



A multipopulation evolutionary framework with Steffensen's method for dynamic multiobjective optimization problems

Tianyu Liu¹ · Lei Cao¹ · Zhu Wang²

Received: 11 August 2020 / Accepted: 20 October 2021 / Published online: 13 November 2021
© The Author(s) 2021

Abstract

Dynamic multiobjective optimization problems (DMOPs) require the evolutionary algorithms that can track the moving Pareto-optimal fronts efficiently. This paper presents a dynamic multiobjective evolutionary framework (DMOEF-MS), which adopts a novel multipopulation structure and Steffensen's method to solve DMOPs. In DMOEF-MS, only one population deals with the original DMOP, while the others focus on single-objective problems that are generated by the weighted summation of the original DMOP. Then, Steffensen's method is used to control the evolving process in two ways: prediction and diversity-maintenance. Particularly, the prediction strategy is devised to predict the next promising positions for the individuals that handle single-objective problems, and the diversity-maintenance strategy is used to increase population diversity before the environment changes and reinitialize the multiple populations after the environment changes. This paper gives a comprehensive comparison of DMOEF-MS with some state-of-the-art DMOEAs on 14 DMOPs and the experimental results demonstrate the effectiveness of the proposed algorithm.

Keywords Dynamic multiobjective optimization · Steffensen's method · Multipopulation structure · Prediction strategy · Diversity-maintenance strategy

Mathematics Subject Classification 68T01 · 68T20

1 Introduction

With multiple conflicting objectives, multiobjective optimization problems (MOPs) [1] have been successfully solved by various evolutionary algorithms (EAs), such as NSGA-II [2], SPEA2 [3], MOPSO [4], MOEA/D [5], ACO [6], and so forth. Since lots of real-world MOPs need to be optimized in dynamic environments [7–11], how to extend multiobjective evolutionary algorithms (MOEAs) to solve dynamic multiobjective optimization problems (DMOPs) has attracted more

and more attention. For static MOEAs, the goal is to find accurate and well-distributed Pareto-optimal fronts (PFs). However, with time-varying optimization environments, a dynamic multiobjective evolutionary algorithm (DMOEA) is expected to find the ideal PF at the current environment and locate the new PF efficiently after the environment changes. Therefore, a promising DMOEA should take into account the two following issues:

- Convergence speed. In a dynamic optimization environment, the PF may change over time. Therefore, a DMOEA is required to converge rapidly before the environment changes.
- Population diversity. A DMOEA should be capable of locating the new PF after the environment changes. A poor population diversity may hinder a DMOEA from tracking the moving PF, especially after the environment changes.

Most of the existing DMOEAs are derived from MOEAs, which are originally designed to solve MOPs in static

✉ Tianyu Liu
liuty@shmtu.edu.cn

Lei Cao
lcao@shmtu.edu.cn

Zhu Wang
zhuwang@shmtu.edu.cn

¹ School of Information Engineering, Shanghai Maritime University, Shanghai, China

² School of Logistics Engineering, Shanghai Maritime University, Shanghai, China

environments. In recent years, plenty of dynamic handling strategies have been inserted into the classic MOEAs to solve DMOPs. According to their inherent behaviors, the existing techniques can be classified into different categories, such as diversity introduction/maintenance strategies [12–15], memory-based strategies [16–20], Multipopulation-based strategies [21–25], and prediction-based strategies [18,26–29]. Among all of these techniques, multipopulation-based and prediction-based strategies have received great concern. Particularly, multipopulation-based strategies carry out parallel exploration in searching space to maintain population diversity, prediction-based strategies are used to accelerate the convergence speed by predicting the next most likely positions of individuals.

In this paper, a multipopulation framework which cooperates with Steffensen's method [30], namely DMOEF-MS, is introduced to solve DMOPs. In many existing multipopulation-based methods, the multiple populations cooperate to explore the searching space and focus on the same DMOP. In this case, each population evolves with less diversity-related guiding information obtained from its optimization problem. This paper proposes a multipopulation framework, in which one population deals with the original DMOP and the others handle the single-objective problems that are generated by the weighted summation method. The single-objective problems, which are different from each other, are presented in forms of nonlinear functions. Therefore, Steffensen's method, which is a classical and efficient algorithm for solving nonlinear equations, is inserted into the proposed multipopulation framework to improve the performance of DMOEF-MS. The detailed description of Steffensen's method is presented in Sect. 2.3. Steffensen's method is adopted to control the evolving process of DMOEF-MS in two ways: prediction and diversity-maintenance strategies. Particularly, the prediction strategy is devised to predict the new location of the individuals that handle single-objective problems, and the diversity-maintenance strategy is used to increase the population diversity at fixed intervals of generations and reinitialize the multiple populations after the environment changes. DMOEF-MS was compared with some state-of-the-art DMOEAs on 14 benchmark DMOPs, and the experimental results demonstrate the effectiveness of the proposed algorithm. The main contributions of this paper are listed as follows.

- (1) A multipopulation framework is proposed. This framework contains $M + 1$ populations, and M is the number of objectives in a DMOP. Each of the first M populations handles a single-objective problem, which is obtained as the weighted sum of the original M objective functions. The last population (the $(M + 1)$ th population) optimizes the original DMOP. The interaction of populations is achieved by a common repository population

(REP), which stored the Pareto-optimal solutions found by all the $M + 1$ populations

- (2) A prediction strategy which is inspired by Steffensen's method is introduced to update the individuals in the first M populations.
- (3) A diversity-maintenance strategy is used to increase the population diversity at fixed intervals of generations before the environment changes and reinitialize the multiple populations after the environment changes.

The rest of this paper is organized as follows. Section 2 introduces the background knowledge of DMOPs and the basic knowledge of Steffensen's method. In Sect. 3, the detailed description of the proposed DMOEF-MS is given. Section 4 presents the comparative experiments and the obtained results. Section 5 is the concluding remarks.

2 Background knowledge

A MOP can be stated as follows:

$$\min_x F(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})] \quad (1)$$

where $\mathbf{x} = [x_1, x_2, \dots, x_N] \in \Omega$ and Ω is the searching space, $F : \Omega \rightarrow R^M$ and R^M is the objective space. Since a MOP consists of M conflicting objective functions, one cannot find a solution that can optimize all the M objective functions simultaneously. Suppose $a, b \in R^M$, a is said *dominate* b if and only if $a_i \leq b_i$ for all $i = 1, 2, \dots, M$ and $a \neq b$. $\mathbf{x}^* \in \Omega$ is called a *Pareto-optimal solution* if there is no $\mathbf{x} \in \Omega$ such that $F(\mathbf{x})$ dominate $F(\mathbf{x}^*)$. For Multi-objective evolutionary algorithms, the purpose is finding a *Pareto-optimal set* (PS), which contains a number of Pareto-optimal solutions. The set of all the objective vectors of PS is the *Pareto-optimal front* (PF).

The continuous DMOPs [27] adopted in this paper are defined as follows:

$$\min_x F(\mathbf{x}, t) = [f_1(\mathbf{x}, t), f_2(\mathbf{x}, t), \dots, f_M(\mathbf{x}, t)] \quad (2)$$

where $\mathbf{x} = [x_1, x_2, \dots, x_N]$ is the candidate solution, N is the dimension of the searching space. M is the number of objective functions. t is the index of the current environment, and $F(\mathbf{x}, t)$ is the objective vector of \mathbf{x} in the t th environment.

For DMOPs, the challenge is designing efficient dynamic handling strategies, which are used to make a balance between population diversity and convergence speed throughout the whole dynamic optimization process. Besides, DMOEAs should also be able to detect environmental changes if they are not assumed to be knowable. The following sections present some representative studies of the

change detection and dynamic handling strategies as well as the basic knowledge of Steffensen's method.

2.1 Change detection

There are two widely used methods for change detection.

2.1.1 Reevaluation of dedicated detectors

If the reevaluated objective value is different from the original one, then the environment is considered to be changed [12,31]. As an easy-to-implement method, it is also a robust detection if enough detectors are adopted. However, this detection method requires additional function evaluations and may generate inaccurate detection results for a noisy optimization function.

2.1.2 Assessment of algorithms' behaviors

In this method, the environmental change is detected if there is a discrepancy between an algorithm's behaviors and the statistical information obtained from its evolving population [32]. This method does not require additional function evaluations. However, it may need problem-related parameters and lead to an overreaction when no environmental change occurs.

2.2 Dynamic handling strategies

Many efficient dynamic handling strategies have been proposed in recent years. Most existing techniques can be classified into the following four categories: diversity introduction/maintenance strategies, memory-based strategies, multipopulation-based strategies, and prediction-based strategies.

2.2.1 Diversity introduction/maintenance strategies

The following techniques are widely used to introduce or maintain population diversity. Deb et al. [12] proposed a DNGA-II, which replaced a portion of the population individuals with mutated or randomly generated ones. This technique can be easily implemented, but the introduction of population diversity may at the cost of the loss of useful information. To solve this problem, several problem-related immigration strategies, which include hybrid immigration [33], memory-based immigration [15], and elitism-based immigration [13], are presented. Besides, local searching is also an efficient way to introduce population diversity. Vavak et al. [34] proposed a variable local search strategy (VLS) to increase the population diversity. In VLS, the mutation rate is increased gradually as the algorithm runs. Ruan et al. [14] devised a diversity maintenance strategy (DMS) to increase

the population diversity by generating new individuals in the estimated regions, which were determined according to the previous information of the decision vectors.

2.2.2 Memory-based strategies

For memory-based strategies, the basic principle is reusing past information to improve the performance of a DMOEA in the new environment after changes and thus memory-based strategies are suitable for DMOPs with periodically changing environments. Branke [16] proposed a memory scheme, which adopted a memory archive to store the best individuals in the population. After the environment changes, the individuals in the memory archive will be reused to initialize the population. Goh and Tan [17] devised a dynamic competitive-cooperation co-evolutionary algorithm (dCOEA), in which the memory archive was modified by replacing the outdated solutions. Wang and Li [20] proposed two memory schemes to utilize the individuals stored in the memory archive. In the first scheme, the stored individuals are randomly selected as the members of the initial population after the environment changes. While in the second scheme, the individuals are modified before used. Recently, memory-based strategies have usually cooperated with other strategies to improve the performance of DMOEAs. For instance, Peng et al. [19] presented a novel prediction and memory strategy (PMS). In PMS, the stored individuals can be used more efficiently since they are reevaluated before used. Liang et al. [18] proposed a hybrid of memory and prediction strategies (HMPS), which devised a memory-based technique to predict the new locations of the individuals.

