Solving dynamic multi-objective problems with an evolutionary multi-directional search approach

Yaru Hu[a], Junwei Ou[b,*], Jinhua Zheng[b,c], Juan Zou[b], Shengxiang Yang[d], Gan Ruan[e]

[a]*Department of Mathematics and Computational Science, Xiangtan University, Xiangtan, Hunan, 411105, China*
[b]*Key Laboratory of Intelligent Computing and Information Processing (Ministry of Education), Xiangtan University, Xiangtan, Hunan, 411105, China*
[c]*Hunan Provincial Key Laboratory of Intelligent Information Processing and Application, Hengyang Normal University, Hengyang, 421002, China*
[d]*School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K.*
[e]*School of Computer Science, University of Birmingham, Birmingham B15 2TT, U.K.*

Abstract

The challenge of solving dynamic multi-objective optimization problems is to effectively and efficiently trace the varying Pareto optimal front and/or Pareto optimal set. To this end, this paper proposes a multi-direction search strategy, aimed at finding the dynamic Pareto optimal front and/or Pareto optimal set as quickly and accurately as possible before the next environmental change occurs. The proposed method adopts a multi-directional search approach which mainly includes two parts: an improved local search and a global search. The first part uses individuals from the current population to produce solutions along each decision variable's direction within a certain range and updates the population using the generated solutions. As a result, the first strategy enhances the convergence of the population. In part two, individuals are generated in a specific random method along every dimension's orientation in the decision variable space, so as to achieve good diversity as well as guarantee the avoidance of local optimal solutions. The proposed algorithm is measured on several bench-

*Corresponding author

Email addresses: huyaru1199@gmail.com (Yaru Hu), junweiou@163.com (Junwei Ou), jhzheng@xtu.edu.cn (Jinhua Zheng), zoujuan@xtu.edu.cn (Juan Zou), syang@dmu.ac.uk (Shengxiang Yang), GXR847@cs.bham.ac.uk (Gan Ruan)

mark test suites with various dynamic characteristics and different difficulties. Experimental results show that this algorithm is very competitive in dealing with dynamic multi-objective optimization problems when compared with four state-of-the-art approaches.

Keywords: Dynamic, Dynamic multi-objective optimization, Local search, multi-directional search strategy

1. Introduction

Dynamic multi-objective optimization problems (DMOPs) have conflicting objectives and criteria which may vary with time, such as objective functions, constraints and/or related problem parameters [1]. The main difficulty in solving DMOPs is balancing the convergence and diversity during the optimization process in a dynamic environment. Consequently, evolutionary algorithms (EAs) [2] have been widely applied to address DMOPs. EAs used to solve DMOPS are called dynamic multi-objective evolutionary algorithms (DMOEAs), and they have been used in problems such as scheduling [3], management [4, 5], control [6], distribution feeder reconfiguration [7] and network routing [2].

Over the past decade, researchers have solved DMOPs using static multi-objective evolutionary algorithms (MOEAs). However, the shortcoming of this approach is the lack of diversity, which can prevent the population from converging towards the POF. In other words, traditional MOEAs have low-speed convergence to the POF when an environmental change occurs. The main reason is that the diversity of the whole population is inadequate, which may lead the population to get stuck in local optimum. Facing this limitation, some researchers have proposed improvements to enhance the performance of EAs based on static MOEAs, so as to make them adapt to the dynamic environment. These improved methods include diversity approaches [8][9], prediction approaches [1][10], multiple population [11][12] and memory approaches[13][14].

Although considerable research has been done in the dynamic optimization field, there are still many drawbacks that require more response strategies to improve the diversity and the convergence of the DMOEAs. For example, some methods based on prediction may misguide the population away from the true POF in cases when the prediction is inaccurate. The memory strategy loses its advantage in the beginning of the optimization, especially in dealing with non-periodic problems.

Considering the characteristics of DMOPs and the shortcomings of present methods, this paper proposes a new DMOEA, which includes a multi-directional search strategy (MSS) and a regularity model-based multi-objective estimation of distribution algorithm(RM-MEDA) [15]. MSS can search the variable space from multiple directions to update outdated solutions, reducing the influence of the environmental change and improving the performance of the algorithm. When the environment changes, MSS is utilized to explore in multiple directions and discover new optimal solutions, so as to enhance the diversity of the population and update the population using solutions which have been found. The main contributions of this research are as follows.

1. In an improved local search, each individual in the population is applied to produce a solution along each dimension's direction in the decision space within a specific range. The current population's solution is replaced if the generated individual is non-dominated. Thus, the population is updated through the search toward multiple directions, which can improve the convergence of the population toward the new POF when the environment varies.
2. The global search sets up a mutation probability for each dimension in the decision space first. Then a solution is generated using a mutation operator in each dimension of the decision space based on every individual of the population so as to update the population. This increases the diversity of the population so that it does not get stuck in the local optimum.
3. When MSS is combined with RM-MEDA [15], MSS can achieve a balance between convergence and diversity as well as a fast response to the

environmental change when the environment changes during the process of optimization.

The rest of this paper is structured as follows. Background is presented in Section 2. Section 3 presents the proposed algorithms. Section 4 presents the experimental results and analysis. Section 5 presents the influence of the different components of MSS. Lastly, conclusions are summarized in Section 6.

2. Background

2.1. Dynamic multi-objective optimization

Several kinds of methods define DMOPs in the dynamic multi-objective optimization community according to the features of dynamisms[16]. The following definition of DMOPs is considered in this paper:

$$\begin{cases} \min \mathbf{F}(\mathbf{x},\mathbf{t}) = \{f_1(\mathbf{x},t), f_2(\mathbf{x},t), \dots f_M(\mathbf{x},t)\}, \\ s.t. x \in \Omega, \end{cases} \tag{1}$$

where t is the time step; M represents the objective number; $x = (x_1, x_2, \cdots, x_n)$ is the n-dimension decision vector within the decision space Ω; and $\mathbf{F}(\mathbf{x}, t)$ is the M-dimension objective function vector to be minimized at time t.

Definition 1. *Pareto Dominance*: Given two individuals p and q in the population, p is said to dominate q, written as $f(p) \prec f(q)$ if $f_i(p) \leq f_i(q)$ $\forall i \in 1, 2, ..., m$ and $f_j(p) < f_j(q)$ $\exists j \in 1, 2, ..., m$.

Definition 2. *Pareto Optimal Set (POS)*: A solution is defined to be a nondominated solution if it is not dominated by any other solutions in ϕ. Thus, the POS [1] is the set of all nondominated solutions and can be defined mathematically as follows:

$$POS := \{x \in \phi | \neg \exists x^* \in \phi, F(x^*) \prec F(x)\}. \tag{2}$$

Definition 3. *Pareto Optimal Front (POF)*: The POF is the set of all non-dominated solutions in the objective space and can be defined mathematically as follows:

$$POF := \{y = F(x) | x \in POS\}. \quad (3)$$

Considering the varying forms of the POS and POF, DMOPs generally are classified into four different types [1, 17]:

- *Type I*: The POS changes with time but the POF is fixed.
- *Type II*: Both the POS and POF change with time.
- *Type III*: The POS remains fixed, while the POF changes with time.
- *Type IV*: Both the POS and POF remain fixed.

All of these exist in real-world optimization problems. In this paper, we mainly take into consideration the first three types.

2.2. Related Work

In order to further improve the performance of DMOEAs, researchers have studied DMOPs. Existing DMOEAs can be classified into the following categories [10][12] based on how they deal with DMOPs.

a. *Diversity Enhancement* : The main purpose of diversity enhancement is to increase the population diversity and to help an algorithm jump out of current optimum when an environmental change occurs. Therefore, a good technology of diversity enhancement can propel an algorithm quickly to track the varying POF and/or POS. For example, NSGA-II [18] was extended to dynamic NSGA-II(DNSGA-II)[3] by either generating random solutions or developing some mutated solutions. However, too much diversity may result in stagnation [19].

b. *Memory Mechanism* : The memory strategy mainly utilizes some good solutions of the historical population to accelerate convergence whenever the environment changes. For example, Peng et al. proposed novel prediction and memory strategies (PMS) [13]. These kinds of strategies perform well on periodic problems, while they are less effective for enhancing convergence in non-periodic problems.

c. *Prediction Strategy* : In the prediction strategy, one specific prediction model is established first, and then solutions are generated by using the prediction model after each environmental change, in which the population is guided to evolve towards the POF. For instance, the prediction methods based on forecasting models include the Kalman Filter model [20], the inverse modeling approach [21] and the autoregressive model [1]. However, the prediction strategy in DMOEAs may misguide the convergence of the population during the early period of searching if the prediction is inaccurate.

d. *Multipopulation* : The multipopulation strategy uses more than two populations to result in good performance. The strategy can attain the goal of balancing convergence and diversity during the optimization process. For example, Chen [12] proposed a dynamic two-archive EA (DTAEA) to handle DMOPs. The two populations are complementary to each other to obtain a good approximation to the POF.

Search methods are effective [22] in solving single-objective optimization problems. Therefore, there is interest in applying them to MOEAs. Particularly, the search methods in MOEAs date back to 1996 according to the best of the authors knowledge, like multi-objective genetic local search algorithm [23] is proposed by Ishibuchi et. al.. In addition, search methods have been applied to real life. For example, they are used in scheduling [24, 25], the performance of cellular [26] and power flow problems [22, 27]. Due to the advantage of search methods [25], search methods have been applied to constrained and unconstrained problems [28].

Aside from the aforementioned approaches, other methods exist to improve the convergence rate and enhance diversity [29, 30]. For instance, Wu et al. proposed a directed search strategy (DSS) [31]. There are two main mechanisms in DSS, which are denoted by DSS1 and DSS2. The DSS1 aims to apply the predicted moving direction of the center points and the orthogonal direction of the center points, and the DSS2 is utilized to produce some solutions in the predicted region of the next generation's POS, so as to improve the convergence speed of the population. Whenever there is an environmental change, DSS1 is conducted to achieve good outputs in terms of convergence and diversity. However, DSS2 is carried out at the end of each generation. Chai et al. put forward an extended evolutionary algorithm [32], which differs from the search approaches. This evolutionary algorithm extended the original NSGA-III algorithm by embedding a discretization scheme for handling system dynamics and producing optimal trajectories in the multi-objective spacecraft optimal control problem.

3. Proposed algorithm

In this section, the proposed MSS for solving DMOPs is presented in detail. The overall framework of MSS is illustrated, which mainly includes three steps. In the first step, based on each decision variable's direction, some solutions of the population are applied to search for better individuals and update the current population, and then some better produced individuals are used to update the external population. The new population is expected to approach the new POF. Algorithm 1 provides the implementation of the first step. On the basis of the second step, an individual in the population is used to stochastically generate a solution with a mutation method in each decision's dimension, and then it is applied to renew the external population. In the third step, the population is selected and obtained using the non-dominated sort [3] on the population which is formed from combining the external population with the population acquired in step 2. Thus, MSS can achieve good diversity and fast convergence towards the new POF when there is an environmental change. The overall framework of MSS is illustrated in Algorithm 4.

3.1. Local search strategy based on multi-direction

In this section, the local search strategy based on multi-direction is used to find more individuals with good convergence and diversity, enhancing the performance in terms of convergence and the speed of response to the environmental change.

In the local search strategy, every solution of the population is applied to search for the best individual and update the current population. Suppose C_t is the center of POS_t, and POS_t is the non-dominated solution obtained at the end of time t. Then C_t can be calculated by the following formula:

$$C_t = \frac{1}{|POS_t|} \sum_{x_t \in POS_t} x_t, \tag{4}$$

where $|POS_t|$ represents the number of non-domination sets at time t, and x_t is the non-dominated individual at time t. Thus, the moving direction of center points referred to as D_t at time t can be expressed as follows:

$$D_t = C_t - C_{t-1}, \tag{5}$$

where C_t and C_{t-1} are the center points of the decision space of the non-dominated solutions at the end of time t and $t-1$, respectively. Suppose that $D_t = (D_t^1, D_t^2, \cdots, D_t^n)$, is set as

$$d_t = (d_t^1, d_t^2, \cdots, d_t^n), \tag{6}$$

where $d_t^i = |D_t^i|$ represents the absolute value of D_t^i, $i = 0, 1, \cdots, n$.

