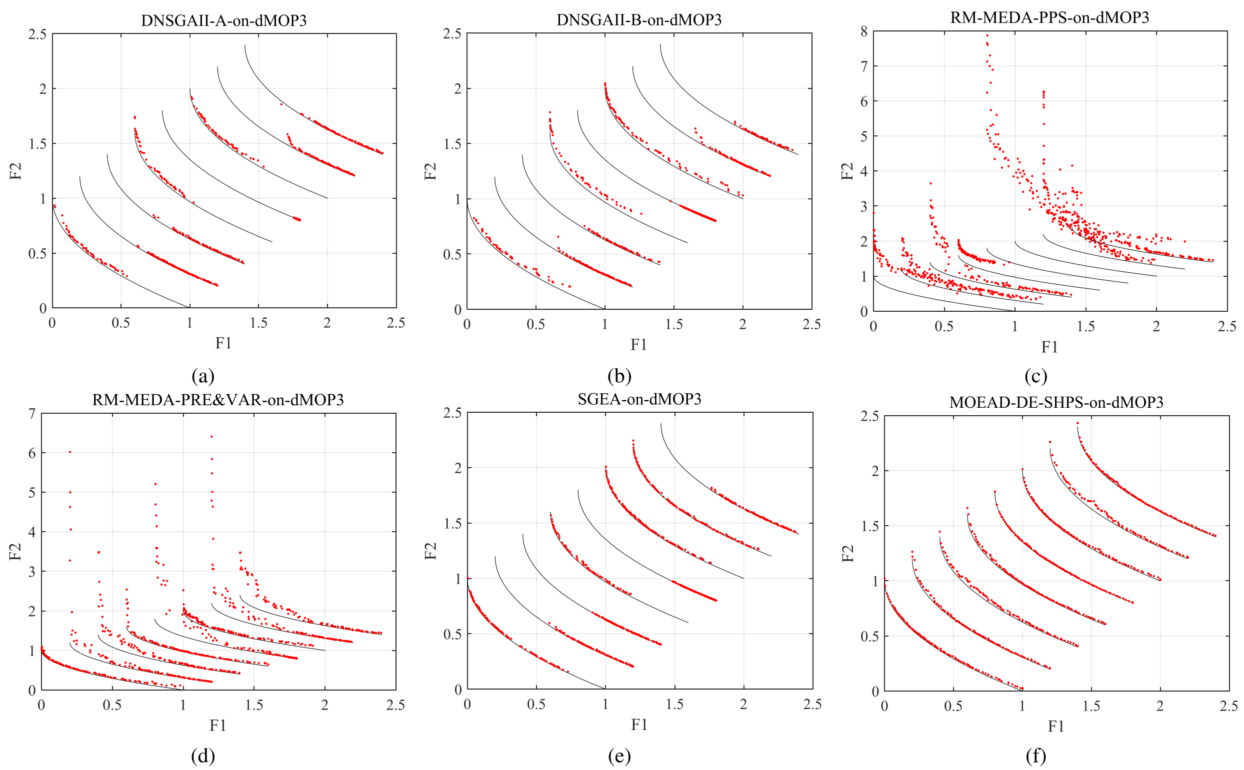

FIGURE 10. The *PF* of DMOP3: (a) DNSGA-II; (b) DNSFA-II-B; (c) RM-MEDA-PPS; (d) RM-MEDA-PRE&VAR; (e) SGEA; (f) MOEA/D-DE-SHPS.

TABLE 3. The statistical results of *MIGD* of all three algorithms.

