



Multiobjective multitasking optimization assisted by multidirectional prediction method

Qianlong Dang¹ · Weifeng Gao¹ · Maoguo Gong²

Received: 6 August 2021 / Accepted: 10 December 2021 / Published online: 4 January 2022
© The Author(s) 2021

Abstract

Multiobjective multitasking optimization (MTO) is an emerging research topic in the field of evolutionary computation, which has attracted extensive attention, and many evolutionary multitasking (EMT) algorithms have been proposed. One of the core issues, designing an efficient transfer strategy, has been scarcely explored. Keeping this in mind, this paper is the first attempt to design an efficient transfer strategy based on multidirectional prediction method. Specifically, the population is divided into multiple classes by the binary clustering method, and the representative point of each class is calculated. Then, an effective prediction direction method is developed to generate multiple prediction directions by representative points. Afterward, a mutation strength adaptation method is proposed according to the improvement degree of each class. Finally, the predictive transferred solutions are generated as transfer knowledge by the prediction directions and mutation strengths. By the above process, a multiobjective EMT algorithm based on multidirectional prediction method is presented. Experiments on two MTO test suits indicate that the proposed algorithm is effective and competitive to other state-of-the-art EMT algorithms.

Keywords Evolutionary multitasking (EMT) · Multiobjective optimization · Evolutionary algorithm · Knowledge transfer strategy · Multidirectional prediction method

Introduction

Evolutionary computing is an important research direction in the field of artificial intelligence, and its algorithm, evolutionary algorithm, is a population-based search algorithm. Because of the excellent performance of evolutionary algorithm, it has been applied in diverse fields, such as path planning [1], shop scheduling [2], multi-agent system [3], networks planning [4], bi-level optimization [5], and symbolic regression [6].

Multiobjective multitasking optimization (MTO) [7] is a new research topic in the field of evolutionary computing, which can solve multiple different multiobjective optimization problems (MOPs) simultaneously and improve the quality of the solutions through knowledge transfer. As pioneers, Gupta et al. [8] first proposed a multiobjective evolutionary multitasking (EMT) algorithm named MO-MFEA to tackle MOPs, which can improve the performance of each task through useful knowledge transfer. Thereafter, many multiobjective EMT algorithms have been proposed. According to the way of knowledge transfer, these algorithms can be divided into the following three categories: (1) adaptive knowledge transfer strategy [9–12]; (2) search space mapping strategy [13–17]; (3) valuable knowledge selection strategy [18–20].

The adaptive knowledge transfer strategy is to adaptively control the intensity of knowledge transfer according to the search process. For adaptive knowledge transfer, Bali et al. [9] changed the intensity of knowledge transfer by learning the similarity between tasks, so as to improve the performance of the algorithm. Yang et al. [10] proposed an algorithm to divide decision variables into diversity variables and convergence variables. By increasing the transfer inten-

✉ Weifeng Gao
gaoweifeng2004@126.com

Qianlong Dang
xidiansldang@163.com

Maoguo Gong
gong@ieee.org

¹ School of Mathematics and Statistics, Xidian University, Xi'an 710126, China

² Key Laboratory of Intelligent Perception and Image Understanding, International Research Center for Intelligent Perception and Computation, Ministry of Education, Xidian University, Xi'an 710071, China

sity of the diversity variables, the solution distribution on Pareto front (PF) is uniform, and reducing the knowledge transfer on the convergence variables makes the population converge to PF. Zheng et al. [11] defined a capability vector to control the intensity of knowledge transfer for improving the performance of the algorithm. Zhou et al. [12] proposed the MFEA-AKT in which the crossover operators of knowledge transfer are controlled through the evolutionary state.