Next, the direction of local search in each dimension is randomly generated. D_t' represents local search direction vector at time t. Suppose that D_t' is presented as follows:

$$D_t' = (a_1, a_2, \cdots, a_n), \tag{7}$$

where a_i is randomly generated using the following formula:

$$a_i = \begin{cases} 1, & random(0,1) \leq 0.5; \\ -1, & random(0,1) > 0.5; \end{cases} \tag{8}$$

Then, some individuals $x_t = (x_1, x_2, \cdots, x_n)$ in population P_t are used to generate an individual $y_i = (y_1, y_2, \cdots, y_n)$ along one direction of each dimension in the decision space. The $random(0, 1)$ is a random function and returns a random value between 0 and 1. The i-th element of y_i is calculated according to the following equation:

$$y_i = x_i + a_i * |N(0, d_i)|, \tag{9}$$

where x_i is the i-th dimension variable of x_t in P_t; d_i is the i-th element of d_t; $N(0, d_i)$ represents a normally distributed random number with mean zero and standard deviation d_i; a_i denotes the i-th dimension variable of local search direction vector D_t.

Similarly, some individuals x_t in population P_t can be applied to produce an individual y_i along another direction of each dimension in the decision space, which is illustrated in the following formula:

$$y_i = x_i - a_i * |N(0, d_i)|, \tag{10}$$

where the parameters are similar to all parameters in (9). The produced solution y_i is evaluated, and the external population is updated using y_i. If y_i is not dominated by x_i, x_i is replaced by y_i; otherwise, equation (10) is used to produce a solution y_i and the external population is renewed by y_i. If y_i is not dominated by x_i, x_i is replaced by y_i; otherwise, the next dimension of the decision space is searched with the same approach.

In Algorithm 1, n is the size of the decision space; P_t is the population at time t; N is the population size, and C_{t-1} is the center point of POS_{t-1} at time $t - 1$. In step 2 and step 17 of Algorithm 1, the final population after the environmental change is selected. Bearing this in mind, 20% of the individuals in the population are selected by the crowded-comparison operator [18] and saved in P_{local} to trace the varying POF or POS. $|P_{local}|$ is the size of the set P_{local} which is smaller than the population size. For each dimension of each individual x_j in P_{local} in the decision space, the search process is carried out along the selected dimension in order to find a better solution and replace it,

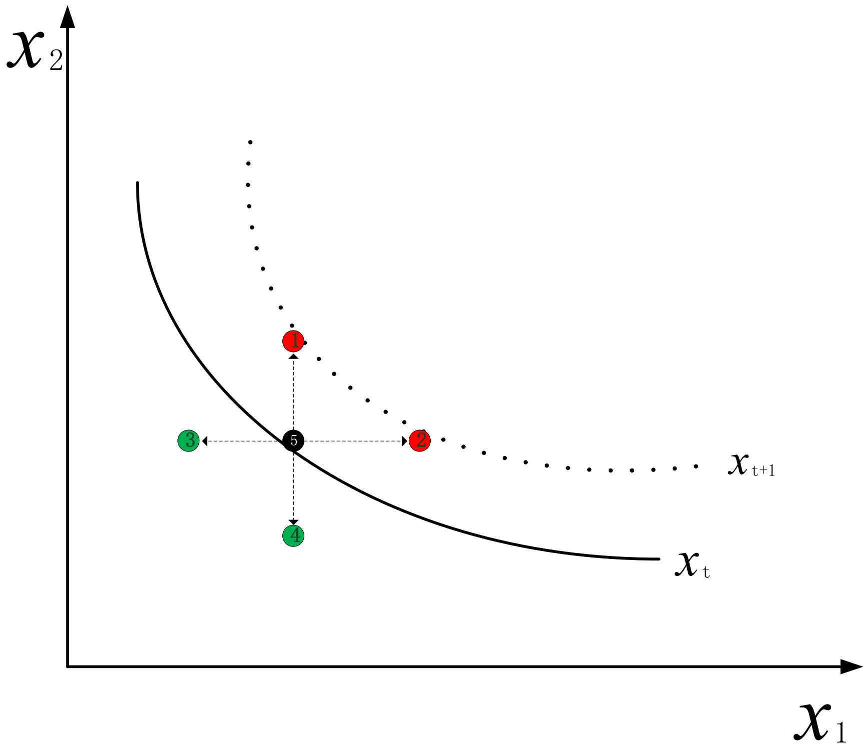


Figure 1: The local search in the decision space. Note that $\mathbf{x}^i$ is denoted as its index i for short.

which is presented from step 4 to step 15 in Algorithm 1. At the same time, every newly produced individual will be used to update the external population, the details of which are shown in Algorithm 3.

A brief search process is illustrated in Figure 1. Take the 2-dimensional search space in the decision space as an example. Assume that the black circle indicates a solution in P_{local} and the green and red circles indicate the solutions generated by a local search strategy. The solution $\mathbf{x}^5$ in P_{local} is first used to search along the first dimension in the decision space. If the solution $\mathbf{x}^2$ is dominated by $\mathbf{x}^5$ (on the contrary, $\mathbf{x}^5$ is replaced by $\mathbf{x}^2$ and $\mathbf{x}^3$ will not be produced), $\mathbf{x}^3$ will be produced and compared to $\mathbf{x}^5$ using the same comparison

Algorithm 1 The pseudo-code of the local search methods

Input: P_t, C_{t-1}, n (the size of the decision space);

Output: P_{local};

1: Calculate C_t, d_t and D'_t according to formulas (4) (6) and (7). P_{local} =crowded-comparison operator(P_t) [18];

2: **for** $j = 1, \cdots, |P_{local}|$ **do**

3:     **for** $i = 1, \cdots, n$ **do**

4:         Generate a new solution y according to formula (9), and conduct boundary check and the evaluation on y.

5:         Individual y renews the external population, which is illustrated in Algorithm 3.

6:         **if** y is not dominated by x_j **then**

7:             $x_j = y$, go to 3;

8:         **else**

9:             go to 11;

10:        **end if**

11:        Produce another individual y according to formula (10); and conduct boundary check and the evaluation on y.

12:        Carry out the same process on y with step 5.

13:        **if** y is not dominated by x_j **then**

14:            $x_j = y$.

15:        **end if**

16:    **end for**

17: **end for**

18: **return** P_{local}

method, which is illustrated in Figure 1. Then, $\mathbf{x}^5$ is applied to search along the second dimension in the decision space using the same method.

3.2. *Global search strategy using mutation*

In the global search strategy, the mutation operation is carried out in each decision variable's direction, achieving the goal of maintaining good diversity and avoiding getting stuck in the local optimum. The detailed steps are shown in Algorithm 2. The individual of the $P_{mutation}$ can be randomly selected from the population P_t. Every individual $x_j = (x_1, x_2, \cdots, x_n)$ in population $P_{mutation}$ is used to generate an individual $k = (k_1, k_2, \cdots, k_n)$ at each dimension in the decision space. This is seen as a mutation method. The i-th element of k_i is calculated according to the following equation:

$$k_i = x_i + a_i * N(0, d_i), \tag{11}$$

where x_i is the i-th dimension variable of x_j in P_t; a_i is calculated according to equation (8); and d_i is the i-th element of d_t which is shown in (6).

Algorithm 2 describes each step of the global search strategy in detail. $random(0, 1)$ is a random function and returns a random value between 0 and 1; $|P_{mutation}|$ is smaller than the population size N; N is the population size and the mutation probability is set as r. In step 1 of Algorithm 2, $0 < w < 1$ is a ratio. Therefore, we set $w = 0.05$ according to [11]. Note that, the $|P_{mutation}|$ is set to $|P_{mutation}| = \lceil w * N \rceil$ when $w * N$ is a non-integer value; within $\lceil . \rceil$ means integer up. k is the individual produced by the mutation operation. The generated individual k is first applied to update the external population. If $k \prec x_i, x_i$ is replaced by k. Lastly, the population is returned.

3.3. *Update of the external population*

In Algorithm 3, $|S_t|$ is the size of S_t; in step 1 to step 3 of Algorithm 3, the individual y is added directly to S_t when the size of the external population is empty; if y is not dominated by any individual in S_t then y is added to S_t and the other individuals dominated by y are deleted, which is presented from step 4 to step 6 in Algorithm 3.

Newly generated individuals in these strategies need to be boundary checked in order to check whether the solutions are within the boundaries of the given

Algorithm 2 The pseudo-code of the global search strategy

Input: P_t, N, C_{t-1};

Output: $P_{mutation}$;

1: The center point of POS_t C_t is obtained by the local search method. Copy the w*N individuals are selected from P_t to $P_{mutation}$.
2: **for** $j = 1, \cdots, |P_{mutation}|$ **do**
3:    **for** $i = 1, \cdots, n$ **do**
4:       **if** random (0, 1) <r **then**
5:          the i-th element of x_j is used to calculate the i-th element of the produced solution according to formula (11).
6:       **end if**
7:    **end for**
8:    The generated solution k is composed by each element produced in step 4. Then boundary check and the evaluation for k are carried out.
9:    Solution k is used to update the external population using Algorithm 3.
10:    **if** $k \prec x_j$ **then**
11:       $x_j = k$
12:    **end if**
13: **end for**
14: **return** $P_{mutation}$

decision space. The detailed procedure of the boundary check is as follows:

$$y_i = \begin{cases} x_i & \text{if } low_i \leq x_i \leq upper_i \\ random(low_i, 0.5(low_i + upper_i)) & \text{if } x_i < low_i \\ random(0.5(low_i + upper_i), upper_i) & \text{if } x_i > upper_i \end{cases} \quad (12)$$

where $i = 1, ..., n$. n is the dimension of the test problems' decision space; $random(x, y)$ is a random function and returns a random value between x and y; low_i is the minimum boundary of the i-th dimension, and $upper_i$ is the maximum boundary of the i-th dimension in the decision space.

Algorithm 3 The pseudo-code for updating the external population

Input: S_t (the external population), y;

Output: S_t;

1: **if** $|S_t| = 0$ **then**
2: y is added to S_t;
3: **end if**
4: **if** y is not dominated by any individual in S_t **then**
5: Add y to S_t and delete other individuals dominated by y.
6: **end if**
7: **return** S_t

Algorithm 4 A multi-directional search strategy

Input: P_t(current population), S_t(external population), τ_t(frequency of change), n_t(severity of changes), N(population size).

Output: P_{t+1}(updated population).

1: Initialize a population named as P_t, set time period $t = 0$, set iteration generation $genIter = 0$.
2: Detect the environmental change, if there is no environmental change, go to 7; else select the non-dominated solutions and calculate the center C_t of non-dominated solution according to formula 4 and calculate d_t according to formula 6, set $t = t + 1$.
3: The P_{local} is obtained by Algorithm 1.
4: The $P_{mutation}$ is obtained by Algorithm 2.
5: Set current population $P_t = P_t \cup S_t$.
6: Sort $P_{archive}$ using non-dominated sort [3] and select N individuals. The iteration is set $genIter = genIter + \lceil 20\% * n \rceil$.
7: Optimize the population with RM-MEDA [15].
8: If the termination is satisfied, then output the P_t; else set $genIter = genIter + 1$, go to 2.

3.4. Overall framework of the proposed algorithm

The initialization of the algorithm is shown in step 1 of Algorithm 4. MSS starts with an initial population P and the initial parameters are set. In step 2, if an environmental change is detected [6], then the response mechanism from step 3 to step 6 is carried out. Step 5 of Algorithm 4 is to integrate the external population and the obtained mutation population from step 3 and step 4 into P_t. Then, the non-dominated sort [3] is carried out on P_t in step 6, and N solutions are obtained and reserved in P_{t+1}. If there is no environmental change, then step 7 is carried out, where the population is optimized with RM-MEDA [1]. In step 6, the iteration generation genIter is set $genIter = genIter + \lceil 20\% * n \rceil$ since the MSS consumes a certain number of evaluations. It is fair for other algorithms to obtain convinced results.

3.5. Computational complexity analysis

This paper adopts the same optimization algorithm RM-MEDA or NSGA-II/DE in different response strategies as the comparison experiments. The difference of computational complexity is mainly reflected in different response strategies. The computational complexity is analyzed as follows:

(1) The computational complexity of MSS: The response strategy proposed in this paper consists of two parts: the local search method and the mutation method. The time complexity of the local search method is $O(nN)$. The computational complexity of the mutation method is $O(nN)$. Consequently, the overall costs of the computational complexity of the MSS is $O(nN)$.

