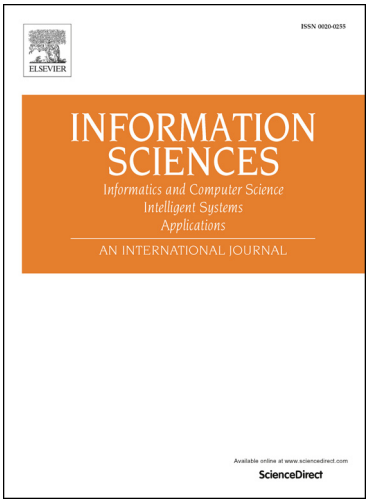
Journal Pre-proof



Dynamic Multi-objective Evolutionary Algorithm Based On Knowledge Transfer

Linjie Wu, Di Wu, Tianhao Zhao, Xingjuan Cai and Liping Xie

PII: S0020-0255(23)00434-6
DOI: https://doi.org/10.1016/j.ins.2023.03.111
Reference: INS 18886

To appear in: *Information Sciences*

Received date: 29 April 2022
Revised date: 16 March 2023
Accepted date: 19 March 2023

Please cite this article as: L. Wu, D. Wu, T. Zhao et al., Dynamic Multi-objective Evolutionary Algorithm Based On Knowledge Transfer, *Information Sciences*, doi: https://doi.org/10.1016/j.ins.2023.03.111.

Dynamic Multi-objective Evolutionary Algorithm Based On Knowledge Transfer

Linjie Wu[a], Di Wu[b], Tianhao Zhao[a], Xingjuan Cai[a,c,*], Liping Xie[a,*]

[a]*School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan, Shanxi, 030024, China*
[b]*Faculty of Information Technology, Beijing University of Technology, Beijing, 100124, China*
[c]*State Key Lab. for Novel Software Technology, Nanjing University, P.R. China*

Abstract

Dynamic multi-objective optimization problems (DMOPs) are mainly reflected in objective changes with changes in the environment. To solve DMOPs, a transfer learning (TL) approach is used, which can continuously adapt to environmental changes and reuse valuable knowledge from the past. However, if all individuals are transferred, they may experience negative transfers. Therefore, this paper proposes a novel knowledge transfer method for the dynamic multi-objective evolutionary algorithm (T-DMOEA) to solve DMOPs, which consists of a multi-time prediction model (MTPM) and a manifold TL algorithm. First, according to the movement trend of historical knee points, the MTPM model uses a weighted method to effectively track knee points after environmental changes. Then, the knowledge of the suboptimal solution is reused in the non-knee point set using the manifold TL technique, which yields more high-quality individuals and speeds up the convergence. In the dynamic evolutionary process, the knee points and high-quality solutions are combined to guide the generation of the initial population in the next environment, ensuring the diversity of the population while reducing the computational cost. The experimental results show that the proposed T-DMOEA algorithm can converge

*Corresponding Author
Email addresses: xingjuancai@163.com (Xingjuan Cai), xieliping@tyust.edu.cn (Liping Xie)

rapidly in solving DMOPs while obtaining better-quality solutions.

Keywords: Knowledge transfer, Predictive model, Dynamic multi-objective optimization, Manifold transfer learning

1. Introduction

Many static engineering optimization problems usually involve multiple requirements that still have conflicts [1, 2, 3]. Therefore, research on multi-objective optimization problems is becoming increasingly urgent [4]. Multi-objective evolutionary algorithm with powerful meta-heuristic search capability are used to successfully solve multi-objective problems. Examples include the elite hybrid metaheuristic optimization algorithm [5], a novel two-stage evolutionary algorithm [6], and a hyperplane-assisted evolutionary algorithm [7]. With the increasing complexity of engineering problems, static optimization is far from satisfying current engineering optimization requirements. Therefore, DMOPs have emerged recently. Dynamic multi-objective optimization problems have mutually constrained objective functions, where the change in time affects the objective function, constraints, and parameters [8]. In recent years, an increasing number of researchers have devoted themselves to research on DMOPs. Dynamic multi-objective optimization has important theoretical research value and broad application prospects in industrial production and real life [9, 10]. Guo et al. [11] established a mathematical model for a complex dynamic vehicle path-planning problem with hard time windows and dynamic users. In addition, there is also the DMOPs problem from in the field of production scheduling: under the ever-changing market demand, how can product companies maximize profits, minimize costs, and minimize environmental pollution in the process of producing products?

There are several ways to solve DMOPs. Lin et al. [12] proposed a reference-point-based strategy to divide the population; this method can enhance the diversity of the population. However, there is still an obstacle in generating multiple populations when the environment changes significantly. Zou

et al. [13] proposed a dynamic optimization method based on center point and knee point prediction. This method only predicts the center and knee points of the population after the environmental changes, rather than the entire population, which speeds up the convergence speed of the population and reduces the computational complexity. Simultaneously, an adaptive strategy was proposed to maintain population diversity. However, its early experience is not sufficient; therefore, there is still much room for improvement in the three-objective optimization problem.

In addition, researchers have combined evolutionary algorithm (EA) solvers with TL to achieve better optimization performance. In 2017, Jiang et al. [14] proposed a transfer-learning-based dynamic multi-objective evolutionary algorithm (TR-DMOEA), which is a framework for solving DMOPs. They proposed the hypothesis that the solution spaces of DMOPs are independent of each other in different environments and have different spatial distributions, but they are related. However, this method wastes considerable time in decomposing the eigenvalues and has the problem of high time complexity. Li et al. [15] proposed a new algorithm based on the TL. The most suitable individuals were selected to compose a new population through TL. Jiang et al. [16] proposed a knee-point-based imbalanced TL for dynamic multi-objective optimization (KT-DMOEA). However, when the knee point is regarded as high quality, the number of individuals is very small. It only transfers the knee point and wastes high-quality individuals that may exist. It may be necessary to generate numerous dominant solutions. Therefore, there is still much room for improvement in the quality of the population.

At present, many scholars have proven that the combination of TL with dynamic multi-objective optimization can effectively solve DMOPs. This is because the state of an environment changes over time. It can make decisions on the state change after prediction, so that the algorithm can make adaptive changes. This not only reduces computational costs, but also improves performance. For the efficiency of transfer, the purpose of this paper is to improve the quality of the solutions and ensure the diversity of the population. We propose

a dynamic multi-objective optimization algorithm based on knowledge transfer. The motivation of this research is that we believe that some high-quality solutions can produce better populations; therefore, we need to select as many high-quality individuals as possible. Moreover, it makes the transfer of learning more comprehensive so that negative transfer can be avoided as much as possible. The primary contributions of this study are as follows.

- A novel dynamic multi-objective evolutionary algorithm is proposed. This algorithm includes a predictive model and manifold transfer technique, which has significant improvements in accelerating convergence and maintaining population diversity.
- An MTPM model is proposed, which can solve the problem of knee point prediction in the new environment. According to the movement trend of the high-quality solution in the previous environment, the method can effectively predict the high-quality solution in the next environment with the dynamic environment change, which can significantly improve the population convergence speed.
- A manifold transfer algorithm is presented to solve the DMOPs. The geodesic flow kernel (GFK) technique is used to obtain high-quality solutions in the target space, which determines the number of subspaces by integrating the kernel in the streamline. This technique not only reduces the occurrence of negative transfer but also prevents bad individuals from consuming a lot of computational resources.

The remainder of this paper is organized as follows. We introduce some fundamental definitions of DMOP and discuss the current work in Section 2. Section 3 proposes a dynamic multi-objective optimization algorithm for knowledge transfer, called T-DMOEA, which can be used to solve complex dynamic multi-objective optimization problems. Section 4 presents the experimental results for different benchmark functions and contrasts them with the latest dynamic multi-objective evolutionary algorithms (DMOEAs). In Section 5, we

summarize the study and outline future work.

2. Background

This section presents the definition of DMOP. We then describe the knowledge of knee points. Finally, the related work on DMOA is discussed, and the design idea of our algorithm is introduced.

2.1. DMOPs

DMOPs are defined as follows:

$$\begin{aligned} &Minimize F\left(x,t\right)=\left\langle f_1\left(x,t\right),f_2\left(x,t\right),\ldots,f_m\left(x,t\right)\right\rangle \\ &s.t\, x\in\Omega_x \end{aligned} \tag{1}$$

where x is the decision vector, and $x \in R^n$ is the decision space. $f_1, f_2, \ldots, f_m$ are objective functions, m is the number of objective functions, t is the environmental variable, and the functions $\mathrm{f}_i\left(x,t\right)$ and $(i=1,2,\ldots,m)$ denote the $i-th$ objective function of at time t. $\Omega=\{x\in R^n\}$ represents the search space of decision variable x. In DMOP, there are dynamic Pareto optimal solutions, and in the solution set, no solution can be dominated by other solutions.

Definition 1 [Dynamic decision vector dominance]: Two candidate solutions x_1 and x_2 (x_1,$x_2 \in \Omega$) are given at time t. Call x_1 dominates x_2, written $x_1 \succ x_2$, when and only when

$$\begin{cases} f_i\left(x_1,t\right)\leq f_i\left(x_2,t\right),\forall i=1,\ldots m \\ f_i\left(x_1,t\right)< f_i\left(x_2,t\right),\exists i=1,\ldots m \end{cases} \tag{2}$$

Definition 2 [Dynamic Pareto-Optimal Set (DPOS)]: All x^* are called dynamic Pareto-optimal solutions. This is denoted $DPOS_t$ at time t, which includes all non-dominated solutions.

$$DPOS_t=\left\{x^*|\nexists x, x\succ_t x^*\right\} \tag{3}$$

Definition 3 [Dynamic Pareto Optimal Frontier (DPOF)]: The mapping of DPOS is called DPOF in the target function space.

$$DPOF_t = \{y^* | y^* = F(x^*, t), \mathrm{x}^* \in DPOS_t\}. \tag{4}$$

It aims to allow the POF to constantly approximate the changing true POF in DMOPs while ensuring the diversity of the solutions.

2.2. Knee Point

In real life, optimization problems have multiple objects that must be considered simultaneously. Because these objects usually conflict with each other, it is necessary to find a set of solutions to balance all objective functions. Instead, there are many Pareto optimal solutions. Their characteristic is that the improvement of any object only needs to sacrifice at least one other object to degrade as a price [17]. In multi-objective optimization, the individuals generated in the later iteration of the algorithm do not dominate each other; therefore, it becomes difficult to select high-quality individuals based on dominance relationships alone. If the user preferences are not clearly given in the algorithm running process, choosing the knee point is the most popular. The most concave point on the Pareto front is called the knee-point. The knee-point is a subset of Pareto solutions; its characteristic is a slight improvement in one goal and a significant decrease in another goal [18]. The hypervolume (HV) indicator tests the comprehensive performance of the algorithm by calculating the volume of the region formed by the solution and the reference point. When dealing with concave optimization problems, the location of the knee point contributes the most to the Pareto solution set. Therefore, the convergence performance of the knee point as a choice option is much greater than that of other neighboring individuals.

Many scholars have proposed different methods of determining the knee point based on its characteristics to increase the selection pressure in the search process. Zou et al. [**?**] used the ideal point and extreme point to form a hyperplane, calculated the vertical distance from all non-dominated solutions

to this hyperplane, and chose the solution with the shortest vertical distance as the knee point. Branke et al. [17] presented an approach based on expected marginal utility (EMU) to determine the role of every weight vector on the solution through EMU. In addition, it is standard to use the angle to identify knee points, and the two neighborhoods of the solution and the solution are formed into two straight lines. The solution with the largest angle among all the solutions is called the knee point. Das et al. [19] proposed a simple method for determining the knee point, which is determined by calculating the distance between the hyperplane and extreme line. The parameters a, b, and c of the extremum line are obtained from the two solutions in the Pareto solution set, and the extremum line L for the bi-objective problem is defined as

$$L : a \cdot f_1 + b \cdot f_2 + c = 0 \tag{5}$$

where a, b, and c are obtained from two extremum points. The distance from solution $S(x_s, y_s)$ to extremum L was calculated as follows:

$$d(S, L) = \begin{cases} \frac{|ax_s + by_s + c|}{\sqrt{a^2+b^2}}, ax_s + by_s + c < 0 \\ -\frac{|ax_s + by_s + c|}{\sqrt{a^2+b^2}}, otherwise \end{cases} \tag{6}$$

Among the non-dominated solutions, the solution with the farthest distance from the extreme line was the knee point. In the two-dimensional space shown in Fig. 1, the target space is divided into three subspaces, where B, D, and G are the regional knee points in each subspace.