2.2.3 Multipopulation-based strategies

The basic idea of multipopulation-based strategies is to conduct simultaneous exploration in searching space by multiple populations. With the strong parallel processing ability, multipopulation-based strategies have demonstrated its effectiveness for DMOPs with multiple peaks. Branke et al. [35] proposed a multipopulation-based algorithm to solve DMOPs. This algorithm uses several smaller populations to track the most promising peaks over time, while a larger population is continuously searching for new peaks. Goh and Tan [17] proposed a competitive and cooperative mechanism to interact between multiple subpopulations for handling DMOPs. Particularly, each subpopulation will compete to represent a particular subcomponent of the original MOP, and the winners will cooperate to generate the eventual solutions. Shang et al. [21] presented a quantum immune clonal co-evolutionary algorithm (QICCA) to solve DMOPs. Having multiple populations, QICCA uses U-measure to control the competition between populations and a new cooperative operation to get better solutions. Liu et al.

[22] proposed a modified coevolutionary multi-swarm particle swarm optimizer (CMPSODMO) to solve DMOPs with rapidly changing environments. In CMPSODMO, the number of swarms is determined by the number of the objective functions, and an information-sharing strategy is adopted to realize the interaction among all swarms. Gong et al. [23] introduced a framework of dynamic interval multi-objective cooperative co-evolutionary optimization. In this framework, the decision variables are divided into two groups according to the interval similarity, and two populations are generated to optimize the above-mentioned two variable groups cooperatively. In a parallel DMOEA [24], the decision variables are divided into several groups according to Spearman rank correlation analysis. Then multiple subpopulations, which are utilized to optimize the variable groups in parallel, cooperate to solve DMOPs with changing variables. Xu et al. [25] proposed a cooperative co-evolutionary strategy for solving DMOPs. In this strategy, the decision variables are divided into two subcomponents according to whether or not they interrelate with the environment. Then two subpopulations, which optimize two subcomponents respectively, are adopted to explore the searching space. It can be seen from the description mentioned above that how to construct multiple populations and interact among the populations has a crucial influence on the performance of the multipopulation-based strategies.

2.2.4 Prediction-based strategies

The basic idea of the prediction-based strategies is reusing as much information from the past environment to speed up the searching process in the new environment. Therefore, prediction-based strategies always cooperate with memory-based strategies and are widely adopted to solve DMOPs with periodic and trending changes. Hatzakis and Wallace [26] proposed a forward-looking approach to solve DMOPs. The proposed approach uses an autoregressive (AR) model to predict the new positions of the individuals according to a sequence of optimum solutions generated from the previous environments. Zhou et al. [27] presented a population prediction strategy (PPS), which divided the Pareto-optimal Set (PS) into two parts: a center point and a manifold. In particular, the next center point is predicted by a univariate AR model according to a series of stored center points, the next manifold is estimated according to the manifolds generated in the past two environments. Wu et al. [28] introduced a directed search strategy (DSS) to accelerate the convergence speed. In DSS, the new individuals are predicted by using the moving directions generated from the PS in the previous two generations. Muruganantham et al. [36] proposed a new DMOEA, which used the Kalman filter (KF) to estimate the next locations of the solutions. Jiang and Yang [37] presented a steady-state and generational evolu-

tionary algorithm (SGEA), which relocate the Pareto-optimal solutions according to the information collected from the previous and current environments. Jiang et al. [38] proposed a Tr-DMOEA, which combined population-based EAs with transfer learning to solve DMOEAs. Li et al. [39] introduced a special points-based hybrid prediction strategy (SHPS) for solving DMOPs. In SHPS, the initial population in a new environment consists of two parts: the predicted special points and the individuals predicted by PPS. Rong et al. [40] adopted multiple prediction models to relocate the Pareto-optimal solutions once the environment changes. Liang et al. [18] proposed an MOEA/D-HMPS, which adopted two prediction strategies to relocate the Pareto-optimal solutions according to whether the current environmental change is similar to the historical changes. Rong et al. [29] proposed a multi-model prediction (MMP) method to tackle continuous DMOPs with more than one type of the unknown PS changes. MMP method utilizes four prediction models to handle different types of PS changes.

2.3 Steffensen's method

As a classical problem in scientific computation, how to find the root of a nonlinear equation $f(a) = 0$ has been widely concerned over the years. Among the variety of root searching methods, Newton's method [41] is probably the most famous one. The iterative process of Newton's method can be implemented according to Eq. (3), with $a(0)$ as the initial estimate of the root. As shown in Eq. (3), $a(g)$ is the current root, $a(g + 1)$ is the root after one iteration, $f'(a(g))$ is the derivation value of $a(g)$. However, the fact that it is difficult to obtain the derivation of a nonlinear function sometimes limits the application of Newton's method.

$$a(g + 1) = a(g) - \frac{f(a(g))}{f'(a(g))}, \quad g = 0, 1, 2, \dots \quad (3)$$

As a notable improvement of Newton's method, Steffensen's method [30] is proposed. It can be seen from Eq. (4), the derivative $f'(a(g))$ is replaced by $\frac{f(a(g)+f(a(g)))-f(a(g))}{f(a(g))}$. In other words, Steffensen's method is derivation free. The theoretical analysis given by Conte and Boor [42] demonstrates that both Newton's method and Steffensen's method are quadratic convergence.

$$a(g + 1) = a(g) - \frac{f(a(g))^2}{f(a(g) + f(a(g))) - f(a(g))}, \quad g = 0, 1, 2, \dots \quad (4)$$

3 Dynamic multipopulation evolutionary framework with Steffensen's method (DMOEF-MS)

This paper proposes a dynamic multipopulation evolutionary framework (DMOEF-MS), which integrates Steffensen's method into a novel multipopulation framework to solve DMOPs. In this section, the motivation and framework of the proposed algorithm are given, followed by a detailed description of the key components in DMOEF-MS.

3.1 Motivations of DMOEF-MS

As described in Sect. 2.2.3, the basic idea of multipopulation-based strategies is that multiple populations cooperate to explore searching spaces and compete to generate promising solutions. With the strong parallel processing ability, multipopulation-based strategies have demonstrated its effectiveness for DMOPs with multiple peaks. However, in many existing multipopulation-based algorithms, the multiple populations focus on the same optimization problem. In this case, two different populations may generate Pareto-optimal solutions that focus on the same or close areas in PFs, which may cause a waste of computations. In the multipopulation structure of DMOEF-MS, multiple populations handle different optimization problems, respectively. Therefore, the adopted multipopulation structure can improve the ability of DMOEF-MS to maintain population diversity and obtain well-distributed PFs, since the multiple populations focus on different areas of PFs by handling different optimization problems.

For DMOEAs with prediction-based strategies, the basic idea is to reuse as much information from the past environment to speed up the searching process in the new environment. Thus prediction-based strategies always cooperate with memory-based strategies and are widely adopted to solve DMOPs with periodic and trending changes. However, the selection and utilization of memory information from past environments have a crucial influence on the performance of DMOEAs. The improper use of past information may directly weaken the performance of a DMOEA with prediction-based strategies and memory strategies. In DMOEF-MS, Steffensen's prediction strategy is used to predict the next likely positions of the individuals that handle single-objective problems. The proposed prediction strategy is implemented according to the populations in the current environment. On the one hand, Steffensen's prediction strategy can easily find more accurate solutions with the help of Steffensen's method, which is a classical and efficient root-searching method for solving nonlinear equations. On the other hand, without reusing the information from the past environment, DMOEF-MS does not need to consider the selection and utilization of the past information.

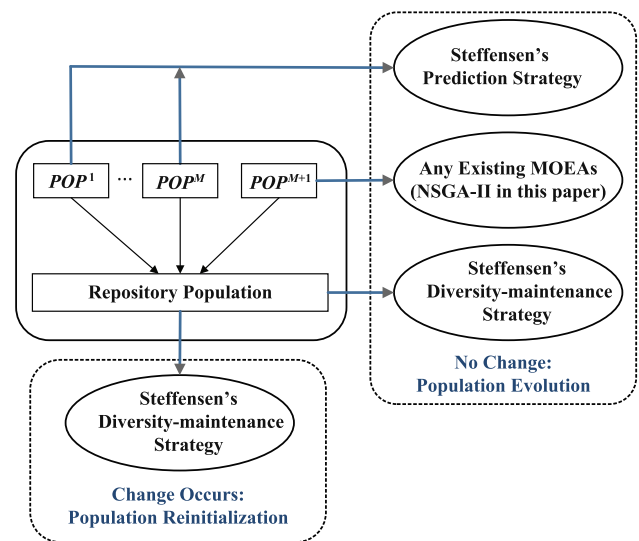


Fig. 1 Theoretical block diagram of DMOEF-MS

Moreover, Steffensen's diversity-maintenance strategy is proposed to increase the population diversity during the whole run in DMOEF-MS. In this strategy, a closeness metric is introduced to measure the closeness between a solution and the weight vectors, which are evenly distributed in the objective space. With the guidance of the closeness metric, Steffensen's diversity-maintenance strategy try to reduce the waste of computations while increasing the population diversity in DMOEF-MS.

The detailed description of the framework and key components of DMOEF-MS are given in Sect. 3.2.

3.2 Overall framework of DMOEF-MS

The overall framework of DMOEF-MS is illustrated in Fig. 1, and the detailed procedure of the proposed algorithm can be found in Algorithm 1. M is the dimension of the objective space, t is the index of the current environment, g is the number of generations. DMOEF-MS starts with initial multiple populations and a repository population (REP) which stores the Pareto-optimal solutions found so far. Among the multiple populations, each of the first M populations handles a single-objective problem that is generated by the weighted summation method. The last population (the $(M + 1)$ th population) optimizes the original DMOP. For a DMOP, if the environmental change is not detected, then the next positions of the individuals in the first M populations can be updated by Steffensen's prediction strategy (line 9). Since (POP^{M+1}) deals with the original DMOP, which can be regarded as a static MOP when no environmental change occurs, thus POP^{M+1} can be evolved according to any existing MOEAs. In this paper, the underlying MOEA is NSGA-II [2] (line 10), which has received significant atten-

Algorithm 1 Overall framework of DMOEF-MS**Input:** M (dimension of the objective space) n_1 (size of the first M populations) n_2 (size of the last population)**Output:** $REP(g)$

```

1: Set  $t = 0, g = 0, REP(g) = \emptyset$ ;
2: Randomly initialize the multiple populations  $P(g) = \{POP^1(g), POP^2(g), \dots, POP^{M+1}(g)\}$ , where  $|POP^i(g)| = n_1$  ( $i = 1, 2, \dots, M$ ) and  $|POP^{M+1}(g)| = n_2$ ;
3: while stopping criterion is not met do
4:    $g = g + 1$ ;
5:   if the environmental change occurs then
6:     Reinitialize the multiple populations according to Steffensen's diversity-maintenance strategy and  $REP(g)$ ;
7:      $t = t + 1$ ;
8:   end if
9:   Obtain  $POP^i(g)$  ( $i = 1, 2, \dots, M$ ) by Steffensen's prediction strategy;
10:  Obtain  $POP^{M+1}(g)$  according to NSGA-II;
11:  Obtain  $REP(g)$  as the nondominated solutions found by all populations so far;
12:  if  $mod(g, s) = 0$  then
13:    Update  $REP(g)$  by Steffensen's diversity-maintenance strategy;
14:  end if
15: end while

```

tion due to its good optimization performance in solving MOPs since proposed. To increase the population diversity, Steffensen's diversity-maintenance strategy is adopted to update REP at fixed intervals of generations, which is determined by a preset parameter s (line 12). Moreover, Steffensen's diversity-maintenance strategy is also used to reinitialize the multiple populations in DMOEF-MS when the environment changes (line 6).