(2) The computational complexity of EGS: The computational complexity of prediction strategy is $O(MN^2)$. The truncation operator is used when the mutation method is performed, so the mutation costs $O(N^2 logN)$ computations. Therefore, the overall computational complexity of EGS is $O(N^2 logN)$.

(3) The computational complexity of PPS [1]: The computational complexity of the center point prediction in PPS is $O(nN)$; the manifold prediction costs $O(nN^2)$ computational complexity, so the two parts take $O(nN^2)$

computational complexity. Therefore, the computational complexity of PPS is $O(nN^2)$ in one generation of the response mechanism.

(4) The computational complexity of DMS: The computational complexity of DMS has mainly three parts: the prediction method, the gradual search strategy and random diversity maintenance. The first one costs $O(N)$ computational resources. The second and third take $O(N)$ computational complexity. The individual selection procedure spends $O(NlogN)$ computations on elitist preservation in DMS. Therefore, the overall computational complexity of DMS for the response to an environmental change is $O(NlogN)$.

(5) The computational complexity of DSS: The computational cost of response mechanism (DSS1) in DSS is mainly produced by the prediction of the POS moving direction and the directed local search [31]. Both need $O(nN/2)$. The computational complexity of DSS2 in each generation costs $O(nN)$. For a better calculation and comparison, the extra computational complexity cost of DSS spent on the optimization algorithm is added to the response mechanism. As a result, the total computational complexity of DSS is O(nN)+O(τ_t*nN), where τ_t is the frequency of environmental change.

Based on this analysis, we can conclude that MSS requires less computational complexity than some comparison strategies, implying MSS is more efficient compared with other methods in solving DMOPs.

4. Experiments

4.1. Test instances and performance indicators

In order to evaluate the performance of the proposed algorithm, we adopt a variety of benchmark test problems with different characteristics. The selected test instances include the FDA test suite [33], DMOP test suite [33], and the F5-F10 test suite [11], respectively. The two test suites have linear correlations between decision variables. In addition, the DMOP test suite is an upgraded version of the FDA test suite. Many optimization problems are no longer just linear, but include more complex characteristics; thus, the F5-F10 test suite proposed by Zhou et al [15] has nonlinear correlation between decision variables. To display the advantages and the disadvantages of the presented algorithm, performance metrics in dynamic optimization are used to test the performance of the algorithm in terms of convergence, distribution and diversity. These performance metrics can be found in (Section 1 of the supplementary file).

4.2. Parameter settings

In PPS [34], EGS [35], DSS [31], DMS [8] and MSS, which are described in Section 2.1 of the supplementary file, the common parameters of the five tactics are set. These are integrated into the framework of DMOEAs where RM-MEDA is indicated as RM, and NSGA-II/DE [31] is denoted as NSDE. The severity of environmental change was set to $N_t = 10$ and 120 environmental changes were tracked for all strategies. The frequency of environmental change was $\tau_t = 20$, $\tau_t = 25$ and $\tau_t = 30$, respectively. As for change detection, 5% of the population were randomly selected and reevaluated to detect the environmental change at every generation. The dimension of the decision space was $n = 20$. The population size was set to $N = 100$, and the algorithm ran 20 times independently for all test problems. The parameters in RM-MEDA were the same as the original paper [15]. The crossover and mutation possibility in NSDE were 1 and 0.05, respectively. The control parameter in the DE operator was set to 0.5. The parameters of the compared algorithms were referenced from their original papers. Note that the parameters of MSS were set as follows: the size of the external population (S_t) was N, and the mutation rate at each dimension of individual was $r = 0.25$ in the global search method. Since the proposed MSS in this paper needs to consume a certain number of evaluations in the prediction, to be fair, the algorithm iterations require removing the number of evaluations consumed at every environmental change and reducing the corresponding number of iterations. Therefore, set $genIter = genIter + \lceil 20\% * n \rceil$ when the environment changes. $\lceil 20\% * n \rceil$ stands for an integer up.

4.3. Experimental results

In order to study the effect of change frequency on the algorithm in dynamic environments, the severity of change is set to a fixed value of 10, and the frequency of change is set to 20, 25 and 30, respectively. Then the statistical results are carried out by means of the Wilcoxon ranksum test [36], whose significance level is set to 0.05. First, the five algorithms being compared are integrated into the framework of DMOEA where the optimization algorithm is RM. The experimental results including the mean and standard deviations values of SP, MS and GD are respectively shown in Table 1 and Tables 1, 2 of the supplementary file. Secondly, as shown in Table 2, two optimization algorithms, namely RM and NSDE, are combined into DMOEA and the statistical results about IGD are presented. Additionally, the selected dynamic multi-objective optimization strategies independently run 20 times to ensure the experiments are fair, and the best values obtained by the five algorithms are highlighted in bold face.

4.3.1. Results on FDA and DMOP Problems

From Table 1 of the supplementary file, we can conclude that MSS achieved the best results in most cases. As a result, MSS obtained a better distribution in most selected FDA and DMOP problems than the other four algorithms. However, MSS was slightly inferior to PPS and EGS on FDA2, but it was superior to DMS and DSS on this test. In general, PPS, EGS and DMS did not show encouraging results on the SP indicator. DSS seems to strive to maintain the uniform distribution of dynamically optimized POFs, since the SP values of DSS are the largest in most cases as seen in Table 1. For problem DMOP1, MSS presents worse results than PPS, EGS and DMS when the change severity is relatively low (i.e., $\tau_t = 20$ and 25). The main reason is that DMOP1 is a dynamic test problem whose POF changes over time while the POS remains fixed. PPS, EGS and DMS have the best even distribution as the severity of change increases from 20 to 25 and to 30.

As shown in Table 2 of the supplementary file, MSS has the best value of the MS metric in most FDA and DMOP instances, implying that the obtained solutions set by MSS covers the POF more extensively than the other strategies in most cases, although MSS was weaker than others on the FDA2. It should also be noted that MSS is a litte worse than EGS on FDA4 when the severity of change is 30. Except for these two cases, MSS exhibits an overall most extensive distribution among FDA and DMOP problems.

Table 1: Mean and SD of GD indicator obtained by five algorithms.

Prob.	τ_t	PPS	EGS	DMS	DSS	MSS
FDA1	20	1.172e-1(4.692e-2)‡	4.163e-2(9.080e-3)‡	1.244e-2(1.014e-3)‡	2.827e-2(1.509e-3)‡	**5.678e-3(4.996e-4)**
	25	9.351e-2(4.571e-2)‡	2.454e-2(5.710e-3)‡	8.562e-3(4.537e-4)‡	1.527e-2(8.057e-4)‡	**4.582e-3(1.992e-4)**
	30	5.799e-2(2.920e-2)‡	1.445e-2(1.991e-3)‡	6.872e-3(5.993e-4)‡	1.110e-2(4.496e-4)‡	**4.145e-3(1.469e-4)**
FDA2	20	1.435e-2(9.453e-4)‡	1.416e-2(4.714e-4)‡	1.693e-2(5.452e-4)‡	6.997e-2(2.937e-3)‡	**1.157e-2(1.330e-4)**
	25	1.380e-2(5.358e-4)‡	1.434e-2(1.901e-3)‡	1.650e-2(5.421e-4)‡	6.764e-2(1.219e-3)‡	**1.158e-2(1.170e-4)**
	30	1.396e-2(3.602e-4)‡	1.336e-2(1.050e-3)‡	1.608e-2(4.509e-4)‡	6.813e-2(1.725e-3)‡	**1.147e-2(1.837e-4)**
FDA3	20	2.497e-1(1.017e-1)‡	8.760e-2(1.542e-2)‡	**3.015e-2(2.658e-3)**	3.738e-2(2.315e-3)	8.564e-2(3.757e-3)
	25	1.829e-1(7.452e-2)‡	9.053e-2(7.004e-3)	**2.595e-2(9.290e-4)**	2.668e-2(1.041e-3)	9.775e-2(2.963e-3)
	30	1.807e-1(8.237e-2)‡	1.026e-1(4.901e-3)	**2.794e-2(8.046e-4)**	2.976e-2(6.765e-4)	1.058e-1(3.629e-3)
FDA4	20	1.870e-1(4.535e-3)‡	2.220e-1(1.332e-2)‡	1.690e-1(6.832e-3)‡	9.138e-2(1.242e-3)‡	**6.805e-2(3.684e-3)**
	25	1.632e-1(6.700e-3)‡	1.823e-1(9.575e-3)‡	1.496e-1(5.517e-3)‡	7.900e-2(2.916e-3)‡	**6.193e-2(2.855e-3)**
	30	1.473e-1(4.081e-3)‡	1.503e-1(7.620e-3)‡	1.329e-1(4.598e-3)‡	7.267e-2(1.008e-3)‡	**5.878e-2(4.051e-3)**
DMOP1	20	1.029e-1(1.854e-1)‡	1.307e-1(1.427e-2)‡	2.950e-2(1.106e-2)‡	2.316e-2(3.921e-3)‡	**2.153e-2(8.804e-3)**
	25	1.878e-2(7.058e-3)‡	1.508e-2(6.147e-3)‡	1.551e-2(5.872e-3)‡	7.790e-2(1.886e-3)‡	**1.012e-2(3.576e-3)**
	30	1.643e-1(2.109e-1)‡	1.101e-2(4.995e-3)‡	9.516e-3(2.619e-3)‡	6.301e-3(5.356e-4)‡	**5.763e-3(1.883e-3)**
DMOP2	20	1.885e-1(8.437e-2)‡	6.537e-2(2.472e-2)‡	1.661e-2(1.783e-3)‡	3.330e-2(2.405e-3)‡	**6.159e-3(4.153e-4)**
	25	1.385e-1(6.568e-2)‡	3.188e-2(8.126e-3)‡	1.044e-2(9.374e-4)‡	1.940e-2(1.292e-3)‡	**5.265e-3(2.325e-4)**
	30	1.222e-1(8.097e-2)‡	2.331e-2(6.264e-3)‡	8.207e-3(6.726e-4)‡	1.297e-2(7.126e-4)‡	**4.550e-3(1.757e-4)**
DMOP3	20	1.121e-1(7.061e-2)‡	4.430e-2(1.184e-2)‡	1.256e-2(1.165e-3)‡	2.740e-2(2.399e-3)‡	**5.844e-3(5.739e-4)**
	25	7.753e-2(3.855e-2)‡	2.263e-2(4.168e-3)‡	8.749e-3(5.662e-4)‡	1.633e-2(1.007e-3)‡	**4.697e-3(3.067e-4)**
	30	6.909e-2(3.132e-2)‡	1.448e-2(2.955e-3)‡	6.611e-3(2.997e-4)‡	1.147e-2(7.629e-4)‡	**4.067e-3(1.117e-4)**
F5	20	5.178e-1(2.915e-1)‡	7.496e-1(4.060e-1)‡	4.576e-2(1.391e-2)‡	3.584e-1(4.720e-2)‡	**2.099e-2(4.928e-3)**
	25	3.704e-1(2.661e-1)‡	4.625e-1(4.068e-1)‡	2.929e-2(6.558e-3)‡	2.105e-1(3.380e-2)‡	**1.646e-2(2.488e-3)**
	30	2.216e-1(2.089e-1)‡	3.716e-1(2.426e-1)‡	2.253e-2(6.032e-3)‡	1.318e-1(1.212e-2)‡	**1.464e-2(3.772e-3)**
F6	20	3.550e-1(7.154e-2)‡	3.373e-1(1.384e-1)‡	5.352e-2(2.251e-2)‡	1.250e-1(2.252e-2)‡	**3.189e-2(1.057e-2)**
	25	2.269e-1(1.087e-1)‡	2.228e-1(1.268e-1)‡	5.214e-2(5.046e-2)‡	6.894e-2(1.058e-2)‡	**3.515e-2(2.162e-2)**
	30	2.403e-1(1.037e-1)‡	1.857e-1(1.126e-1)‡	**3.449e-2(2.380e-2)**	4.325e-2(1.543e-2)‡	4.107e-2(2.105e-2)
F7	20	2.053e-1(1.620e-1)‡	1.102e-1(6.259e-2)‡	3.224e-2(1.340e-2)†	8.318e-2(1.136e-2)‡	**2.709e-2(2.006e-2)**
	25	1.480e-1(7.687e-2)‡	7.300e-2(3.545e-2)‡	2.776e-2(1.378e-2)†	5.560e-2(7.539e-3)‡	**2.646e-2(1.545e-2)**
	30	1.261e-1(7.495e-2)‡	7.331e-2(3.790e-2)‡	3.434e-2(3.392e-2)‡	3.364e-2(9.507e-3)‡	**1.822e-2(5.862e-3)**
F8	20	1.111e+0(4.644e-1)‡	8.442e-1(5.067e-1)‡	2.286e-1(1.330e-1)‡	3.638e-1(2.923e-2)‡	**1.439e-1(2.843e-2)**
	25	6.594e-1(2.941e-1)‡	4.605e-1(2.841e-1)‡	**1.220e-1(8.482e-2)**	2.272e-1(3.256e-2)‡	1.413e-1(3.318e-2)
	30	4.813e-1(2.929e-1)‡	3.873e-1(2.189e-1)‡	**5.978e-2(3.524e-2)**	1.502e-1(1.776e-2)‡	1.264e-1(4.925e-2)
F9	20	1.109e+0(2.460e-1)‡	9.566e-1(3.554e-1)‡	8.641e-1(4.087e-1)‡	3.149e-1(5.346e-2)‡	**4.387e-2(4.493e-3)**
	25	7.217e-1(1.474e-1)‡	5.987e-1(2.897e-1)‡	3.285e-1(2.553e-1)‡	2.686e-1(3.437e-2)‡	**3.929e-2(7.102e-3)**
	30	6.784e-1(1.992e-1)‡	3.284e-1(1.570e-1)‡	2.169e-1(1.911e-1)‡	1.346e-1(1.076e-2)‡	**3.431e-2(5.089e-3)**
F10	20	2.925e-1(2.440e-2)‡	2.635e-1(2.298e-2)‡	1.998e-1(1.065e-2)‡	1.463e-1(3.176e-3)‡	**7.729e-2(6.983e-3)**
	25	2.290e-1(1.496e-2)‡	2.050e-1(1.541e-2)‡	1.799e-1(8.975e-3)‡	1.198e-1(4.320e-3)‡	**6.829e-2(7.126e-3)**
	30	2.090e-1(1.035e-2)‡	1.670e-1(7.883e-3)‡	1.704e-1(8.313e-3)‡	1.078e-1(3.681e-3)‡	**6.286e-2(3.176e-3)**