The search space mapping strategy is to use the learned mapping method to align the distribution in the subspace for reducing the differences between the subpopulations of different tasks. Bali et al. [13] used the linear domain adaptive method to map two tasks to a higher order representation space which is easier to share similarity, to achieve more efficient knowledge transfer. To solve the vehicle routing problem, Feng et al. [14] improved search space mapping and explicit knowledge transfer using automatic encoder to solve the problem of transfer mapping. Ding et al. [15] implemented an EMT algorithm to solve expensive optimization problems, which can promote knowledge transfer between optimization problems with different optimal locations and different number of decision variables. Liang et al. [16] introduced an algorithm called MOMFEA-SADE which builds the transfer matrix based on the subspace alignment strategy. Zhou et al. [17] proposed an algorithm named KAES which designs a kernelized autoencoding to capture nonlinearity between problems in pursuit of efficient knowledge transfer across domains.

The valuable knowledge selection strategy can find the valuable solutions to improve the efficiency of positive transfer. Feng et al. [18] proposed an algorithm named EMEA which adopts the nondominated solutions in each task as valuable knowledge to transfer, whereas this algorithm was only applicable to solve highly related tasks. Lin et al. [19] selected the new transferred solutions near the positive transferred solutions, and this method can improve the efficiency of positive knowledge transfer. In addition, Lin et al. [20] introduced an EMT algorithm according to incremental learning. In this paper, the Naive Bayes classifier finds the valuable areas through the positive transferred solutions, and selects the new transferred solutions in these areas.

The efficient knowledge transfer strategy can significantly improve the convergence speed and the quality of the solutions. Different from the previous work, this paper designs a new transfer strategy, i.e., the transfer strategy based on multidirectional prediction method. First, according to the method of binary clustering, multiple representative points are generated for the population, and the population is divided into multiple classes. Second, the prediction direction of each class is generated based on the representative points. Then the adaptive mutation strength is designed for each class according to the degree of population improvement. Finally, the predictive transferred solutions are generated by prediction

directions and mutation strengths. Based on the above facts, this paper proposes a multiobjective EMT algorithm with multidirectional prediction method (EMT-MPM). The main contributions are listed as follows:

1. This paper is the first attempt to design an efficient transfer strategy by multidirectional prediction method.
2. An effective prediction direction generation method is developed to generate multiple prediction directions.
3. A mutation strength adaptation method is proposed to flexibly control the range of transfer.
4. The performance of EMT-MPM is investigated on two MTO test suits. Compared with the state-of-the-art EMT algorithms, the experimental results indicate that EMT-MPM is effective and competitive.

The rest of this paper is arranged as follows. The next section gives the related work and research motivation. The subsequent section presents the details of EMT-MPM followed by which the experimental results are presented. The final section concludes this paper and proposes several future works.

Related work

MTO problem

MTO [7] has attracted more and more attention in EMT community. MTOs aim to solve multiple different MOPs simultaneously for better solutions through knowledge transfer between tasks. Generally, an MTO problem with K tasks can be defined as follows:

$$\text{Minimize : } \begin{cases} F_1(\mathbf{x}_1) = (f_1^1(\mathbf{x}_1), \dots, f_1^{m_1}(\mathbf{x}_1)) \\ F_2(\mathbf{x}_2) = (f_2^1(\mathbf{x}_2), \dots, f_2^{m_2}(\mathbf{x}_2)) \\ \vdots \\ F_j(\mathbf{x}_j) = (f_j^1(\mathbf{x}_j), \dots, f_j^{m_j}(\mathbf{x}_j)) \\ \vdots \\ F_K(\mathbf{x}_K) = (f_K^1(\mathbf{x}_K), \dots, f_K^{m_K}(\mathbf{x}_K)) \end{cases} \quad (1)$$

$$\text{subject to } \mathbf{x}_j \in \Omega^{d_j}, \quad j = 1, 2, \dots, K,$$

where F_j denotes the j th multiobjective task which consists of m_j objective functions; \mathbf{x}_j represents the solution in the j th task; Ω^{d_j} denotes the search space of j th task whose dimension is d_j .

Prediction-based methods

Many prediction-based methods have been proposed to solve dynamic multiobjective optimization problems, which can

be classified into three categories, namely, individual-based prediction [21], population-based prediction [22–25], and multidirectional prediction [26].