3.3 Key components in DMOEF-MS

The following section gives a detailed description of the key components in DMOEF-MS, including the construction of single-objective problems, change detection, Steffensen's prediction strategy, Steffensen's diversity-maintenance strategy, and reinitialization strategy.

3.3.1 Construction of single-objective problems

For the i th ($i = 1, 2, \dots, M$) population, its optimization function can be obtained according to Eq. (5), where M is the dimension of the objective space, f_j is the j th objective function of the original DMOP, λ_j is the j th element of λ , $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_M]$ is a randomly generated vector and $\sum_{j=1}^M \lambda_j = 1$. By exchanging values between λ_i and the largest member in λ , λ_i is set to be the largest one in λ for the i th ($i = 1, 2, \dots, M$) population POP^i , as shown in Eq. (5). In

other words, POP^i ($i = 1, 2, \dots, M$) places emphasis on the i th objective function. Therefore, for a DMOP with M objective functions, $POP^1, POP^2, \dots, POP^M$ will focus on different objective functions during their evolving process. The first M populations and the last population POP^{M+1} , which handles the original DMOP, will share the nondominated solutions found in their evolving process in order to get well-distributed Pareto-optimal solutions.

$$\min_{\mathbf{x}} G^i(\mathbf{x}) = \sum_{j=1}^M \lambda_j f_j(\mathbf{x}) \quad (5)$$

$$\lambda_i = \max\{\lambda_j | j = 1, 2, \dots, M\}$$

3.3.2 Change detection

Since there is no noise in the test DMOPs adopted in this paper, the environmental change is detected by reevaluating the selected detectors. Specifically, several individuals are selected as detectors, and the objective values of the detectors are re-calculated at each generation. Please note that it will cost additional function evaluations to detect environmental changes. If the new objective values are different from the previous ones, then the environmental change can be thought to be detected.

3.3.3 Prediction strategy based on Steffensen's method

In DMOEF-MS, the first M populations deal with single-objective problems, which are obtained according to Eq. (5). Suppose the single-objective problem for POP^i ($1 \leq i \leq M$) is $G^i(\mathbf{x})$. Then, Steffensen's method can be utilized to update the individuals in POP^i . Suppose $\mathbf{x}(g)$ is an individual in POP^i at the g th generation, $G^i(\mathbf{x}(g))$ is the objective value of $\mathbf{x}(g)$. As shown in Eq. (6), r is randomly generated in $(0, 1)$. Therefore, l is smaller than $G^i(\mathbf{x}(g))$. The goal of the proposed prediction strategy is to decrease the value of $G^i(\mathbf{x}(g))$, since the original DMOP is a minimization problem. In this case, $G^i(\mathbf{x}(g))$ can be decreased by reducing $G^i(\mathbf{x}(g))$ to l . Suppose $x_j(g)$ is the j th dimension which is chosen to be predicted. Thus, the problem becomes finding the root of the nonlinear equation $G^i(x_j) - l = 0$, with $x_j(g)$ as its initial root. Therefore, the next position of $x_j(g)$, namely $x_j(g+1)$, can be estimated by the proposed prediction strategy, which is based on the above-mentioned Steffensen's method. Steffensen's method can be implemented according to Eq. (7), where $x_j(g+1)$ is the updated value after using the proposed prediction strategy, N is the dimension of the searching space. In this case, Steffensen's prediction strategy will cost at most $2 \times N$ function evaluations for producing each offspring. The detailed procedure

of the Steffensen's prediction strategy is shown in Algorithm 2.

Algorithm 2 Steffensen's prediction strategy

Input:

N (dimension of the searching space)

$\mathbf{x}(g)$

Output:

$\mathbf{x}(g+1)$

1: Set $j = 1$, $stop_flag = 0$;

2: Obtain l by Eq. (6);

3: **while** $stop_flag = 0$ **do**

4: Obtain $x_j(g+1)$ by Eq. (7);

5: **if** $G^i(x(g+1)) \leq l$ or $j \geq N$ **then**

6: $stop_flag = 1$;

7: **else**

8: $j = j + 1$;

9: **end if**

10: **end while**

$$l = r \cdot G^i(\mathbf{x}(g)), r \in (0, 1) \quad (6)$$

$$H^i(x_j(g)) = G^i(x_j(g)) - l$$

$$x_j(g+1) = x_j(g)$$

$$-\frac{H^i(x_j(g))^2}{H^i(x_j(g) + H^i(x_j(g))) - H^i(x_j(g))}, \quad 1 \leq j \leq N \quad (7)$$

3.3.4 Diversity-maintenance strategy based on Steffensen's method

A DMOEA must have a fast convergence speed in order to find the ideal Pareto-optimal solutions before the environmental change occurs. In this paper, Steffensen's prediction strategy is proposed to accelerate the convergence speed. However, fast convergence speed may also lead to a rapid loss of population diversity. As shown in Fig. 2, the obtained Pareto-optimal solutions may concentrate in a small area of the whole PF, which makes it difficult to track the ideal PF after the environment changes.

This paper devises a diversity-maintenance strategy, which is also based on Steffensen's method, to increase the population diversity. In Steffensen's diversity-maintenance strategy, the objective space is divided by a set of evenly distributed weight vectors. Suppose the dimension of the objective space is 2. As shown in Fig. 3, $\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^u$ are u weight vectors and the solid points represent the Pareto-optimal solutions found by algorithms so far. Steffensen's diversity-maintenance strategy is designed to maintain population diversity by obtaining u well-distributed solutions, which are as close as possible to the u weight vectors respectively.

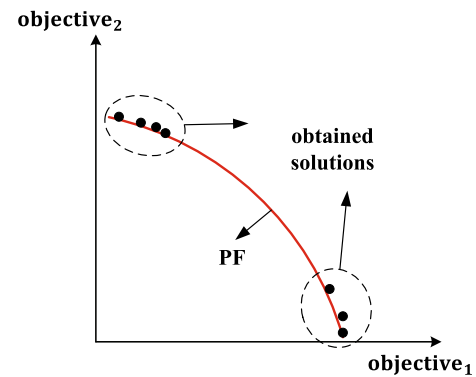


Fig. 2 Representation of unevenly distributed solutions

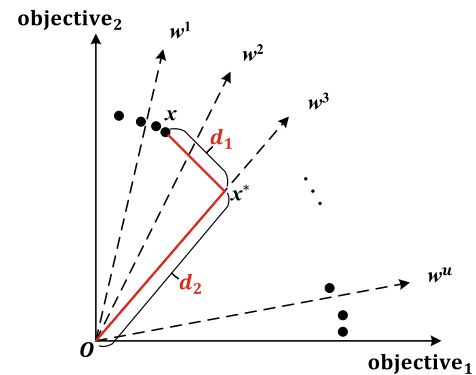


Fig. 3 Steffensen's diversity-maintenance strategy

The most important step in Steffensen's diversity-maintenance strategy is to evaluate the closeness between a solution and a weight vector. To demonstrate the closeness metric used in this paper, \mathbf{x} and \mathbf{w}^3 in Fig. 3 are taken as an example. Specifically, the closeness metric c between \mathbf{x} and \mathbf{w}^3 can be calculated according to Eq. (8). Let \mathbf{x}^* be the projection of \mathbf{x} on \mathbf{w}^3 . As illustrated in Fig. 3, d_1 is the distance between \mathbf{x} and \mathbf{x}^* , and d_2 is the distance between \mathbf{x}^* and \mathbf{O} . $\theta > 0$ is a preset accuracy parameter and is set to 0.5 in this paper. Therefore, the smaller the value of c , the closer \mathbf{x} to \mathbf{w}^3 . r is randomly generated in $(0, 1)$, hence $c_1 = r \cdot c$ is smaller than c . In this case, a solution that is closer to \mathbf{w}^3 can be obtained by finding the root of the nonlinear equation $c - c_1 = 0$, as shown in Eq. (9). Similarly to Sect. 3.3.3, the nonlinear equation can be solved according to Steffensen's method. In Eq. (9), x_j and \hat{x}_j are the j th elements of \mathbf{x} before and after using Steffensen's diversity-maintenance strategy. The detailed procedure of Steffensen's diversity-maintenance strategy is given in Algorithm 3.

$$c = d_1 + \theta \cdot d_2$$

$$c_1 = r \cdot c, \quad r \in (0, 1) \quad (8)$$

$$L(x_j) = c - c_1$$

$$\hat{x}_j = x_j - \frac{L(x_j)^2}{L(x_j + L(x_j)) - L(x_j)}, \quad 1 \leq j \leq N \quad (9)$$

Algorithm 3 Steffensen's diversity-maintenance strategy

Input:
 REP
 N (dimension of the searching space)
 u (number of the weight vectors)
 w^1, w^2, \dots, w^u (the weight vectors)

Output:
 REP

```

1: Set  $newREP = REP$ 
2: for  $i = 1$  to  $u$  do
3:   Obtain  $x$ , which is the closest to  $w^i$  from  $REP$ ;
4:   Obtain  $c$  and  $c_1$  by Eq. (8), respectively;
5:   Set  $j = 1, stop\_flag = 0$ ;
6:   while  $stop\_flag = 0$  do
7:     Obtain  $a_j$  by Eq. (9);
8:     if  $L(x) \leq c_1$  then
9:       Replace the  $j$ th dimension of  $x$  with  $\hat{x}_j$ ;
10:    end if
11:     $j = j + 1$ ;
12:    if  $j \geq N$  then
13:       $stop\_flag = 1$ ;
14:    end if
15:  end while
16:   $newREP = \{newREP, x\}$ ;
17: end for
18: Obtain  $REP$  as the nondominated solutions in  $newREP$ .

