An environmental selection and transfer learning-based dynamic multiobjective optimization evolutionary algorithm

Qiang He · Zheng Xiang · Peng Ren

Received: 11 March 2021 / Accepted: 26 September 2021 / Published online: 12 January 2022
© The Author(s), under exclusive licence to Springer Nature B.V. 2022

Abstract In recent years, the dynamic multiobjective optimization problems (DMOPs), whose major strategy is to track the varying PS (Pareto Optimal Solution, PS) and/or PF (Pareto Optimal Frontier), caused a great deal of attention worldwide. As a promising solution, reusing of “experiences” to establish a prediction model is proved to be very useful and widely used in practice. However, most existing methods overlook the importance of environmental selection in the evolutionary processes. In this paper, we propose a dynamic multiobjective optimal evolutionary algorithm which is based on environmental selection and transfer learning (DMOEA-ESTL). This strategy makes full use of the environmental selection and transfer learning techniques to generate individuals for a new environment by reusing the experience to maintain the diversity of the population and speed up the evolutionary process. As experimental validation, we embed this new scheme in the NSGA-II (non-dominated sorting genetic algorithm). We test the proposed algorithm with the help of six benchmark functions as well as compare it with the other two prediction based strategies FPS (Forward-looking Prediction Strategy, FPS) and PPS (Population Prediction Strategy, PPS). The experimental results testify that the proposed strategy can deal with the DMOPs effectively.

Keywords Artificial intelligence · Dynamic multiobjective optimization problems (DMOPs) · Evolutionary algorithms (EAs) · Environmental selection · Transfer learning

1 Introduction

A multiobjective optimization problem (MOP), which means an optimization problem with multiple objectives, is a major and widespread problem in both industrial applications and scientific research. Because of the changing of the objective functions and/or constraints, we call these problems as dynamic multiobjective optimization problems (DMOPs). Moreover, the change of PSs in decision space and/or the PFs in objective space is unpredictable or not observable, which makes the optimization problems more complex.

In recent years, there has been a large amount of literature on DMOPs. These literature show that it is widely used in engineering and research on DMOPs in various application areas [1–10]. Meanwhile, some other scientific problems, such as intelligent learning [11], bi-level optimization [12], can be solved by means of DMOPs. The DMOPs have been one of the
hottest research topics, and many advanced control algorithms have been proposed in the late years. As so far, many new strategies have been proposed to tackle DMOPs, such as initialization, reference points, Kalman filter, dynamic learning et al. All these new methods are static methods based. However, various new theoretical and practical have proved that traditional optimization strategies may be hard to solve these complex problems efficiently.

Recently, evolutionary algorithms (EAs) are more and more popular in dealing with DMOPs partly because of the successful application of EAs in dealing with static MOPs. The EAs, which were proposed by Fogel in the 1960s [13], have been widespread concerned in the last few years as they can perform the global search without differentiability of the objective functions [14]. In addition, EAs are very efficient in dealing with DMOPs [15, 16], such as video surveillance systems, control systems [17, 18], design, and so on. According to [19], a Pareto-based evolutionary algorithm using decomposition and truncation for dynamic multi-objective optimization is proposed. In order to solve the DMOPs by the evolutionary algorithm, this paper proposes a novel mating selection strategy, an efficient environmental selection technique, and an effective dynamic response mechanism. In reference [20], an evolutionary algorithm based on Minkowski distance is proposed. This new strategy can solve the problem of individual selection caused by a large number of nondominated solutions in the evolutionary algorithm. The algorithms combined with EAs are named dynamic multi-objective optimization evolutionary algorithms (DMOEAs). As so far, most of the current DMOEAs are in their early stage of development.

As we know, the environment plays three different functions in the evolutionary process, and they are constraint, facilitating, and guiding. In one word, the main function of the environment is to guide the population to evolve in a better direction. At the same time, the evolutionary population also affects the evolutionary environment, such as the changes of the current evolutionary state, the update of environmental knowledge, etc. In the process of dynamic evolution, how to achieve the diversity and distribution of the population is the key to deal with the DMOPs. So, the method how to help the population to be reinitialized according to the new environment and enhance population diversity is important.

Significantly, the prediction-based strategy performs very well in dealing with DMOPs. The cardinal idea of these methods is following the tracks and reusing the good solutions when the environmental change occurs. It was proven theoretically and practically that this method was effective to deal with the DMOPs especially for those problems that the solutions obey an identical distribution.

The field application of traditional machine learning already confirmed that it has a poor performance to predict the data when the data that used to train and the predicted data are not conform to the independent identical distribution. In recent years, a new prediction strategy based on transfer learning [21] was proposed. In transfer learning strategy, the distribution of data that used in training and testing is different. However, prediction performance has been greatly improved by this strategy, and this method has been more and more popular. Therefore, transfer learning, as an effective method in dealing with these complex problems, has played a significant role in solving of DMOPs.

In this paper, we propose a new strategy entitled DOMEA-ESTL, which integrates the environmental selection and transfer learning schemes. Our motivation is to get the solutions of a dynamic multiobjective optimization problem in a new environment by the advantages of environmental selection and transfer learning, which can not only maintain the distribution characteristics of the population but also accelerate the convergence of the population. Firstly, we introduce the environmental selection into this system, which can help to get some guide individuals by the environmental evaluate mechanism, and then divide individuals into two parts (with change and without change) according to the environmental regulation. Secondly, we predict the guide individuals in a new environment for the optimization function according to the individual characteristics (predict the individual in a new environment by transfer learning for individuals with change or use the individual as the new guide individuals in a new environment after correction for individuals without change). Thirdly, we get the optimal population in a new environment by the guide-individuals generated in the second step with NSGA-II scheme. Experimental results show that this new strategy can effectively solve DMOPs. Compared with traditional method, this new scheme can not only enhance population diversity, and make the population respond quickly to the
different degrees of environmental changes, but also
can accelerate the convergence of population by
“experiences”.

The main contributions of this paper are summa-
rized as follows:

(1) We consider an environmental selection strat-
cy which can maintain population diversity.
The environment, which plays a role of con-
straint, facilitating and guiding in the evolution
process of the population, can help to establish a
dynamic environment evolutionary model
which can help to improve the initiative and
instruction of population search. Therefore, the
environmental selection strategy improves the
performance by guided fashion, and it makes the
population can respond quickly to the environ-
mental changes. So, in this paper, we take
advantage of the environmental selection strat-
cy to guide the evolution of population in a
new environment.