1. *Individual-based prediction (INBP)*: The movement trajectory of each individual in the population is predicted.
2. *Population-based prediction (POBP)*: One single prediction model is established for the trajectory of the entire population.
3. *Multidirectional prediction (MULP)*: The population is divided into several subpopulations by clustering method, and several prediction models are established for the subpopulations.

In INBP, Zhou et al. [21] proposed an individual-based prediction method, which used individual historical information to predict the possible positions of each individual. By contrast, a population-based prediction method, named POBP, was proposed in [22], where only one single prediction model was employed for predicting the whole population. Inspired by the work in [22], some population-based prediction methods have been proposed. For example, Jiang et al. [23] presented a steady-state and generational evolutionary algorithm, which predicted the evolution direction by the center points of two adjacent moments. Peng et al. [24] introduced a method based on central point prediction combined with memory strategy for dynamic multiobjective optimization. Wu et al. [25] implemented a directed search strategy based on the center points of Pareto set. This strategy used the center point and linear time series model to predict the population movement direction and its vertical direction, and then obtained the predicted new position. In MULP, Rong et al. [26] proposed a dynamic multiobjective evolutionary algorithms based on multidirectional prediction method, which established several prediction models for the subpopulations.

Motivation

Although many transfer strategies such as adaptive knowledge transfer strategy [9–12], search space mapping strategy [13–17], and valuable knowledge selection strategy [18–20] have been applied to MTO, there is no literature that attempts to transfer solutions to promising regions through the prediction method. The core of MTO is to improve the convergence speed and the quality of the solutions through the transferred solutions between tasks. In other words, the transferred solutions are used to guide the evolution of the current task. Therefore, transferring the solutions to promising regions can improve the convergence speed and the quality of the solutions.

As an illustrative example, Fig. 1 shows the results of transferred solutions with both the general transfer strategy

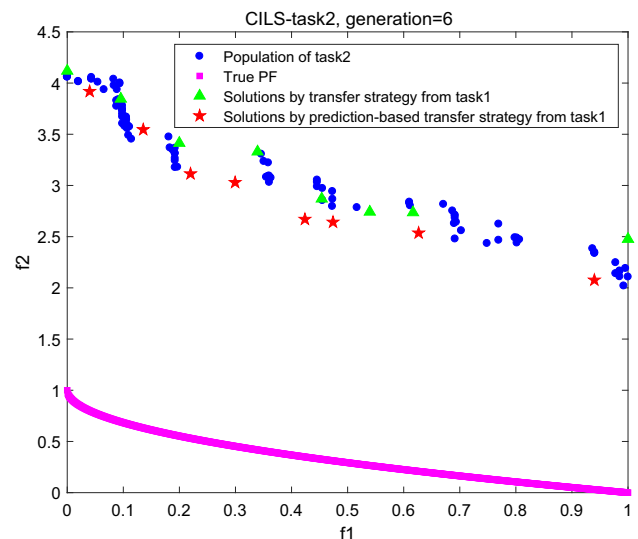


Fig. 1 The transferred solutions obtained by transfer strategy and prediction-based transfer strategy on CILS-task2

and the prediction-based transfer strategy. In Fig. 1, the pink rectangle represents the true PF of the task, the blue circle represents the current population, the green triangle represents the transferred solutions obtained by the general transfer strategy, and the red pentacle represents the transferred solutions obtained by the prediction-based transfer strategy. It can be observed that the transferred solutions obtained by the prediction-based transfer strategy are closer to the true PF, which indicates that the prediction method can transfer the solutions to more promising regions. Therefore, it is a very significant research problem to transfer the solutions to the promising region using the prediction method. In this paper, we are the first attempt to design an efficient knowledge transfer strategy based on multidirectional prediction method.

Proposed algorithm

This section presents the transfer strategy based on multidirectional prediction method. Moreover, the main framework of EMT-MPM is given. Finally, the complexity of EMT-MPM is analyzed.