test functions	(τ_t, n_t)	statistic	MOEA/D-DE-PPS	MOEA/D-DE-PRE&VAR	MOEA/D-DE-SHPS
FDA1	(5,10)	Mean	0.0383(3)	0.0215(2)	**0.0167(1)**
		Std	0.0040	0.0020	0.0019
	(10,10)	Mean	0.0110(2)	0.0146(3)	**0.0082(1)**
		Std	6.6823e-4	7.5417e-4	8.9158e-4
	(15,10)	Mean	0.0076(2)	0.0093(3)	**0.0063(1)**
		Std	4.3490e-4	3.7591e-4	4.1175e-4
FDA2	(5,10)	Mean	0.0623(3)	0.0259(2)	**0.0186(1)**
		Std	0.0312	0.0253	0.0030
	(10,10)	Mean	0.0166(3)	0.0103(2)	**0.0093(1)**
		Std	0.0015	0.0017	5.8960e-4
	(15,10)	Mean	0.0110(3)	0.0077(2)	**0.0076(1)**
		Std	0.0013	7.5584e-4	0.0012
FDA3	(5,10)	Mean	0.0503(3)	**0.0298(1)**	0.0302(2)
		Std	0.0040	0.0024	0.0022
	(10,10)	Mean	0.0268(3)	0.0254(2)	**0.0194(1)**
		Std	0.0018	0.0022	0.0012
	(15,10)	Mean	0.0208(3)	0.0204(2)	**0.0156(1)**
		Std	0.0020	0.0015	0.0014
FDA4	(5,10)	Mean	0.0740(3)	0.0626(2)	**0.0623(1)**
		Std	0.0018	4.3056e-4	5.8707e-4
	(10,10)	Mean	0.0557(2)	0.0564(3)	**0.0528(1)**
		Std	2.9375e-4	1.6480e-4	1.9084e-4
	(15,10)	Mean	0.0512(2)	0.0518(3)	**0.0499(1)**
		Std	1.8968e-4	1.7149e-4	9.9890e-5
FDA5	(5,10)	Mean	0.0804(3)	**0.0357(1)**	0.0362(2)
		Std	0.0111	6.5347e-4	3.9025e-4
	(10,10)	Mean	0.0476(3)	0.0314(2)	**0.0291(1)**
		Std	0.0085	2.0738e-4	1.9203e-4
	(15,10)	Mean	0.0457(3)	0.0280(2)	**0.0272(1)**
		Std	0.0101	1.6093e-4	1.8989e-4
DMOP1	(5,10)	Mean	0.0318(3)	0.0244(2)	**0.0241(1)**
		Std	0.0116	0.0120	0.0086
	(10,10)	Mean	0.0090(3)	0.0088(2)	**0.0087(1)**
		Std	0.0023	0.0022	0.0021
	(15,10)	Mean	0.0058(2)	0.0059(3)	**0.0057(1)**
		Std	0.0011	0.0011	0.0014
DMOP2	(5,10)	Mean	0.0455(3)	0.0275(2)	**0.0221(1)**
		Std	0.0046	0.0060	0.0045
	(10,10)	Mean	0.0129(2)	0.0177(3)	**0.0095(1)**
		Std	9.6868e-4	0.0020	0.0013
	(15,10)	Mean	0.0083(2)	0.0107(3)	**0.0069(1)**
		Std	0.0010	6.7880e-4	7.9911e-4
DMOP3	(5,10)	Mean	0.1289(3)	0.0917(2)	**0.0481(1)**
		Std	0.1062	0.0082	0.0041
	(10,10)	Mean	0.0273(3)	0.0229(2)	**0.0137(1)**
		Std	0.0016	0.0027	0.0013
	(15,10)	Mean	0.0115(3)	0.0109(2)	**0.0081(1)**
		Std	6.6023e-4	7.7750e-4	5.4320e-4

DNSGA-II-A and DNSGA-II-B performs the worst, and can hardly converge.

Fig. 10 presents the *PF* obtained by the six algorithms on dMOP3 when $t = 30, 40, 50, 60, 70, 80, 90, 100$, and (τ_T, n_T) is set to (10,10). Due to the characteristic of dMOP3 that *PS* changes with time and *PF* remains fixed, so we move the obtained *PF* and true *PF* simultaneously. From Fig. 10, we can see that MOEA/D-DE-SHPS has the absolute predominance. Among the six algorithms, just MOEA/D-DE-SHPS can converge to the true *PF*. There are no comparability for other algorithms.

D. COMPARISON BETWEEN SHPS AND THE OTHER TWO PREDICTION STRATEGIES

This section compares SHPS with other two prediction strategies including PRE&VAR and PPS to prove the performance of SHPS. PPS and PRE&VAR are also integrated into MOEA/D-DE, so all the three algorithms are referred to as MOEA/D-DE-PPS, MOEA/D-DE-PRE&VAR and MOEA/D-DE-SHPS. The parameter settings for all algorithms are the same as those in Table 1 shown in Section IV-C.1.

Table 3 gives the statistical results of *MIGD* over 20 runs for different algorithms, in which, the black represents the best result of all the three comparison algorithms.