(2) We introduce transfer learning into this strategy,
which can take advantage of the nonindepend-
et and identically distributed nature of data, to
construct a prediction model. According to the
transfer learning strategy, it can accelerate the
convergence of the population. This new
scheme makes full use of the “experiences”,
which is also used in memory-based method,
and it significantly improves the search effi-
ciency in solving the DMOPs.

This paper is organized as follows. Section 2
presents some preliminary studies and related works.
Section 3 proposes the strategy based on environ-
mental selection and transfer learning in detail. Test
instances and performance metrics will be given in
Sect. 4. In Sect. 4, experimental results are also
presented and analyzed. Finally, Sect. 5 sums up
some conclusions and gives some suggestions as the
future research topics.

2 Preliminary studies and research

In this section, some basic theories used in this paper
are introduced. The purpose is to introduce some basic
knowledge and give certain background for readers to
understand the sequel.

2.1 Dynamic multiobjective optimization

According to the characteristics of the nondetermi-
nacy in dynamic multiobjective optimization prob-
lems, it can be divided into different categories
\cite{22, 23}. Here, we focus on DMOPs that can be
described as formula (1).

\[
\begin{align*}
\text{minimize} & \quad F(x, t) = (f_1(x, t), f_2(x, t), \ldots, f_m(x, t))^T \\
\text{subject to} & \quad x \in \prod_{i=1}^{n} [a_i, b_i]
\end{align*}
\]

in which \( t \in \mathbb{N} \) represents time instants, \( m \) is the
number of objective functions, \( x = (x_1, x_2, \ldots, x_n)^T \) is
the decision vector, \(-\infty < a_i < b_i < +\infty\) for all
\( i = 1, 2, \ldots, n \), \( \prod_{i=1}^{n} [a_i, b_i] \subset \mathbb{R}^n \) gives the interval ranges
of the decision space. The objective vector is
\( F(x, t) : \mathbb{R}^n \times T \rightarrow \mathbb{R}^m \) which consists \( m \) time varying objective
functions \( f_i(x, t), i = 1, \ldots, m \). \( \mathbb{R}^m \) is the objective
space.

Definition 1 Pareto Dominance. \( p \) and \( q \) are two
values in the decision space, \( p \) is said to dominate \( q \)
which can be expressed as \( f(p) < f(q) \), if and only if
\( f_i(p) \leq f_i(q), \forall i = \{1, 2, \ldots, m\} \) and
\( f_j(p) < f_j(q), \exists j \in \{1, 2, \ldots, m\} \).

Definition 2 Pareto Optimal Solution (PS). \( x \) is the
decision variable, \( \Omega \) is the objective space, and \( F \) is the
objective vector function. Then, the PS, which is the
set of all nondominated solutions, can be denoted by
formula (2).

\[ \text{PS} := \{ x \in \Omega | \nexists x^* \in \Omega, (F(x^*) < F(x)) \} \] (2)

Definition 3 Pareto Optimal Frontier (PF). \( x \) is the
decision variable, \( F \) is the objective vector function.
The PF, which is the set of nondominated solutions
with respect to the objective space, can be denoted by
formula (3).

\[ \text{PF} := \{ y = F(x) | x \in \text{PS} \} \] (3)

2.2 Dynamic multiobjective optimization

evolutionary algorithms (DMOEAs)

It is important that a search algorithm can quickly
adapt to environmental changes in dealing with
DMOPs. Therefore, population convergence and diversity is the key issue, which is also the central research hotspots and directions for designing EAs for solving DMOPs [22]. DMOEAs can solve the problem mentioned above successfully.

Detection and reflection of environmental change, optimization calculation are considered as three major steps of most existing DMOEAs. Algorithm 1 gives a general framework of a DMOEA [23–26]. Then, we will focus on a few major technologies used in the framework of a DMOEA.

Algorithm 1: A General Framework of A DMOEA

1. Initialize the parameters, and set the time \( t = 0 \);
2. Initialize a population \( PS \);
3. while termination condition is not reached do
   4. if change==1 then /*Remark 1*/
      5. \( t = t + 1 \);
      6. \( PS_{new} = \text{MOEA}(PS) \); /*Remark2*/
   7. else
     8. Optimize the \( t \)-th MOP by a MOEA for one generation; /*Remark3*/
   9. end
10. end

Remark 1 Change detection (line 4).

A change point \( t_{cp} \) is defined as [17]:

\[
F(x, t_{cp}) \neq F(x, t_{cp} + 1) \quad (4)
\]

where \( F(x, t_{cp}) \) and \( F(x, t_{cp} + 1) \) are the objective functions at the time \( t_{cp} \) and \( t_{cp} + 1 \), respectively. With this step, a change can be detected when it occurs.

Remark 2 Change Reaction (Lines 5 and 6).

In response to the change in the environment, the system will make some corresponding responses, which can be found in this step. There are many actions described in [27–37].

Remark 3 Optimization Calculation (Line 8).

This step is used to optimize the current MOPs for one generation. A multiobjective optimization evolutionary algorithm (MOEA) is usually applied to solve stationary MOPs. The purpose of using MOEA directly or with some modifications is to enhance the diversity of the population. Three environmental elements are used to actualize the functions listing above, which are environmental knowledge, environmental evaluation, and environmental regulation in a new environment [24].

First of all, environmental knowledge is the environmental attribute in the process of evolution. There are two kinds of environmental knowledge which are static and dynamic, respectively. Static knowledge, which is constant values all over the process of evolution, consisted of environmental capacity and many other constant values. Dynamic knowledge, which is variable values, consisted of the direction of the evolution and the individuals obtained in the new environment, etc.

Then, the environmental evaluation mechanism, which is used to evaluate the living environment of the population or individuals which contain the individual location in the environment, the entire population...
distribution, etc. Environmental evaluation is used for guiding evolution.

Thirdly, environmental regulation means that individuals need to make some change in a new environment as well as the environment change.

With these three environmental elements mentioned above, it can guide and promote the evolution by the functions of constraint, facilitating and guiding when an environmental change occurs. Above all, the environmental selection mechanism can help to evolve to the desired direction according to the new environment.

3 Environmental selection and transfer learning-based dynamic multiobjective optimization evolutionary algorithm

In this section, we follow with interest how to treat the change of environment by the proposed algorithm when a change is detected. In this section, a dynamic multiobjective optimization evolutionary algorithm based on environmental selection and transfer learning (DMOEA-ESTL) is proposed.

The solutions of a dynamic optimization problem always change with the environment, and the probability distributions are also different from each other. Even though these distributions are not identical, but are correlated. It is assumed that if we transfer these individuals into a new space, in which the individuals are as “similar” as possible, we can construct an initial population by these available solutions, such that we can get the solutions for a new environment with low computational cost. In view of the above, we refer to transfer learning, which can deal with the problems in different but related fields, to improve the DMOEAs.