Transfer strategy based on multidirectional prediction method

The transfer strategy based on multidirectional prediction method is proposed. First, the adaptive clustering method is used to divide the population into M classes. The distribution mean of each class is used as the representative point, where the number of representative points is M . Second, accord-

ing to two successive generations of representative points, the evolutionary direction of each class is generated by constructing a natural gradient. Thirdly, the mutation strength of each class is calculated based on the relative improvement of two successive generations. Finally, the transferred solutions are divided into the nearest class, and the predictive transferred solutions are generated by the prediction directions and mutation strengths.

Generation representative points

The binary clustering method [26] and weighted sum method are introduced to generate representative points. First, according to binary clustering method, the center point of the population and all m extreme solutions¹ of the PF are selected and stored in the class center set C . The center point of the g generation population P_g is calculated as follows:

$$\bar{x}^g = \frac{1}{N} \sum_{i=1}^N x_i^g, \quad (2)$$

where N is the population size; x_i^g is the i th solution of the g generation population. The m th extreme solution p_m is

$$p_m = \arg \min_{x_i^g \in P_g} f_m(x_i^g), \quad (3)$$

where f_m is the m th objective function of the task; P_g is the g generation population. Second, the center point of the population and m extreme solutions are used as the initial class centers, and the Euclidean distances between the solutions of the population and $m + 1$ class centers are calculated. If $M > m + 1$, the farthest point in each class from the class center is the candidate class center, and this distance is defined as the class radius. Moreover, the class with the largest radius is selected, and the candidate class center of this class is used as the new center point. The process is repeated until the population is divided into M classes. Finally, the distribution mean of each class is used as the representative point by the weighted sum method and is calculated as follows:

$$\bar{x}_i^g = \sum_{j=1}^{N_i^g} w_j^g x_j^g, \quad (4)$$

where N_i^g is the number of the solutions in generation g of i -th class; x_j^g is the j -th solution in i th class; w_j^g is a weight

and can be defined as follows:

$$w_j^g = \frac{\ln(N_i^g + \frac{1}{2}) - \ln R_j^g}{\sum_{k=1}^{N_i^g} (\ln(N_i^g + \frac{1}{2}) - \ln R_k^g)}, \quad 1 \leq j \leq N_i^g, \quad (5)$$

where R_j^g is the ranking of solution x_j^g in i th class, and the calculation method is

$$R_j^g = N_i^g - j + 1, \quad (6)$$

where j is the number of layers for solution x_j^g in the nondominated sorting of the i th class. For instance, for a nondominated solution x_j^g , j is equal to 1. According to Eq. (5), a better solution can get more weight, and the method of weight calculation emphasizes the top ranked solutions. In addition, it is clear that there is $\sum_{j=1}^{N_i^g} w_j^g = 1$. Specifically, the corresponding algorithm flow is shown in Algorithm 1.

Algorithm 1 The generation of representative points

Input: g generation population $P_g = \{x_1^g, x_2^g, \dots, x_N^g\}$, number of representative

points M , number of objective function m .

Output: M representative points $\bar{x}_1^g, \bar{x}_2^g, \dots, \bar{x}_M^g$.

- 1: Calculate the center point of P_g and m extreme solutions as the initial class
 - 2: centers by 2 and 3, and store them in the class center set C ;
 - 3: **While** ($|C| \leq M$) **do**
 - 4: Calculate the Euclidean distance between a solution x_i^g in P_g and each of
 - 5: the class centers in C , and take x_i^g into its corresponding class L_j , $j = 1,$
 - 6: $2, \dots, |L|$ is the label of a class;
 - 7: **For** $j = 1$ to $|L|$ **do**
 - 8: Find the solution with the furthest distance from the class center of L_j ,
 - 9: denoted as h_j ;
 - 10: **End For**
 - 11: Select the solution with the farthest distance from all h_j , denoted as the
 - 12: new class center, and store it in C ;
 - 13: **If** $|C| = M$ **then**
 - 14: Calculate the distribution mean of each class L_j by (4), and obtain M
 - 15: representative points $\bar{x}_1^g, \bar{x}_2^g, \dots, \bar{x}_M^g$;
 - 16: **End If**
 - 17: **End While**
-