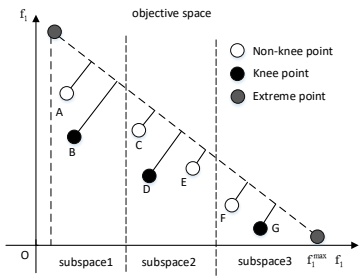


Figure 1: Regional knee points are B, D, and G in the three subspaces.

2.3. Related work

Comprehensive research has been conducted in the literature related to DMOEAs. In recent years, significant progress has been made in solving DMOPs.

The diversity-based mechanism is a strategy that can be used to maintain population diversity in a changing environment, and is also an effective way to help the algorithm jump out of the local optimum [20]. To promote research on dynamic constrained multi-objective optimization, Chen et al. [21] presented a new dynamic solution method in 2020. The designed selection operator can obtain diverse nondominated solutions when environmental changes are detected. Deb et al. [22] proposed a dynamic NSGA-II, which contains two versions. It uses random or mutated individuals to enhance diversity. However, the Pareto-optimal frontier (POF) obtained is unrestricted by this method, which may reduce the robustness of the algorithm. Camara et al. [23] proposed parallel evolution to solve the DMOPs. It uses Pareto solutions as selection and mutation operators and applies the crowding mechanism to the Pareto frontier. This method solves the divergence problem of convergent populations. Cao et al. [24] proposed a first-order difference model based on DMOEA (FD-DMOEA), which can track the changing Pareto-optimal set (POS) or POF faster. At present, most methods still have room for improvement in terms of population diversity.

Existing studies have shown that the memory-based approach can use external storage to explicitly or implicitly store information from the previous generation, which can better guide future searches. It is also more effective in DMOP that change with environmental cycles. For example, Jiang et al. [26] proposed a memory-driven manifold TL-based dynamic multi-objective optimization algorithm (MMTL-DMOEA) that combines a manifold TL and a memory mechanism to obtain Pareto solutions. Peng et al. [27] used a dynamic evolution model that used different types of knowledge to continuously guide the search in a dynamic environment. When the environment changes, it is difficult to quickly and accurately search for a high-quality and uniformly distributed POS using the above method. Guo et al. [28] introduced a variable number of objectives that implemented a dynamic dual-file evolution algorithm that

enabled the co-evolution of two populations simultaneously. These two populations focus on diversity and convergence. When the environment changes, these two populations are rebuilt adaptively. This method avoids the shortcomings of previous methods.

The study of dynamic problems has become a trend, and methods based on predictions have attracted increasing attention from scholars [29, 30] who used their past experience to forecast the best solution for the next environment. Jiang et al. [31] proposed a dynamic multi-objective optimization algorithm based on the transfer of individuals (IT-DMOEA), which requires identifying a set of guidance population groups and then using transfer Adaboost to generate predicted individuals in a new environment. [32] proposed a dynamic predicting strategy based on the analysis of decision variables. It constructs a comprehensive prediction method to estimate the fitness value in the next environment. The disadvantage of this method is that the estimation of historical information is inaccurate, which may lead to poor convergence. Zou et al. [33] proposed a knee-guided prediction method that reduced the burden on the evolution process by finding solutions to knee points and the area near the knee point. This method enhances the accuracy of the predictive model in the information search process. Liang et al. [34] proposed a predictive model based on the feedback mechanism. Correction feedback is used to revise the initial predictive model. Effectiveness feedback was used to enhance the effectiveness of the re-initialization. The applicability of these mechanisms to environmental changes is not considered in most existing prediction-based methods; therefore, valuable environmental information is sometimes wasted.

The method based on the TL is a branch of prediction. It is important that the distribution of solutions vary with the environment in DMOPs; that is, solutions are non-independent and identically distributed at different times. Jiang et al. [35] used a filtering strategy to deal with populations, and used Adaboost to build a predictive model. Although this method avoids the aggregation of individuals, the waste of a large number of high-quality solutions would slow convergence. Feng et al. [36] proposed a method for solving DMOPs using

automatic coding evolution. The non-dominant solution in the auto-encoder can be used to predict the movement of the POS. Jiang et al. [26] combined a memory mechanism that preserves elite individuals in past environments with manifold characteristics, and it predicts the best individuals for the next environment in the evolutionary process. However, the number of subspaces is uncertain and can affect the transfer efficiency. In [37], a distribution estimation algorithm based on domain adaptation and non-parametric estimation was proposed. However, this method does not consider the distribution of solutions in the previous and subsequent environments, and can easily lead to local optimization and negative transfer.

Therefore, this method is briefly summarized as follows: After obtaining the non-dominated solution of the initial population, we divided the target space into multiple subspaces according to the method of finding the knee point. Then, it is necessary to select the regional knee point and the remaining non-dominated solutions on each subspace and use different transfer methods to reuse the knowledge of these two sub-populations, which can predict the sub-population at the next moment. First, it was necessary to store the knee points of the area in the external archive set. Once a change in the environment is detected, these regional knee points are input into MTPM to obtain special individuals at the new moment, which are called the prediction knee points. In the remaining non-dominated solutions, we can screen out relatively poor individuals and retain the solutions second only to the knee point, which we call suboptimal solutions. They can reduce the probability of negative transfers and ensure the quality of understanding. We can then predict the solution at the next moment by constructing the geodesic stream core. Finally, the predicted individuals of the two parts are combined to increase the diversity of the population and provide a better initial population for the next environment, which keeps the population moving in the right direction.

3. Proposed method

In this section, detailed information about the overall algorithm is described, which includes the recognition of regional knee points, MTPM predictive model, and non-knee point TL algorithm.

3.1. Overall algorithm

This paper presents a dynamic multi-objective optimization algorithm based on knowledge transfer, which combines the kernel-based manifold TL method with a predictive model to generate more high-quality solutions to guide the population and make the algorithm adapt to environmental changes as quickly as possible. The prediction model can effectively track the Pareto solution after changing the environment, ensuring that the diversity of the population avoids the local optimization of the algorithm. In a kernel-based manifold TL approach to obtain evolutionary processes with high-quality solutions in a new environment, the reuse of historical information improves the searching ability of the algorithm, avoids the occurrence of negative transfer, and speeds up the convergence rate of the population. We used two archive sets in a nondominant solution. These are the knee point archive set P_1 and the non-knee point archive set P_2 respectively. The two archive sets use different methods to generate the knee-point set and the high-quality solution set for the next moment. In this study, regional knee points were determined by dividing the subspace and determining the distance from the extreme line or hyperplane. Therefore, it is necessary to use the proposed predictive model MTPM in the knee-point archive. For non-knee-point archives, we used TL to obtain new environments of high-quality solutions in the evolutionary process. This study uses different methods to find two elite solutions to constitute the initial population, which leads the population to evolve in a better direction and accelerates the convergence speed.

Algorithm 1 describes the overall flow of this algorithm. In the first three environments, the initial population was randomly generated, and this population-based static multi-objective optimization algorithm (SMOA) is used to optimize

the initial population to obtain their respective POS. Then, the knee point set and non-knee point set in each environment are stored in two external archive sets. The key to this study is to quickly optimize DMOP and enhance the quality of the solution using the proposed prediction approach. When detecting an environment greater than three in dynamic change, individuals in the two external archive sets use different strategies to make predictions. In the knee-point archive set P_1, it is assumed that the length of the vector of the knee point is always the same. The positions of the knee points in the next environment were obtained by weighting the trend of the historical inflection movement. The archive set P_2 is the set of historical environment nonknee points, and there are some bad solutions. Therefore, the support vector regression (SVR) technique is used to obtain the set of solutions with estimator E, which are defined as suboptimal solutions. A kernel-based manifold transfer technique is used to map the solution sets into subspaces in the geodesic flow and obtain high-quality solutions by reusing the knowledge of suboptimal solutions. Any population-based SMOA can be used to optimize the merged population, which guides the population search for a true POF.

Algorithm 1 T-DMOEA

Input: DMOP objective function is $F(x,t)$, number of clusters L, size of individuals N, number of subspaces p, external memory P_1, P_2

Output: the POS of $F(x,t)$ in the current environment and the generation of initial populations.

1: **while** the environment has changed **do**
2: $t = t + 1$;
3: **if** $t == 1 | t == 2 | t == 3$ **then**
4: $POS_{\mathrm{t}} = SMOA\left(F(x,t)\, initPop\right)$;
5: $P_1(t) = knee_t$;
6: $P_2 = n - knee_t$;
7: **else**;
8: $Knee_{predict} = MTPM\left(P_1, F(x,t), P\right)$;
9: Use SVR to structure the estimator E with $\{P_2, F(x,t)\}$;

Estimate values of P2: $X = E(P_2)$;
Find sub-optimal solutions in X;
$Trans - nknee = Transfer(Best - nknee, L, F(x,t), N)\sqrt{b^2 - 4ac}$;
$POS_{estimated} = [Transf - nknee \cup Knee_{predict}]$;
$POS_t = SMOA(F(x,t), POS_{estimated})$;
Return POS_t;
end if
end while

3.2. Regional Knee Points

We divided the decision space into a few subspaces, and it was necessary to calculate the corresponding knee point in each subspace. This is also known as a regional knee point. There was only one regional knee point in each subspace. When time t is greater than three, we input the calculated regional knee point into the designed MTPM. It can be used to forecast knee-point sets in the next environment. To calculate the regional knee points, the target space was evenly divided into p subspaces, and the $m-th$ objective function was randomly selected. When the time is t, $f_{m,t}^{\max}$ and $f_{m,t}^{\min}$ are the maximum and minimum values of the $m-th$ objective function, respectively. The subspace sizes were as follows:

$$size_{m,t,i} = \frac{f_{m,t}^{\max} - f_{m,t}^{\min}}{p}. \tag{7}$$

The calculation method for the lower limit of the subspace is as follows.

$$LB_{m,t,i} = f_{m,t}^{\max} + (i-1) \times size_{m,t} \tag{8}$$

The calculation method for the upper limit of the subspace is as follows.

$$UB_{m,t,i} = f_{m,t}^{\min} + i \times size_{m,t} \tag{9}$$

3.3. Predictive model

In this study, we design an MTPM predictive model in which the movement of the optimal solution is tracked by the non-dominant solution at the first three

consecutive moments of the current moment. To better maintain the characteristics of the historical solution, the model uses the motion trend of the two historical environments to predict the motion trend of the knee point through weighting in the new environment. However, these two movement trends require the identification of three environmental knee points. Algorithm 2 presents the detailed steps of the MTPM. This prediction model has 4 steps in total. First, a vector can be obtained in the subspace i, where the vector represents the movement trend of the regional knee point from the previous environment to the current environment.

$$\vec{V}_i^{t-2} = k_i^{t-2} - k_i^{t-1} \, (1 \leq t \leq 4) \tag{10}$$

$$\vec{V}_i^{t-3} = k_i^{t-3} - k_i^{t-2} \, (1 \leq t \leq 4) \tag{11}$$

k_i^{t-3}, k_i^{t-2} and k_i^{t-1} represent regional knee points at $t-3$, $t-2$ and $t-1$. Second, we must use the knowledge of three adjacent moments to predict the movement trend of the new moment. In theory, we believe that the use of multi-time knowledge learning can better maintain the characteristics of the previous solution, and the motion trend at $t-2$ has a greater impact than the motion trend at $t-3$ at the new time; therefore, we set the weights ω_1 and ω_2 on the motion vector. The latter value is larger than that of the former. The following equation was used:

$$\vec{V}_i^{t-1} = \omega_1 \cdot \vec{V}_i^{t-3} + \omega_2 \cdot \vec{V}_i^{t-2} \, (\omega_1 = 0.3, \omega_2 = 0.7, t > 3) \tag{12}$$

In the third step, we can use this trend to obtain the movement trend of the predicted knee point in the $i-th$ subspace in the new environment. The prediction of the knee point can be calculated using (13).

$$k_i^t = k_i^{t-1} + \vec{V}_i^t \tag{13}$$

Algorithm 2 Multi-time Predictive Model

Input: The objective function of the DMOP is: $F(x,t)$, three regional knee point sets k_i^{t-3}, k_i^{t-2} and k_i^{t-1}, and the size of subspaces p

Output: obtained Knee Point $k_{predict}$.