‡ and † indicate MSS performed significantly better than and equivalently to the corresponding algorithm, respectively. N_t is set to 10.

As can be seen from Table 1, the convergence of MSS is better on most test problems than all other response strategies. However, for FDA3, the GD value of MSS is better than PPS and EGS when the severity of change is 20. Consequently, the conclusion can be made that the prediction in these strategies can benefit the convergence of the population to some extent when addressing problems having similar characteristics as FDA3.

The IGD is a comprehensive metric, so it can evaluate an algorithm's perfor-

Table 2: Mean and SD of IGD indicator obtained by five algorithms.

Prob.	EA+τ_t	PPS	EGS	DMS	DSS	MSS
FDA1	RM+20	8.081e-2(3.167e-2)‡	3.391e-2(6.533e-3)‡	1.207e-2(6.848e-4)‡	2.103e-2(8.289e-4)‡	**6.691e-3(2.054e-4)**
	RM+25	6.618e-2(3.253e-2)‡	2.089e-2(4.222e-3)‡	9.173e-3(3.372e-4)‡	1.350e-2(6.119e-4)‡	**6.115e-3(1.683e-4)**
	RM+30	4.168e-2(1.908e-2)‡	1.349e-2(1.405e-3)‡	7.885e-3(4.256e-4)‡	1.078e-2(1.951e-4)‡	**5.804e-3(2.010e-4)**
	NSDE+20	4.247e-2(9.240e-3)‡	2.101e-2(8.079e-4)‡	1.773e-2(7.800e-4)‡	2.000e-2(2.503e-4)‡	**1.129e-2(4.837e-4)**
	NSDE+25	2.492e-2(2.112e-3)‡	1.613e-2(8.567e-4)‡	1.410e-2(4.035e-4)‡	1.432e-2(3.285e-4)‡	**9.647e-3(4.770e-4)**
	NSDE+30	1.795e-2(1.205e-3)‡	1.309e-2(1.244e-3)‡	1.293e-2(1.267e-3)‡	1.092e-2(1.394e-4)‡	**9.815e-3(6.337e-4)**
FDA2	RM+20	7.573e-3(5.189e-4)‡	6.567e-3(2.016e-4)‡	9.694e-3(3.148e-4)‡	4.887e-2(3.028e-3)‡	**5.681e-3(8.010e-5)**
	RM+25	6.872e-3(2.302e-4)‡	6.560e-3(5.870e-4)‡	8.756e-3(2.266e-4)‡	4.488e-2(1.407e-3)‡	**5.653e-3(4.460e-5)**
	RM+30	6.683e-3(1.767e-4)‡	6.216e-3(3.280e-4)‡	8.124e-3(2.238e-4)‡	4.381e-2(2.763e-3)‡	**5.621e-3(5.702e-5)**
	NSDE+20	8.784e-3(2.240e-4)‡	6.292e-3(8.289e-5)‡	6.156e-3(4.377e-4)‡	3.650e-2(4.269e-4)‡	**5.755e-3(2.561e-5)**
	NSDE+25	7.534e-3(2.497e-4)‡	6.156e-3(1.791e-5)‡	5.904e-3(5.159e-4)‡	3.539e-2(8.512e-5)‡	**5.714e-3(7.525e-5)**
	NSDE+30	6.955e-3(2.596e-5)‡	6.010e-3(3.543e-5)‡	**5.526e-3(8.467e-5)**	3.464e-2(1.339e-4)‡	5.638e-3(6.410e-5)
FDA3	RM+20	1.724e-1(8.969e-2)‡	2.152e-2(9.171e-3)‡	1.215e-2(2.017e-3)‡	1.350e-2(3.341e-4)‡	**6.499e-3(1.946e-4)**
	RM+25	1.119e-1(6.476e-2)‡	1.264e-2(2.307e-3)‡	8.470e-3(7.577e-4)‡	9.421e-3(4.781e-4)‡	**6.107e-3(2.730e-4)**
	RM+30	1.098e-1(7.433e-2)‡	1.040e-2(1.718e-3)‡	7.438e-3(6.713e-4)‡	7.918e-3(5.938e-4)‡	**5.768e-3(1.521e-4)**
	NSDE+20	1.368e-1(4.399e-2)‡	2.126e-2(1.673e-3)‡	2.067e-2(1.313e-3)‡	1.813e-2(3.227e-4)‡	**1.024e-2(5.910e-4)**
	NSDE+25	8.930e-2(2.053e-2)‡	1.873e-2(1.155e-3)‡	1.586e-2(2.139e-3)‡	1.326e-2(5.207e-4)‡	**9.595e-3(8.921e-5)**
	NSDE+30	5.253e-2(1.532e-2)‡	1.696e-2(9.879e-4)‡	1.380e-2(9.924e-4)‡	1.029e-2(3.480e-4)‡	**8.891e-3(1.272e-4)**
FDA4	RM+20	1.481e-1(3.134e-3)‡	1.3353e-1(4.093e-3)‡	1.397e-1(4.022e-3)‡	9.607e-2(6.241e-4)‡	**8.822e-2(1.666e-3)**
	RM+25	1.335e-1(3.507e-3)‡	1.2083e-1(2.871e-3)‡	1.274e-1(2.654e-3)‡	9.078e-2(1.057e-3)‡	**8.495e-2(1.729e-3)**
	RM+30	1.245e-1(1.963e-3)‡	1.1130e-1(2.670e-3)‡	1.182e-1(2.271e-3)‡	8.845e-2(2.717e-4)‡	**8.264e-2(2.276e-3)**
	NSDE+20	2.158e-1(2.953e-3)‡	1.365e-1(4.825e-3)‡	1.683e-1(4.732e-3)‡	1.038e-1(1.633e-3)‡	**9.424e-2(2.234e-3)**
	NSDE+25	1.996e-1(2.253e-3)‡	1.325e-1(3.127e-3)‡	1.597e-1(8.142e-3)‡	9.974e-2(8.593e-4)‡	**9.220e-2(2.023e-3)**
	NSDE+30	1.830e-1(2.421e-3)‡	1.265e-1(2.619e-3)‡	1.521e-1(4.018e-3)‡	9.617e-2(4.786e-4)‡	**9.170e-2(1.733e-3)**
DMOP1	RM+20	8.859e-2(1.612e-1)‡	1.4251e-1(5.024e-3)‡	1.808e-2(4.897e-3)‡	9.364e-2(1.047e-3)‡	**1.067e-2(2.152e-3)**
	RM+25	1.856e-2(4.595e-3)‡	1.5180e-2(6.345e-3)‡	1.469e-2(3.132e-3)‡	1.659e-2(8.615e-4)‡	**1.018e-2(1.790e-3)**
	RM+30	1.647e-1(2.067e-1)‡	1.4890e-2(6.419e-3)‡	1.302e-2(3.027e-3)‡	9.167e-3(1.373e-4)‡	**8.780e-3(1.126e-3)**
	NSDE+20	1.687e-1(2.321e-2)‡	7.550e-2(5.812e-3)‡	8.095e-2(1.734e-2)‡	9.790e-2(1.983e-2)‡	**6.146e-2(1.261e-2)**
	NSDE+25	1.060e-1(1.991e-2)‡	5.973e-2(8.150e-3)‡	4.704e-2(3.398e-3)‡	5.386e-2(2.203e-2)‡	**3.696e-2(3.386e-3)**
	NSDE+30	6.081e-2(4.659e-3)‡	4.504e-2(1.339e-2)‡	3.777e-2(7.265e-3)‡	3.154e-2(6.396e-3)‡	**2.806e-2(4.642e-3)**
DMOP2	RM+20	1.807e-1(9.705e-2)‡	7.4641e-2(4.800e-2)‡	1.584e-2(1.737e-3)‡	2.413e-2(9.974e-4)‡	**7.964e-3(6.225e-4)**
	RM+25	1.459e-1(7.759e-2)‡	3.2133e-2(1.646e-2)‡	1.117e-2(1.296e-3)‡	1.617e-2(7.707e-4)‡	**7.129e-3(5.224e-4)**
	RM+30	1.336e-1(8.924e-2)‡	2.6387e-2(2.201e-2)‡	9.182e-3(7.077e-4)‡	1.156e-2(3.063e-4)‡	**6.763e-3(8.708e-4)**
	NSDE+20	5.556e-2(1.467e-2)‡	3.868e-2(7.676e-3)‡	2.526e-2(1.928e-3)‡	2.672e-2(5.697e-4)‡	**1.560e-2(1.335e-3)**
	NSDE+25	3.524e-2(3.980e-3)‡	2.110e-2(2.422e-3)‡	1.882e-2(1.567e-3)‡	1.774e-2(2.802e-4)‡	**1.547e-2(1.609e-3)**
	NSDE+30	2.573e-2(1.326e-3)‡	2.187e-2(3.398e-3)‡	1.610e-2(3.570e-3)‡	2.297e-2(8.058e-5)‡	**1.418e-2(1.233e-3)**
DMOP3	RM+20	1.336e-1(8.924e-2)‡	2.638e-2(2.201e-2)‡	9.1827e-3(7.077e-4)‡	2.036e-2(9.716e-4)‡	**6.763e-3(8.708e-4)**
	RM+25	5.556e-2(2.693e-2)‡	1.964e-2(3.191e-3)‡	9.5044e-3(5.264e-4)‡	1.406e-2(5.946e-4)‡	**6.235e-3(1.866e-4)**
	RM+30	4.863e-2(2.068e-2)‡	1.366e-2(2.147e-3)‡	7.8289e-3(2.178e-4)‡	1.090e-2(3.626e-4)‡	**5.775e-3(1.139e-4)**
	NSDE+20	4.280e-2(1.657e-2)‡	2.224e-2(1.405e-3)‡	1.735e-2(1.633e-3)‡	2.056e-2(2.620e-4)‡	**1.142e-2(3.689e-4)**
	NSDE+25	2.654e-2(5.699e-3)‡	1.623e-2(1.566e-4)‡	1.307e-2(6.700e-4)‡	1.423e-2(1.106e-4)‡	**1.005e-2(4.905e-4)**
	NSDE+30	2.009e-2(3.445e-3)‡	1.307e-2(9.312e-4)‡	1.188e-2(6.653e-4)‡	1.092e-2(1.135e-4)‡	**8.999e-3(7.096e-4)**
F5	RM+20	2.317e-1(1.090e-1)‡	3.552e-1(1.354e-1)‡	2.9284e-2(4.395e-3)‡	7.088e-2(3.586e-3)‡	**1.718e-2(1.369e-3)**
	RM+25	1.494e-1(1.049e-1)‡	2.052e-1(1.759e-1)‡	1.9839e-2(2.170e-3)‡	4.705e-2(3.420e-3)‡	**1.370e-2(7.233e-4)**
	RM+30	8.327e-2(6.119e-2)‡	1.528e-1(9.551e-2)‡	1.5034e-2(1.526e-3)‡	3.397e-2(2.693e-3)‡	**1.153e-2(4.026e-4)**
	NSDE+20	2.269e-1(1.872e-3)‡	9.979e-2(1.282e-2)‡	7.578e-2(1.124e-3)‡	1.161e-1(2.805e-3)‡	**4.548e-2(1.673e-3)**
	NSDE+25	1.301e-1(4.212e-3)‡	6.914e-2(5.164e-3)‡	5.386e-2(5.710e-4)‡	7.272e-2(1.267e-3)‡	**3.951e-2(1.621e-3)**
	NSDE+30	1.030e-1(1.713e-2)‡	5.864e-2(8.453e-3)‡	4.253e-2(1.593e-3)‡	4.911e-2(1.355e-3)‡	**3.316e-2(1.652e-3)**
F6	RM+20	1.879e-1(3.053e-2)‡	1.480e-1(4.083e-2)‡	3.649e-2(8.791e-3)‡	4.273e-2(3.661e-3)‡	**3.092e-2(8.000e-3)**
	RM+25	1.087e-1(4.196e-2)‡	9.361e-2(2.838e-2)‡	2.817e-2(4.367e-3)†	**2.674e-2(2.716e-3)**	2.892e-2(9.061e-3)
	RM+30	9.852e-2(3.636e-2)‡	7.881e-2(2.611e-2)‡	2.180e-2(4.208e-3) †	**1.924e-2(1.910e-3)**	2.571e-2(5.148e-3)
	NSDE+20	2.392e-1(1.052e-2)‡	1.051e-1(5.774e-3)‡	7.580e-2(1.044e-3)‡	1.053e-1(2.709e-3)‡	**3.610e-2(2.167e-3)**
	NSDE+25	1.262e-1(9.155e-3)‡	6.401e-2(9.477e-3)‡	6.275e-2(4.755e-3)‡	6.343e-2(1.722e-3)‡	**3.418e-2(5.598e-3)**
	NSDE+30	1.158e-1(1.000e-2)‡	6.257e-2(9.769e-3)‡	4.388e-2(1.267e-2)‡	4.135e-2(4.306e-4)‡	**3.281e-2(2.021e-4)**
F7	RM+20	9.317e-2(4.873e-2)‡	5.546e-2(1.108e-2)‡	2.322e-2(2.281e-3)‡	3.166e-2(1.186e-3)‡	**2.033e-2(4.406e-3)**
	RM+25	7.099e-2(3.534e-2)‡	4.167e-2(9.941e-3)‡	**1.934e-2(2.513e-3)**	2.242e-2(9.053e-4)‡	1.953e-2(3.879e-3)
	RM+30	6.170e-2(2.796e-2)‡	3.674e-2(9.315e-3)‡	1.780e-2(3.242e-3)†	1.604e-2(9.461e-4)‡	**1.509e-2(3.713e-3)**
	NSDE+20	2.088e-1(2.155e-2)‡	7.398e-2(1.947e-3)‡	6.188e-2(4.098e-3)‡	7.889e-2(2.023e-3)‡	**2.761e-2(4.134e-3)**
	NSDE+25	1.228e-1(5.060e-3)‡	5.562e-2(6.181e-3)‡	4.288e-2(4.777e-3)‡	4.857e-2(6.685e-4)‡	**2.720e-2(5.680e-3)**
	NSDE+30	8.048e-2(3.774e-3)‡	5.725e-2(2.747e-3)‡	3.511e-2(2.791e-3)‡	3.367e-2(8.828e-4)‡	**2.201e-2(3.128e-3)**
F8	RM+20	4.959e-1(2.284e-1)‡	3.260e-1(1.492e-1)‡	1.037e-1(5.245e-2)	**7.146e-2(3.187e-3)**	1.320e-1(2.598e-2)
	RM+25	2.983e-1(1.368e-1)‡	2.203e-1(1.119e-1)‡	6.872e-2(5.119e-2)	**5.007e-2(2.871e-3)**	1.143e-1(2.667e-2)
	RM+30	2.081e-1(9.320e-2)‡	1.500e-1(6.361e-2)‡	3.653e-2(2.038e-2)	**3.534e-2(1.637e-3)**	8.832e-2(2.830e-2)
	NSDE+20	5.424e-1(1.099e-1)‡	1.417e-1(4.255e-2)‡	3.047e-1(3.237e-2)‡	1.207e-1(2.214e-3)‡	**9.787e-2(2.189e-2)**
	NSDE+25	3.099e-1(5.298e-2)‡	8.109e-2(1.184e-2)	2.322e-1(3.429e-2)‡	**7.529e-2(1.662e-3)**	1.010e-1(1.391e-2)
	NSDE+30	1.878e-1(5.659e-3)‡	6.847e-2(1.092e-2)	2.000e-1(3.247e-2)‡	**5.013e-2(1.948e-3)**	7.217e-2(1.014e-2)
F9	RM+20	6.145e-1(1.303e-1)‡	6.379e-1(2.057e-1)‡	6.372e-1(2.945e-1)‡	7.125e-2(6.754e-3)‡	**5.061e-2(3.773e-3)**
	RM+25	3.618e-1(7.905e-2)‡	4.025e-1(1.511e-1)‡	2.483e-1(1.609e-1)‡	5.288e-2(3.893e-3)‡	**3.737e-2(2.923e-3)**
	RM+30	3.039e-1(8.803e-2)‡	2.276e-1(7.025e-2)‡	1.455e-1(1.016e-1)‡	3.248e-2(1.693e-3)‡	**2.872e-2(2.361e-3)**
	NSDE+20	4.838e-1(9.638e-3)‡	1.755e-1(1.979e-2)‡	1.682e-1(2.280e-2)‡	1.332e-1(6.479e-3)‡	**5.585e-2(1.422e-3)**
	NSDE+25	2.985e-1(2.747e-2)‡	1.357e-1(9.438e-3)‡	1.065e-1(4.663e-3)‡	7.872e-2(5.780e-4)‡	**4.751e-2(1.411e-3)**
	NSDE+30	2.061e-1(1.244e-2)‡	9.345e-2(3.488e-3)‡	7.540e-2(6.977e-3)‡	5.324e-2(2.167e-3) ‡	**4.022e-2(1.442e-3)**
F10	RM+20	1.677e-1(7.141e-3)‡	1.432e-1(5.328e-3)‡	1.320e-1(2.327e-3)‡	1.088e-1(1.093e-3)‡	**8.917e-2(1.490e-3)**
	RM+25	1.437e-1(4.091e-3)‡	1.272e-1(2.801e-3)‡	1.235e-1(1.931e-3)‡	1.004e-1(1.424e-3)‡	**8.801e-2(1.187e-3)**
	RM+30	1.337e-1(2.987e-3)‡	1.172e-1(1.634e-3)‡	1.195e-1(1.984e-3)‡	9.673e-2(6.182e-4)‡	**8.733e-2(7.048e-4)**
	NSDE+20	6.941e-1(7.530e-3)‡	2.563e-1(6.102e-3)‡	4.958e-1(2.764e-2)‡	1.725e-1(2.112e-3)‡	**1.070e-1(1.006e-3)**
	NSDE+25	5.693e-1(1.019e-2)‡	2.379e-1(5.527e-3)‡	3.692e-1(1.836e-2)‡	1.495e-1(1.210e-3)‡	**1.061e-1(1.256e-3)**
	NSDE+30	4.845e-1(6.242e-3)‡	2.175e-1(2.898e-3)‡	4.305e-1(5.220e-2)‡	1.372e-1(1.770e-3)‡	**1.045e-1(1.695e-3)**