Figure 2 depicts the process of selecting four class centers from fifteen solutions. First, the center point of the population and two extreme solutions are obtained by Eqs. 2 and 3, respectively. Second, the other solutions in the population are assigned to three classes, Class 1, Class 2, and Class 3 according to the Euclidean distance from the class center points A, B, and \bar{x}^g . Because points C, D, and E are the solutions farthest from the class center among the three classes, respectively, and the distance from C to the class center A is

¹ Extreme solutions refer to the solutions with minimum value on an objective function, and they can make a good description of the location of the PF. Moreover, the definition of extreme solution is given in Eq. (3).

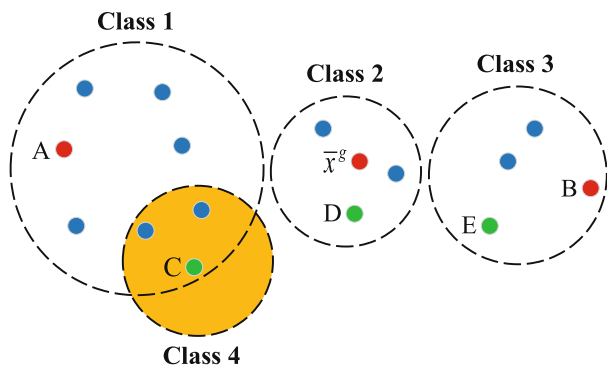


Fig. 2 Schematic diagram of class centers selection process

farther than that from D to \bar{x}^g and E to B. Therefore, C is chosen as the new class center point. Thirdly, the solutions in Class 1 are divided into two classes based on the distance from A and C. Finally, four class centers are selected, and the population is divided into four classes.

Generation prediction directions

A multidirectional prediction method is proposed to predict the corresponding evolutionary direction of each class. Suppose $\bar{X}^{g-1} = \{\bar{x}_1^{g-1}, \bar{x}_2^{g-1}, \dots, \bar{x}_M^{g-1}\}$ and $\bar{X}^g = \{\bar{x}_1^g, \bar{x}_2^g, \dots, \bar{x}_M^g\}$ are class representative points of generation $g - 1$ and generation g , respectively. First, for any representative point \bar{x}_i^g in \bar{X}^g , the nearest representative point \bar{x}_i^{g-1} in \bar{X}^{g-1} is selected as the parent of \bar{x}_i^g . Second, the natural gradient² Δx_i^g in i th class is constructed and is expressed as

$$\Delta x_i^g = \bar{x}_i^g - \bar{x}_i^{g-1}, \tag{7}$$

where \bar{x}_i^{g-1} and \bar{x}_i^g are the representative points of generation $g - 1$ and generation g in i -th class. Finally, the prediction direction $\hat{\Delta} x_i^g$ is defined as follows:

$$\hat{\Delta} x_i^g = (1 - c_p)\hat{\Delta} x_i^{g-1} + c_p\Delta x_i^g \tag{8}$$

where c_p is a parameter that controls the weight of the historical prediction direction; $\hat{\Delta} x_i^{g-1}$ is the prediction direction of generation $g - 1$ in i th class; Δx_i^g is the natural gradient of generation g in i th class. Clearly, $\hat{\Delta} x_i^g$ is a weighted combination of the historical prediction direction $\hat{\Delta} x_i^{g-1}$ and the current natural gradient Δx_i^g . Thus, $\hat{\Delta} x_i^g$ is a weighted combination of all the historical natural gradients and is similar to the evolution path in [29]. According to Eqs. (7) and (8),

² The natural gradient is proposed in the covariance matrix adaptation evolution strategy [27,28], and is accumulated to form the evolution path as the momentum term.