1: **for** $i = 1$ to p **do**
2: Calculate the trend of motion of $\vec{V}_i^{t-2}$ and $\vec{V}_i^{t-3}$
3: **end for**
4: Calculate $\vec{V}_i^{t-1}$ by equation (12);
5: Compute $k_{predict}$ according to equation (13);
6: Return $k_{predict}$

3.4. Geodesic Flow Kernel

This study uses the manifold TL technique, which maps information in the source and target domains into a manifold geometric space composed of multidimensional vector spaces and reuses valuable knowledge in the source domain to obtain the required information in the target domain [38]. As shown in Figure 2, we can map the source and target datasets to the Grassmann manifold, where the red dot indicates the source domain, which means the already trained dataset, denoted by $\phi(0)$. The yellow dot denotes the target domain, which indicates the dataset waiting for training and is denoted by $\phi(1)$. The orange line represents the geodesic flow constructed between two points $\phi(0) - \phi(1)$, and $\phi(k)$ represents the function of the geodesic flow, which contains several subspaces. The right side of the figure shows that the original data are mapped to the subspace $\phi(k)$, which forms feature vectors $z^{\infty} \in H^{\infty}$, and the inner product of these feature vectors can define the kernel function G. H^{∞} denotes infinite-dimensional feature space. By mapping the data of the source and target domains to the geodesic flow, the target domain mapping data on the stream shape are made infinitely close to the mapping data in the source domain. This method parameterizes the smooth change from the source domain to the target domain.

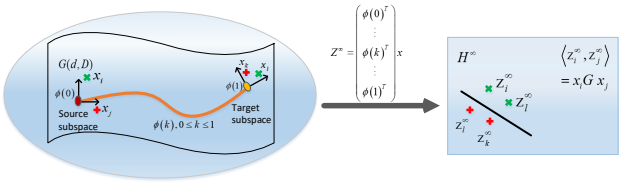


Figure 2: The approach of the domain adaptive method based on the geodesic flow kernel.

The detailed steps of the TL technique based on the geodesic flow core are as follows. In the proposed method, the principal component analysis (PCA) technique is used to process the data, which can obtain the subspace basis sets of the source domain P_S and the target domain P_T about the d-dimensional eigenvalues, where $P_S, P_T \in R^{n\times d}$, let $R_S \in R^{n\times(nd)}$ be the orthogonal complement of P_S and let $\phi(k)$ be the constructed geodesic flow, $\phi(0) = P_S$ and $\phi(1) = P_T$ when $0 < k < 1$, $\Phi(k) = P_S U_1 \Gamma(k) - R_S U_2 \Sigma(k)$, where U_1 and U_2 are a set of standard orthogonal matrices, they are given by a set of singular value decomposition;

$$P_S^T P_T = U_1 \Gamma V^T, R_S^T P_T = -U_1 \Sigma V^T \tag{14}$$

Where $\Gamma(k)$ and $\Sigma(k)$ are $d \times d$ dimensional diagonal matrices composed of $\cos(k\theta_i)$ and $\sin(k\theta_i)$, respectively, $i = 1, 2, ..., d$, and θ_i is the protagonist between the source domain and the target domain, $\theta_i \in [0, \pi/2]$.

The geodesic flow connects the source and target domains, $\phi(\cdot)$ represents a subspace on this geodesic flow, $t \in (0, 1)$, and the projection of the sample x in this subspace is $\phi(t)^T x$. Projecting the data in the two domains continuously onto the subspace on the flow is described as the process of change from the source domain transfer to the target domain. $\phi(1)$ corresponds to the projection of the target domain and $\phi(0)$ corresponds to the projection of the source domain.

The innovation of GFK projects the two original data x_i and x_j in an infinite number of subspaces from 0 to 1 on the flow. Then, the projections of these infinite subspaces are stitched together into an infinite-dimensional vector

Z^{∞}, where the inner product between Z_i and Z_j is defined as

$$\left\langle z_i^{\infty}, z_j^{\infty} \right\rangle = \int_0^1 \left(\phi(t)^T x_i\right)^T \left(\phi(t)^T x_j\right) dt = x_i^T G x_j \tag{15}$$

Where $G \in \mathbb{R}^{D \times D}$ denotes a positive semi-definite matrix. The so-called "kernel technique" is that the kernel function generates an inner product between features of infinite dimensions.

Matrix G is calculated as follows:

$$G = [P_S U_1 R_S U_2] \begin{bmatrix} \Lambda_1 \Lambda_2 \\ \Lambda_2 \Lambda_3 \end{bmatrix} \begin{bmatrix} U_1^T P_S^T U_2^T P_S^T \end{bmatrix} \tag{16}$$

where Λ_1, Λ_2 and Λ_3 are diagonal matrices and the diagonal elements are as follows:

$$\lambda_{1i} = 1 + \frac{\sin(2\theta_i)}{2\theta_i}, \lambda_{2i} = \frac{\cos(2\theta_i) - 1}{2\theta_i}, \lambda_{3i} = 1 - \frac{\sin(2\theta_i)}{2\theta_i}. \tag{17}$$

After obtaining the geodesic core, we calculated the point mapped from the original data to the geodesic.

$$\bar{x} = x^T \sqrt{G} \tag{18}$$

where $x \in R^{\mathrm{n}}$ is the initial data from the source or target domain and $\bar{x}$ is the mapping data of the initial data projected onto the manifold.

Algorithm 3 Transfer

Input: Sub-optimal solutions: DMOP's objective function $F(x,t)$, population size N, number of clusters L

Output: Transfer of individuals: Trans-nknee.

Initialization;
Clustering sub-optimal solutions: $Nknee^1, ..., Nknee^L$ by LPCA;
for $j = 1$ to L **do**
    $P_S = PCA(Nknee^j)$;
    Generate set T containing N individuals of $F(x,t)$;
    $P_T = PCA(T)$;
    Construct the geodesic flow kernel G through equation (16);
    for $x \in Nknee^j$ **do**
        The mapped data $\bar{x}$ is obtained by combining the original data x with kernel G;
        $x^{'} = argmin_{x^{'}} \left\| x^{'T}\sqrt{G} - \bar{x} \right\|$;
        Trans−nknee $= x^{'} \cup$Trans−nknee;
    end for
    Return Trans−nknee;
end for

The specific transfer steps are presented in Algorithm 3. First, the sub-optimal solutions are clustered into m-one-dimensional L-segmented manifolds through Local PCA(LPCA) [39]: $Nknee^1, ..., Nknee^L$. Then the geodesic flow core G can be constructed in each cluster. The source P_S and target domains P_T in the manifold transfer are $Nknee^j$ and a randomly generated new population after processing using PCA. Then, we map the data x in $Nknee^j$ to the geodesic. The mapping data can be obtained according to Equation 18. In a dynamic environment, the source and target domains may contain similar mapping data on the manifold. Therefore, we can find that the target domain data $x^{'}$ and its mapping data $x^{'T}\phi(\cdot)$ on the manifold are infinitely close to the mapping data $\bar{x}$ of the source domain. This study used the interior point

method to solve the single-objective optimization problem.

4. Experiments

This chapter uses commonly used dynamic benchmark sets to verify the effectiveness of the proposed T-DMOEA in solving DMOPs and makes a comprehensive comparison with several recent DMOEAs. The comparative algorithms, parameter settings, evaluation metrics, and benchmarking suite of issues involved in this experiment are described in section 4.1. Different evaluation metrics were used to measure the effectiveness of the algorithm and the impact of the parameter settings in the predictive model on the performance of the algorithm, as described in Section 4.2. The running times of different algorithms are compared in Section 4.3. A group of ablation experiments was performed to demonstrate the superiority of the proposed method, as described in Section 4.4.

4.1. Performance Indicators and Test Problems

In this section, we compare the proposed T-DMOEA with some relatively novel dynamic algorithms developed in recent years, which can verify the efficiency of the algorithm design. The test function used in the comparison experiment was the DF series [40], and the DF benchmark test suite contained 14 questions (DF1-DF14). The dynamics of the DF suite are reflected in the possible existence of a dynamic PF or PS in different environments, and the 14 different benchmark functions can better represent different problems in real scenarios, which can effectively compare the performance of dynamic multi-objective optimization algorithms. Current DMOPs are divided into three categories: type I is POS/POF both dynamic, type II is POF dynamic, POS is static, type III is POS dynamic, and POF does not change.

To enhance the persuasiveness of the algorithm and guarantee fairness of the experiment, we used MOEA/D as the benchmark optimization algorithm. This has been extensively demonstrated experimentally [14, 16]. When solving this

problem, the algorithm MOEA/D has a high adaptability in dynamic environments. These algorithms include Tr-DMOEA, KT-DMOEA, FD-DMOEA [24], MMTL-DMOEA [26], IT-DMOEA [31], and static MOEA/D [41] were revised into the dynamic optimization algorithm RI-MOEA/D. Similarly, ten percent of the population must be randomly reinitialized in the face of environmental changes.

The t parameter in the DMOP's objective function $F(x,t)$ denotes the environment instant, where $t = (1/n_t)\lfloor \tau/\tau_t \rfloor$, among them τ, τ_t and n_t are the generation counter, the frequency of environmental changes, and the severity of environmental change. The parameters of the comparison algorithm are set according to the original literature, which comes from the CEC 2018 DMO DF benchmark suite. In this study, three evaluation indicators were used to measure the performance of the algorithm. The following is a description of these three indicators.

- 1) Inverted generational distance (IGD) is an efficient method to evaluate the convergence and diversity of algorithms. The smaller IGD value means more diversity and better convergence of the algorithm. The equation for this evaluation index is:

$$IGD(POF^*, POF) = \frac{1}{n} \sum_{p^* \in POF^*} \min_{p \in POF} \|p^* - p\|^2 \tag{19}$$

  Where POF^* represents the true POF in DMOP; POF is the front formed by the POS obtained from the algorithm, and n is the population size. The MIGD is adapted from IGD, and it is defined as the average of the IGD values running at time T.

$$MIGD(POF^*, POF) = \frac{1}{T} \sum_{i=1}^{T} IGD({POF_t}^*, POF_t) \tag{20}$$

  Where T is the total running time.

- 2) Maximum Spread (MS): It is used to measure the diversity of the solutions acquired by the algorithm. A larger MS value means a higher degree

of coverage. The metric equation of MS is defined as:

$$MS=\sqrt{\frac{1}{M}\sum_{\mathrm{k}=1}^{M}\left[\frac{\min\left[\overline{P_k^t},\overline{P_k^*}\right]-\max\left[\underline{P_k^t},\underline{P_k^*}\right]}{\overline{P_k^t}-\underline{P_k^t}}\right]^2} \tag{21}$$

$\overline{P_k^t}$ and $\overline{P_k^*}$ represent the maximum value of the obtained POF and the true POF, respectively,

- 3) HV: The convergence and distribution of the solution are measured by the volume enclosed by the solution set and reference points. The equation of HV is defined as follows:

$$HV\left(POF,ref\right)=\Lambda\left(\bigcup_{p\in POF}\left\{p^{'}|ref\succ p^{'}\succ p\right\}\right) \tag{22}$$

Where $\wedge$ is the Lebesgue measure and $ref\in R^M$ is the reference point for the computation of HV.

The MHV metric is a variant of HV that represents the average of HV values over certain time steps in a given run.

$$MHV=\left(POF,ref\right)=\frac{1}{|T|}\sum_{t\in T}HV\left(POF_t,ref\right) \tag{23}$$

In the experiment, we set the dimensions of the decision variable to 10, the population size of the bi-objective optimization problem to 100, and the population size of the tri-objective optimization problem to 150. In the experiments, the three dynamic sets of settings most often used by the DF benchmark function were chosen. $(n_t=10,\tau_t=10),(n_t=10,\tau_t=5),(n_t=5,\tau_t=10)$. The generation counter τ was set to $50*\tau_{\mathrm{t}}$.

The parameter setting of the T-DMOEA algorithm was based on the literature [22]. The experiments show that the difficulty of predicting the inflection point differs for different test sets. The method performs well on most test functions; however, when faced with a complex dynamic POF, the acquisition of knee points becomes difficult. Setting more than ten knee points does not significantly improve the convergence performance. Therefore, it is reasonable to set P to 10.

4.2. Performance On Benchmark Problems

The statistical results for MIGD, MHV, and MS are shown in Tables 1, 2, and 3, respectively. Table 4 lists the significant differences between the compared algorithms, and Figure 4 provides a detailed analysis of the impact of the parameters on the performance of the algorithms.