‡ and † indicate MSS performed significantly better than and equivalently to the corresponding algorithm, respectively. N_t is set to 10.

mance in terms of the closeness, distribution, and coverage of an approximation to the true POF. For the overall analysis of MSS when we utilize RM, we can combine the IGD with SP, GD and MS to deeply and extensively reveal the algorithm's performance on the test instances. As can be seen in Table 2, MSS performed best among all the compared algorithms on FDA and DMOP. As for FDA2, the distribution of MSS is slightly weaker than that of PPS (shown in Table 1 of the supplementary file) and the spread coverage of MSS is slightly weaker than DMS (presented in Table 2 of the supplementary file). Nevertheless, MSS is the best in terms of GD and IGD metrics, probably because a better GD value has more positive effects on IGD than SP and MS to some extent. The spread coverage of DMS is the best but its convergence is the worst. It can be concluded that good SP or MS values do not necessarily result in a satisfying IGD metric, which can be obviously observed from the experiments on FDA2. To summarize the metrics of SP, MS, GD and IGD, frequency of change has a significant effect on the algorithms' performance. In other words, as the changing frequency increases from 20 to 30, four metrics get better and better in most test instances. Similarly, the conclusion can be made from Table 2 that all the strategies integrating NSGA-II-DE present the same performance as those integrating RM-MEDA, except for the situation that MSS ranks the second on FDA2 when the changing severity is 30.

4.3.2. Results on F Problems

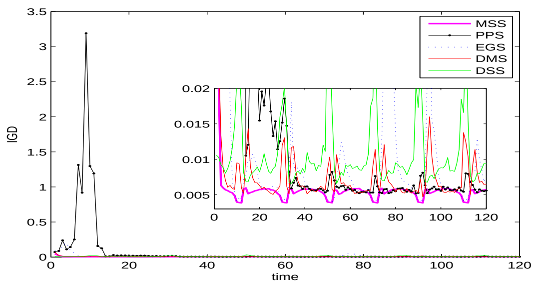
The F test instances are different from the FDA and DMOP. F5-F10 test instances are nonlinear linkages between decision variables, which have some new features in the dynamic environment.

From Table 1 of the supplementary file, the experimental results on the SP metric can be seen. The considerably small SP values of MSS suggest the POF approximations have good distribution on the true POF. This demonstrates that MSS can maintain better distribution of its approximations over changes on most of the test problems compared to the other algorithms. In addition, Table 2 of the supplementary file shows the spread coverage of MSS is significantly better in dealing with most problems. MSS performed a little worse than DMS and DSS on F7 and F8. More than that, on F6, MSS was only worse than DSS.

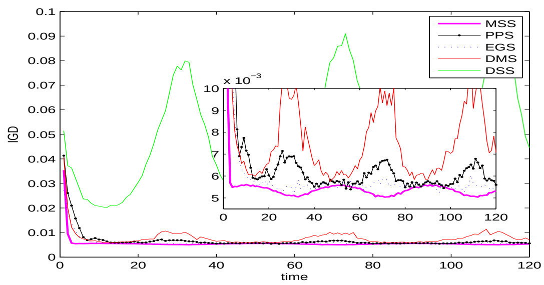
And MSS performed better than DMS when $\tau_t = 20$ and equivalent to DSS when $\tau_t = 25$ and 30. It should be mentioned that DSS was superior to MSS on F6, F7 and F8. MSS may have better capacity to remove dominance resistant solutions (DRS) than DSS, which can be showed by the IGD values obtained by MSS on F7 which are smaller than DSS whenever the frequency of change (τ_t). For F6 and F8, the spread coverage obtained by MSS is worse than the other algorithms, indicating that MSS needs to be improved for dealing with F6 and F8 well.