As mentioned above, because of the different probability distributions of the solutions of a DMOP, an invalid or even negative transfer learning will be led to if we transfer the characteristics of the individuals in a dynamic environment directly [47]. In this paper, we consider an environmental selection strategy before the transfer learning, which is used to get the environment-adapted individuals to transfer learning.

The basic principle of the DMOEA-ESTL strategy is shown in Fig. 1.

3.1 Environmental selection

The traditional dynamic multiobjective optimization algorithm can track the new PSs/PFs quickly, but it ignores the role of the dynamic environment in population evolution and lacks guidance search. From the perspective of evolution, “environment” is the sum of external conditions that an individual lives in. In this paper, the environment can be seen as a group of entities that can guide and promote the evolution of the population. In this paper, there are two reasons for using the environmental selection strategy. Firstly, this strategy guides the search of the population in a new environment by the information generated by the population in two different environments before and after the environmental change. Secondly, in order to avoid invalid or even negative transfer learning, we can select individuals with change as the source domain of transfer learning.

We can store the population in a dynamic environment by an idea which concurs with the grid environment domain [46]. In this paper, we call the whole simulated environment as environment domain which consists of many same grids that called the unit domain. The grid environment domain records the convergence and distribution information generated during population evolution jointly. The individuals in the environment domain and unit domain are shown in Fig. 2. In this figure, the dimension of the environment domain is the same as the objective dimension which also equals the unit domain. In order to get the location of the individual, we should get the bottom and top boundaries of each dimension firstly, and they are denoted by formula (5).

\[
B_i = \min(P_i) - (\max(P_i) - \min(P_i)) / (2 \times \text{num})
\]

\[
T_i = \max(P_i) - (\max(P_i) - \min(P_i)) / (2 \times \text{num})
\]

(5)

where \(B_i\) and \(T_i\) are the bottom and top boundary of the \(i\)-dimension, respectively. \(P_i\) is an individual of the population on the \(i\)-dimension, \(\max(P_i)\) is the maximum of \(P_i\), while \(\min(P_i)\) is the minimum of \(P_i\). \(\text{num}\) is the number of unit domain on the \(i\)-dimension. Then, the size of unit domain on the \(i\)-dimension \(\text{US}_i\) can be defined as formula (6).

\[
\text{US}_i = (T_i - B_i) / \text{num}
\]

(6)
The unit domain $UD_i$ of an individual $x_i$ can be denoted by formula (7).

$$UD_i = \frac{F_i - B_i}{US_i}$$

where $F_i$ is the $i$-dimension objective value.

In this paper, the dynamic environmental selection mechanism saves the information of the grid environment domain by environmental knowledge firstly. The location and the distribution of individuals in Fig. 2 will change as well as the environment. Then it can evaluate the population with the environmental evaluation mechanism, which can calculate the difference of the unit domain before and after the environmental change. After that, we can get a serial of guide-individuals by these individuals with good evaluation results. These guide-individuals can be defined as:

$$g_i^t = x_i^t + \frac{W_i^t - W_i^{t-1}}{G}, \text{ if } W_i^t - W_i^{t-1} > 0$$

$$g_i^t = x_i^t - \frac{W_i^t - W_i^{t-1}}{G}, \text{ if } W_i^t - W_i^{t-1} < 0$$

(8)

where $g_i^t$ is the guide-individual, $x_i^t$ denotes the individuals obtained at time $t$. $i = 1, 2, \ldots, n$ is the dimensions of the decision space. $G$ is a random number with a mean value of 0 and a variance of 1. $W_i^t$ is the center of non-dominated solutions obtained at time $t$.

For a changed environment, there may be different regulations. So the population should make the corresponding change. According to [46], in order to make the corresponding change for different individuals, the population in an environment was divided into three parts which are shown in Fig. 3. Then, in this paper, we consider two subpopulations which are subpopulation1 (with change) and subpopulation2 (without change). The individuals of subpopulation2 (without change) are non-dominated individuals who have the largest crowding distance. The rest individuals of the population belong to subpopulation2. With the subpopulation1 (with change), we can get the guide-individuals in a new environment by the transfer learning strategy. With the subpopulation2 (without change), we can get the guide-individuals in a new environment corrected by formula (8).

3.2 Transfer learning

In this section, we adopt a new machine learning method, which is called transfer learning (TL). In principle, TL is a new method that memory-based which can reuse the knowledge that we have already obtained. So, by this model, it can generate individuals for the optimization problem at a new time by the gained knowledge of finding PSs. Based on this strategy, the solutions of a DMOP can be obtained, and the process will be efficiently and effectively. For this new strategy, it is important to determine the transfer component.

1. Transfer Component Analysis (TCA)

TCA is a dimension reduction-based method that can transform the problem of learning an entire kernel matrix to a low-rank matrix $W$ instead. Then, the rest of this part will introduce how to get the matrix $W$ by TCA.

It is assumed that $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$ are two random variables whose probability distributions are $P$ and $Q$ on a domain $\mathcal{X}$, respectively. So far, there are many criteria to measure the difference between different distributions, which are related to, but distinct from each other. Gretton, Smola, and their other partners proposed a nonparametric distance estimation scheme called maximum mean discrepancy (MMD) to distinguish distributions in the reproducing kernel Hilbert space (RKHS).
According to \cite{48–50}, the MMD is defined as:

\[
\text{Dist } X, Y = \frac{1}{n_1} \sum_{i=1}^{n_1} \phi(x_i) - \frac{1}{n_2} \sum_{i=1}^{n_2} \phi(y_i)
\]

where \(H\) is the RKHS. By the so-called "kernel trick", and we introduce a kernel function \(K\), then we can rewrite (9) as formula (10).

\[
\text{Dist}(X_s, Y) = \frac{1}{n_1} \sum_{i=1}^{n_1} \sum_{j=1}^{n_1} k(x_s, x_y) + \frac{1}{n_2} \sum_{i=1}^{n_2} \sum_{j=1}^{n_2} k(x_t, x_t) - \frac{1}{n_1n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} k(x_s, x_t)
\]

\[
= \text{tr}(KL)
\]

\[
K = \begin{bmatrix}
K_{S,S} & K_{S,T} \\
K_{T,S} & K_{T,T}
\end{bmatrix}
\]

where \(X_s = \{x'_s\} = \{\phi(x_t')\}\) and \(X_t = \{x'_t\} = \{\phi(x_t)\}\) are the source domain and target domain, respectively. \(K_{X,Y}\) is a kernel matrix with \(k_{ij} = \kappa(x_i, y_i) = \phi(x_i)^T \phi(y_i)\). By the so-called "empirical kernel map", we can rewrite (10) as formula (13).