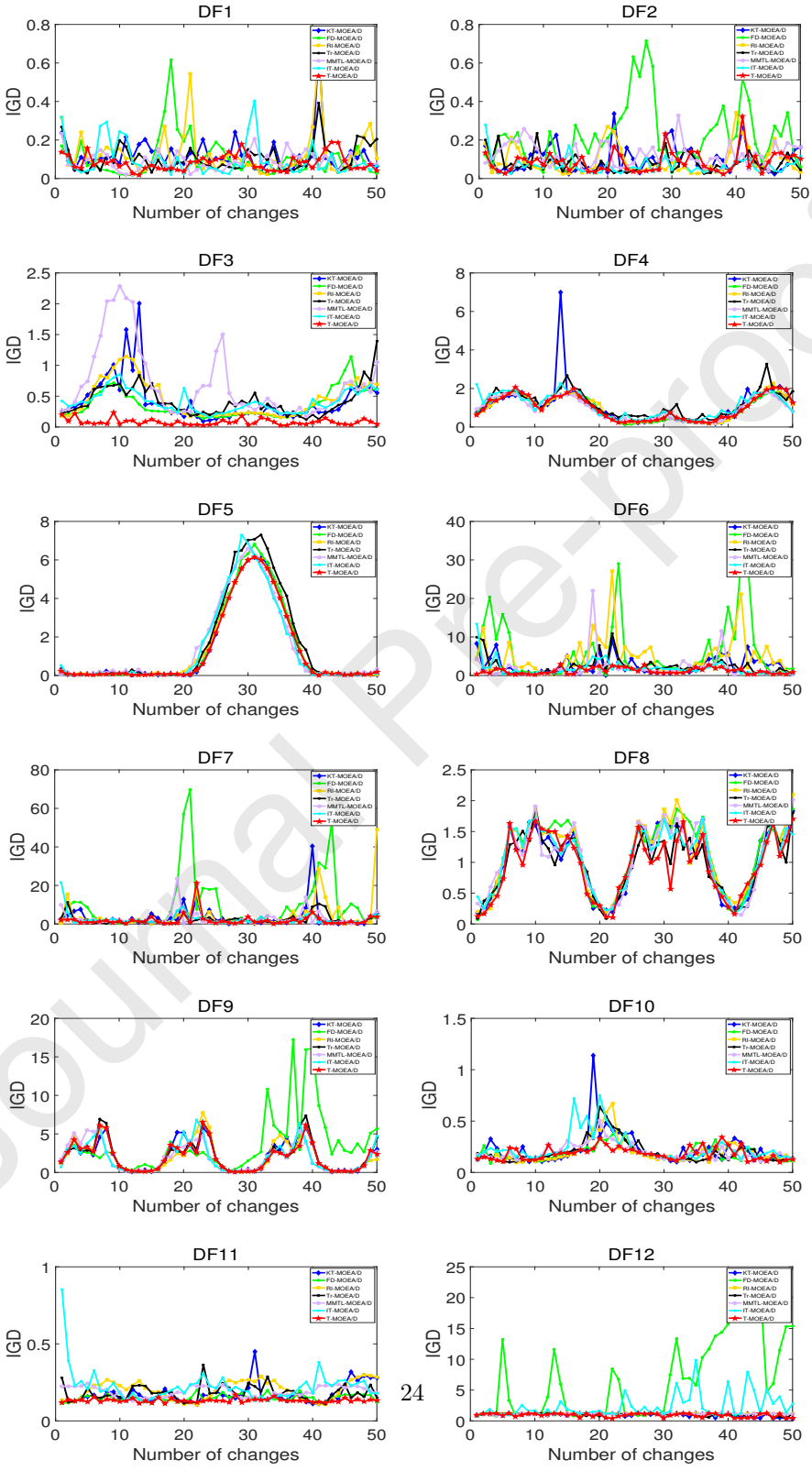
Table 1: Experimental results of MIGD obtained by different dynamic multi-objective optimization algorithms with different dynamic settings benchmark

Problems	Metrics	T-MOEA/D	KT-MOEA/D	Tr-MOEA/D	RI-MOEA/D	FD-MOEA/D	MMTL-MOEA/D	IT-MOEA/D
DF1	(10,10)	**0.08188±6.961e-02**	0.0855±6.29e-02	0.1152±1.09e-01	0.1222±1.70-01	0.09860±5.773e-02	0.08689±5.027e-02	0.097±1.191e-01
	(10,5)	0.1745±6.372e-02	0.1093±8.37e-02	**0.1065±4.70e-02**	0.1217±7.73e-02	0.7104±1.314e-01	0.1327±6.940e-02	0.132±8.661e-02
	(5,10)	**0.09490±4.048e-02**	0.1323±8.96e-02	0.1778±9.09e-02	0.1906±1.06e-01	0.1165±7.633e-02	0.5470±1.249e-01	0.0984±1.038e-01
DF2	(10,10)	**0.07312±4.464e-02**	0.09823±7.086e-02	0.0796±4.25e-02	0.0898±5.03e-02	0.2218±1.574e-01	0.09854±5.532e-02	0.104±2.261e-01
	(10,5)	0.161±3.602e-02	**0.0698±4.87e-12**	0.0851±4.13e-02	0.0851±4.88e-02	0.8288±1.253e-01	0.1654±9.366e-02	0.1020±5.756e-02
	(5,10)	**0.09104±4.714e-02**	0.1240±5.82e-02	0.1517±7.88e-02	0.1446±7.92e-02	0.1054±1.467e-01	0.09653±6.337e-02	0.0980±1.253e-01
DF3	(10,10)	0.3449±2.053e-01	0.4194±3.576e-01	**0.3239±1.43e-01**	0.4207±2.15e-01	0.5107±1.733e-01	0.5719±4.540e-01	0.3622±1.610e-01
	(10,5)	0.5067±2.227e-01	0.3946±2.41e-01	0.3340±1.43e-01	0.4065±1.85e-01	**0.2603±1.998e-01**	0.6102±4.242e-01	0.4173±1.437e-01
	(5,10)	0.3977±1.710e-01	0.4053±2.04e-01	0.4008±1.32e-01	0.4666±2.08e-01	0.5329±2.826e-01	0.5795±5.720e-01	**0.3866±1.670e-01**
DF4	(10,10)	1.007±6.301e-01	1.116±1.041e+00	1.1678±6.00e-01	1.1112±5.86e-014	0.9497±6.279e-01	**0.942±5.48E-01**	1.032±5.962e-01
	(10,5)	1.474±2.967e-01	0.9741±5.90e-01	1.1431±6.50e-01	1.1639±5.85e-01	**0.3032±1.160e-01**	0.983±5.32E-01	1.095±5.477e-01
	(5,10)	1.388±4.304e-01	1.0266±6.49e-01	1.3115±6.34e-01	1.2736±5.20e-01	1.414±4.312e-01	**0.978±5.26E-01**	1.097±6.065e-01
DF5	(10,10)	**1.281±2.028e+00**	1.306±2.048e+00	1.5886±2.49e+00	1.5440±2.38e+00	1.368±2.171e+00	1.460±2.292e+00	1.421±2.321e+00
	(10,5)	**0.9322±7.118e-01**	1.3107±2,15e+00	1.6223±2.46e+00	1.5720±246e+00	0.9797±3.451e-02	1.700±2.565e+00	1.561±2.394e+00
	(5,10)	**0.96889±3.287e-01**	1.2956±2.03+00	1.8847±2.75e+00	1.8507±2.70e+00	1.435±3.410e-00	1.490±2.310e+00	1.485±2.301e+00
DF6	(10,10)	**1.655±3.557e+00**	2.3480±2.87e+00	1.8157±2.66e+00	2.7378±3.09e+00	6.055±7.707e+00	2.279±4.392e+00	2.526±4.057e+00
	(10,5)	3.326±1.597e+00	1.9362±3.61e+00	3.1819±5.92e+00	**1.5022±9.67e-01**	1.999±2.236e+00	3.812±5.769e+00	4.198±6.069e+00
	(5,10)	**1.499±1.231e+00**	3.0427±3.25e+00	3.8332±4.75e+00	4.8446±5,15e+0	3.356±6.024e+00	2.312±5.007e+00	2.507±4.394e+00
DF7	(10,10)	**1.219±4.178e-01**	2.2345±2.32e+00	2.0291±2.42e+00	2.9849±3.85e+00	9.836±1.481e+01	2.275±4.072e+00	2.469±4.409e+00
	(10,5)	2.192±1.598e+00	**2.0195±1.48e+00**	2.1043±2.12e+00	2.3055±2.34e+00	5.933±1.551e+01	3.743±5.533e+00	3.915±5.939e+00
	(5,10)	**0.7533±3.222e-01**	3.1397±4.24e+00	3.1823±4.22e+00	3.3120±3.08e+00	1.717±2.373e+00	1.963±2.013e+00	3.818±8.995e+00
DF8	(10,10)	**0.9499±4.966e-01**	1.031±5.379e-01	0.9792±5.19e-01	0.9517±5.08e-01	1.123±5.735e-01	1.024±5.255e-01	1.002±4.920e-01
	(10,5)	0.9810±4.992e-01	0.8749±4.29e-01	0.9949±4.79e-01	0.9940±4.59e-01	**0.7699±3.205e-01**	0.9160±4.914e-01	0.9104±4.606e-01
	(5,10)	0.9517±5.282e-01	**0.8212±4.69e-01**	0.9122±4.65e-01	0.8663±4.45e-01	0.9904±4.936e-01	1.026±5.169e-01	0.9749±4.530e-01
DF9	(10,10)	3.040±1.476e+00	2.133±1.874e+00	**1.6306±1.60e+00**	1.6587±1.62e+00	3.599±3.879e+00	1.753±1.640e+00	1.780±1.736e+00
	(10,5)	2.917±1.491e+00	1.7776±1.77e+00	1.4271±1.48e+000	1.4032±1.13e+00	**0.8455±2.593e-01**	1.723±1.689e+00	1.738±1.793e+00
	(5,10)	1.7813±1.352e+00	1.8248±1.83e+0	1.8170±2.06e+00	1.7829+1.99e+00	2.388±1.954e+00	**0.1034±1.676e+01**	1.666±1.818e+00
DF10	(10,10)	**0.1310±2.424e-02**	0.1880±1.34e-01	0.1883±7.86e-02	0.2036±8.49e-02	0.1845±7.901e-02	0.176±6.358e-02	0.2059±1.331e-01
	(10,5)	**0.1470±3.030e-02**	0.1950±1.39e-01	0.2167+734e-02	0.2108±8.63e-02	0.8789±6.808e-02	0.1895±9.393e-02	0.2211±1.087e-01
	(5,10)	**0.1810±8.291e-02**	0.2000±1.12e-01	0.2196±1.02e-01	0.2210±1.28e-01	0.2022±1.211e-01	0.2077±9.194e-02	0.2665±2.082e-01
DF11	(10,10)	**0.1311±1.345e-02**	0.1399±1.756e-02	0.1953±4.96e-02	0.1916±4.77e-02	0.1496±3.728e-02	0.1835±3.918e-02	0.2030±8.966e-02
	(10,5)	**0.1686±2.161e-02**	0.1820±2.721e-02	0.2622±8.64e-02	0.2560+9.15e-02	0.9904±1.508e-02	0.1981±5.005e-02	0.2436±1.245e-01
	(5,10)	0.1520±2.293e-02	0.1627±5.807e-02	0.2373±8.16e-02	0.2289±6.99e-02	**0.1476±1.732e-02**	0.264±8.353e-02	0.3030±1.745e-01
DF12	(10,10)	**0.9197±2.873e-01**	1.0082±2.45e-01	1.0084±2.53e-01	0.9390±2.83e-01	6.818±3.198e+00	1.130±1.008e-01	1.692±1.586e+00
	(10,5)	**0.8981±2.873e-01**	1.0520±2.14e-01	1.0046±247e-01	0.9286±2.84e-01	1.075±1.131e+00	1.084±1.287e-01	1.983±1.830e+00
	(5,10)	**0.8994±2.612e-01**	0.9962±2.68e-01	0.9777+2.76e-01	0.9198±2.97e-01	4.575±3.353e+00	1.138±9.321e-02	1.426±1.313e+00
DF13	(10,10)	1.371±1.806e+00	**1.3474±1.87e+00**	1.5068±192e+00	1.4751±194e+00	1.378±1.798e+00	1.410±1.828e+00	1.433±1.824e+00
	(10,5)	**0.9457±4.515e-02**	1.3327±1.81e+00	1.5078±1,99+00	1.5023±1.98e+00	0.9996±3.639e-01	1.490±1.945e+00	1.516±1.928e+00
	(5,10)	**1.3150±1.043e+00**	1.4130±1.93e+00	1.7077±2.21+00	1.7365±2.38e+00	1.3706±3.643e+00	1.402±1.817e+00	1.415±1.852e+00
DF14	(10,10)	0.8906±1.354e+00	**0.8381±1.38e+00**	0.9316±142e+00	0.9239±142e+00	0.09374±2.027e-02	0.8799±1.339e+00	0.9138±1.401e+00
	(10,5)	**0.7989±1.583e-01**	0.8482±1.37e+00	0.9383±142e+00	0.9500+14le+00	0.9250±1.122e-01	0.8857±1.342e+00	0.9977±1.508e+00
	(5,10)	**0.8605±2.653e-01**	0.8818±1.38e+00	1.0686±1.58e+00	1.0668±1.60e+00	0.08808±1.501e-02	0.8787±1.330e+00	0.9340±1.403e+00