Table 1 clearly shows that the GD values obtained by MSS are usually much lower than the other strategies in most cases, which means that it has better convergence than the other algorithms. For F8 test problems, MSS presents a more uniform distribution than the other strategies (Table 1 of the supplementary file), while the convergence and extensive distribution of MSS are slightly worse than DMS (Tables 1 and 2 of the supplementary file). The overall performance (i.e., the metric IGD) of MSS on F8 was worse than DMS and DSS. Clearly, the uncompetitive coverage (i.e., slightly small MS metric) and poor convergence (i.e., relatively large GD metric) of obtained approximations are the main reasons for the low performance of MSS on F8, respectively.

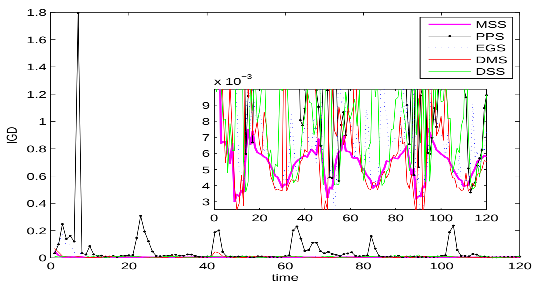
As shown in Table 2, MSS is significantly superior to the other algorithms on most selected test problems when the four methods are incorporated into RM-MEDA, although DMS and DSS provide slightly better IGD values than MSS for F8. For F8, the IGD value of MSS is weaker than the DMS and DSS in spite of its good spread distribution (SP). This may be because the comprehensive performance of MSS was severely affected by its poor coverage (MS). However, good SP, GD and MS do not necessarily result in satisfying the IGD metric, and this can be particularly observed on F6, implying that the IGD metric is slightly worse than DMS although it provides the almost best three metrics except for the MS on this problem. From Table 1, 2 and Table 1, 2 of the supplementary file, it can be seen there is slight fluctuation in different performance metrics shown in the experiments, while the overall comprehensive performance of MSS was significantly better than the other three algorithms. Overall, MSS had



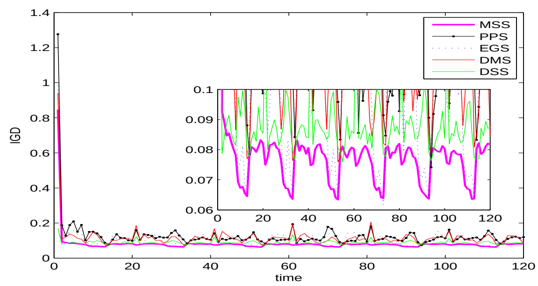
(a) FDA1



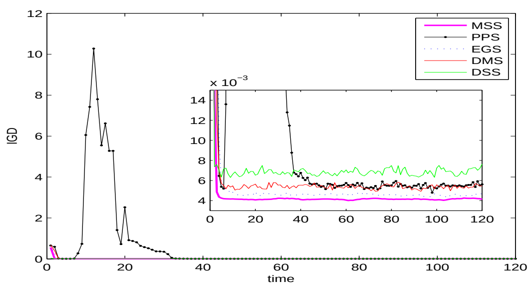
(b) FDA2



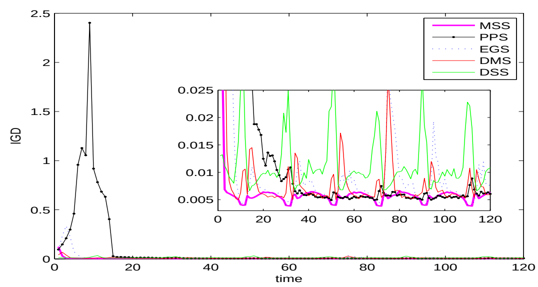
(c) FDA3



(d) FDA4



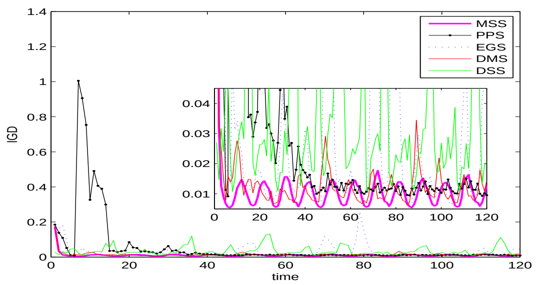
(e) DMOP1



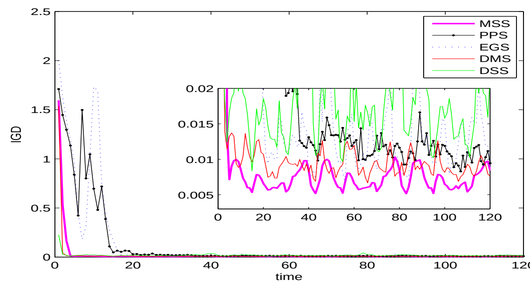
(f) DMOP2

Figure 2: Evolution curves of average IGD values for eight problems with N_t =10 and τ_t=30.

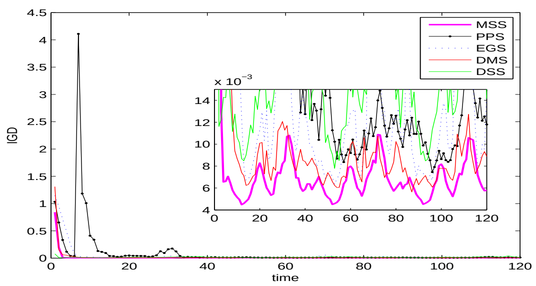
less sensitivity to the frequency of change (i.e., τ_t = 20 to 30) in coping with DMOPs, as can be seen from their gradual improvement on three metrics when τ_t increases from 20 to 30. On the other hand, with the variation of frequency, there were great improvements in the other four methods in most of the test



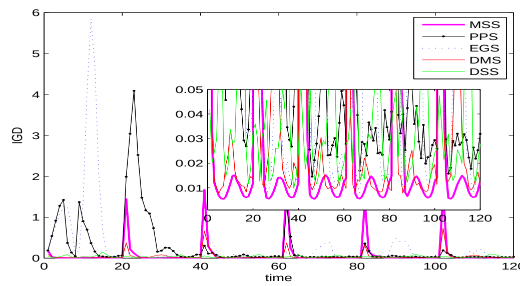
(a) F5



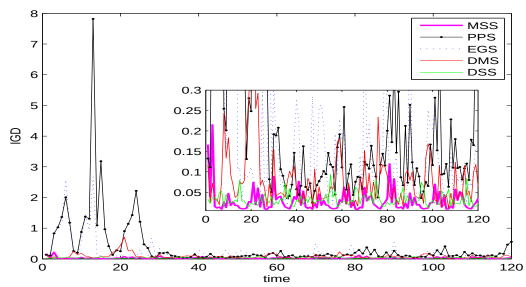
(b) F6



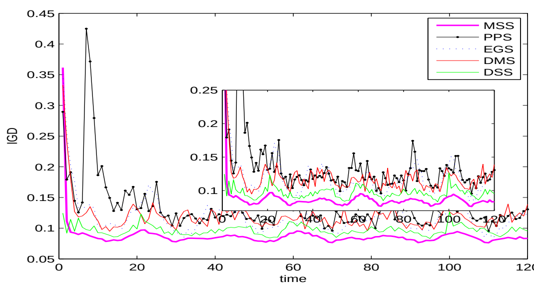
(c) F7



(d) F8



(e) F9



(f) F10

Figure 3: Evolution curves of average IGD values for eight problems with N_t =10 and τ_t=30.

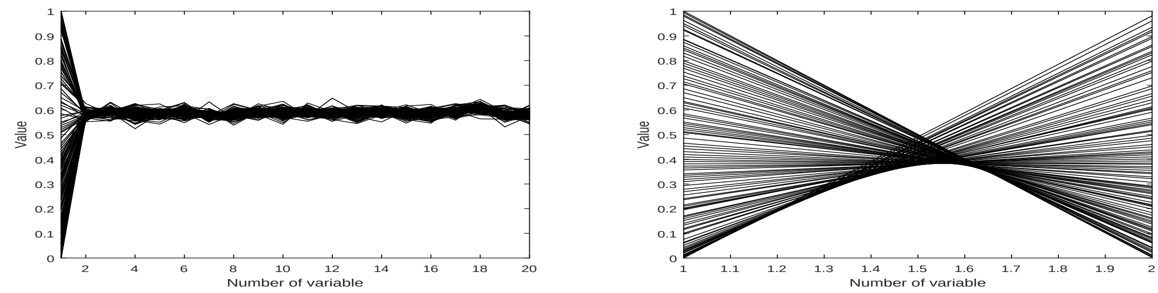
446 instances. It can be also inferred from Table 2 that MSS that incorporates
447 NSGA-II-DE performs still and even a little better than the algorithms that
448 integrate RM-MEDA when compared with the four other approaches.

4.3.3. Comparison of the distribution and convergence of the final obtained population

For a more intuitive understanding, apart from the experimental results given in the table, the evolution curves of the average IGD value are provided in Figures 2 and 3. As can be seen from these figures, the MSS responds to the environmental changes faster than the others overall. Therefore, the MSS had better convergence and distribution performance. The only exception is that DMS performed best on F8. The reason for that may be the lack of population diversity (i.e., poor MS values) when a change occurs, and the IGD values obtained by MSS fluctuate widely on this problem. Despite that, in order to clearly show MSS's performance, FDA1, DMOP2, F5, F8 and F9 were chosen. The distribution of the final population gained by the four strategies at different time steps is shown in Figures 1, 2, 3, 4 and 5 of the supplementary material.

From the Figures 1, 2, 3, 4 and 5 of the supplementary material, it is obvious that MSS performed better than the other four strategies at the early time steps, which shows that MSS is able to respond to environmental changes more accurately and quickly. EGS performed better than PPS, for the reason that the prediction of PPS needs more historical message accumulation than EGS. More than that, EGS's gradient prediction strategy and memory strategy can help to achieve good convergence at the first stage. DMS performed better than PPS and EGS due to the fact that the prediction in DMS can facilitate the convergence. Clearly, on convergence and distribution, MSS is superior to PPS, EGS and DMS on both linear and nonlinear problems. From Figures 1, 2, 3, 4 and 5 of the supplementary material, MSS has clearly significant advantages on convergence and distribution compared with the other strategies, implying that MSS is able to make the obtained population track the POS rapidly and accurately when coping with DMOPs. Therefore, MSS can better respond to environmental changes and track the POF over time.

In order to analyze the relationships between the the decision variables and the objectives, we select FDA1 to draw the final distribution of decision space



(a) decision space (b) objective space

Figure 4: The solution set obtained by MSS on FDA1 with $t = 5$.

and objective space on MSS. At time $t = 5$, it can be seen from Figures 4 that MSS obtained good distribution in objective space, while it gained slightly worse distribution in decision space. The possible reason may be that the proposed algorithm only consider the distribution in objective space. In most cases, there is bigger real-world application to find POF in objective space [37, 38].

4.4. Comparison under different setting

Table 3: Mean and SD of IGD obtained by two algorithms.

		$0 \leqslant t \leqslant 20$					$20 < t \leqslant 160$				
instance	strategy	mean	1stQ	median	3rdQ	t-test	mean	1stQ	median	3rdQ	t-test
FDA1	PPS	0.0560	0.0144	0.0354	0.0686	+	0.0053	0.0052	0.0053	0.0053	
	MSS	**0.0067**	**0.0064**	**0.0074**	**0.0067**		0.0053	0.0052	0.0053	0.0053	
DMOP1	PPS	0.1996	0.0979	0.1303	0.2058	+	0.0052	0.0052	0.0052	0.0052	+
	MSS	**0.0173**	**0.0154**	**0.0144**	**0.0151**		**0.0042**	**0.0041**	**0.0042**	**0.0042**	

+(−) supports (fails to support).

Table 4: Mean and SD of IGD obtained by two algorithms.