\[
L = [L_{ij}] = \begin{cases}
\frac{1}{n_1} & x_i, x_j \in X \\
\frac{1}{n_2} & x_i, x_j \in Y \\
-\frac{1}{n_1n_2} & \text{otherwise}
\end{cases}
\]

\[
\text{Dist}(X'_s, X'_t) = \text{tr}((KWKW^TK)L) = \text{tr}(W^TKKW)
\]

When minimizing formula (13), it needs a regularization term \(\text{tr}(W^TW)\). Thus, the problem can be changed into formula (14).

\[
\begin{align*}
\min_W & \text{tr}(W^TW) + \mu \cdot \text{tr}(W^TKKW) \\
\text{s.t.} & \quad W^TKKW = I
\end{align*}
\]

\[
H = I_{n_1+n_2} - \frac{1}{n_1+n_2} PP^T
\]

where \(I\) is a \((m + n) \times (m + n)\) unit matrix. \(W^TW\) is a regularization term. \(P\) is a \((m + n) \times 1\) matrix whose elements are all one. \(m\) and \(n\) are the number of samples in the source and objective domains, respectively. \(\mu\) is the tradeoff parameter.

2. DMOEA-TL

In this section, the pseudo-codes of DMOEA-TL algorithm, which is a transfer learning-based scheme that can help to search for the PFs at a new time, is given. More specifically, we take the historical data of PFs that obtained at time \(t\) that generated by transfer learning method as a source domain, the object solutions of the next time (time \(t + 1\)), as the objective domain, and then construct a mapping function \(\phi(.)\) by the domain adaptation approach. This mapping function will embed the distributions that the source and objective domain obey separately.
into a latent space, and in that space, the difference between the two distributions will be as small as possible.

The pseudo-code of DMOEA-TL algorithm can be found in algorithm 2. According to the DMOEA-TL scheme, the PFs at a new time can be obtained.

### Algorithm 2: DMOEA-TL

**Input:** The function to be optimized \( F_{t+1}(\cdot) \); the Optimal solution of \( F_t(\cdot) \) at time \( t \), \( PF_t = \{p_1, \ldots, p_m\} \), a kernel function \( \kappa(\cdot, \cdot) \).

**Output:** A population Pop-reinit.

1. Initialization, parameter setting;
2. For \( F_t(\cdot) \) and \( F_{t+1}(\cdot) \), randomly set two sets of the solutions \( X_S \) and \( Y_T \); /*Remark 1*/
3. Calculating the values of \( F_t(X_S) \) and \( F_{t+1}(Y_T) \);
4. \( W = TCA(\{F_t(X_S)\}, \{F_{t+1}(Y_T)\}, \kappa) \);
5. Suppose that \( PLS = \emptyset \); /*Remark 2*/
6. for every \( p \in PF_t \) do
7. \( \kappa_p = [\kappa(F_t(X_S(1)), p), \ldots, \kappa(F_{t+1}(Y_T(n_t)), p)]^T \);
8. \( \phi(p) \leftarrow W^T \kappa_p \);
9. \( PLS = PLS \cup \{\phi(p)\} \);
10. end
11. for every \( j \in PLS \) do
12. \( x = \arg \min \|\phi(F_{t+1}(x)) - j\| \); /*Remark 3*/
13. Pop-reinit = Pop-reinit \( \cup \{x\} \);
14. end
15. return Pop-reinit;

**Remark 1** Generally speaking, the number of the components of \( X_S \) and \( Y_t \) are predefined. It is assumed that \( |X_S| = n_S \) and \( |Y_t| = n_t \). In general, the more number of sampling, the better prediction result but higher cost will be, so the number of solutions to be generated in this step usually depends on the available resources.

**Remark 2** \( PLS \) is the abbreviation for particle in the latent space. It can be regarded as a set of mapped solutions in the latent space.

**Remark 3** In this step, our motivation is to get an individual \( x \), such that the result \( \phi(F_{t+1}(x)) \) is as close as possible to \( j \in PLS \) in the latent space. In this re-

gard, it is worthy to first solve a single objective optimization problem, and the solutions can be used in dealing with DMOPs.

Figure 4 illustrates the key steps of the DMOEA-TL algorithm, which can help readers quickly catch the key thought of this strategy. What needs to be emphasized is that the input to the TCA are the samples of the solutions at time \( t \) and \( t+1 \), and its
output is a transformation matrix $W$, by which we can construct the latent space. Step 1, it depicts the method how to map the PF at time $t$ into the latent space, then steps 2 and 3 describe the process of searching for an initial population which can be used as the solution of the dynamic optimization problem at a new time of $t + 1$.

### 3.3 DMOEA-ESTL

In this section, an environmental selection and transfer learning-based DMOEA which is called DMOEA-ESTL will be given. Based on DMOEA-TL, the DMOEA-ESTL introduces the environmental selection mechanism before transfer learning. This method combines the advantages of environmental selection and transfer learning. Theoretically, the environmental selection scheme can guide the evolution of individuals by the function of constraint, facilitating and guiding, and the transfer learning makes full use of the “past experience” to get the new individuals when the change is detected. Therefore, the DMOEA-ESTL is a very effective strategy to solve the DMOPs. The principle of DMOEA-ESTL is shown in Fig. 5. As a case of study, we select a well-represented algorithm NSGA-II to verify our approach. The pseudo-codes of DMOEA-ESTL are given in algorithm 3.
4 Empirical study

4.1 Test instances

The benchmark function is important for the test and design of the algorithm. There are many typical DMOPs in [14] and [51], and many variants of these problems in [26] and [52]. In this experiment, there are six standard functions, and all of these functions are selected from various DMOPs types [53] which can be used to verify the performance of DMOPs.

These functions can be divided into two categories. The first one belongs to DMOP functions, that are DMOP1 ~ DMOP3, which are functions with the linear correlation between the decision variables. The second one belongs to FDA functions, that are FDA4, FDA6, and FDA7, which are three functions with the linear correlation and two functions with nonlinear correlation between the decision variables, respectively. Table 1 gives all these six standard test functions and their PF and PS in detail.