It can be seen from Table 1 that the algorithm T-MOEA/D obtained better results for the MIGD metric. This algorithm obtained the 24 best results in total, KT-MOEAD obtained five best results, TR-MOEA/D obtained three good results, FD-MOEA/D obtained five good results, MMTL-MOEA/D obtained three good results, RI-MOEA/D and IT-MOEA/D each had one better

result. In most benchmark sets, the performance of T-MOEA/D was significantly better than that of the other competing algorithms. In particular, in DF1, DF2, DF5, and DF6, the algorithm still adapts to changes in the environment as quickly as possible. However, the performance of T-MOEA/D is poor in DF8 and DF9, where both POS and POF change drastically with the environment, leading to high complexity of the problem. The algorithm FD-MOEA/D achieved five better results on IGD evaluation metrics. The algorithm is designed to determine the center of mass of an individual by designing a time-series model from the perspective of the decision space to track the movement changes of an individual with the environment. The algorithm achieves a good performance when the POF changes are less drastic. The algorithm KT-MOEA/D uses linear prediction using the motion trend of the knee points under the target space and knowledge transfer for a small number of samples using a support vector machine binary classification model combined with the unbalanced TrAdaboost classifier. In DF14, the geometric structure of POF appears as variable linkages and knee point regions as the environment changes, so that the algorithm can better capture dynamic changes. In DF4, the position of POS and the length of POF change with the environment, and the algorithm MMTL-MOEA/D uses historical solutions to predict the position and length of solutions after environmental changes, which have good performance.

Figure 3 shows the IGD evaluation metrics for each environment for the seven algorithms compared on the DF benchmark. By drawing these curves, it can be seen that the curve obtained is at the bottom of the image in most cases. It can be observed that the curve of this method is smoother. This proves that the algorithm we designed not only exhibits good performance, but is also more stable.



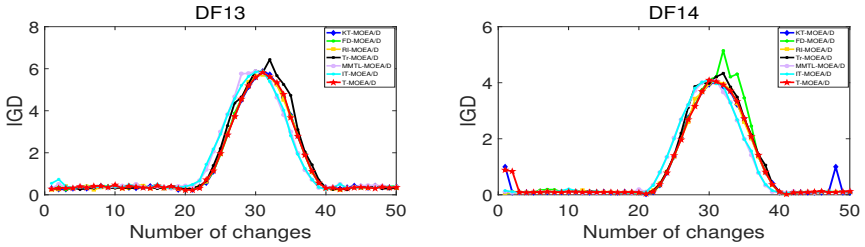


Figure 3: Approach of the domain adaptive method based on the geodesic flow kernel.

The MS values for the five algorithms are listed in Table 2. The proposed T-MOEA/D showed better performance than the other competing algorithms in terms of uniformity. The T-MOEA/D algorithm performed well on 27 of the 42 test cases shown in the figure. Tr-MOEA /D achieved six better performances. MMTL-MOEA/D performed four better performances, FD-MOEA/D showed two good performances, KT-MOEA/D and IT-MOEA/D had one good result each. We analyze all the dynamic settings in detail. T-MOEA/D performed best in DF3, DF4, DF6, and DF11. For most test cases, the performance of T-MOEA/D was better than those of the other methods. However, the diversity performance of this algorithm was weak for DF8 and DF12. DF8 will show mixed concavity and convexity changes with the change in the environment; in DF12 where the POF is relatively stable and the POS is time-varying, the search for the optimal Pareto solution set slows down with the degree of complexity of the dynamic problem. The algorithm Tr-MOEA/D obtained good results for the test set DF8. The method uses domain-adaptive learning to gain knowledge of the optimal Pareto solution, which ensures that the algorithm searches in the direction of better development in the next environment.

Table 2: Experimental results of MS obtained by different dynamic multi-objective optimization algorithms with different dynamic settings benchmark

Problems	Metrics	T-MOEA/D	KT-MOEA/D	Tr-MOEA/D	RI-MOEA/D	FD-MOEA/D	MMTL-MOEA/D	IT-MOEA/D
DF1	(10,10)	**0.9050±5.765e-02**	0.6748±2.923e-01	0.8122±1.292e-01	0.7791±7.569e-02	0.8181±1.634e-01	0.8686±1.112e-01	0.8332±1.515e-01
	(10,5)	0.7632±1.104e-01	0.6496±2.000e-01	0.7774±1.283e-01	0.7103±1.158e-01	0.7340±1.692e-01	**0.8113±1.291e-01**	0.7733±1.325e-01
	(5,10)	**0.8669±6.458e-02**	0.7049±1.440e-01	0.8007±1.025e-01	0.8004±1.082e-01	0.7990±1.302e-01	0. 8809±9.199e-02	0.8266±1.701e-01

DF2	(10,10)	**0.8886±7.195e-02**	0.7181±1.162e-01	0.7542±1.821e-01	0.6116±2.787e-01	0.7820±1.330e-01	0.7809±1.340e-01	0.8092±1.376e-01
	(10,5)	0.8095±1.082e-01	0.6388±1.912e-01	**0.8265±1.049e-01**	0.6604±2.057e-01	0.7443±1.359e-01	0.7472±1.494e-01	0.7689±1.326e-01
	(5,10)	**0.8652±1.113e-01**	0.6985±1.370e-01	0.8416±1.076e-01	0.6975±1.984e-01	0.7970±1.311e-01	0. 8147±1.324e-01	0.732±4.689e-02
DF3	(10,10)	0.4573±2.831e-01	0.2143±1.820e-01	0.2577±1.270e-01	0.1459±9.359e-02	0.1759±1.716e-01	**0.4677± 2.945e-01**	0.4263±3.073e-01
	(10,5)	**0.4363±2.356e-01**	0.2146±1.493e-01	0.2509±1.748e-01	0.2835±1.625e-01	0.1365±1.485e-01	0.4134± 2.951e-01	0.3855±2.962e-01
	(5,10)	**0.4727±2.764e-01**	0.2295±1.652e-01	0.2806±1.510e-01	0.2668±1.916e-01	0.2065±1.143e-01	0.4294 ± 2.857e-01	0.3953±2.907e-01
DF4	(10,10)	**0.3480±3.106e-01**	0.1882±1.492e-01	0.2373±1.432e-01	0.1621±1.997e-01	0.1995±1.634e-01	0.312±1.83E-01	0.3109±1.852e-0
	(10,5)	**0.6821±1.179e+00**	0.2369±1.710e-01	0.2600±1.763e-01	0.2426±1.249e-01	0.2493±1.528e-01	0.282±1.93E-01	0.3086±1.830e-01
	(5,10)	**0.3259±2.029e-01**	0.1965±1.349e-01	0.2069±1.482e-01	0.1337±1.545e-01	0.1764±1.426e-01	0.312±1.91E-01	0.3148±2.282e-01
DF5	(10,10)	**0.9691±3.542e-02**	0.7733±2.293e-01	8.822e-01±1.276e-01	0.8957±9.408e-02	0.9167±1.017e-01	0.9072±1.283e-01	0.8915±1.653e-01
	(10,5)	0.95001±5.202e-02	0.7856±1.680e-01	8.728e-01±1.478e-01	0.8887±4.531e-02	0.9367±5.555e-02	**0. 9621±1.073e-01**	0.9595±3.507e-01
	(5,10)	0.9603±4.252e-02	0.8698±1.924e-01	**0.9727±3.725e-02**	0.9331±5.555e-02	0.9308±5.148e-02	0.9174±8.991e-02	0.9158±2.759e-01
DF6	(10,10)	0.666±2.881e-01	0.6798±3.696e-01	**0.8125±2.980e-01**	0.7441±2.430e-01	0.6411±3.619e-01	0.6962 ± 3.125e-01	0.6215 ± 3.314e-01
	(10,5)	0.8335±2.407e-01	0.8541±1.657e-01	8.000e-01±3.079e-01	6.553e-01±2.443e-01	**0.8669±1.865e-01**	0.7411± 3.734e-01	0.7689 ± 2.949e-01
	(5,10)	**0. 8406±2.650e-01**	0.7842±2.183e-01	0.8330±2.567e-01	0.7530±2.430e-01	0.7611±3.525e-01	0.7688 ± 2.954e-01	0.6725 ± 2.865e-01
DF7	(10,10)	0.8531±2.503e-01	6.917e-01±3.929e-01	8.645e-01±3.135e-01	4.348e-01±3.769e-01	0.7608±3.795e-01	**0. 8676± 2.839e-01**	0.735 ± 3.669e-01
	(10,5)	**0. 8411±3.303e-01**	7.172e-01±4.007e-01	8.226e-01±3.566e-01	3.491e-01±3.637e-01	0.7899±3.436e-01	0.841 ± 3.040e-01	0.8143 ± 3.700e-01
	(5,10)	**0.8755±2.855e-01**	6.859e-01±4.080e-01	0.8735±3.276e-01	0.3099±3.638e-01	0.6687±3.874e-01	0.8357 ± 2.637e-01	0.8432 ± 2.789e-01
DF8	(10,10)	0.3731±3.225e-01	0.2331±2.139e-01	**0.3746±3.064e-01**	0.1772±2.625e-01	0.3083±2.294e-01	0.2986±2.541e-01	0.3208±2.071e-01
	(10,5)	**0.4263±3.389e-01**	0.2619±3.052e-01	0.4078±2.755e-01	0.1482±1.231e-01	0.3343±2.387e-01	0.3937±2.312e-01	0.3570±2.382e-01
	(5,10)	0.3202±3.316e-01	0.2148±2.875e-01	**0.3555±2.356e-01**	0.1113±6.904e-02	0.2832±2.336e-01	0.3008±2.664e-01	0.3146±2.244e-01
DF9	(10,10)	0.6994±3.383e-01	0.4275±4.040e-01	0.7224±2.645e-01	0.6144±2.650e-01	0.6799±3.839e-01	0.716±2.735e-01	**0.7301±2.261e-01**
	(10,5)	**0.8752±7.667e-01**	0.4870±3.856e-01	0.6895±2.526e-01	0.6754±3.318e-01	0.7321±3.442e-01	0.7622±2.370e-01	0.7284±2.579e-01
	(5,10)	**0.7874±2.538e-01**	0.5394±3.760e-01	0.7359±2.928e-01	0.5605±3.794e-01	0.5949±4.122e-01	0.7312±2.607e-01	0.7025 ± 2.649e-01
DF10	(10,10)	**0.9147±9.867e-02**	0.7405±1.809e-01	0.8820±9.017e-02	0.7588±1.816e-01	0.7648±1.791e-01	0.8921±1.232e-01	0.8758±1.362e-01
	(10,5)	0.8366±1.270e-01	0.8078±1.389e-01	**0.9169±8.348e-02**	0.5944±2.423e-01	0.7650±1.979e-01	0.9113±9.547e-02	0.8833±1.569e-01
	(5,10)	**0.9131±9.781e-02**	0.8581±1.527e-01	0.8849±1.224e-01	0.7866±2.066e-01	0.8549±1.498e-01	0.8934 ± 1.429e-01	0.8541±2.240e-01
DF11	(10,10)	**0.982±1.544e-02**	0.9817±1.270e-02	0.9788±2.241e-02	0.9678±2.073e-02	0.9715±2.566e-02	0.9369±4.365e-02	0.9273±5.267e-02
	(10,5)	**0.9751±1.938e-02**	0.9513±3.115e-02	0.9202±7.293e-02	0.9551±4.445e-02	0.9496±7.433e-02	0.9146±4.943e-02	0.8896±9.934e-02
	(5,10)	**0.9847±1.590e-02**	0.9747±2.554e-02	0.9626±2.122e-02	0.9686±2.608e-02	0.9709±2.567e-02	0.8882±8.001e-02	0.8807±8.723e-02
DF12	(10,10)	**0.2499±2.661e-01**	0.1975±7.649e-01	0.1459±1.984e-01	0.01857±5.770e-04	0.5604±3.617e-01	0.4443 ± 2.409e-01	0.4732±2.875e-01
	(10,5)	0.2711±2.600e-01	0.4824±7.532e-01	0.1164±1.949e-01	0.03659±5.864e-03	**0.5761±4.465e-01**	0.4476 ± 3.210e-01	0.5504±2.365e-01
	(5,10)	0.3438±3.567e-01	**0.780±1.674e-01**	0.1775±2.506e-01	0.01735±5.960e-04	0.5445±3.306e-01	0.4638± 2.847e-01	0.5306±2.483e-01
DF13	(10,10)	**0.9090±4.583e-02**	0.8104±9.350e-02	0.8648±8.288e-02	0.8226±7.328e-02	0.8481±8.134e-02	0.844±8.901e-02	0.859±9.277e-02
	(10,5)	**0.9075±1.026e-01**	0.8423±1.022e-01	0.8666±1.003e-01	0.7892±9.255e-02	0.8384±9.157e-02	0.8253±1.191e-01	0.861±8.215e-02
	(5,10)	**0.9375±4.624e-02**	0.8049±5.635e-02	0.8536±8.723e-02	0.8092±9.470e-02	0.8137±9.026e-02	0.8555±1.016e-01	0.8665±7.784e-02
DF14	(10,10)	**0.9732±3.161e-02**	0.9573±4.875e-02	0.9162±2.92e-03	0.9731±2.637e-02	0.9559±3.749e-02	0.8226 ± 2.423e-01	0.8281±2.377e-01
	(10,5)	**0.8937±1.340e-01**	0.8162e±1.696e-01	0.8167±1.346e-01	0.8519±1.382e-01	0.8014±2.480e-01	0.7817 ± 2.485e-01	0.8228±2.465e-01
	(5,10)	0.9662±2.361e-02	0.9637±3.301e-02	0.9039±1.78e-03	**0.9906±1.543e-02**	0.9327±4.608e-02	0.7546 ± 2.873e-01	0.7731±2.978e-01