```

Algorithm 4 Reinitialization strategy

Input:
 REP
 M (dimension of the objective space)
 n_1 (size of the first M populations)
 n_2 (size of the last population)
 w^1, w^2, \dots, w^u (the weight vectors)

Output:
 $POP^1, POP^2, \dots, POP^M, POP^{M+1}$

```

1: Reevaluate the members of  $REP$  in the new environment;
2:  $POP \leftarrow$  the nondominated solutions in  $REP$ ;
3: for  $i = 1$  to  $M + 1$  do
4:    $POP^i = POP$ ;
5:   if  $i \leq M$  then
6:      $n = n_1$ ;
7:   else
8:      $n = n_2$ ;
9:   end if
10:  while  $|POP^i| < n$  do
11:    Select a weight vector  $w$  from  $w^1, w^2, \dots, w^u$  randomly;
12:    Obtain the individual  $a$ , which is the closest to  $w$  from  $POP$ ;
13:    Update  $a$  according to Steffensen's diversity-maintenance
      strategy (Algorithm 3 lines 4–15);
14:     $POP^i \leftarrow \{POP^i, a\}$ 
15:  end while
16:  if  $|POP^i| > n$  then
17:    Reduce  $|POP^i|$  to  $n$  by removing members from  $POP^i$ 
      randomly;
18:  end if
19: end for

```

3.3.5 Reinitialization strategy

In this paper, the multiple populations will be reinitialized by reusing the nondominated solutions in REP obtained in the last environment when the environmental change occurs. Steffensen's diversity-maintenance strategy is adopted in the reinitialization process to increase the population diversity after the environment changes. The detailed procedure is shown in Algorithm 4.

4 Experimental study**4.1 Test problems and parameter settings**

According to whether the PFs or PSs change with the environment, DMOPs fall into four types.

- Type1: PS changes while PF remains fixed.
- Type2: PS changes while PF changes.
- Type3: PS remains fixed while PF changes.
- Type4: PS remains fixed while PF remains fixed.

In this paper, the proposed DMOEF-MS is tested on 14 DMOPs, whose detailed information is given in Table 1. The FDA test problems are frequently adopted in the performance assessment of DMOEAs and the dMOP test problems are an extension of the FDA problems. The above two test suits consist of 8 problems, which pertain to different types of DMOPs and have linear linkages between the decision variables. For FDA3 and FDA5 in the FDA test suite, the density distribution of the Pareto-optimal solutions along PF is time-varying, which makes it difficult for DMOEAs to find the Pareto-optimal solutions that can maintain a good distribution over time. For dMOP3 in the dMOP test suite, the variables that control the spread of the PF are time-changing, which makes it difficult for DMOEAs to maintain population diversity, especially when the environment changes. The F test suite is composed of ten problems and four of them (F1–F4) are chosen from the FDA and dMOP test suites. This paper utilizes the other test instances (F5–F10), which have nonlinear linkages between the decision variables. Unlike the test problems which have smoothly changing environments, the environmental changes are sharp and irregular for F9 and F10.

As described in Sect. 3.2, the underlying MOEA adopted in the proposed DMOEF-MS is NSGA-II. To evaluate the effectiveness of the proposed strategies, DMOEF-MS is compared with two dynamic NSGA-II algorithms, namely DNSGA-II-A and DNSGA-II-B [12]. Furthermore, for comprehensive assessment, the proposed DMOEF-MS is also compared with four state-of-the-art DMOEAs: MOEA/D with a hybrid of memory and prediction strategies (MOEA/D-

Table 1 Detailed information of test problems

Test problem	Number of objectives	Type of the DMOP	References
FDA1	2	Type1	[43]
FDA2	2	Type3	[43]
FDA3	2	Type2	[43]
FDA4	3	Type1	[43]
FDA5	3	Type2	[43]
dMOP1	2	Type3	[17]
dMOP2	2	Type2	[17]
dMOP3	2	Type1	[17]
F5	2	Type2	[27]
F6	2	Type2	[27]
F7	2	Type2	[27]
F8	3	Type1	[27]
F9	2	Type2	[27]
F10	2	Type2	[27]

HMPS) [18], quantum immune clonal coevolutionary algorithm based on competition and cooperation among multiple populations (QICCA) [21], steady-state and generational evolutionary algorithm (SGEA) [37], and Kalman Filter prediction implemented in MOEA/D-DE (MOEA/D-KF) [36].

The parameter settings used in the test problems and the comparative algorithms are as follows. For all test problems, the index of environments t is calculated as Eq. (10). Where g is the number of generations, n_t and τ_t are the severity and frequency of the environmental changes, respectively. In this paper, n_t is set to 10, and τ_t is set to 300. The maximal number of function evaluations is set to 30,000 before the environmental change occurs and the environment will change 40 times in each run. To get statistical results, each algorithm will run 20 times for each test problem independently. For all algorithms, the population size is set to 100 for 2-objective problems and 105 for 3-objective problems. Specifically, the population sizes of the multiple populations in DMOEF-MS are shown in Table 2. Following MOEA/D-KF and MOEA/D-HMPS, 10 individuals, which are randomly generated in searching space is reevaluated to detect the environmental changes in DMOEF-MS. The construction method of the weight vectors used in DMOEF-MS can refer to Das and Dennis [44], which provided an efficient

way to produce a set of points that are evenly distributed in a multidimensional space. In the proposed algorithm, the empirical setting of the number of weight vectors is shown in Table 2. Moreover, the parameter s in Algorithm 1 is set to 5, which means Steffensen's diversity-maintenance strategy is implemented at intervals of 5 generations. In DNSGA-II-A and DNSGA-II-B, the re-initialization parameter ζ is set to 20, and the hypermutation rate is 0.5.

$$t = \frac{1}{n_t} \lfloor \frac{g}{\tau_t} \rfloor \quad (10)$$

4.2 Performance metrics

In this paper, the following metrics are adopted to assess the performance of algorithms in terms of convergence and diversity.

4.2.1 Modified inverted generational distance

Modified inverted generational distance (*MIGD*) [27] is derived from the inverted generational distance (*IGD*) [45,46], which can be calculated according to Eq. (11) and is widely used to evaluate the performance of MOEAs in static environments. In Eq. (11), $F(\mathbf{x}^i, t)$ is the objective value of \mathbf{x}^i in the t th environment, $\|F(\mathbf{x}^i, t) - F(\mathbf{x}^j, t)\|$ is the Euclidean distance between $F(\mathbf{x}^i, t)$ and $F(\mathbf{x}^j, t)$. P_t^* is a set of points which are evenly distributed along the ideal PF in the t th environment, and $|P_t^*|$ is the number of the members in P_t^* . P_t is the Pareto-optimal solutions obtained by the algorithm that needs to be assessed. *IGD* assesses the performance of an algorithm in terms of diversity and accuracy. If *IGD* = 0 then each solution in P_t^* will be found in P_t . In this case, the obtained Pareto-optimal solutions are just on the ideal PF and do not miss any part of the whole PF. As shown in Eq. (12), $|T|$ is the number of the environmental changes in a single run.

$$IGD(P_t^*, P_t) = \frac{\sum_{\mathbf{x}^i \in P_t^*} d(\mathbf{x}^i, P_t)}{|P_t^*|} \quad (11)$$

$$d(\mathbf{x}^i, P_t) = \min_{\mathbf{x}^j \in P_t} \{\|F(\mathbf{x}^i, t) - F(\mathbf{x}^j, t)\|\}$$

Table 2 Parameter settings in DMOEF-MS

		Population size	Number of weight vectors
2-Objective problems	The first 2 populations	30	21
	The last population	40	28
3-Objective problems	The first 3 populations	25	21
	The last population	30	28

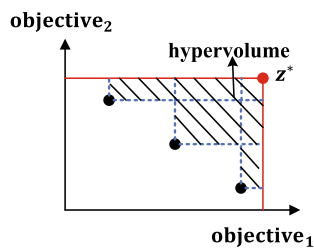


Fig. 4 Computation of HV in 2-dimensional objective space

$$MIGD(P_t^*, P_t) = \frac{1}{|T|} \sum_{1 \leq t \leq |T|} IGD(P_t^*, P_t) \quad (12)$$

4.2.2 Modified hypervolume

Modified hypervolume (MHV) [18] is an extension of HV [47] and can be calculated as shown in Eq. (13). HV computes the area of the hypervolume which is constructed by the obtained P_t and a reference point z^* . For minimization problems, $z^* = [z_1 + 0.1, z_2 + 0.1, \dots, z_M + 0.1]$, z_i ($1 \leq i \leq M$) is the maximum objective value for the i th objective function in the t th environment, M is the dimension in the objective space. Suppose $M = 2$, the computation of HV is shown in Fig. 4.

$$MHV = \frac{1}{|T|} \sum_{1 \leq t \leq |T|} HV(P_t, z^*) \quad (13)$$

4.3 Investigation of parameter s in DMOEF-MS

In DMOEF-MS, s controls the frequency of using Steffensen's diversity-maintenance strategy. If s is too large, Steffensen's diversity-maintenance strategy will function inadequately. However, if s is too small, too much function evaluations will be spent on maintaining population diversity. To demonstrate the influence of s on the performance of DMOEF-MS, FDA1 and dMOP2 are taken as examples. In this section, s is set to 1, 3, 5, 7, and 9, respectively.

Figure 5 shows the mean $MIGD$ and MHV obtained by DMOEF-MS with s changing from 1 to 9 for FDA1 and dMOP2 over 20 independent runs. It can be seen from Fig. 5, as s decreases, $MIGD$ and MHV tend to become better at first, and then become worse severely. If s is too small, DMOEF-MS will spend too much function evaluations on Steffensen's diversity-maintenance strategy and the other strategies proposed in DMOEF-MS will function inadequately, since the maximum number of function evaluations has been given for each environment in this paper. If s is too large, DMOEF-MS will have a poor performance in maintaining population diversity. As Fig. 5 shows, DMOEF-MS with $s = 5$ performs best in terms of both $MIGD$ and MHV

for FDA1. For dMOP2, DMOEF-MS with $s = 5$ also has the best performance in terms of MHV . Therefore, s is set to 5 in this paper.

4.4 Performance comparison with DNSGA-II-A and DNSGA-II-B

Tables 3 and 4 present the statistical results of MIGD and MHV values obtained by DMOEF-MS, DNSGA-II-A and DNSGA-II-B for all test problems. The boldface numbers in Tables 3 and 4 represent the best results obtained by all 3 algorithms.