4.2 Performance indications for dynamic optimization

In order to evaluate the performance of the new scheme, a modified version of the inverse generational distance, abbreviated as IGD, is used here. It is a widely used metric for evaluating the performance of DMOEAs [14, 15] which can measure both convergence and distribution of the solution set obtained by a DMOEA. The IGD metric is defined by the formula (16).

$$\text{IGD}(R^c, R^l) = \frac{\sum_{v \in R^c} d(v, R^l)}{|R^c|}$$  

(16)

where $R^c$ is a set that contains the solutions uniformly sampled from the true PF, and $R^l$ is the non-dominated solution set achieved by a DMOEA. $d(v, R^l)$ is the
| Instance | Search space | Objectives, PS and PF | Remarks |
|----------|--------------|-----------------------|---------|
| DMOP1    | $[0, 1] \times [-1, 1]^{n-1}$ | $f_1(x, t) = x_1, f_2(x, t) = g \times h$ | Two objectives, PF changes |
|          |              | $g = 1 + 9 \sum_{i=2}^{n} x_i^2, h = 1 - \left( \frac{f_1}{g} \right)^{\frac{1}{2}}$ | PS fixed |
|          |              | $H = 2.5 + 0.75 \sin(0.5 \pi t), t = [\pi/\tau]/n_T$ | |
|          |              | $PS(t) : 0 \leq x_1 \leq 1, x_i = 0, i = 2, \ldots, n$ | |
|          |              | $PF(t) : f_1 = 1 - \sqrt{f_2}, 0 \leq f_1 \leq 1$ | |
| DMOP2    | $[0, 1] \times [-1, 1]^{n-1}$ | $f_1(x, t) = x_1, f_2(x, t) = g \times h, t = [\pi/\tau]/n_T$ | Two objectives, PF changes |
|          |              | $g = 1 + \sum_{i=2}^{n} (x_i - G)^2, h = 1 - \left( \frac{f_1}{g} \right)^{\frac{1}{2}}$ | PS changes |
|          |              | $G = \sin(0.5 \pi t), H = 2.5 + 0.75 \sin(0.5 \pi t)$ | |
|          |              | $PS(t) : 0 \leq x_1 \leq 1, x_i = G, i = 2, \ldots, n$ | |
|          |              | $PF(t) : f_2 = 1 - \sqrt{f_1}, 0 \leq f_1 \leq 1$ | |
| DMOP3    | $[0, 1] \times [-1, 1]^{n-1}$ | $f_1(x, t) = x_1, f_2(x, t) = g \times h, t = [\pi/\tau]/n_T$ | Two objectives, PF fixed |
|          |              | $g = 1 + \sum_{i=2}^{n} (x_i - G)^2, h = 1 - \left( \frac{f_1}{g} \right)^{\frac{1}{2}}$ | PS changes |
|          |              | $G = \sin(0.5 \pi t), H = 2.5 + 0.75 \sin(0.5 \pi t)$ | |
|          |              | $PS(t) : 0 \leq x_1 \leq 1, x_i = G, i = 2, \ldots, n$ | |
|          |              | $PF(t) : f_2 = 1 - \sqrt{f_1}, 0 \leq f_1 \leq 1$ | |
| FDA4     | $[0, 1]^n$ | $f_1(x, t) = (1 + g) \cos(0.5 \pi x_1) \cos(0.5 \pi x_1)$ | Three objectives, PF is fixed |
|          |              | $f_2(x, t) = (1 + g) \cos(0.5 \pi x_1) \sin(0.5 \pi x_1)$ | |
|          |              | $f_3(x, t) = (1 + g) \sin(0.5 \pi x_1), t = [\pi/\tau]/n_T$ | |
|          |              | $g(x, t) = \sum_{i=2}^{n} (x_i - G)^2, G = \sin(0.5 \pi x_1)$ | |
|          |              | $PS(t) : 0 \leq x_1 \leq 1, x_i = G, i = 2, \ldots, n$ | |
|          |              | $PF(t) : f_1 = \cos(\mu) \cos(\nu), f_2 = \cos(\mu) \sin(\nu), f_3 = \sin(\nu), 0 < \mu, \nu < \pi/2$ | |
| FDA6     | $[0, 5]^n$ | $f_1(x, t) = |x_1 - a|^{\frac{3}{2}} + \sum_{i=2}^{n} y_i^2, a = 2 \cos(1.5 \pi t) \sin(0.5 \pi t) + 2$ | Two objectives, PF changes |
|          |              | $f_2(x, t) = |x_1 - a|^{\frac{3}{2}} + \sum_{i=2}^{n} y_i^2, b = 2 \cos(1.5 \pi t) \cos(0.5 \pi t) + 2$ | |
|          |              | $y_i = x_i - b = 1 + |x_1 - a|^{1/2}, H = 2.5 + 0.75 \sin(0.5 \pi t), t = [\pi/\tau]/n_T, i = 2, \ldots, n$ | |
|          |              | $L_i = [i]1 \leq i \leq n, \text{isodd}, H_i = [i]1 \leq i \leq n, \text{iseven}$ | |
|          |              | $PS(t) : a \leq x_1 \leq a + 1, x_i = b + 1 - |x_1 - a|^{1/2}$ | |
|          |              | $PF(t) : f_2 = (1 - S)^{H_i} f_1 = S^H, 0 \leq S \leq 1$ | |
| FDA7     | $[0, 5]^n$ | $f_1(x, t) = |x_1 - a|^{\frac{3}{2}} + \sum_{i=2}^{n} y_i^2, f_2(x, t) = |x_1 - a|^{\frac{3}{2}} + \sum_{i=2}^{n} y_i^2$ | Two objectives, PF changes |
|          |              | $y_i = x_i - b = 1 + |x_1 - a|^{1/2}, H = 2.5 + 0.75 \sin(0.5 \pi t)$ | |
|          |              | $a = 1.71(1 - \sin(\pi t)) + 3.4, b = 1.4(1 - \sin(\pi t)) \cos(\pi t) + 2.1$ | |
|          |              | $L_i = [i]1 \leq i \leq n, \text{isodd}, H_i = [i]1 \leq i \leq n, \text{iseven}, t = [\pi/\tau]/n_T$ | |
|          |              | $PS(t) : a \leq x_1 \leq a + 1, x_i = b + 1 - |x_1 - a|^{1/2}, i = 2, \ldots, n$ | |
|          |              | $PF(t) : f_2 = (1 - S)^{H_i} f_1 = S^H, 0 \leq S \leq 1$ | |
minimum distance from \( v \) and a solution in \( R^s \). The IGD metric is also called D-metric. The ideal IGD value is zero, which means that it has obtained the best convergence and diversity.

One variant of the IGD, called MIGD, can also be used to evaluate dynamic multiobjective optimization algorithms [25], and it takes the mean of the IGD values in some time steps over a run as the performance metric, given by formula (17).

\[
\text{MIGD}(R^s, R^t) = \frac{1}{|T|} \sum_{t \in T} \text{IGD}(R^s, R^t)
\]

(17)

where \( T \) is a set of discrete-time points in a run, and \(|T|\) is the cardinality of \( T \).