The experimental results for the MHV values are recorded in Table 3. Compared to the IGD metric, the HV metric can better quantify the convergence and diversity of the solutions obtained by the algorithm, and the experimental results show that T-MOEA/D has 23 better performance results. In most cases, the algorithm was able to obtain more accurate estimates in the new environment. KT-MOEA/D has 7 good results, algorithm MMTL-MOEA/D has 6 good results, and IT-MOEA/D has 6 good results. The performance of the test sets DF2, DF5, DF6, and DF7 in algorithm T-MOEA/D is significantly better than the other algorithms. Because the PS and PF of these problems have regular variations, the knee points can be better tracked in the changing environment. However, the performance in DF12 and DF14 is slightly weaker than other algorithms, because the large deformation brings obstacles to individual prediction in a dynamic environment. In IT-MOEA/D, good performance was achieved in the benchmark test set DF12. The method designs a pre-search

mechanism from a decision space perspective to obtain high-quality individuals, and then constructs a sample-based transfer learning classifier TrAdaboost to predict the initial population after environmental changes, which can accurately capture the dynamic changes of POS in DF12.

Table 3: Experimental results of MHV obtained by different dynamic multi-objective optimization algorithms with different dynamic settings benchmark

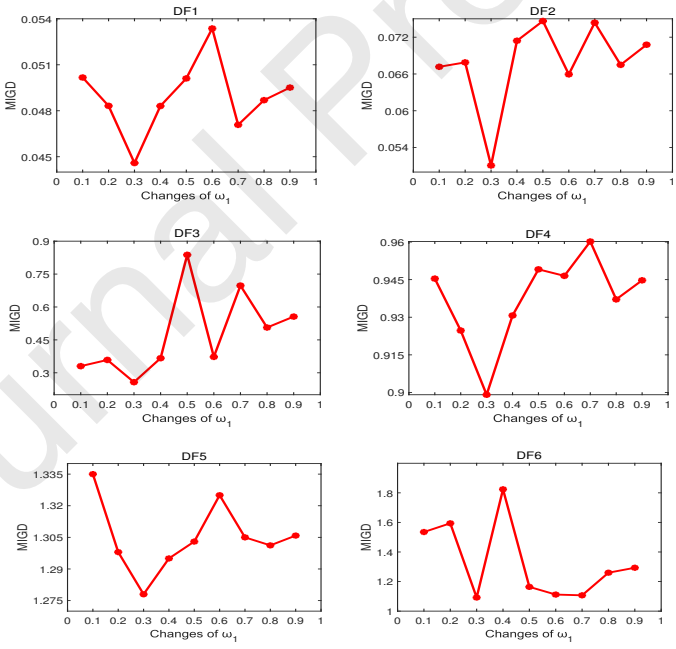
Problems	metrics	T-MOEA/D	KT-MOEA/D	Tr-MOEA/D	RI-MOEA/D	FD-MOEA/D	MMTL-MOEA/D	IT-MOEA/D
DF1	(10,10)	**0.5317±1.427e-01**	0.5187±1.740e-01	0.5250±1.430e-01	0.5292±1.403e-01	0.5315±1.369e-01	0.5172±1.258e-01	0.5149±1.780e-01
	(10,5)	**0.4438±1.526e-01**	0.3998±1.625e-01	0.4141±1.468e-01	0.0072±3.592e-02	0.4350±1.152e-01	0.4342±1.277e-01	0.4405±1.781e-01
	(5,10)	0.5171±1.459e-01	0.4870±1.724e-01	0.5120±1.494e-01	0.0125±6.636e-02	0.5182±1.290e-01	**0.5541±1.361e-01**	0.5220±1.693e-01
DF2	(10,10)	**0.7472±6.526e-02**	0.6477±1.140e-01	0.6328±7.081e-01	0.6401±0.082e-01	0.7429±8.158e-02	0.7262±7.841e-02	0.7393±1.707e-01
	(10,5)	0.6560±9.931e-02	0.6230±1.152e-01	0.6555±9.622e-02	0.0313±1.192e-01	0.6528±1.111e-01	0.6159±1.282e-01	**0.6989±7.528e-02**
	(5,10)	**0.7534±6.283e-02**	0.7411±6.278e-02	0.7518±7.091e-02	0.0218±1.079e-01	0.7249±7.772e-02	0.7267±9.077e-02	0.0732±4.689e-02
DF3	(10,10)	**0.2675±1.661e-01**	0.2600±1.887e-01	0.2512±1.491e-01	0.0228±7.736e-02	0.2299±1.636e-01	0.1852±1.583e-01	0.2387±1.669e-01
	(10,5)	0.2088±1.515e-01	**0.2348±2.022e-01**	0.1923±1.295e-01	0.0112±4.981e-02	0.1779±1.384e-01	0.1495±1.304e-01	0.1830±1.323e-01
	(5,10)	0.2580±1.719e-01	0.2750±2.022e-01	0.2479±1.476e-01	0.0148±6.786e-02	0.2183±1.560e-01	0.1782±1.484e-01	0.2227±1.616e-01
DF4	(10,10)	**3.0860±2.648e+00**	0.3640±2.672e-01	2.8820±2.547e+00	0.7112±1.794e+00	2.9160±2.711e+00	3.0390±2.514e+00	2.7730±2.402e+00
	(10,5)	2.7200±2.505e+00	**3.0150±2.614e+00**	2.5760±2.472e+00	0.3998±1.230e+00	2.7110±2.401e+00	3.0100±2.46E-00	2.5680±2.393e+00
	(5,10)	**3.0210±2.609e+00**	2.6150±2.478e+00	2.8630±2.659e+00	1.2180±2.003e+00	2.6930±2.575e+00	3.0190±2.650e+00	2.7510±2.580e+00
DF5	(10,10)	**0.3969±2.935e-01**	0.3919±2.887e-01	0.3801±2.891e-01	0.0262±1.095e-01	0.2973±2.800e-01	0.3686±2.822e-01	0.3901±2.920e-01
	(10,5)	**0.3283±2.560e-01**	0.3117±2.396e-01	0.3061±2.470e-01	0.0176±7.051e-02	0.2705±2.344e-01	0.3103±2.398e-01	0.3067±2.561e-01
	(5,10)	**0.3923±2.959e-01**	0.3836±2.904e-01	0.3786±2.896e-01	0.0199±9.846e-02	0.3625±2.821e-01	0.3596±2.840e-01	0.3713±2.889e-01
DF6	(10,10)	**0.2667±3.240e-01**	0.1538±2.322e-01	0.2336±3.021e-01	0.1529±1.081e-01	0.1297±2.477e-01	0.2105±2.969e-01	0.1748±2.535e-01
	(10,5)	**0.1179±2.140e-01**	0.2574±2.454e-01	0.0429±9.927e-02	0.0728±5.151e-02	0.0681±1.768e-01	0.1003±2.258e-01	0.0686±1.681e-01
	(5,10)	**0.2517±3.266e-01**	0.2368±2.651e-01	0.2382±2.686e-01	0.0028±1.978e-02	0.189±3.147e-01	0.2125±2.809e-01	0.1966±3.086e-01
DF7	(10,10)	**2.6870±5.343e+00**	2.3520±4.732e+00	2.1850±4.305e+00	0.0250±1.765e-01	2.007±4.610e+00	2.1660±4.651e+00	1.7220±3.756e+00
	(10,5)	**1.7640±4.078e+00**	0.0408±1.232e-01	1.3260±3.636e+00	0.0078±5.514e-02	2.001±4.738e+00	1.4450±3.753e+00	1.0370±3.057e+00
	(5,10)	**2.8750±5.440e+00**	2.3920±5.047e+00	2.4690±4.969e+00	1.1550±8.168e+00	2.302±4.729e+00	2.3160±5.605e+00	1.6120±4.542e+00
DF8	(10,10)	42.63±3.303e+01	42.68±3.315e+01	42.6200±3.314e+01	26.1600±2.837e+01	42.10±3.363e+01	**43.11±3.309e+01**	42.65±3.322e+01
	(10,5)	42.32±3.261e+01	42.01±3.335e+01	42.13±3.302e+01	28.8700±2.916e+01	32.08±3.327e+01	**43.08±3.328e+01**	42.17±3.320e+01
	(5,10)	43.05±3.255e+01	43.12±3.286e+01	43.38±3.286e+01	26.7800±2.928e+01	43.43±3.296e+01	43.56±3.281e+01	**43.65±3.309e+01**
DF9	(10,10)	18.13±2.665e+01	**18.40±2.682e+01**	18.09±2.697e+01	0.3953±2.795e+00	17.71±2.581e+01	15.65±2.455e+01	15.81±2.522e+01
	(10,5)	**17.63±2.625e+01**	2.0290±1.804e+00	17.20±2.626e+01	0.190±1.900e+00	17.50±2.631e+01	14.74±2.309e+01	14.23±2.194e+01
	(5,10)	**17.57±2.552e+01**	15.9800±2.366e+01	14.90±2.263e+01	13.47±1.052e+01	17.350±2.521e+01	11.56±1.880e+01	11.83±2.010e+01
DF10	(10,10)	0.8733±3.769e-01	**0.8908±3.847e-01**	0.8595±4.078e-01	0.6277±4.536e-01	0.8836±3.841e-01	0.8830±3.873e-01	0.8775±3.784e-01
	(10,5)	0.8222±4.029e-01	0.2980±2.183e-01	0.8100±4.176e-01	2.3110±9.798e+00	0.7332±4.110e-01	**0.8398±4.127e-01**	0.8207±4.081e-01
	(5,10)	**0.8081±3.960e-01**	0.7941±4.038e-01	0.8002±4.151e-01	0.5288±4.659e-01	0.7243±4.011e-01	0.8001±4.073e-01	0.7885±4.228e-01
DF11	(10,10)	**0.7805±2.022e-01**	0.7278±2.098e-01	0.7116±1.902e-01	0.3137±1.849e-01	0.7661±1.923e-01	0.7257±3.574e-01	0.7271±3.745e-01
	(10,5)	**0.6907±2.031e-01**	0.6744±2.174e-01	0.5928±1.645e-01	0.2105±1.884e-01	0.6903±2.080e-01	0.6623±3.292e-01	0.6269±3.050e-01
	(5,10)	0.7737±2.089e-01	0.7387±2.179e-01	0.7044±2.039e-01	0.3861±1.814e-01	0.7531±1.959e-01	**0.8711±6.748e-01**	0.8118±6.310e-01
DF12	(10,10)	9.2590±2.110e+00	0.9873±2.780e-01	8.9960±2.105e+00	9.2930±2.115e+00	9.4750±2.157e+00	9.2930±2.115e+00	**19.29±1.282e+01**
	(10,5)	9.2850±2.123e+00	0.9892±2.817e-01	8.9870±2.188e+00	6.4740±3.993e+00	1.2420±2.695e+01	9.2930±2.115e+00	**17.97±1.801e+01**
	(5,10)	8.8590±2.579e+00	8.6210±2.570e+00	8.7130±2.608e+00	8.1800±2.492e+00	7.3380±1.167e+00	8.9110±2.583e+00	**15.53±1.426e+01**
DF13	(10,10)	1.7090±1.194e+00	1.7150±1.201e+00	1.6690±1.182e+00	0.1300±4.152e-01	1.7730±1.239e+00	**1.8410±1.266e+00**	1.7320±1.256e+00
	(10,5)	1.5330±1.097e+00	1.5350±1.093e+00	1.4520±1.095e+00	0.9819±3.381e-01	1.5230±1.070e+00	1.5590±1.082e+00	**1.6310±1.207e+00**
	(5,10)	**1.7740±1.210e+00**	1.7170±1.183e+00	1.7380±1.225e+00	1.1140±4.153e-01	1.7410±1.180e+00	1.7240±1.245e+00	1.8080±1.255e+00
DF14	(10,10)	**0.2022±2.359e-01**	0.2015±2.379e-01	0.1531±1.762e-01	0.0004±2.472e-03	0.2008±2.347e-01	0.1998±2.311e-01	0.1977±2.258e-01
	(10,5)	0.1890±2.167e-01	**0.3209±1.968e-01**	0.3170±1.928e-01	0.0011±5.299e-03	0.1865±2.182e-01	0.1826±2.109e-01	0.1763±2.001e-01
	(5,10)	0.2028±2.362e-01	**0.3804±2.025e-01**	0.3780±2.055e-01	0.2786±2.039e-01	0.2006±2.329e-01	0.2023±2.292e-01	0.2048±2.335e-01