It can be seen from Tables 3 and 4 that the proposed algorithm has achieved the best performance for most of the test problems in terms of $MIGD$ and MHV . Since the underlying MOEA in DMOEF-MS is NSGA-II, the fact that DMOEF-MS performs better than DNSGA-II-A and DNSGA-II-B may indicate the effectiveness of the proposed strategies in DMOEF-MS. In DNSGA-II-A, 20% of the population are reinitialized randomly in the searching space when the environmental change occurs. However, in DNSGA-II-B, 20% of the population are mutated to adapt to a new environment. Therefore, DNSGA-II-A may perform better for problems with large environmental change. As shown in Tables 3 and 4, DNSGA-II-A performs better than DNSGA-II-B for F9 and F10, in which the environment changes sharply and irregularly. For FDA3, DMOEF-MS is ranked in second and third place in terms of $MIGD$ and MHV , respectively. For FDA5, DMOEF-MS is ranked in third place in terms of $MIGD$. For FDA3 and FDA5, the density distribution of the Pareto-optimal solutions along PF changes over time. However, in Steffensen's diversity-maintenance strategy adopted in DMOEF-MS, a set of evenly distributed weight vectors is used to obtain the Pareto-optimal solutions. As a result, Steffensen's diversity-maintenance strategy may weaken the performance of DMOEF-MS, as shown in Fig. 6b. For F8, DMOEF-MS performs worse than DNSGA-II-B in terms of $MIGD$. However, the MHV value obtained by DMOEF-MS is the best among the 3 algorithms.

To better compare the performance of the 3 (k) algorithms on all the 14 (N) test problems, the Friedman test and the pairwise post hoc Bonferroni–Dunn test [48] are conducted on the statistical results. Table 5 gives the average ranks achieved by the 3 algorithms on all test problems in terms of $MIGD$ and MHV . The boldface numbers give the best rank values obtained by DNSGA-II-A, DNSGA-II-B, and DMOEF-MS. As shown in Table 5, for $MIGD$ and MHV , the Friedman statistic (F_F) values are greater than the critical value of the F-distribution with $(k - 1)$ and $(k - 1) \times (N - 1)$ degrees of freedom at 90% confidence level ($F_{0.1}(2, 26) = 2.52$). In this case, the Friedman test reports significant differences among the comparative algorithms for

Fig. 5 *MIGD* and *MHV* obtained by DMOEF-MS with different s

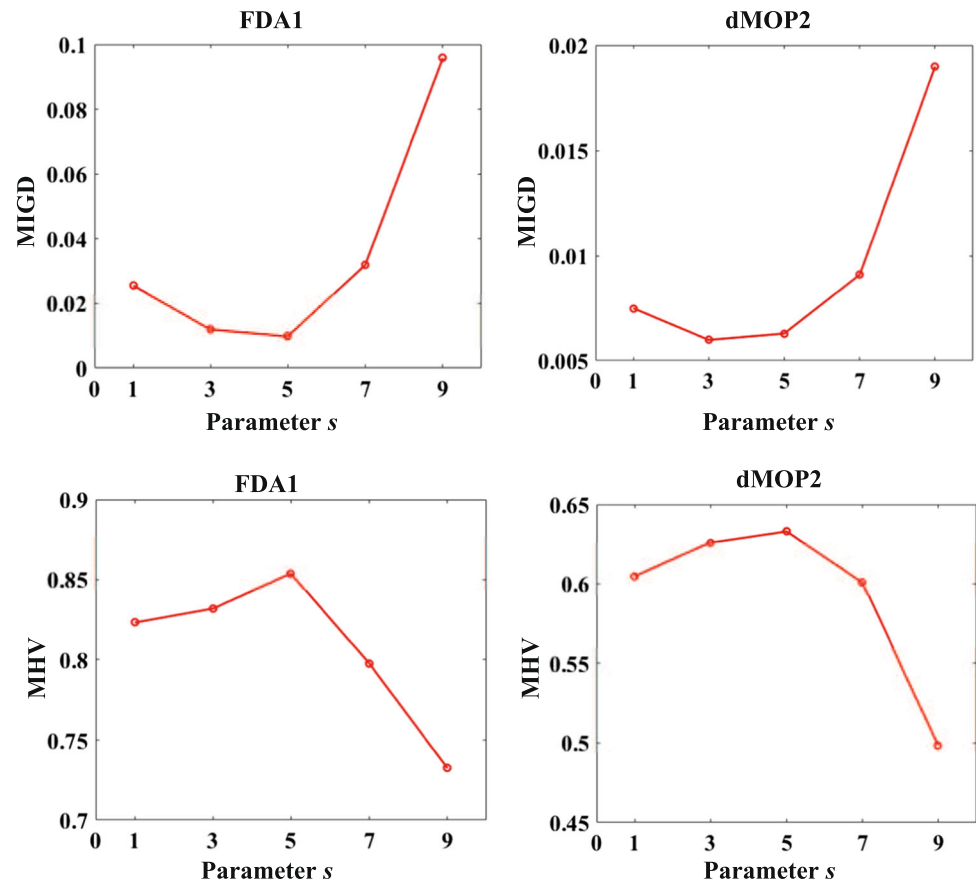


Table 3 Statistical values of *MIGD* obtained by DMOEF-MS, DNSGA-II-A and DNSGA-II-B

Problem	DNSGA-II-A	DNSGA-II-B	DMOEF-MS
FDA1	2.66E−02 ± 4.75E−03	2.69E−02 ± 1.26E−03	9.78E−03 ± 4.43E−04
FDA2	8.92E−02 ± 9.72E−02	8.94E−02 ± 6.06E−03	1.12E−02 ± 3.70E−03
FDA3	1.94E−02 ± 5.22E−02	1.15E−01 ± 8.40E−02	2.00E−02 ± 1.94E−03
FDA4	8.51E−02 ± 1.66E−02	8.43E−02 ± 2.04E−03	6.25E−02 ± 1.62E−03
FDA5	1.31E−01 ± 4.31E−02	1.33E−01 ± 3.18E−02	1.45E−01 ± 5.85E−02
dMOP1	6.80E−02 ± 3.99E−02	3.34E−02 ± 1.31E−02	8.94E−03 ± 6.99E−04
dMOP2	5.62E−01 ± 9.49E−02	6.95E−02 ± 8.67E−02	6.27E−03 ± 4.93E−04
dMOP3	4.38E−01 ± 6.83E−02	3.63E−01 ± 6.20E−02	5.75E−02 ± 4.73E−03
F5	9.63E−01 ± 3.48E−01	5.77E−01 ± 1.09E−01	4.66E−02 ± 4.07E−03
F6	6.05E−01 ± 9.39E−02	5.72E−01 ± 3.29E−02	9.55E−02 ± 2.54E−03
F7	5.90E−01 ± 7.71E−02	4.23E−01 ± 2.71E−01	4.76E−02 ± 7.14E−03
F8	2.09E−01 ± 2.25E−02	1.92E−01 ± 1.37E−01	1.97E−01 ± 2.43E−02
F9	1.35E+00 ± 5.81E−01	2.10E+00 ± 5.06E−01	4.65E−01 ± 9.29E−03
F10	1.32E−01 ± 4.37E−01	1.49E−01 ± 2.53E−01	5.80E−02 ± 3.50E−03

both *MIGD* and *MHV*. After that, the pairwise post hoc Bonferroni-Dunn test is conducted for pairwise comparison between DMOEF-MS and the other comparative algorithms. The critical difference (CD) according to the Bonferroni-Dunn test at 90% confidence level is 0.74. As shown in Table 5, the differences between the average ranks of DMOEF-MS and the other comparative algorithms are greater than 0.74,

which means DMOEF-MS performs better than DNSGA-II-A and DNSGA-II-B at 90% confidence level in terms of both *MIGD* and *MHV*. In other words, as shown in Fig. 6, DMOEF-MS generates solutions more efficiently and stably than the other two algorithms for most test problems.

Table 4 Statistical values of *MHV* obtained by DMOEF-MS, DNSGA-II-A and DNSGA-II-B

Problem	DNSGA-II-A	DNSGA-II-B	DMOEF-MS
FDA1	7.96E-01±4.62E-02	7.79E-01±5.00E-02	8.32E-01±4.50E-04
FDA2	1.09E+00±5.57E-01	1.08E+00±1.61E-01	1.12E+00±1.56E-03
FDA3	1.23E+00±9.21E-02	1.02E+00±5.77E-02	9.05E-01±1.31E-01
FDA4	6.55E-01±9.30E-03	6.70E-01±9.58E-03	7.30E-01±8.53E-03
FDA5	1.22E+00±1.27E-01	1.38E+00±2.43E-01	2.85E+00±1.23E-03
dMOP1	6.30E-01±3.61E-02	6.67E-01±3.55E-02	6.69E-01±3.69E-02
dMOP2	5.11E-01±4.52E-02	4.69E-01±3.79E-01	6.36E-01±7.06E-03
dMOP3	7.81E-01±1.84E-01	8.12E-01±5.14E-02	8.47E-01±6.49E-03
F5	5.17E-01±2.76E-02	4.96E-01±7.13E-02	7.52E-01±1.83E-02
F6	6.32E-01±1.93E-02	6.64E-01±3.19E-02	6.97E-01±4.46E-02
F7	4.17E-01±3.89E-02	5.45E-01±9.16E-02	7.05E-01±5.32E-02
F8	4.43E-01±9.04E-03	4.73E-01±5.19E-02	5.78E-01±4.70E-03
F9	5.68E-01±9.23E-02	5.20E-01±3.79E-02	6.84E-01±3.31E-02
F10	3.78E-01±3.18E-02	3.12E-01±1.97E-02	6.91E-01±2.61E-02

4.5 Effects of the key operators in DMOEF-MS

The key operators adopted in DMOEF-MS are Steffensen's prediction strategy, Steffensen's diversity-maintenance strategy, and the reinitialization strategy. Since the multiple populations in DMOEF-MS are reinitialized based on Steffensen's diversity-maintenance strategy, the contributions of Steffensen's prediction and diversity-maintenance strategies are studied in this section. In this section, two variants of DMOEF-MS are adopted to demonstrate the effectiveness of the two strategies mentioned above. One variant is DMOEF-MS without Steffensen's prediction strategy and the other one is DMOEF-MS without Steffensen's diversity-maintenance strategy. In DMOEF-MS without Steffensen's diversity-maintenance strategy, the multiple populations are reinitialized randomly in searching space when the environment changes. Two problems, namely FDA1 and dMOP2, are tested in this section. Figure 7 demonstrates the mean MIGD values obtained by 3 algorithms over 20 independent runs. It can be seen from Fig. 7 that the performance of DMOEF-MS without Steffensen's prediction strategy is significantly worse than that of DMOEF-MS for FDA1 and dMOP2. This may indicate that the proposed prediction strategy does help the algorithm find more promising solutions throughout the whole run. The performance of DMOEF-MS without Steffensen's diversity-maintenance strategy is also worse than that of DMOEF-MS for FDA1 and dMOP2. Therefore, Steffensen's diversity-maintenance strategy also can improve the performance of DMOEF-MS in terms of IGD. As shown in Fig. 7, Steffensen's prediction strategy may play a greater role in improving the performance of DMOEF-MS.