4.3 Parameter settings

The performance of the algorithm must be proved by experiments, so two recently proposed prediction based methods, namely, the population prediction strategy (PPS) [26], and a forward-looking prediction strategy (FPS) [54], for dealing with DMOPs is compared with the proposed strategy DMOEA-ESTL in the experimental studies. For a fair comparison, all these three strategies are embedded into NSGA-II.
4.4 Numerical tests and analysis

The shape of PSs and PFs of each test function is different from each other, and they belong to different types. Figures 6 and 7 describe the true PS and PF of the six testing functions. In Fig. 6, the horizontal axis and the vertical axis are $x_1$ and $x_2$ which can be found in $\text{PS}(t)$. According to the specific benchmark function, the $\text{PS}(t)$ is different from each other. The detailed value of $\text{PS}(t)$ is shown in Table 1. Similarly, in Fig. 7, the horizontal axis and the vertical axis are $f_1$ and $f_2$ which can be found in $\text{PF}(t)$. According to the specific benchmark function, the $\text{PF}(t)$ is different from each other. The detailed value of $\text{PF}(t)$ is shown in Table 1.

| Algorithm name          | DMOP1       | FDA6       |
|-------------------------|-------------|------------|
| FPS                     | 7.32E–3     | 3.69E–2    |
| PPS                     | 5.26E–3     | 3.61E–2    |
| DMOEA-ESTL              | 4.98E–3     | 3.27E–2    |
Fig. 10  Average MIGD of Three Algorithms.

a DMOP1.  b DMOP2.
c DMOP3.  d FDA4.
e FDA6.  f FDA7
Figure 8 gives the results of population obtained by FPS, PPS, and DMOEA-ESTL at $t = 125, 130, 135, 140, 145, 150$ among 20 runs on DMOP1. Then, Fig. 9 gives the results of population obtained by FPS, PPS and DMOEA-ESTL at $t = 143, 145, 147, 149, 151, 153$ among 20 runs on FDA6. In order to further verify the performance of these three strategies, Table 3 gives the average error values of these three algorithms on DMOP1 and FDA6. The average error represents the average deviation of the obtained PFs from their true values. According to Figs. 8, 9, and Table 3, we can draw that the DMOEA-ESTL can approximate the PFs very well for these two functions, and the performance of the FPS and PPS is slightly worse than DMOEA-ESTL.

Figure 10 shows the average MIGD of the three algorithms FPS, PPS, and DMOEA-ESTL on six test functions over 20 runs. In this simulation system, we consider 300 evolutionary generations, and the environmental change occurs every 30 generations. Then, all these experimental results of the statistical average MIGD are given in Table 4.

It must be emphasized that all of these results are obtained without tuning any one of the parameters, which means that the experimental conditions are the same. The main reason for not adjusting the parameters is that we can get better performance by tuning any one of the parameters. So, we can get a fair result with the same parameters.

According to the experimental results, we can draw the following conclusions:

1. For this initial phase, the performance of these three algorithms is relatively poor. The reason might be that the lacking of historical information at the beginning. So, when $1 < t \leq 20$, the MIGD of these three algorithms is much larger than that of other time. When $t > 20$ the performance of these three algorithms is gradually improved with the passage of time. The reason might be that there is sufficient data after evolutionary operation.

2. An obvious conclusion is that the average MIGD values of DMOEA-ESTL are less than PPS and FPS, especially in the range of $0 < t \leq 20$. The observation means that DMOEA-ESTL can perform better performances when environmental changes are detected.

3. As time goes on, the environment will change periodically, and the experience will be accumulated gradually. So, the convergence and diversity of PPS will gradually stabilized, and the reason is the combination of memory scheme. The performance of PPS will be much better than that of DMOEA-ESTL. In a word, the difference of these two strategies is small, and the performance of DMOEA-ESTL is better than that of FPS.

4. DMOP1 is the problem with a fixed PS. Theoretically, the diversity and convergence of the population are contradictory, and they can not reach the optimal at the same time. The performance is much better than that of other testing functions which indicates that the using of historical data can improve the performance of convergence.

5. For these three algorithms, the performance of DMOEA-ESTL scheme is not worse than that of the other two strategies, which is because that the DMOEA-ESTL takes advantage of the transfer learning which can achieve better learning results in the absence of source data to generate new individuals in a new environment.

6. As shown in Fig. 10, the convergence time of the DMOEA-ESTL strategy is the shortest for all these experiments. For example, the convergence time of DMOEA-ESTL on DMOP1 is close to $t = 5$, and the convergence time of the other four testing functions are also in the range of $t < 20$, which is shorter than that of the other two (FPS and PPS) strategies.

The DMOEA-ESTL strategy combines environmental selection and transfer learning, and then introduces them into an evolutionary algorithm NSGA-II. The main computation time is focused on the transfer learning strategy and evolutionary algorithm. According to the transfer learning scheme, the

| MIGD  | FPS     | PPS     | DMOEA-ESTL |
|-------|---------|---------|------------|
| DMOP1 | 7.56E-3 | 5.14E-3 | 5.12E-3    |
| DMOP2 | 5.46E-2 | 5.16E-2 | 3.31E-2    |
| DMOP3 | 3.68E-2 | 3.02E-2 | 2.98E-2    |
| FDA4  | 9.76E-2 | 9.32E-2 | 9.19E-2    |
| FDA6  | 3.87E-2 | 3.58E-2 | 3.22E-2    |
| FDA7  | 6.88E-2 | 3.88E-2 | 2.95E-2    |
Fig. 10 continued

(d) FDA4

(e) FDA6

(f) FDA7
computation time complexity is $O(m \times (n_1 \times n_2 + n_2 \times n_3))$, where $m$ is the number of training samples, $n_1, n_2, n_3$ are the number of neurons of input layer, middle layer, and output layer. According to the evolutionary algorithm, the maximum computation time complexity is $O(n^3)$, where $n$ is the number of individuals. In this paper, According to Fig. 10, when the MIGD is stable, the iterations number of the proposed algorithm is 5 which is almost 20 for the FPS and PPS. So, we can draw that the DMOEA-ESTL scheme has a better performance of computation time complexity.

In summary, these experimental results show that the DMOEA-ESTL performs better than that of the other two algorithms on the six testing functions. It also shows that the combination of environmental selection and transfer learning can improve the performance of the existing multiobjective EAs appreciably without significant modifications for solving the DMOPs. There we summed up the reason for the results. The environmental selection of DMOEA-ESTL divides the current population into two parts (with change and without change) according to the different behavioral characteristics of each individual when a change is detected. According to different environmental changes, these two subpopulations can be treated differently. Meanwhile, with the transfer learning, the subpopulation with change will be guided to evolve toward their desired environments according to the new environmental knowledge and regulation. Therefore, DMOEA-ESTL is able to respond more quickly to environmental change and can perform better than the other two algorithms.