From a statistical perspective, as shown in Table 4, the p-values of the algorithms proposed in this study are less than 5% compared to the other algorithms. This indicates that there is a significant difference, and this algorithm is superior to the other algorithms. In T-MOEA/D, parameter ω plays a key role in determining the location of the predicted knee points. The choice of parameter ω affects the prediction performance. Figure 4 shows the effect of

the choice of ω on the IGD of the KT-MOEA/D algorithm. In the T-MOEA/D, a weighted approach was used to predict the set of knee points. The choice of parameter ω affects the prediction performance; the sum of ω_1 and ω_2 is always one. When ω_1 is 0.3 and ω_2 is 0.7, the convergence effect is the best. The accurate tracking of historical information can more precisely predict the knee point position at the next moment, which enhances the convergence of the algorithm.

Table 4: test results of Wilcoxon

T-MOEAD vs.	p-value
KT-MOEAD	1.75E-06
Tr-MOEAD	1.70E-04
RI-MOEAD	3.69E-06
FD-MOEAD	1.27E-06
MMTL-MOEA/D	1.70E-03
IT-MOEA/D	1.40E-04



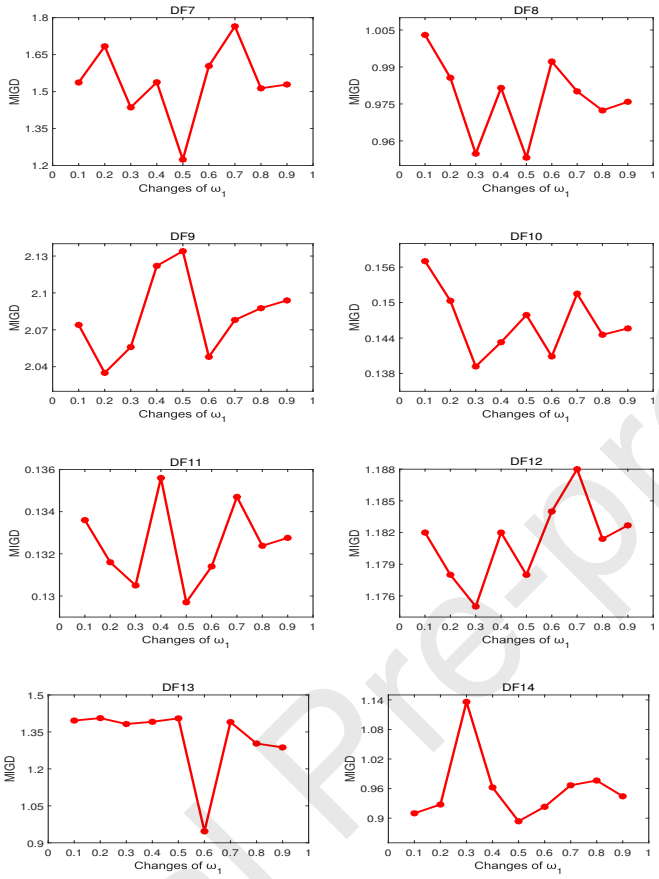


Figure 4: The effect of ω parameter on IGD indicators

4.3. CPU time

To demonstrate the advantages of the T-MOEA/D algorithm in terms of convergence and evolution speed. We used the hardware configuration of AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz to compare the CPU time, and the system version used was Windows 10. The running times of T-MOEA/D, KT-MOEA/D, Tr-MOEA/D, MMTL-MOEA/D, and IT-MOEA/D in the DF series test set are listed in Table 5. Experiments show that the running time of T-MOEA/D is much shorter than that of other algorithms. After identifying individuals in the target domain, Tr-MOEA/D and IT-MOEA/D spend a lot of time searching for the corresponding individuals in the source domain. Using the

knee point to perform the imbalanced transfer in KT-MOEA/D wastes a large number of high-quality individuals and increases the computational cost. The algorithm MMTL-MOEA/D maps elite individuals in the sample geodesic flow (SGF), but the number of subspaces in the SGF cannot be determined, which affects the accuracy of the algorithm in obtaining estimates. In the algorithm design of this study, the prediction of knee points in the target space and the transfer of suboptimal solutions were used to generate better Pareto solutions in the decision space. The running times of DF10, DF11, DF13, and DF14 were the shortest, which proves that this algorithm is more suitable for dealing with three-objective optimization problems. In summary, the algorithm is still the most competitive in terms of IGD, MS, HV performance metrics, and running timecombined.

Table 5: Running time of five algorithms on different functions (Unit: second)

Problems	T-MOEA/D	KT-MOEA/D	Tr-MOEA/D	MMTL-MOEA/D	IT-MOEA/D
DF1	**1.10e+02**	1.30e+02	4.28e+02	1.42e+02	1.28e+02
DF2	**9.77e+01**	9.81e+01	4.21e+02	1.41e+02	1.37e+02
DF3	**7.43e+01**	1.28e+02	3.63e+02	1.09e+02	1.09e+02
DF4	**7.26e+01**	1.17e+02	4.15e+02	8.68e+01	1.01e+02
DF5	**9.05e+01**	1.38e+02	4.33e+02	1.33e+02	1.28e+02
DF6	**8.09e+01**	8.84e+01	3.57e+02	1.12e+02	9.70e+01
DF7	**7.48e+01**	8.65e+01	3.25e+02	1.11e+02	9.81e+01
DF8	**8.51e+01**	2.56e+02	3.71e+02	1.29e+02	1.17e+02
DF9	**8.71e+01**	9.52e+01	3.48e+02	1.19e+02	1.10e+02
DF10	**1.79e+02**	3.07e+02	4.31e+02	2.11e+02	3.73e+02
DF11	**2.05e+02**	3.29e+02	4.43e+02	2.26e+02	3.76e+02
DF12	**1.59e+02**	4.27e+02	3.73e+02	1.97e+02	2.82e+02
DF13	**1.86e+02**	3.06e+02	4.41e+02	2.44e+02	3.62e+02
DF14	2.88e+02	3.19e+02	4.31e+02	**2.37e+02**	3.84e+02

4.4. Ablation Study

DMOEA's main idea is to use the knee point knowledge obtained in the previous three environments combined with the non-knee point knowledge, which can generate the initial population in the next environment using prediction and manifold transfer methods, respectively. It can predict the best solutions in one environment and retain them to provide a better initial population for

the next environment. The effectiveness of the proposed method was verified by conducting ablation experiments to compare the MIGD and MS results of the four algorithms. Among the 28 datasets, the algorithm T-MOEA/D obtained 18 better performance results. As listed in Table 6, the ablation experiments in this study used T-DMOEA, RI-MOEA/D, T-MOEA/DP, and T-MOEA/DG. The algorithm T-MOEA/DP means that only the MTPM model is used to solve DMOPs, and the algorithm T-MOEA/DG means that only the manifold TL method is used to predict the initial population in the new environment. The dynamic setting ($n_t = 10, \tau_t = 10$) was chosen as the benchmark for this experiment.

Table 6: Experimental results of metrics obtained by RI-MOEA/D, T-MOEA/DG, T-MOEA/DP, and T-MOEA/D

Problems	metrics	T-MOEA/D	T-MOEA/DG	T-MOEA/DP	RI-MOEA/D
DF1	MIGD	**0.08188±6.961e-02**	0.1176±7.908e-02	0.9025±6.964e-02	0.1222±1.70-01
	MS	**0.9050±5.765e-02**	0.6961±1.855e-01	0.9001±6.964e-02	0.7791±7.569e-02
DF2	MIGD	**0.07312±4.464e-02**	0.08352±4.952e-02	0.07763±4.914e-02	0.0898±5.03e-02
	MS	**0.8886±7.195e-02**	0.7850±1.234e-01	0.8061±1.440e-01	0.6116±2.787e-01
DF3	MIGD	**0.3449±2.053e-01**	4.357e-01±3.156e-01	0.3779±2.325e-01	0.4207±2.15e-01
	MS	0.3905±2.477e-01	**0.6137±3.615e-01**	0.4106±3.573e-01	0.1459±9.359e-02
DF4	MIGD	1.007±6.301e-01	1.118±4.664e-01	**0.9946±5.521e-01**	1.1112±5.86e-014
	MS	0.3480±3.106e-01	**0.4248±3.198e-01**	0.2850±2.090e-01	0.1621±1.997e-01
DF5	MIGD	**1.281±2.028e+00**	1.476±2.300e+00	1.301±2.089e+00	1.5440±2.38e+00
	MS	**0.9691±3.542e-02**	0.8982±1.215e-01	0.9028±1.143e-01	0.8957±9.408e-02
DF6	MIGD	**1.655±3.557e+00**	2.973±5.813e+00	1.712±2.198e+00	2.7378±3.09e+00
	MS	2.434±3.744e+00	**4.074±1.076e+01**	1.283±2.032e+00	0.6949±6.294e-01
DF7	MIGD	**1.219±4.178e-01**	2.838±5.497e+00	1.991±3.917e+00	2.9849±3.85e+00
	MS	0.1526±3.524e+01	**0.1531±3.524e+01**	0.1344±1.956e+01	0.1356±3.764e+01
DF8	MIGD	0.9499±4.966e-01	0.9788±5.297e-01	**0.9423±5.106e-01**	0.9517±5.08e-01
	MS	**0.3731±3.225e-01**	0.2971±2.780e-01	0.2506±2.337e-01	0.1772±2.625e-01
DF9	MIGD	3.040±1.476e+00	2.091±1.930e+00	2.059±1.800e+00	**1.6587±1.62e+00**
	MS	**0.6994±3.383e-01**	0.6344±2.638e-01	0.6668±2.714e-01	0.6144±2.650e-01
DF10	MIGD	**0.1310±2.424e-02**	0.2165±7.348e-02	0.1409±4.817e-02	0.2036±8.49e-02
	MS	0.8809±9.959e-02	0.6600±2.620e-01	**0.9237±1.126e-01**	0.7588±1.816e-01
DF11	MIGD	0.1591±4.256e-02	0.1674±4.994e-02	**0.1329±1.735e-02**	0.1916±4.77e-02
	MS	**0.9820±1.544e-02**	0.9248±8.311e-02	0.9757±3.095e-02	0.9678±2.073e-02
DF12	MIGD	0.9197±2.873e-01	**0.7666±2.801e-01**	1.162±9.129e-02	0.9390±2.83e-01
	MS	**0.2499±2.661e-01**	0.2035±1.836e-01	0.05537±1.638e-01	0.001857±5.770e-04
DF13	MIGD	**1.371±1.806e+00**	1.478±1.915e+00	1.383±1.791e+00	1.4751±1.94e+00
	MS	**0.909±4.583e-02**	0.7977±1.043e-01	0.8605±6.966e-02	0.8226±7.328e-02
DF14	MIGD	**0.8906±1.354e+00**	0.9556±1.354e+00	0.9009±1.341e+00	0.9239±1.42e+00
	MS	**0.9732±3.161e-02**	0.8732±9.383e-02	0.9231±2.312e-02	0.9731±2.637e-02

When using the algorithm T-MOEA/D to provide the initial population for the next environment, the comparison shows that the experimental results of MIGD and MS are significantly improved compared to the other algorithms, and

the combination of both mechanisms is better than using only one mechanism. The algorithm T-MOEA/DP obtains the solution in the new environment by tracking the historical movement trend of the knee point; the number of knee points is relatively small. Moreover, owing to the characteristics of the knee points, it becomes difficult to solve the problem with concave and convex variations, which causes the algorithm to converge more slowly. In the algorithm T-MOEA/DG, the technique of TL is used to respond to changes in the environment. When all individuals are transferred, some bad individuals create a large computational burden during the transfer process and cause negative transfer, which makes the algorithm unable to obtain an accurate prediction of individuals in the new environment. Thus, when the environment changes, the combination of these two methods allows the evolutionary algorithm to perform better in the new environment and also improves the convergence and diversity of the population.