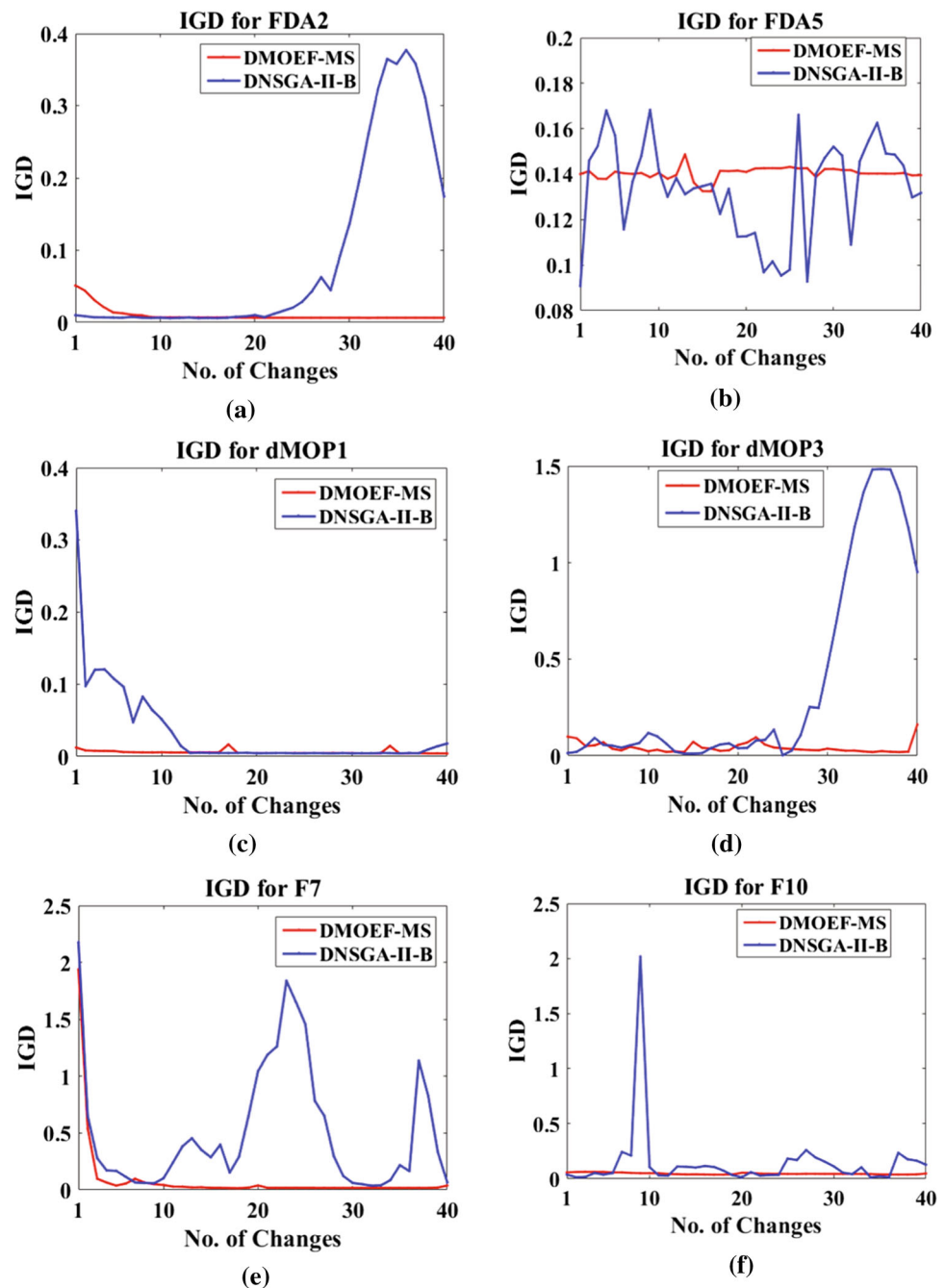
4.6 Influence of the severity and frequency of environmental changes

To investigate the performance of DMOEF-MS on DMOPs with different change severity and frequency, FDA1 and dMOP2 are tested in this section. To study the influence of n_t (severity of the environmental change), τ_t (frequency of the environmental change) is fixed to 30 and n_t is set to 5, 10, and 20, respectively. In this section, the number of environmental changes in a single run is set to 120. The mean values and the standard deviations of *MIGD* of 20 independent runs on FDA1 and dMOP2 are demonstrated in Fig. 8. It can be observed from Fig. 8, the performance of DMOEF-MS becomes better as n_t increases. Similarly, to investigate the influence of τ_t , n_t is fixed to 10 and τ_t is set to 10, 20 and 30, respectively. The mean values and the standard deviations of *MIGD* of 20 independent runs on FDA1 and dMOP2 are demonstrated in Fig. 9. As Fig. 9 shows, the performance of DMOEF-MS also becomes better as τ_t increases.

4.7 Performance comparison with other DMOEAs

Tables 6 and 7 present the statistical results of *MIGD* and *MHV* obtained by MOEA/D-HMPS, QICCA, SGEA, MOEA/D-KF, and the proposed DMOEF-MS. According to Tables 6 and 7, DMOEF-MS achieves the best *MIGD* values in 7 out of 14 problems and the best *MHV* values in 8 out of 14 problems. The statistical results may indicate the effectiveness of the proposed strategies adopted in DMOEF-MS. On the one hand, Steffensen's prediction strategy improves the ability to find promising solutions. As described in Sect. 3.2, most individuals in DMOEF-MS deal with single-objective problems, which can be optimized by solving nonlinear functions. Therefore, Steffensen's pre-

Fig. 6 Average *IGD* values obtained by DMOEF-MS and DMOEF-MS over the number of environmental changes for 20 runs



diction strategy can obtain the next positions of individuals efficiently with the help of Steffensen's method. On the other hand, the adopted multipopulation structure and Steffensen's diversity maintenance strategy are effective for maintaining population diversity. Moreover, the reinitialization strategy improves the ability to adapt to the new environments in DMOEF-MS.

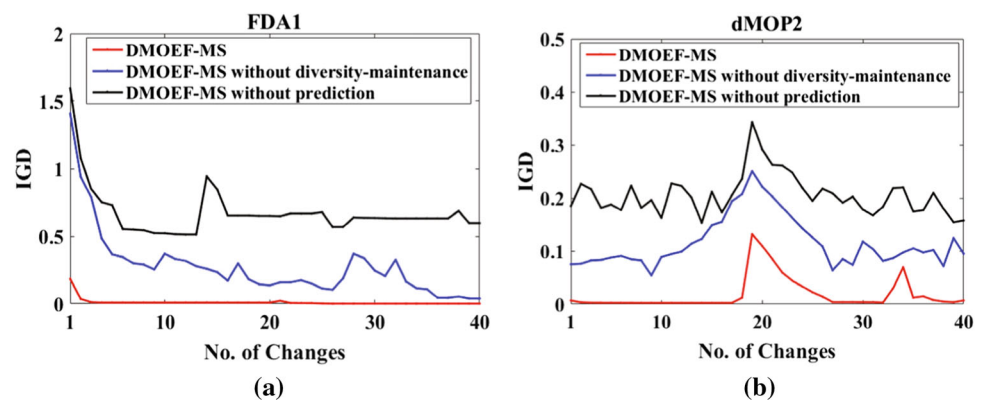
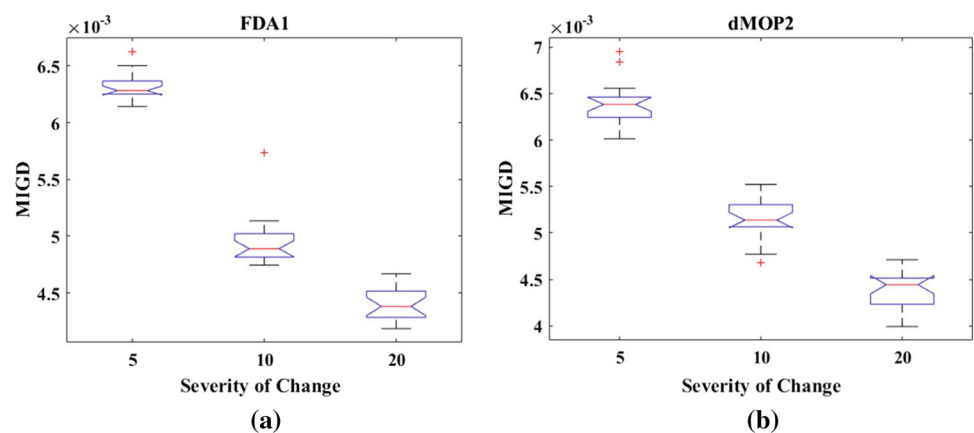
Table 8 shows the average ranks of MOEA/D-HMPS, QICCA, SGEA, MOEA/D-KF, and DMOEF-MS by the Friedman test. The boldface numbers are the best results achieved by all 5 algorithms. It can be seen from Table 8

that the statistic value F_F is greater than $F_{0.1}(5 - 1, (5 - 1) \times (14 - 1))$. Therefore, there are significant differences among the 5 algorithms at 90% confidence level. As shown in Table 8, the proposed DMOEF-MS achieves the best and the second-best average ranks in terms of *MIGD* and *MHV*, respectively.