5 Conclusion and future work

In this paper, a new prediction-based strategy, which is called DMOEA-ESTL, is proposed. The motivation of this method is to track the changing PS and/or PF of dynamic multiobjective optimization problems. The main idea of this strategy is to take advantage of the information of the population in a new environment and transfer the main characteristics of the population to accelerate tracking capability for PS and/or PF. There are three main steps for this algorithm. The first one is to get some guide-individuals by the environmental evaluation mechanism of the environment selection described in Sect. 3.1. Secondly, according to the characteristics of the different individuals, the guide individuals in a new environment are predicted by the transfer learning scheme, and the detailed method is described in Sect. 3.2. Thirdly, the optimal population in a new environment will be get by the guide individuals generated in the above two steps with the help of the NSGA-II scheme. The DMOEA-ESTL is designed to construct the initial population in order to realize the balance between convergence and diversity. The main contribution of this paper is the combination of the environmental selection and transfer learning method which can not only make full use of the guiding role of environmental selection in the process of population evolution but also can solve the problem of reusing the “experience” of the population before and after environmental changes by the transfer learning scheme.

For experimental verification, it shows that the DMOEA-ESTL scheme performs better than the other two prediction-based strategies FPS and PPS. Although DMOEA-ESTL has been demonstrated can outperform the other two strategies on problems with dramatic changes in PF and/or PS or with complex correlation relationships between the decision variables, but it is also expected to examine its performance against more other types of DMOEAs. Therefore, it is called an urgent need to design a strategy that can solve many different types of multiobjective optimization problems simultaneously.

Author contributions For Conceptualization, QH and PR; methodology, all authors have contributed to the methodology presented in this article; validation, PR and ZX; formal analysis, QH; investigation, QH; resources, ZX; writing, all authors have contributed to writing of this article.

Funding This research received no external funding.

Availability of data and material The datasets used in this paper are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare no conflict of interest.
References

1. Jiao, R., Zeng, S., Li, C., Pedrycz, W.: Evolutionary constrained multi-objective optimization using NSGA-II with dynamic constraint handling. In: 2019 IEEE Congress on Evolutionary Computation (CEC). IEEE (2019)

2. Abello, M.B., Bui, L.T., Michalewicz, Z.: An adaptive approach for solving dynamic scheduling with time-varying number of tasks-Part II. In: 2011 IEEE Congress of Evolutionary Computation (CEC). IEEE (2011)

3. Ma, X., Yang, J., Sun, H., Hu, Z., Wei, L.: Multiregional co-evolutionary algorithm for dynamic multiobjective optimization. Inf. Sci. 545, 1–24 (2020)

4. Tang, J., Alam, S., Lokan, C., Abbass, H.A.: A multi-objective evolutionary method for dynamic airspace re-territorization using sectors clipping and similarities. In: 2012 IEEE Congress on Evolutionary Computation. IEEE (2012)

5. Cao, L., Xu, L., Goodman, E.D., Li, H.: Decomposition-based evolutionary dynamic multi-objective optimization using a difference model. Appl. Soft Comput. 76, 473–490 (2019)

6. Bui, L.T., Michalewicz, Z.: An evolutionary multi-objective approach for dynamic mission planning. In: Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2010, Barcelona, Spain. IEEE (2010)

7. Curtis, S.K., Mattson, C.A., Hancock, B.J.: Divergent exploration in design with a dynamic multiobjective optimization formulation. Struct. Multidiscip. Optim. 47(5), 645–657 (2013)

8. Jun, H., Liqian, X., Cong-Cong, X., Qiang, D.: A novel bioinspired multiobjective optimization algorithm for designing wireless sensor networks in the internet of things. J. Sens. 2015, 1–16 (2015)

9. Macedo, L.L., Godinho, P., Alves, M.J.: Mean-semivariance portfolio optimization with multiobjective evolutionary algorithms and technical analysis rules. Expert Syst. Appl. 79, 33–43 (2017)

10. Zhang, Z.: Multiobjective optimization immune algorithm in dynamic environments and its application to greenhouse control. Appl. Soft Comput. J. 8(2), 959–971 (2008)

11. Kim, K., Mckay, R., Moon, B.R.: Multiobjective evolutionary algorithms for dynamic social network clustering. In: Genetic & Evolutionary Computation Conference, p.1179. DBLP (2010)

12. Linmala, M.: Bi-level optimization for a dynamic multi-objective problem. Eng. Optim. 44(2), 195–207 (2012)

13. Fogel, L.J., Owens, A.J.: Artificial intelligence through simulated evolution. Biophys. Cybern. Syst. (1966)

14. Liu, R., Yang, P., Liu, J.: A dynamic multi-objective evolutionary algorithm for complex environmental changes. Knowl.-Based Syst. 2, 106612 (2021)

15. Yang, S., Nguyen, T.T., Li, C.: Evolutionary dynamic optimization: test and evaluation environments. In: Evolutionary Computation for Dynamic Optimization Problems, pp. 3–37 (2013)

16. Rong, M., Gong, D., Pedrycz, W., Wang, L.: A multi-model prediction method for dynamic multi-objective evolutionary optimization. IEEE Trans. Evol. Comput. 24(2), 290–304 (2019)

17. Zhou, A., Qu, B.Y., Li, H., Zhao, S.Z., Suganthan, P.N., Zhang, Q.: Multiobjective evolutionary algorithms: a survey of the state of the art. Swarm Evol. Comput. 1(1), 32–49 (2011)

18. Cheng, R., Jin, Y., Narukawa, K., Sendhoff, B.: A multi-objective evolutionary algorithm using gaussian process-based inverse modeling. IEEE Trans. Evol. Comput. 19(6), 838–856 (2015)

19. Ou, J., Zheng, J., Gan, R., et al.: A pareto-based evolutionary algorithm using decomposition and truncation for dynamic multi-objective optimization. Appl. Soft Comput. 85, 105673 (2019)

20. Hang, X., Zeng, W., Zeng, X., et al.: An evolutionary algorithm based on minkowski distance for many-objective optimization. IEEE Trans. Cybern. 49(11), 3968–3979 (2019)

21. Alyafeai, Z., Alshaibani, M.S., Ahmad, I.: A survey on transfer learning in natural language processing, 1–19 (2020)

22. Hatzakis, I., Wallace, D.: Dynamic multi-objective optimization with evolutionary algorithms: a forward-looking approach. In: Conference on Genetic & Evolutionary Computation. ACM (2006)