5. Conclusion

We conclude that the use of TL techniques to solve DMOPs is an effective approach. However, these methods face various challenges in terms of convergence. To solve these problems, we propose a dynamic multi-objective optimization algorithm based on knowledge transfer to solve DMOPs. Based on changes in the dynamic environment, the MTPM model was used to predict the knee point of the new environment. The populations in the two environments before and after the change were used as the source and target domains, respectively. Sub-optimal solutions are transferred using kernel-based manifold transfer techniques. These solutions can guide the generation of better initial populations in new environments by merging high-quality individuals of these two components during dynamic evolution. In future work, we will apply this method to real-life dynamic problems and continue to explore the potential of TL techniques in dynamic scenarios.

Author contributions

Linjie Wu and Tianhao Zhao proposed the improved method and wrote the first draft.

Di Wu was responsible for reviewing the paper. Xingjuan Cai and Liping Xie made grammatical revisions and finalized the paper.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this study.

Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant No.YDZJSX2021A038 the National Natural Science Foundation of China under Grant No.61806138.

References

[1] Huangke Chen, Guohua Wu, Witold Pedrycz, Ponnuthurai Nagaratnam Suganthan, Lining Xing, and Xiaomin Zhu. An adaptive resource allocation strategy for objective space partition-based multiobjective optimization. *IEEE TRANSACTIONS ON SYSTEMS MAN CYBERNETICS-SYSTEMS*, 51(3):1507–1522, MAR 2021.

[2] X. Cai, Z. Hu, and J. Chen. A many-objective optimization recommendation algorithm based on knowledge mining. *Information ences*, 2020.

[3] A Zc, Z. A. Peng, A Zh, A Xc, B Wz, and C Jc. An improved matrix factorization based model for many-objective optimization recommendation. *Information Sciences*, 2021.

[4] A Zc, A Jz, W. A. Di, A Xc, W. B. Hui, C Wz, and D Jc. Hybrid many-objective particle swarm optimization algorithm for green coal production problem. *Information Sciences*, 518:256–271, 2020.

[5] H. Wang, K. Li, and W. Pedrycz. An elite hybrid metaheuristic optimization algorithm for maximizing wireless sensor networks lifetime with a sink node. *IEEE Sensors Journal*, PP(99):1–1, 2020.

[6] Huangke Chen, Ran Cheng, Witold Pedrycz, and Yaochu Jin. Solving many-objective optimization problems via multistage evolutionary search. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(6):3552–3564, 2019.

[7] H. Chen, Y. Tian, W. Pedrycz, G. Wu, R. Wang, and L. Wang. Hyperplane assisted evolutionary algorithm for many-objective optimization problems. *IEEE Transactions on Cybernetics*, pages 1–14, 2019.

[8] F. Zou, G. G. Yen, L. Tang, and C. Wang. A reinforcement learning approach for dynamic multi-objective optimization. *Information Sciences*, 546(4):815–834, 2021.

[9] W. A. Feng, A Fl, A Yl, and W. B. Hui. A new prediction strategy for dynamic multi-objective optimization using gaussian mixture model. *Information Sciences*, 2021.

[10] W. Du, W. Zhong, Y. Tang, W. Du, and Y. Jin. High-dimensional robust multi-objective optimization for order scheduling: A decision variable classification approach. *IEEE Transactions on Industrial Informatics*, PP(99):1–1, 2018.

[11] Y. N. Guo, J. Cheng, S. Luo, and D. W. Gong. Robust dynamic multi-objective vehicle routing optimization method. *IEEE/ACM Transactions on Computational Biology & Bioinformatics*, pages 1–1, 2017.

[12] T. Lin and H. Zha. Riemannian manifold learning. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 30(5):796, 2008.

[13] Juan Zou, Qingya Li, Shengxiang Yang, Hui Bai, and Jinhua Zheng. A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization. *APPLIED SOFT COMPUTING*, 61:806–818, DEC 2017.

[14] Min Jiang, Zhongqiang Huang, Liming Qiu, Wenzhen Huang, and Gary G. Yen. Transfer learning-based dynamic multiobjective optimization algorithms. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, 22(4):501–514, AUG 2018.

[15] Erchao Li and Xiangqi Ma. Dynamic multi-objective optimization algorithm based on transfer learning for environmental protection. *EKOLOJI*, 28(107):2509–2519, 2019.

[16] Min Jiang, Zhenzhong Wang, Haokai Hong, and Gary G. Yen. Knee point-based imbalanced transfer learning for dynamic multiobjective optimization. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, 25(1):117–129, FEB 2021.

[17] J. Branke, K. Deb, H. Dierolf, and M. Osswald. Finding knees in multi-objective optimization. *DBLP*, 2004.

[18] Min Jiang, Zhenzhong Wang, Haokai Hong, and Gary G. Yen. Knee point-based imbalanced transfer learning for dynamic multiobjective optimization. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, 25(1):117–129, FEB 2021.

[19] Indraneel Das. On characterizing the "knee" of the pareto curve based on normal-boundary intersection. *Structural Optimization*, 18:107–115, 10 1999.

[20] Ruwang Jiao, Sanyou Zeng, Changhe Li, and Witold Pedrycz. Evolutionary constrained multi-objective optimization using nsga-ii with dynamic constraint handling. pages 1634–1641, 06 2019.

[21] Qingda Chen, Jinliang Ding, Shengxiang Yang, and Tianyou Chai. A novel evolutionary algorithm for dynamic constrained multiobjective optimization problems. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, 24(4):792–806, AUG 2020.

[22] Kalyanmoy Deb, Udaya N, and Karthik Sindhya. Dynamic multi-objective optimization and decision-making using modified nsga-ii: A case study on hydro-thermal power scheduling. pages 803–817, 05 2007.

[23] Mario Camara, Julio Ortega, and Francisco de Toro. A single front genetic algorithm for parallel multi-objective optimization in dynamic environments. *NEUROCOMPUTING*, 72(16-18, SI):3570–3579, OCT 2009.

[24] Leilei Cao, Lihong Xu, Erik D Goodman, and Hui Li. A first-order difference model-based evolutionary dynamic multiobjective optimization. In *Simulated Evolution and Learning: 11th International Conference, SEAL 2017, Shenzhen, China, November 10–13, 2017, Proceedings 11*, pages 644–655. Springer, 2017.

[25] Radhia Azzouz, Slim Bechikh, and Lamjed Ben Said. A dynamic multi-objective evolutionary algorithm using a change severity-based adaptive population management strategy. *SOFT COMPUTING*, 21(4):885–906, FEB 2017.

[26] Min Jiang, Zhenzhong Wang, Liming Qiu, Shihui Guo, Xing Gao, and Kay Chen Tan. A fast dynamic evolutionary multiobjective algorithm via manifold transfer learning. *IEEE Transactions on Cybernetics*, 51(7):3417–3428, 2020.

[27] Zhou Peng, Jinhua Zheng, and Juan Zou. A population diversity maintaining strategy based on dynamic environment evolutionary model for dynamic multiobjective optimization. In *2014 IEEE Congress on Evolutionary Computation (CEC)*, pages 274–281. IEEE, 2014.

[28] Yinan Guo, Huan Yang, Meirong Chen, Jian Cheng, and Dunwei Gong. Ensemble prediction-based dynamic robust multi-objective optimization methods. *SWARM AND EVOLUTIONARY COMPUTATION*, 48:156–171, AUG 2019.

[29] Miao Rong, Dunwei Gong, Yong Zhang, Yaochu Jin, and Witold Pedrycz. Multidirectional prediction approach for dynamic multiobjective optimization problems. *IEEE TRANSACTIONS ON CYBERNETICS*, 49(9):3362–3374, SEP 2019.

[30] Miao Rong, Dunwei Gong, Witold Pedrycz, and Ling Wang. A multimodel prediction method for dynamic multiobjective evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 24(2):290–304, 2019.

[31] Min Jiang, Zhenzhong Wang, Shihui Guo, Xing Gao, and Kay Chen Tan. Individual-based transfer learning for dynamic multiobjective optimization. *IEEE TRANSACTIONS ON CYBERNETICS*, 51(10):4968–4981, OCT 2021.

[32] Elaine Guerrero-Pena and Aluizio F. R. Araujo. Dynamic multi-objective evolutionary algorithm with objective space prediction strategy. *APPLIED SOFT COMPUTING*, 107, AUG 2021.

[33] Fei Zou, Gary G. Yen, and Lixin Tang. A knee-guided prediction approach for dynamic multi-objective optimization. *INFORMATION SCIENCES*, 509:193–209, JAN 2020.

[34] Zhengping Liang, Ya Zou, Shunxiang Zheng, Shengxiang Yang, and Zexuan Zhu. A feedback-based prediction strategy for dynamic multi-objective evolutionary optimization. *EXPERT SYSTEMS WITH APPLICATIONS*, 172, JUN 15 2021.

[35] Min Jiang, Zhenzhong Wang, Shihui Guo, Xing Gao, and Kay Chen Tan. Individual-based transfer learning for dynamic multiobjective optimiza-

tion. *IEEE TRANSACTIONS ON CYBERNETICS*, 51(10):4968–4981, OCT 2021.

[36] Liang Feng, Wei Zhou, Weichen Liu, Yew-Soon Ong, and Kay Chen Tan. Solving dynamic multiobjective problem via autoencoding evolutionary search. *IEEE TRANSACTIONS ON CYBERNETICS*, 52(5):2649–2662, MAY 2022.

[37] Min Jiang, Liming Qiu, Zhongqiang Huang, and Gary G. Yen. Dynamic multi-objective estimation of distribution algorithm based on domain adaptation and nonparametric estimation. *INFORMATION SCIENCES*, 435:203–223, APR 2018.

[38] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2066–2073. IEEE, 2012.

[39] Qingfu Zhang, Aimin Zhou, and Yaochu Jin. Rm-meda: A regularity model-based multiobjective estimation of distribution algorithm. *IEEE Transactions on Evolutionary Computation*, 12(1):41–63, 2008.

[40] Shouyong Jiang, Shengxiang Yang, Xin Yao, Kay Chen Tan, Marcus Kaiser, and Natalio Krasnogor. Benchmark functions for the cec'2018 competition on dynamic multiobjective optimization. Technical report, Newcastle University, 2018.

[41] Qingfu Zhang and Hui Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731, 2007.

[42] Joaquin Derrac, Salvador Garcia, Daniel Molina, and Francisco Herrera. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *SWARM AND EVOLUTIONARY COMPUTATION*, 1(1):3–18, MAR 2011.

Dear Professor

It is a great pleasure for us to submit a research paper entitled:

"Dynamic Multi-objective Evolutionary Algorithm Based On Knowledge Transfer"

This article has not been published previously and it is not under consideration for publication elsewhere. This publication is approved by all authors and if accepted, it will not be published elsewhere in the same form, in English or in any other language, without the written consent of the copyright-holder.

Conflict of Interests

No conflict of interest exits in the submission of this manuscript, and manuscript is approved by all authors for publication.

We will be happy to hear your valuable comments.

Best regards,

Xingjuan Cai

School of Computer Science and Technology, Taiyuan University of Science and Technology, China

Credit authorship contribution statement

Linjie Wu: Conceptualization, Methodology, Experiment - Operation & Analysis, Writing - Original Draft.
Di Wu: Conceptualization, Methodology, Supervision, Validation, Writing - Review & Editing.
Tianhao Zhao: Methodology, Experiment - Operation, Writing - Review & Editing.
Xingjuan Cai & Liping Xie: Methodology, Supervision, Investigation.