Although Steffensen's prediction strategy adopted in DMOEF-MS can help the algorithm find accurate and efficient solutions, the experimental results in Tables 6 and 7 show that DMOEF-MS does not achieve the best performance for some test problems. For FDA3 and FDA5, the

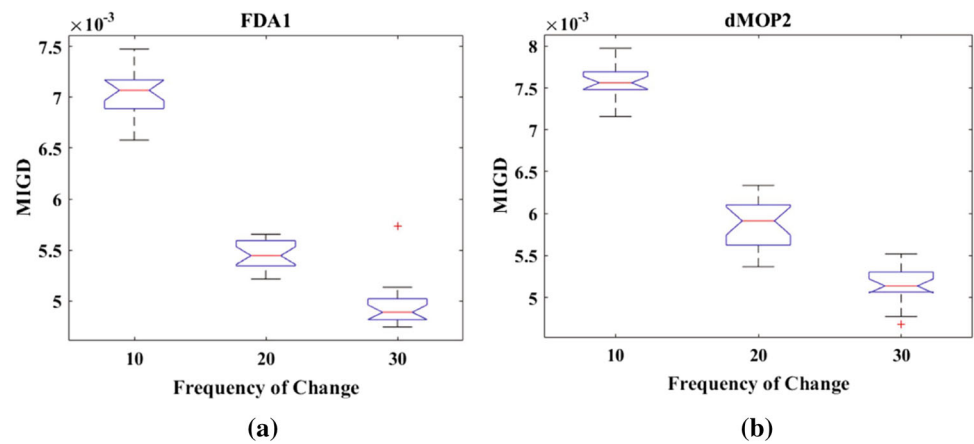
Table 5 Average ranks of algorithms by Friedman test for *MIGD* and *MHV*

Algorithm	Average Rank	Average Rank Difference w.r.t. DMOEF-MS
<i>MIGD</i>		
DNSGA-II-A	2.43	1.14
DNSGA-II-B	2.29	1.00
DMOEF-MS	1.29	—
Statistic Value F_F	8.91	
<i>MHV</i>		
DNSGA-II-A	2.5	1.36
DNSGA-II-B	2.36	1.22
DMOEF-MS	1.14	—
Statistic Value F_F	16.52	

Fig. 7 Average IGD values of DMOEF-MS and two variants**Fig. 8** *MIGD* on FDA1 and dMOP2 with different τ_t 

density distribution of the Pareto-optimal solutions changes over time. In other words, the ideal Pareto-optimal solutions are not evenly distributed along the PFs in some stages of the whole run. Thus the proposed Steffensen's diversity-maintenance strategy, which obtains the Pareto-optimal solutions under the guidance of a set of evenly distributed weight vectors, may weaken the performance of DMOEF-MS sometimes. For dMOP3, population diver-

sity is particularly important since the variables that control the spread of PFs are time-changing. In MOEA/D-KF and MOEA/D-HMPS, the underlying MOEA is MOEA/D, which can help to increase population diversity in the algorithms. In QICCA, the coevolutionary competitive and cooperative operators, which can obtain uniform and diverse solutions, help QICCA find good approximations of the ideal solutions. Therefore, DMOEF-MS performs relatively worse than

Fig. 9 *MIGD* on FDA1 and dMOP2 with different n_t **Table 6** Mean and Standard Deviation values of *MIGD* obtained by DMOEF-MS and other DMOEAs

Problem	MOEA/D-HMPS	QICCA	SGEA	MOEA/D-KF	DMOEF-MS
FDA1	1.27E-02 (1.31E-04)	4.69E-02 (2.35E-03)	4.45E-02 (6.94E-03)	1.16E-02 (9.70E-04)	9.78E-03 (1.50E-04)
FDA2	7.82E-03 (3.44E-04)	3.07E-02 (4.38E-03)	2.29E-01 (2.81E-02)	1.27E-02 (4.65E-03)	1.12E-02 (3.70E-03)
FDA3	1.90E-02 (2.86E-03)	4.33E-02 (2.86E-03)	3.86E-02 (4.45E-03)	2.25E-02 (1.64E-03)	2.00E-02 (1.94E-03)
FDA4	8.84E-02 (3.75E-03)	1.27E-01 (9.76E-03)	9.53E-02 (2.97E-03)	9.14E-02 (5.51E-03)	6.25E-02 (1.62E-03)
FDA5	1.59E-01 (3.40E-02)	1.81E-01 (1.19E-02)	1.15E-01 (3.59E-02)	1.65E-01 (4.89E-02)	1.45E-01 (5.85E-02)
dMOP1	6.90E-03 (9.23E-04)	1.07E-02 (7.51E-03)	9.11E-03 (2.51E-03)	9.19E-03 (5.06E-04)	8.94E-03 (6.99E-04)
dMOP2	7.12E-03 (3.92E-04)	2.63E-02 (5.93E-03)	1.32E-02 (4.72E-03)	9.13E-03 (3.86E-04)	6.27E-03 (4.93E-04)
dMOP3	4.81E-02 (2.57E-04)	5.46E-02 (6.16E-03)	7.85E-02 (2.51E-03)	3.09E-02 (1.96E-03)	5.75E-02 (4.73E-03)
F5	6.49E-02 (3.51E-03)	7.06E-01 (4.83E-02)	5.27E-01 (4.58E-02)	6.56E-02 (5.94E-03)	4.66E-02 (4.07E-03)
F6	9.85E-02 (9.17E-03)	5.19E-01 (2.85E-02)	4.28E-01 (3.75E-02)	1.80E-01 (3.77E-02)	9.55E-02 (2.54E-03)
F7	5.25E-02 (3.80E-03)	7.34E-02 (5.67E-03)	6.57E-01 (7.79E-03)	6.22E-02 (5.40E-03)	4.76E-02 (7.14E-03)
F8	2.84E-01 (1.01E-02)	3.75E-01 (3.79E-02)	3.67E-01 (9.34E-03)	2.93E-01 (1.29E-02)	1.97E-01 (2.43E-02)
F9	3.43E-02 (5.68E-03)	3.92E-01 (4.69E-02)	2.31E-01 (3.37E-02)	3.55E-01 (1.62E-02)	4.65E-01 (9.29E-03)
F10	7.06E-02 (7.94E-03)	3.18E-01 (1.12E-02)	2.77E-01 (5.28E-03)	3.71E-02 (5.28E-03)	5.80E-02 (3.50E-03)

MOEA/D-KF, MOEA/D-HMPS and QICCA for dMOP3 in terms of *MIGD*. F9 and F10 are relatively difficult to optimize among the test problems. In MOEA/D-HMPS, two different response strategies are adopted according to whether the current environmental change is similar to the

historical changes. Therefore, for F9 and F10, MOEA/D-HMPS outperform DMOEF-MS, in which the predication process is only based on the information in the current population.

Table 7 Mean and Standard Deviation values of *MHV* obtained by DMOEF-MS and Other DMOEAs

Problem	MOEA/D-HMPS	QICCA	SGEA	MOEA/D-KF	DMOEF-MS
FDA1	8.26E−01 (2.63E−03)	8.10E−01 (6.54E−04)	8.06E−01 (6.89E−04)	8.11E−01 (7.48E−04)	8.32E−01 (4.50E−04)
FDA2	6.91E−01 (8.25E−04)	6.08E−01 (5.38E−03)	6.15E−01 (9.96E−03)	6.22E−01 (7.82E−04)	1.12E+00 (1.06E−03)
FDA3	1.42E+00 (8.44E−03)	1.19E+00 (3.99E−02)	1.23E+00 (2.59E−02)	1.38E+00 (8.00E−03)	9.05E−01 (1.31E−02)
FDA4	7.09E−01 (8.69E−03)	6.95E−01 (5.79E−02)	6.14E−01 (5.49E−02)	6.92E−01 (1.45E−03)	7.30E−01 (8.53E−03)
FDA5	4.40E+00 (7.60E−03)	3.51E+00 (2.39E−02)	4.35E+00 (1.83E−03)	4.62E+00 (2.40E−03)	2.85E+00 (1.23E−03)
dMOP1	6.63E−01 (4.91E−02)	6.27E−01 (4.89E−02)	6.44E−01 (3.37E−02)	6.49E−01 (9.01E−03)	6.69E−01 (3.69E−02)
dMOP2	6.42E−01 (4.03E−04)	6.11E−01 (9.65E−03)	6.26E−01 (1.32E−03)	6.29E−01 (9.42E−04)	6.36E−01 (7.06E−03)
dMOP3	8.57E−01 (8.21E−03)	7.23E−01 (1.54E−02)	7.59E−01 (4.30E−03)	8.35E−01 (6.69E−03)	8.47E−01 (6.49E−03)
F5	7.32E−01 (2.96E−03)	6.47E−01 (7.44E−03)	7.09E−01 (1.89E−03)	7.11E−01 (6.86E−03)	7.52E−01 (1.83E−02)
F6	6.80E−01 (9.29E−03)	3.68E−01 (7.75E−02)	4.18E−01 (4.86E−02)	6.25E−01 (4.35E−02)	6.97E−01 (4.46E−02)
F7	6.81E−01 (7.94E−03)	3.06E−01 (6.44E−02)	5.08E−01 (3.78E−02)	6.10E−01 (8.11E−03)	7.05E−01 (5.32E−02)
f8	5.59E−01 (6.22E−03)	4.50E−01 (5.87E−02)	4.39E−01 (2.07E−02)	5.50E−01 (3.01E−03)	5.78E−01 (4.70E−03)
f9	7.23E−01 (2.27E−03)	3.94E−01 (4.35E−02)	4.59E−01 (3.11E−02)	6.42E−01 (9.23E−03)	6.84E−01 (3.31E−02)
f10	6.28E−01 (4.08E−02)	3.81E−01 (5.94E−02)	3.11E−01 (2.62E−02)	7.14E−01 (6.02E−02)	6.91E−01 (2.61E−02)

Table 8 Average ranks of algorithms by Friedman Test for *MIGD* and *MHV*

MIGD		MHV	
Algorithm	Average Rank	Algorithm	Average Rank
MOEA/D-HMPS	1.93	MOEA/D-HMPS	1.71
QICCA	4.79	QICCA	4.50
SGEA	3.57	SGEA	4.14
MOEA/D-KF	2.86	MOEA/D-KF	2.71
DMOEF-MS	1.86	DMOEF-MS	1.86
Statistic Value F_F	14.63	Statistic Value F_F	14.97

4.8 Running time

This section demonstrates the running times of the algorithms on three two-objective problems (FDA2, FDA4, and dMOP2) and two three-objective problems (F5 and F8). This experiment is carried out on a personal computer (Intel(R) Core(TM) i7-8700K CPU @ 3.70 GHz, 32.0 G RAM) and each algorithm runs 20 times independently. Table 9 gives the running times of the algorithms.

It can be observed from Table 9, MOEA/D-KF is the most time-consuming, since the prediction model adopted in it is more complex than the others. MOEA/D-HMPS, SGEA, and DMOEF-MS take relatively less time to predict the next locations of the candidate solutions. As shown in Table 9, the time taken by MOEA/D-HMPS and DMOEF-MS are relatively close. Specifically, SGEA takes more time than MOEA/D-HMPS and DMOEF-MS because the generational environmental selection method spends some time on pre-

Table 9 Running Time of Algorithms

Test Problem	MOEA/D-HMPS	QICCA	SGEA	MOEA/D-KF	DMOEF-MS
FDA2	109(s)	1143(s)	546(s)	10252(s)	105(s)
FDA4	114(s)	1207(s)	509(s)	11793(s)	149(s)
dMOP2	106(s)	1124(s)	556(s)	10762(s)	110(s)
F5	179(s)	1403(s)	598(s)	12932(s)	195(s)
F8	223(s)	1570(s)	582(s)	13874(s)	237(s)

serving good solutions for the next generation. QICCA takes more time than DMOEF-MS because the quantum updating operator and the competitive-cooperative operator takes more time to improve the performance of the obtained solutions.

5 Concluding remarks

This paper proposes a dynamic multiobjective evolutionary framework (DMOEF-MS), which adopts a novel multipopulation structure and Steffensen's method to handle DMOPs. Among the multiple populations in DMOEF-MS, one population evolves by the classic NSGA-II while the others evolve according to the proposed prediction strategy which is based on Steffensen's method. Moreover, Steffensen's diversity-maintenance strategy is introduced to increase population diversity at fixed intervals of generations before the environment changes and reinitialize the multiple populations after the environment changes.