23. Von Lücken, C., Barán, B., Brizuela, C.: A survey on multi-objective evolutionary algorithms for many-objective problems. Comput. Optim. Appl. 58, 707–756 (2014)

24. Zhou, A., Jin, Y., Zhang, Q.: A population prediction strategy for evolutionary dynamic multiobjective optimization. IEEE Trans. Cybern. 44(1), 40–53 (2013)

25. Wei, J., & Zhang, M.: Simpex model based evolutionary algorithm for dynamic multi-objective optimization. In: International Conference on Advances in Artificial Intelligence. Springer-Verlag (2011)

26. Liu, M.: Memory enhanced dynamic multi-objective evolutionary algorithm based on decomposition. J. Softw. 24(7), 1571–1588 (2014)

27. Azzouz, R.: A multiple reference point-based evolutionary algorithm for dynamic multi-objective optimization with undetectable changes. In: 2014 IEEE Congress on Evolutionary Computation (CEC). IEEE (2014)

28. Rambabu, R., Vadakkepat, P., Tan, K.C., et al.: A mixture-of-experts prediction framework for evolutionary dynamic multiobjective optimization. IEEE Trans. Cybern. 50(12), 5099–5112 (2019)

29. Jiang, M., Hu, W., Qiu, L., Shi, M., Tan, K.C.: Solving dynamic multi-objective problems via support vector machine. In: 2019 IEEE Congress on Evolutionary Computation (CEC) (2019)

30. Jiang, S., Yang, S.: A steady-state and generational evolutionary algorithm for dynamic multi-objective optimization. IEEE Trans. Evol. Comput. 21(1), 65–82 (2017)

31. Liu, R., Zhang, W., Jiao, L., Liu, F., Ma, J.: A sphere-dominance based preference immune-inspired algorithm for dynamic multiobjective optimization. In: Proc. GECCO (2011)

32. Jiang, S., Yang, S.: Evolutionary dynamic multiobjective optimization: benchmarks and algorithm comparisons. IEEE Trans. Cybern. 47(1), 198–211 (2016)

33. Chun’An, L., Yuping, W.: Multiobjective evolutionary algorithm for dynamic nonlinear constrained optimization problems. Syst. Eng. Electron. 20(1), 204–210 (2009)
34. Jiang, S., Yang, S.: A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization. IEEE Trans. Evol. Comput. 21(1), 65–82 (2017)
35. Chen, J., Lin, Q., Zhen, J.: A hybrid immune multiobjective optimization algorithm. Eur. J. Oper. Res. 204(2), 294–302 (2011)
36. Wei, J., Wang, Y.: Hyper rectangle search based particle swarm algorithm for dynamic constrained multi-objective optimization problems. In: Evolutionary Computation. IEEE (2012)
37. Yang, M., Kang, L., Guan, J.: Multi-algorithm co-evolution strategy for Dynamic Multi-Objective TSP. In: IEEE Congress on Evolutionary Computation. IEEE (2008)
38. Wang, P., Tong, X.: A dimension convergence-based evolutionary algorithm for many-objective optimization problems. IEEE Access 8, 224631–322464 (2020)
39. Azevedo, C.R.B., Araújo. A.F.: Generalized immigration schemes for dynamic evolutionary multiobjective optimization. In: 2011 IEEE Congress of Evolutionary Computation (CEC). IEEE (2011)
40. Wang, Y., Li, B.: Investigation of memory-based multi-objective optimization evolutionary algorithm in dynamic environment. In: 2009 IEEE Congress on Evolutionary Computation, pp. 630–637 (2009)
41. Câmara, M., Ortega, J., de Toro, F.: Approaching dynamic multi-objective optimization problems by using parallel evolutionary algorithms. In: Advances in Multi-Objective Nature Inspired Computing. Springer, Berlin, Heidelberg (2010)
42. Vinek, E., Beran, P.P., Schikuta, E.: A dynamic multi-objective optimization framework for selecting distributed deployments in a heterogeneous environment. Procedia Comput. Sci. 4, 166–175 (2011)
43. Helbig, M., Engelbrecht, A.P.: Analyses of guide update approaches for vector evaluated particle swarm optimisation on dynamic multi-objective optimisation problems. In: 2012 IEEE Congress on Evolutionary Computation. IEEE (2012)
44. Câmara, M., Ortega, J., de Toro, F.: A single front genetic algorithm for parallel multi-objective optimization in dynamic environments. Neurocomputing 72(16–18), 3570–3579 (2009)
45. Li, K., Kwong, S., Deb, K.: A dual-population paradigm for evolutionary multiobjective optimization. Inf. Sci. 309, 50–72 (2015)
46. Peng, Z., Zheng, J., Zou, J.: A population diversity maintaining strategy based on dynamic environment evolutionary model for dynamic multiobjective optimization. In: Evolutionary Computation. IEEE (2014)
47. Gui, L., Lu Q., et al.: Improving transfer learning in cross lingual opinion analysis through negative transfer detection. In: International Conference on Knowledge Science, Engineering and Management, pp. 394–406 (2015)
48. Gretton, A., Borgwardt, K.M., Rasch, M.J., Schlkopf, B., Smola, A.J.: A kernel method for the two-sample-problem. Adv. Neural. Inf. Process. Syst. 1, 513–520 (2007)
49. Jiang, M., Huang, Z., Qiu, L., Huang, W., Yen, G.G.: Transfer learning based dynamic multiobjective optimization algorithms. IEEE Trans. Evol. Comput. 2(4), 501–514 (2018)
50. Smola, A., Gretton, A., Song, L., Schoelkopf, B.: A Hilbert space embedding for distributions. In: International Conference on Discovery Science, pp. 40–41. Springer, Berlin, Heidelberg (2007)
51. Raquel, C., Xin, Y.: Dynamic multi-objective optimization: a survey of the state-of-the-art. In: Evolutionary Computation for Dynamic Optimization Problems. Springer, Berlin, Heidelberg (2013)
52. Goh, C.K., Tan, K.C.: A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. IEEE Press, USA (2009)
53. Helbig, M., Engelbrecht, A.: Benchmark functions for CEC 2015 special session and competition on dynamic multiobjective optimization. Dept. Comput. Sci. Univ. Pretoria Pretoria S. Afr. Rep. (2015)
54. Zheng, J.H., Peng, Z., Zhou, J., Shen, R.M.: A prediction strategy based on guide-individual for dynamic multi-objective optimization. ACTA ELECTON. SIN. 43(9), 1816–1825 (2015)

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