MOEA/D-DE-SHPS significantly outperforms MOEA/D-DE-PPS on all test functions when (τ_T, n_T) is set to (5,10), (10,10) and (15,10). Meanwhile, MOEA/D-DE-SHPS performs better than MOEA/D-DE-PRE&VAR on majority of FDA and dMOP functions, except on FDA3 and FDA5 when (τ_T, n_T) is set to (5,10). The reason may be that, for FDA3 and FDA5, when τ_T is set to 5, environment changes quickly, static algorithm can't find the optimal solutions of current environment, which will lead the history information collected from the previous environment is inaccurate, and will lead the predicted population to be located in the wrong search region.

In a word, the proposed SHPS has a better performance than PRE&VAR and PPS.

V. CONCLUSION

In this paper, we propose a special points-based hybrid prediction strategy (SHPS) which is integrated into the multi-objective optimization algorithm based on decomposition with differential evolution (MOEA/D-DE) to handle DMOPs. In SHPS, when historical information is not abundant to conduct the prediction model (AR model), PRE&VAR method is adopted to generate the initial population of new environment. Meanwhile, owing to PPS has a strong dependence of historical information gathered from previous environments, so the inaccurate historical information may lead the predicted population to be located in the wrong search region. Therefore, this paper introduces the special points-based strategy, in which, the initial population of the new environment consists of two parts of individuals: the predicted special points and the predicted population by PPS (except the special points). The focuses of the special points could avoid the prediction of the population completely depends on the historical information of the central point and manifold, which could reduce the influence of inaccurate prediction of PPS.

Two empirical studies are conducted to verify the effectiveness of MOEA/D-DE-SHPS. Firstly, MOEA/D-DE-SHPS is compared with five well-known DMOOAs, which are NSGA-II-A, NSGA-II-B, RM-MEDA-PPS, RM-MEDA-LPS and SGEA. Empirical results show that MOEA/D-DE-SHPS outperforms comparison algorithms on most test functions except dMOP1 and dMOP2. Secondly, a comparison between SHPS and other two prediction strategies including PRE&VAR and PPS indicates that SHPS outperforms PRE&VAR and PPS on majority test functions.

Although MOEA/D-DE-SHPS is promising for dealing with DMOPs. But there are still some problems needed to be solved. When the historical information is not enough to establish the AR model, a more effective response strategy may be designed to respond to the environmental changes and to promote algorithm converge to true *PF* as far as possible. Because the accuracy of the collected historical information extremely influence the performance of MOEA/D-DE-SHPS.

REFERENCES

[1] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proc. 1st Int. Conf. Genetic Algorithms*, 1985, pp. 93–100.

[2] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[3] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[4] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.

[5] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 256–279, Jun. 2004.

[6] A. Zhou, Q. Zhang, Y. Jin, B. Sendhoff, and E. Tsang, "Global multiobjective optimization via estimation of distribution algorithm with biased initialization and crossover," in *Proc. 9th Annu. Conf. Genetic Evol. Comput. (GECCO)*, 2007, pp. 617–623.

[7] Q. Zhang, A. Zhou, and Y. Jin, "RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm," *IEEE Trans. Evol. Comput.*, vol. 12, no. 1, pp. 41–63, Feb. 2008.

[8] A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, and E. Tsang, "Prediction-based population re-initialization for evolutionary dynamic multi-objective optimization," in *Proc. Int. Conf. Evol. Multi-Criterion Optim.* Berlin, Germany: Springer, 2007, pp. 832–846.

[9] C.-K. Goh and K. C. Tan, "A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 1, pp. 103–127, Feb. 2009.

[10] A. Zhou, Y. Jin, and Q. Zhang, "A population prediction strategy for evolutionary dynamic multiobjective optimization," *IEEE Trans. Cybern.*, vol. 44, no. 1, pp. 40–53, Jan. 2014.

[11] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization by NSGA-II and MOEA/D with large populations," in *Proc. IEEE Int. Conf. Syst., Man Cybern. (SMC)*, Oct. 2009, pp. 1758–1763.

[12] Y. Liu and B. Niu, "A multi-objective particle swarm optimization based on decomposition," in *Proc. Int. Conf. Intel. Comput.* Springer, 2013, pp. 200–205.

[13] C. C. Pei, S. H. Chen, Q. Zhang, and J. L. Lin, "MOEA/D for flowshop scheduling problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2008, pp. 1433–1438.

[14] T. J. Yuen and R. Ramli, "Comparision of compuational efficiency of MOEA\D and NSGA-II for passive vehicle suspension optimization," in *Proc. Eur. Conf. Modelling Simulation (ECMS)*, 2010, pp. 219–225.