DMOEF-MS has been compared with some state-of-the-art DMOPs on several widely used DMOPs. The experimental results demonstrate that DMOEF-MS can track the moving Pareto-optimal solutions efficiently on most of the test problems. Specifically, the multiple populations handle different optimization problems and cooperate to get well-distributed Pareto-optimal solutions. Steffensen's prediction strategy accelerates the convergence speed by predicting the next positions of individuals efficiently. Meanwhile, Steffensen's diversity-maintenance strategy increases the population diversity by using a set of evenly-distributed weight vectors.

However, it can be observed from Table 9, DMOEF-MS takes more time than some of the comparative algorithms. The main reason for this is that the prediction strategy adopted in DMOEF-MS needs to consider all dimensions of an individual sometimes, as shown in Algorithm 2. Thus DMOEF-MS takes a relatively long time to predict the next positions of individuals. To improve the efficiency of DMOEF-MS, the possible directions of this paper are listed below.

1. A potential weakness of this paper is the lack of correlation analysis of decision variables, which can be used to divide the decision variables into several groups. As discussed in Sect. 3.3.3, the proposed prediction strategy will cost at most $2 \times N$ function evaluations for producing each offspring. The combination of Steffensen's prediction strategy and the correlation analysis of decision variables might be a possible way to improve the time efficiency of DMOEF-MS.
2. The main framework of DMOEF-MS is to optimize multiple problems, including the constructed single-objective problems and the original DMOP, by multiple populations. In this case, the construction method of single-objective problems plays an important role in the performance of DMOEF-MS. Therefore, the analysis of different methods of constructing single-objective problems is also one future work of this paper.
3. The maintenance of population diversity plays a crucial role in DMOPs. Therefore, investigating other efficient methods to update the repository population, such as DAA [49], is also a possible work of this paper.

Acknowledgements This work was supported by the National Natural Science Foundation of China [Grant No. 61806122].

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Coello CAC (2006) Evolutionary multi-objective optimization: a historical view of the field. *IEEE Comput Intell Mag* 6(1):28–36
- Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evolut Comput* 6(2):182–197
- Zitzler E, Laumanns M, Thiele L (2002) Spea2: improving the strength pareto evolutionary algorithm. In: *Proceedings on evolution methods design optimization control application industrial problems*, pp 95–100
- Coello CAC, Lechuga MS (2002) MOPSO: a proposal for multiple objective particle swarm optimization. In: *Proceedings of the 2002 congress on evolutionary computation*, 2002. CEC 02
- Zhang QF, Li H (2007) MOEA/D: a multi-objective evolutionary algorithm based on decomposition. *IEEE Trans Evolut Comput* 11(6):712–731
- Xu BL, Wang ZQ (2007) A multi-objective-ACO-based data association method for bearings-only multi-target tracking. *Commun Nonlinear Sci Numer Simul* 12:1360–1369
- Isaacs A, Puttigg V, Ray T, Smith W, Anavatti S (2008) Development of a memetic algorithm for dynamic multi-objective optimization and its applications for online neural network modeling of UAVS. In: *Neural networks, 2008. IJCNN 2008. IEEE world congress on computational intelligence*, pp 548–554
- Deng ZM, Liu MB, Ouyang YF, Lin SJ, Xie M (2018) Multi-objective mixed-integer dynamic optimization method applied to optimal allocation of dynamic var sources of power systems. *IEEE Trans Power Syst* 33(2):1683–1697
- Wang X, Yao PY, Zhang JY, Lujun W, Jiao ZQ (2019) Dynamic resource scheduling for C2 organizations based on multi-objective optimization. *IEEE Access* 7:64614–64626
- Ding JL, Yang C, Xiao Q, Chai TY, Jin YC (2019) Dynamic evolutionary multiobjective optimization for raw ore allocation in mineral processing. *IEEE Trans Emerg Top Comput Intell* 3(1):36–48
- Chen W, Yang H, HAo YF (2019) Scheduling of dynamic multi-objective flexible enterprise job-shop problem based on hybrid QPSO. *IEEE Access* 7:127090–127097
- Deb K, Rao NUB, Karthik S (2007) Dynamic multi-objective optimization and decision-making using modified NSGA-II: a case study on hydro-thermal power scheduling. In: *international conference on evolutionary multi-criterion optimization*, p 803–817
- Mavrovouniotis M, Yang SX (2013) Genetic algorithms with adaptive immigrants for dynamic environments. In: *IEEE congress on evolutionary computation*, pp 2130–2137
- Ruan G, Yu G, Zheng JH, Zou J, Yang SX (2017) The effect of diversity maintenance on prediction in dynamic multi-objective optimization. *Appl Soft Comput* 58:631–647
- Yang SX (2008) Genetic algorithms with memory-and elitism-based immigrants in dynamic environments. *Evol Comput* 16(3):385–416
- Branke J (1999) Memory enhanced evolutionary algorithms for changing optimization problems. In: *Proceedings of the 1999 congress on evolutionary computation-CEC99*, pp 1875–1882
- Goh CK, Tan KC (2009) A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. *IEEE Trans Evol Comput* 13(1):103–127
- Liang ZP, Zheng SX, Zhu ZX, Yang SX (2019) Hybrid of memory and prediction strategies for dynamic multiobjective optimization. *Inf Sci* 485:200–218
- Peng Z, Zheng JH, Zou J, Liu M (2015) Novel prediction and memory strategies for dynamic multiobjective optimization. *Soft Comput* 19(9):2633–2653
- Wang Y, Li B (2009) Investigation of memory-based multi-objective optimization evolutionary algorithm in dynamic environment. In: *IEEE congress on evolutionary computation*, pp 630–637
- Shang RH, Jiao LC, Ren YJ, Li L, Wang LP (2014) Quantum immune clonal coevolutionary algorithm for dynamic multiobjective optimization. *Soft Comput* 18(4):743–756
- Liu RC, Li JX, Fan J, Mu CH, Jiao LC (2017) A coevolutionary technique based on multi-swarm particle swarm optimization for dynamic multiobjective optimization. *Eur J Oper Res* 261(3):1028–1051
- Gong DW, Xu B, Zhang Y, Guo YN, Yang SX (2020) A similarity-based cooperative co-evolutionary algorithm for dynamic interval multiobjective optimization problems. *IEEE Trans Evol Comput* 24(1):142–156
- Xu B, Zhang Y, Gong DW, Wang L (2017) A parallel multi-objective cooperative co-evolutionary algorithm with changing variables. In: *Proceedings of the genetic and evolutionary computation conference companion*, pp 1888–1893
- Xu B, Zhang Y, Gong DW, Guo YN, Rong M (2018) Environment sensitivity-based cooperative co-evolutionary algorithms for dynamic multi-objective optimization. *IEEE/ACM Trans Comput Biol Bioinf* 15(6):1877–1890
- Hatzakis I, Wallace D (2006) Dynamic multi-objective optimization with evolutionary algorithms: a forward-looking approach. In: *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp 1201–1208
- Zhou AM, Jin YC, Zhang QF (2014) A population prediction strategy for evolutionary dynamic multiobjective optimization. *IEEE Trans Cybern* 44(1):40–53
- Wu Y, Jin YC, Liu XX (2015) A directed search strategy for evolutionary dynamic multiobjective optimization. *Soft Comput* 19(11):3221–3235
- Rong M, Gong DW, Pedrycz W, Wang L (2020) A multimodel prediction method for dynamic multiobjective evolutionary optimization. *IEEE Trans Evol Comput* 24(2):290–304
- Jain P (2007) Steffensen type methods for solving non-linear equations. *Appl Math Comput* 194(2):527–533
- Greeff M, Engelbrecht AP (2008) Solving dynamic multi-objective problems with vector evaluated particle swarm optimisation. In: *IEEE congress on evolutionary computation (IEEE world congress on computational intelligence)*, pp 2917–2924
- Richter H (2009) Detecting change in dynamic fitness landscapes. In: *IEEE congress on evolutionary computation*, pp 1613–1620
- Yang SX, Tinós R (2007) A hybrid immigrants scheme for genetic algorithms in dynamic environments. *Int J Autom Comput* 4(3):243–254
- Vavak F, Fogarty TC, Jukes K (1996) A genetic algorithm with variable range of local search for tracking changing environments. In: *International conference on parallel problem solving from nature*, pp 376–385
- Branke J, Kaussler T, Smidt C, Schmeck H (2000) A multipopulation approach to dynamic optimization problems. In: *Evolutionary design and manufacture*, pp 299–307
- Muruganantham A, Tan K, Vadakkepat P (2016) Evolutionary dynamic multiobjective optimization via Kalman filter prediction. *IEEE Trans Cybern* 46(12):2862–2873
- Jiang SY, Yang SX (2017) A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization. *IEEE Trans Evol Comput* 21(1):65–82
- Jiang M, Huang ZQ, Qiu LM, Huang WZ, Yen GG (2018) Transfer learning-based dynamic multiobjective optimization algorithms. *IEEE Trans Evol Comput* 22(4):501–514
- Li JX, Liu RC, Wang RN, Liu J, Mu CH (2019) A special points-based hybrid prediction strategy for dynamic multi-objective optimization. *IEEE Access* 7:62496–62510

40. Rong M, Gong DW, Zhang Y, Jin YC, Pedrycz W (2019) Multi-directional prediction approach for dynamic multiobjective optimization problems. *IEEE Trans Cybern* 49(9):3362–3374
41. Ortega JM, Rheinboldt WC (1970) Iterative solution of nonlinear equations in several variables. Society for Industrial and Applied Mathematics, Philadelphia
42. Conte SD, Boor CD (2017) Elementary numerical analysis: an algorithmic approach. Society for Industrial and Applied Mathematics, Philadelphia
43. Farina M, Deb K, Amato P (2004) Dynamic multiobjective optimization problems: test cases, approximations, and applications. *IEEE Trans Evol Comput* 8(5):425–442
44. Das I, Dennis JE (1998) Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM J Optim* 8(3):631–657
45. Zhang QF, Zhou AM, Jin YC (2008) RM-MEDA: a regularity model based multiobjective estimation of distribution algorithm. *IEEE Trans Evol Comput* 12(1):41–63
46. Camara M, Ortega J, Toro FD (2009) Performance measures for dynamic multi-objective optimization. In: International work-conference on artificial neural networks, pp 760–767
47. Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Trans Evol Comput* 3(4):257–271
48. Demsar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7:1–30
49. Zhang Y, Gong DW, Sun JY, Qu BY (2018) A decomposition-based archiving approach for multi-objective evolutionary optimization. *Inf Sci* 430:397–413

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