Problem	Strategy	$0 \leqslant t \leqslant 120$	$1 \leqslant t \leqslant 40$	$41 \leqslant t \leqslant 80$	$81 \leqslant t \leqslant 120$
FDA1	DMS	0.0079 (0.0004)	0.0096(0.0009)	0.0072 (0.0005)	0.0071 (0.0003)
	MSS	**5.8e-3(2.0e-4)**	**6.7e-3(2.412e-4)**	**5.3e-3(4.865e-5)**	**5.3e-3(4.286e-5)**
DMOP1	DMS	0.0129(0.0043)	0.0281(0.0129)	0.0054(3.85E-05)	0.0054(4.23E-05)
	MSS	**8.8e-3(1.1e-3)**	**2.0e-2(1.9e-3)**	**4.1e-3(1.8e-5)**	**4.2e-3(8.2e-6)**

‡ and † indicate MSS performed significantly better than and equivalently to the corresponding algorithm, respectively.

In order to test whether we have the ability of the exact implementation of the algorithms, we performed a comparison experiment between the proposed

Table 5: Parameter settings about the experiments of Tables 3 and 4.

the name of parameter	PPS	DMS
the number of the environment changes	160	120
N_t	10	10
τ_t	30	30
the size of the population	100	100
the size of the decision space	20	20

algorithm and the other algorithms as per the settings given in the articles of PPS and DMS algorithms in dealing with FDA1 and DMOP1. Particularly, the results of the IGD were obtained by the PPS in Table 3, which come from the original literature. Meanwhile, the results of the IGD were obtained by the MSS (Table 3) according to the parameter settings of the PPS. Similarly, the results of the MSS are obtained in Table 4 according to the parameter settings of the DMS. The parameter settings are described in detail in Table 5. As shown in Tables 3 and 4, the performance of MSS had better competitiveness than PPS and DMS.

5. Discussion

5.1. Influence of Different Components

This subsection shows the effect of different components of MSS. MSS mainly includes two parts: an improved local search and a global search. The two strategies are carried out separately, and the experiments are conducted on FDA and DMOP test suites with the setting of $(N_t, \tau_t) = (10, 30)$. S1 represents the local search strategy and S2 is the global search strategy. IGD, SP, MS and GD are adopted as the performance metrics. The obtained average IGD, SP, MS and GD results of the two divided strategies and their standard deviation values are presented in Table 6. From Table 6, it can be demonstrated that the combined strategy contributes to convergence and diversity, which can be

Table 6: Mean and standard deviation of metrics for the algorithms with different strategies on FDA1, FDA2, FDA3, FDA4, DMOP1, DMOP2 and DMOP3 over 20 runs. Two strategies: local search strategy and global search strategy are denoted by S1 and S2 respectively.

Prob.	metrics	S1	S2	S1+S2
FDA1	IGD	6.930e-3(1.396e-4)‡	9.772e-3(2.622e-4)‡	**5.804e-3(2.010e-4)**
	SP	3.646e-3(1.988e-4)‡	4.224e-3(2.284e-4)‡	**3.415e-3(1.601e-4)**
	MS	9.968e-1(4.979e-4)†	9.940e-1(6.412e-4)‡	**9.969e-1(6.505e-4)**
	GD	5.575e-3(1.807e-4)‡	9.157e-3(2.599e-4)‡	**4.145e-3(1.469e-4)**
FDA2	IGD	5.679e-3(5.374e-5)‡	5.836e-3(9.891e-5)‡	**5.621e-3(5.702e-5)**
	SP	**5.026e-3(1.153e-4)**	5.149e-3(1.298e-4)‡	5.027e-3(9.930e-5)
	MS	7.589e-1(7.266e-4)	**7.600e-1(8.061e-4)**	7.584e-1(5.830e-4)
	GD	1.160e-2(1.396e-4)‡	1.216e-2(2.831e-4)‡	**1.147e-2(1.837e-4)**
FDA3	IGD	6.556e-3(2.349e-4)‡	8.774e-3(1.003e-3)‡	**5.768e-3(1.521e-4)**
	SP	3.640e-3(7.068e-5)‡	4.505e-3(1.337e-4)‡	**3.451e-3(6.999e-5)**
	MS	8.705e-1(5.312e-3)‡	**8.734e-1(2.872e-3)**	8.659e-1(6.093e-3)
	GD	1.082e-1(3.404e-3)‡	1.032e-1(2.365e-3)‡	**1.058e-1(3.629e-3)**
FDA4	IGD	8.658e-2(2.422e-3)‡	1.001e-1(1.478e-3)‡	**8.264e-2(2.276e-3)**
	SP	4.186e-2(1.859e-3)‡	5.387e-2(3.160e-3)‡	**4.319e-2(1.660e-3)**
	MS	1.000e+0(2.112e-6)†	1.000e+0(1.786e-6)†	**1.000e+0(1.970e-6)**
	GD	4.379e-1(6.650e-3)‡	4.877e-1(4.688e-3)‡	**5.878e-2(4.051e-3)**
DMOP1	IGD	9.693e-3(2.091e-3)‡	1.031e-2(2.891e-3)‡	**8.780e-3(1.126e-3)**
	SP	3.579e-3(1.721e-4)†	3.795e-3(6.924e-4)†	**3.685e-3(2.641e-4)**
	MS	9.933e-1(2.383e-3)†	9.914e-1(4.343e-3)†	**9.940e-1(1.111e-3)**
	GD	7.158e-3(1.665e-3)‡	7.216e-3(2.263e-3)‡	**5.763e-3(1.883e-3)**
DMOP2	IGD	7.898e-3(5.067e-4)‡	1.175e-2(4.865e-4)‡	**6.763e-3(8.708e-4)**
	SP	3.590e-3(2.334e-4)‡	4.240e-3(2.149e-4)‡	**3.414e-3(2.189e-4)**
	MS	**9.943e-1(1.420e-3)**	9.884e-1(1.346e-3)‡	9.941e-1(1.996e-3)
	GD	6.357e-3(1.748e-4)‡	1.119e-2(2.404e-4)‡	**4.550e-3(1.757e-4)**
DMOP3	IGD	6.912e-3(1.585e-4)‡	9.809e-3(2.368e-4)‡	**5.775e-3(1.139e-4)**
	SP	3.568e-3(8.893e-5)‡	4.232e-3(1.957e-4)‡	**3.421e-3(1.445e-4)**
	MS	9.967e-1(5.136e-4)†	9.941e-1(6.121e-4)‡	**9.968e-1(5.333e-4)**
	GD	5.545e-3(1.004e-4)‡	9.192e-3(2.299e-4)‡	**4.067e-3(1.117e-4)**

‡ and † indicate S1 + S2 performs significantly better than and equivalently to S1 or S2, respectively. N_t is set to 10. τ_t is set to 30.

seen from the IGD, SP, MS and GD values. The reason is because these two strategies can compensate for the drawback between each other. S2 obtained the worst SP, IGD, MS and GD values in most metrics. The results of S2 obviously suggest that the use of an improved local search can significantly improve the performance of MSS. In addition, influence of the severity of change and study of different dimensions of decision space on the MSS are presented in Section 3 of the supplementary material to show the excellence of the proposed algorithm compared to other algorithms.

These results clearly exhibit the importance of each component in dealing with DMOPs. The combination strategy performed significantly better than the other strategies on the majority of test instances in Table 6, and the local search strategy performed better than the global search strategy in other instances, which indicates that the strategy combining two strategies has some positive influence on performance of MSS when solving dynamic problems with different features.

5.2. Influence of function evaluations

Table 7: Mean and SD of four indicator obtained by two algorithms.

	FDA1		FDA2		DMOP1	
metric	MSS	MSS-Ev	MSS	MSS-Ev	MSS	MSS-Ev
IGD	5.804e-3(2.010e-4)†	5.703e-3(2.010e-4)	5.621e-3(5.702e-5)†	5.624e-3(2.622e-4)	8.780e-3(2.091e-3)†	8.779e-3(2.090e-3)
SP	3.415e-3(1.601e-4)†	3.425e-3(1.601e-4)	5.027e-3(9.930e-5)†	5.020e-3(9.930e-5)	3.685e-3(1.721e-4)†	3.68.e-3(1.711e-4)
MS	9.969e-1(6.505e-4)†	9.979e-1(6.505e-4)	7.584e-1(5.830e-4)†	7.580e-1(5.831e-4)	9.940e-1(2.383e-3)†	9.944e-1(2.384e-3)
GD	4.145e-3(1.469e-4)†	3.957e-3(1.460e-4)	1.147e-2(1.837e-4)†	1.157e-2(1.827e-4)	5.763e-3(1.665e-3)†	5.764e-3(1.665e-3)

‡ and † indicate MSS performs significantly better than and equivalently to the corresponding algorithm, respectively. N_t is set to 10 and τ_t is set to 30.

MSS in each static environment has only evaluation of $5\% * N$ more than other algorithms. For this problem, we have analyzed from two perspectives: math and experiments. Firstly, suppose the change severity is set to $\tau_t = 30$ and the evolution number of the total population would be $N * \tau_t = 30N$ in a certain evolutionary environment. The proportion of the evolution number of the mutation individual to the evolution of the total population would be $(5\% * N)/(N * \tau_t) = 1/3000$ under some static environment. This proportion is very small from the math view, implying that the influence of excess evolutionary number is feeble or can be ignored. Secondly, we did some analysis also. We first supposed that MSS-Ev ensures that the total number of evaluations in the population is the same in each static environment. In order to clearly analyze the performance of the populations obtained by MSS and MSS-Ev at each static environment, three typical test problems including FDA1, FDA2 and DMOP1 were chosen. We can see that the performances obtained by the MSS and MSS-Ev were not significantly different from each other. So MSS performed equivalently to MSS-Ev from the Wilcoxon ranksum test view. Overall, the influence of the excess number of evolutions $5\% * N$ can be ignored in the process of experiment since the result of Table 7 and the mathematical analysis are the same.

5.3. Influence of the solutions size in the local search strategy

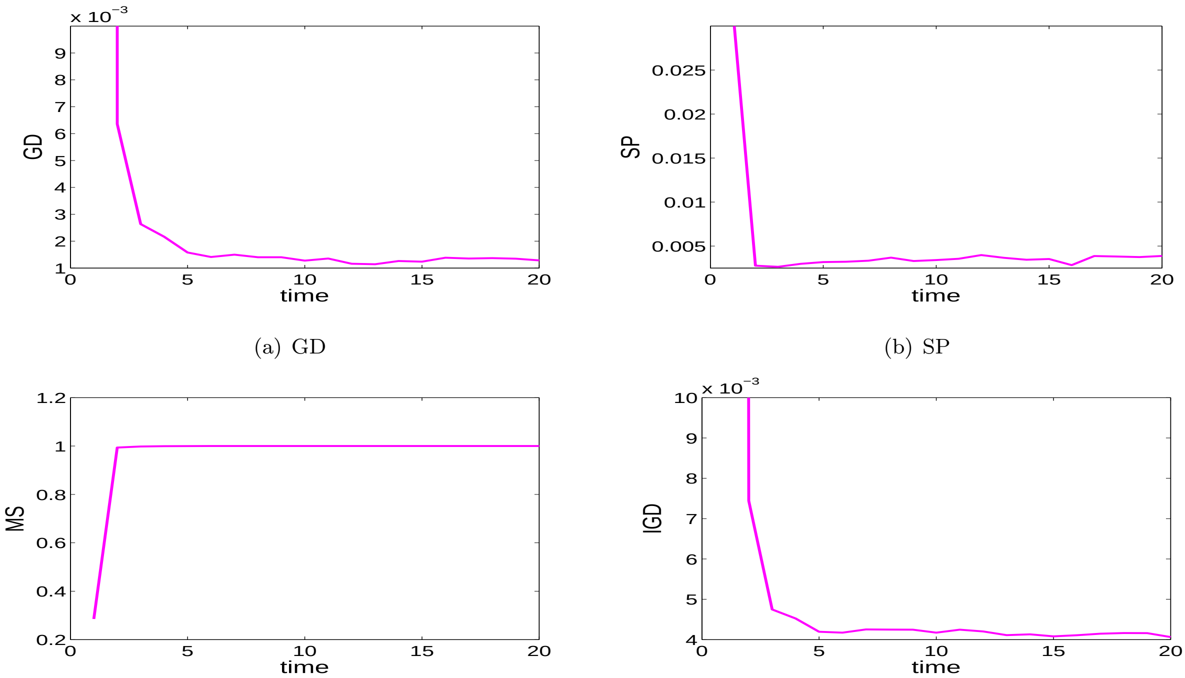
Table 8: Mean and standard deviation of IGD metric for MSS with the solution size in the local search strategy on FDA2, DMOP2 and F5 over 20 runs

Problem	5%	10%	20%	30%
FDA2	8.124e-2(2.238e-4)	8.194e-3(2.458e-4)	5.621e-3(5.702e-5)	5.451e-3(6.002e-5)
DMOP2	3.182e-2(5.077e-3)	1.582e-2(7.077e-4)	6.763e-3(8.708e-4)	6.000e-3(8.708e-4)
F5	8.370e-1(6.119e-2)	6.327e-2(6.119e-2)	1.153e-2(4.026e-4)	9.153e-3(4.026e-4)

We are also interested in examining the influence of the solution size in the local search strategy of MSS. Table 8 shows the mean and standard deviation of the IGD metric for MSS with different settings for 5%, 10%, 20% and 30% on FDA2, DMOP2 and F5 over 20 runs. The performance of MSS when 5% or 10% of individuals ware selected was significantly worse than the other two settings. Although MSS performed increasingly reliably with the increase of the solution's size in the local search strategy, the results are similar to each other when 20% and 30% of individuals are used. In this paper, we choose 20% of individuals in the population, taking computational time into consideration.