[15] M. Farina, K. Deb, and P. Amato, "Dynamic multiobjective optimization problems: Test cases, approximations, and applications," *IEEE Trans. Evol. Comput.*, vol. 8, no. 5, pp. 425–442, Oct. 2004.

[16] K. Yamasaki, "Dynamic Pareto optimum GA against the changing environments," in *Evolut. Algorithm. Dyn. Optim. Probl.*, 2001, pp. 47–50.

[17] K. Deb, U. B. Rao, and S. Karthik, "Dynamic multi-objective optimization and decision-making using modified NSGA-II: A case study on hydro-thermal power scheduling," in *Proc. Int. Conf. Evol. Multi-Criterion Optim.* Berlin, Germany: Springer, 2007, pp. 803–817.

[18] Y. Wu, Y. Jin, and X. Liu, "A directed search strategy for evolutionary dynamic multiobjective optimization," *Soft Comput.*, vol. 19, no. 11, pp. 3221–3235, 2015.

[19] S. Jiang and S. Yang, "A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 1, pp. 65–82, Feb. 2017.

[20] Z. Zhang, "Multiobjective optimization immune algorithm in dynamic environments and its application to greenhouse control," *Appl. Soft Comput.*, vol. 8, no. 2, pp. 959–971, 2008.
[21] X. Li, J. Branke, and M. Kirley, "On performance metrics and particle swarm methods for dynamic multiobjective optimization problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Sep. 2007, pp. 576–583.
[22] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments—A survey," *IEEE Trans. Evol. Comput.*, vol. 9, no. 3, pp. 303–317, Jun. 2005.
[23] F. Vavak, "Adaptive combustion balancing in multiple burner boiler using a genetic algorithm with variable range of local search," in *Proc. 7th Int. Conf. Genetic Algorithms*, 1997, pp. 719–726.
[24] H. G. Cobb, "An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, time-dependent nonstationary environments," Naval Res. Lab, Washington, DC, USA, Tech. Rep. 6760, 1990.
[25] S.-Y. Zeng *et al.*, "A dynamic multi-objective evolutionary algorithm based on an orthogonal design," in *Proc. IEEE Int. Conf. Evol. Comput. (CEC)*, Jul. 2006, pp. 573–580.
[26] Z. Zhang and S. Qian, "Artificial immune system in dynamic environments solving time-varying non-linear constrained multi-objective problems," *Soft Comput.*, vol. 15, no. 7, pp. 1333–1349, 2011.
[27] I. Hatzakis and D. Wallace, "Dynamic multi-objective optimization with evolutionary algorithms: A forward-looking approach," in *Proc. 8th Annu. Conf. Genetic Evol. Comput. (GECCO)*, 2006, pp. 1201–1208.
[28] M. Liu, J. Zheng, J. Wang, Y. Liu, and L. Jiang, "An adaptive diversity introduction method for dynamic evolutionary multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2014, pp. 3160–3167.
[29] I. Hatzakis and D. Wallace, "Topology of anticipatory populations for evolutionary dynamic multi-objective optimization," in *Proc. 11th AIAA/ISSMO Multidisciplinary Anal. Optim. Conf.*, 2006, p. 7071.
[30] X. Zhang, Y. Tian, and Y. Jin, "A knee point-driven evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 761–776, Dec. 2015.
[31] I. Das, "On characterizing the 'knee' of the Pareto curve based on normal-boundary intersection," *Struct. Optim.*, vol. 18, nos. 2–3, pp. 107–115, 1999.
[32] J. Branke, K. Deb, H. Dierolf, and M. Osswald, "Finding knees in multi-objective optimization," in *Proc. Int. Conf. Parallel Problem Solving Nature*, 2004, pp. 722–731.
[33] K. Deb and S. Gupta, "Understanding knee points in bicriteria problems and their implications as preferred solution principles," *Eng. Optim.*, vol. 43, no. 11, pp. 1175–1204, 2011.
[34] H. Richter, "Detecting change in dynamic fitness landscapes," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, May 2009, pp. 1613–1620.
[35] R. Liu, J. Li, F. Jing, C. Mu, and L. Jiao, "A coevolutionary technique based on multi-swarm particle swarm optimization for dynamic multi-objective optimization," *Eur. J. Oper. Res.*, vol. 261, no. 3, pp. 1028–1051, 2017.