5.4. Detailed analysis of MSS by DMOP1

In dynamic multi-objective optimization, we usually use the metrics of SP, MS, GD and IGD as they can help deeply investigate an algorithms's performance regarding convergence, distribution, and diversity in the objective space. More precisely, those performances of MSS have been shown in Fig. 5.4 by the poly-line form in dealing with DMOP1. As indicated in Fig. 5.4, the performances of MSS fluctuates when $t < 5$ (i.e., the GD value obtained goes from 0.01 to 0.0015; the SP value obtained goes from 0.03 to 0; the MS value obtained goes from 0.3 to 1). Due to the influence on convergence, distribution, and diversity when an environmental change occurs, the IGD value obtained by MSS fluctuated widely on this problem when $t < 5$. When $t \geq 5$, the performance of MSS was comparative in handling DMOP1.



(a) GD (b) SP

(c) MS (d) IGD

Figure 5: Evolution curves of GD, SP, MS and IGD values for DMOP1 with N_t =10 and τ_t=30.

5.5. More discussion

This section is devoted to discussing the advantages and limitations of the proposed MSS algorithm. First, the proposed MSS algorithm combines local search strategy and global search strategy to handle DMOPs, which is capable of quickly tracking the changing POS. However, it needs some additional computational cost for promoting population convergence. Therefore, we reduce the corresponding number of iterations introduced in section 4.2. Secondly, MSS also has some drawbacks compared with DMS algorithms on problems F5-F10. This is because DMS can generate more well-diversified solutions, which has much advantage in addressing difficult variable-linkage ZJZ (F5-F10) [15] problems. Therefore, further improvement should be made on the MSS algorithm to increase population diversity.

6. Conclusions and future work

In this paper, a new dynamism handling method (i.e., MSS) has been proposed for handling multi-objective problems with time-varying characteristics. This approach consists of two main components: a two step-size strategy for local search strategy and global search strategy. The local search strategy based on mutliple directions has used individuals from the current population to produce solutions along each decision variable's direction within a certain range and to update the population using the generated solutions, achieving a fast convergence to the Pareto optimal set. On the other hand, the global search strategy using mutation has been proposed to increase the probability of individual variation to improve the diversity of the population. These components are important for creating a good initial population and enhancing either diversity or convergence, when a change occurs in the environment.

The proposed algorithm was measured on several benchmark test suites with various dynamic characteristics and different difficulties. Experimental results show that this algorithm is very competitive in dealing with DMOPs compared with four state-of-the-art approaches. In other words, MSS has the ability to find the dynamic Pareto optimal front and/or Pareto optimal set as quickly and accurately as possible before the next environmental change occurs. However, MSS's performance still should be improved on the test problem F8 by enhancing convergence and diversity.

Although MSS has performed well in dealing with dynamic multi-objective problems, there could be some improvements. For example, MSS can be combined with predictive models to speed up the prediction of the accuracy of the model in the early stages of change. Our future work includes reducing the diversity loss caused by environmental changes according to the feedback of each generation and developing a new change response method and new operators to evolve the population for solving DMOPs with different characteristics.

Acknowledgement

This work was supported by the research projects: the National Natural Science Foundation of China under Grant Nos. 61502408, 61673331, 61379062 and 61772178, the Education Department Major Project of Hunan Province under Grant No. 17A212, the CERNET Innovation Project under Grant No. NGII20150302, the Science and Technology Plan Project of Hunan Province under Grant No. 2016TP1020, the Provinces and Cities Joint Foundation Project under Grant No. 2017JJ4001. the postgraduate research and innovation Project of Hunan Province under Grant No. XDCX2019B057.

References

[1] A. Zhou, Y. Jin, Q. Zhang, A population prediction strategy for evolutionary dynamic multiobjective optimization, Cybernetics, IEEE Transactions on 44 (1) (2014) 40–53.

[2] K. Chi, G. Yin, Y. Dong, H. Dong, Link prediction in dynamic networks based on the attraction force between nodes, Knowledge-Based Systems.

[3] K. Deb, S. Karthik, et al., Dynamic multi-objective optimization and decision-making using modified nsga-ii: a case study on hydro-thermal power scheduling, in: Evolutionary Multi-Criterion Optimization, Springer, 2007, pp. 803–817.

[4] Z. Deng, M. Liu, Y. Ouyang, S. Lin, M. Xie, Multi-objective mixed-integer dynamic optimization method applied to optimal allocation of dynamic var sources of power systems, IEEE Transactions on Power Systems 33 (2) (2018) 1683–1697.

[5] B. Lokeshgupta, S. Sivasubramani, Multi-objective dynamic economic and emission dispatch with demand side management, International Journal of Electrical Power & Energy Systems 97 (2018) 334–343.

[6] Z. Zhang, Multiobjective optimization immune algorithm in dynamic environments and its application to greenhouse control, Applied Soft Computing 8 (2) (2008) 959–971.

[7] A. Azizivahed, H. Narimani, M. Fathi, E. Naderi, H. R. Safarpour, M. R. Narimani, Multi-objective dynamic distribution feeder reconfiguration in automated distribution systems, Energy 147 (2018) 896–914.

[8] R. Gan, G. Yu, J. Zheng, J. Zou, S. Yang, The effect of diversity maintenance on prediction in dynamic multi-objective optimization, Applied Soft Computing.

[9] Y.-J. Zhang, S.-F. Shao, J. Niyongabo, Cloud hyper mutation particle swarm optimization algorithm based on cloud model, Pattern Recognition and Artificial Intelligence 24 (1) (2011) 90–94.

[10] Q. Li, J. Zou, S. Yang, J. Zheng, G. Ruan, A predictive strategy based on special points for evolutionary dynamic multi-objective optimization, Soft Computing 23 (11) (2019) 3723–3739.

[11] C.-K. Goh, K. C. Tan, A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization, Evolutionary Computation, IEEE Transactions on 13 (1) (2009) 103–127.

[12] R. Chen, K. Li, X. Yao, Dynamic multiobjectives optimization with a changing number of objectives, IEEE Transactions on Evolutionary Computation 22 (1) (2018) 157–171.

[13] Z. Peng, J. Zheng, J. Zou, M. Liu, Novel prediction and memory strategies for dynamic multiobjective optimization, Soft Computing 19 (9) (2015) 2633–2653.

[14] M. Liu, W. Zeng, Memory enhanced dynamic multi-objective evolutionary algorithm based on decomposition, Ruan Jian Xue Bao/Journal of Software 24 (7) (2013) 1571–1588.

[15] Q. Zhang, A. Zhou, Y. Jin, Rm-meda: A regularity model-based multi-objective estimation of distribution algorithm, Evolutionary Computation, IEEE Transactions on 12 (1) (2008) 41–63.

[16] Q. Zhang, S. Yang, S. Jiang, R. Wang, X. Li, Novel prediction strategies for dynamic multi-objective optimization, IEEE Transactions on Evolutionary Computation.

[17] S. Jiang, M. Kaiser, S. Yang, S. Kollias, N. Krasnogor, A scalable test suite for dynamic multiobjective optimization.

[18] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multi-objective genetic algorithm: Nsga-ii, IEEE transactions on evolutionary computation 6 (2) (2002) 182–197.

[19] S. Jiang, S. Yang, A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization, IEEE Transactions on Evolutionary Computation PP (99) (2017) 1–1.

[20] A. Muruganantham, Y. Zhao, S. B. Gee, Dynamic multiobjective optimization using evolutionary algorithm with kalman filter, Procedia Computer Science 24 (2013) 66–75.

[21] S. B. Gee, K. C. Tan, C. Alippi, Solving multiobjective optimization problems in unknown dynamic environments: An inverse modeling approach, IEEE transactions on cybernetics 47 (12) (2017) 4223–4234.

[22] N. Daryani, M. T. Hagh, S. Teimourzadeh, Adaptive group search optimization algorithm for multi-objective optimal power flow problem, Applied Soft Computing 38 (2016) 1012–1024.

[23] H. Ishibuchi, T. Murata, Multi-objective genetic local search algorithm, in: Proceedings of IEEE international conference on evolutionary computation, IEEE, 1996, pp. 119–124.

[24] H. Ishibuchi, T. Murata, A multi-objective genetic local search algorithm and its application to flowshop scheduling, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 28 (3) (1998) 392–403.

[25] D.-J. Wang, F. Liu, Y. Jin, A multi-objective evolutionary algorithm guided by directed search for dynamic scheduling, Computers & Operations Research 79 (2017) 279–290.

[26] T. Murata, H. Nozawa, Y. Tsujimura, M. Gen, H. Ishibuchi, Effect of local search on the performance of cellular multiobjective genetic algorithms for designing fuzzy rule-based classification systems, in: Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600), Vol. 1, IEEE, 2002, pp. 663–668.

[27] D. F. Pires, C. H. Antunes, A. G. Martins, Nsga-ii with local search for a multi-objective reactive power compensation problem, International Journal of Electrical Power & Energy Systems 43 (1) (2012) 313–324.

[28] K. Sindhya, A. Sinha, K. Deb, K. Miettinen, Local search based evolutionary multi-objective optimization algorithm for constrained and unconstrained problems, in: 2009 IEEE Congress on Evolutionary Computation, IEEE, 2009, pp. 2919–2926.

[29] J. Ou, J. Zheng, G. Ruan, Y. Hu, J. Zou, M. Li, S. Yang, X. Tan, A pareto-based evolutionary algorithm using decomposition and truncation for dynamic multi-objective optimization, Applied Soft Computing (2019) 105673.

[30] S. Jiang, S. Yang, Evolutionary dynamic multiobjective optimization: Benchmarks and algorithm comparisons, IEEE Transactions on Cybernetics PP (99) (2016) 1–14.

[31] Y. Wu, Y. Jin, X. Liu, A directed search strategy for evolutionary dynamic multiobjective optimization, Springer-Verlag, 2015.

[32] R. Chai, A. Savvaris, A. Tsourdos, Y. Xia, S. Chai, Solving multiobjective constrained trajectory optimization problem by an extended evolutionary algorithm, IEEE transactions on cybernetics.

[33] M. Farina, K. Deb, P. Amato, Dynamic multiobjective optimization problems: test cases, approximations, and applications, Evolutionary Computation, IEEE Transactions on 8 (5) (2004) 425–442.

[34] A. Zhou, Estimation of distribution algorithms for continuous multiobjective optimization, Ph.D. thesis, University of Essex (2009).

[35] W. T. Koo, C. K. Goh, K. C. Tan, A predictive gradient strategy for multiobjective evolutionary algorithms in a fast changing environment, Memetic Computing 2 (2) (2010) 87–110.

[36] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics bulletin 1 (6) (1945) 80–83.

[37] M. Jiang, Z. Huang, L. Qiu, W. Huang, G. G. Yen, Transfer learning-based dynamic multiobjective optimization algorithms, IEEE Transactions on Evolutionary Computation 22 (4) (2018) 501–514.

[38] M. Li, L. Zhen, X. Yao, How to read many-objective solution sets in parallel coordinates [educational forum], IEEE Computational Intelligence Magazine 12 (4) (2017) 88–100.