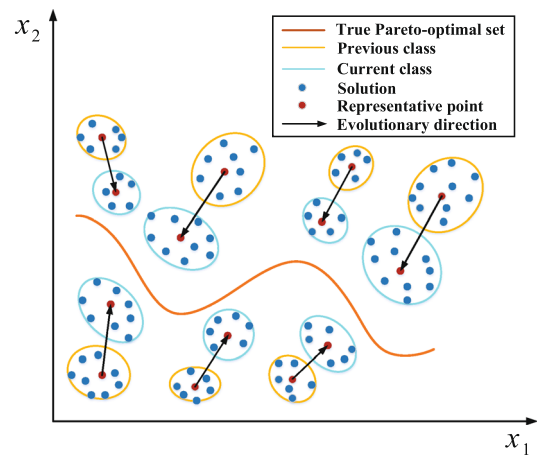


Fig. 3 Example for multidirectional prediction method

the calculation of the prediction direction for each representative point is independent of each other, and its value can be determined by the corresponding representative point and its historical representative point. Moreover, these prediction directions are complementary to each other. By predicting the evolutionary direction of each representative point, multiple prediction directions can work together to more accurately predict the promising directions.

Figure 3 is an example to explain multidirectional prediction method. First, the population is divided into seven classes by the adaptive clustering method, and seven representative points are generated. Second, the evolutionary direction are generated according to current and historical representation points. Finally, the promising directions are predicted by seven evolutionary directions.

Calculation mutation strengths

The mutation strength is calculated to control the step size of transferred solutions. First, the concept of hypervolume (HV) [30] is introduced to compare the solution of multiobjective optimization. As shown in Fig. 4, this is an example to illustrate the concept of HV. The red dot represents the reference point. The green, blue, and yellow matrices represent the HV of points A, B, and C, respectively. Specifically, the larger HV indicates the better quality of the solution. Second, the relative improvement $\Delta_{improv}^{i,g}$ of generation g in i -th class is defined as follows:

$$\Delta_{improv}^{i,g} = \frac{\sum_{i=1}^{N_i^g} HV(x_i^g) - \sum_{i=1}^{N_i^{g-1}} HV(x_i^{g-1})}{\sum_{i=1}^{N_i^{g-1}} HV(x_i^{g-1})}, \tag{9}$$

where N_i^g and N_i^{g-1} represent the number of the solutions in generation g and generation $g - 1$ of i -th class, respectively; $HV(x_i^g)$ and $HV(x_i^{g-1})$ denote the HV of solutions

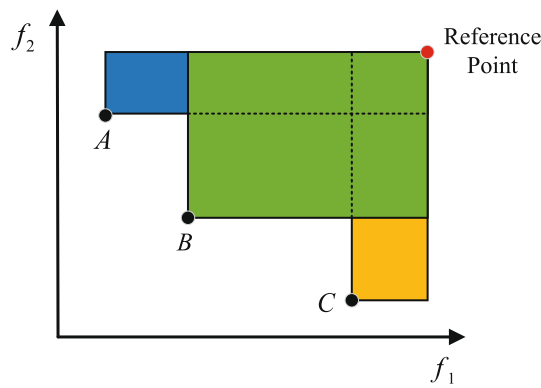


Fig. 4 Example for understanding the hypervolume

\mathbf{x}_i^g and \mathbf{x}_i^{g-1} in the objective space, respectively. Finally, the mutation strength σ_i^g is calculated

$$\sigma_i^g = \sigma_i^{g-1} \exp \left[\frac{1}{d_\sigma} \left(\frac{\Delta_{\text{improv}}^{i,g}}{\Delta_{\text{improv}}^{i,g-1}} - 1 \right) \right], \quad (10)$$

where d_σ is a damping factor, and $d_\sigma \geq 1$; $\Delta_{\text{improv}}^{i,g}$ and $\Delta_{\text{improv}}^{i,g-1}$ are the relative improvements of generation g and generation $g-1$ in i -th class; for initial mutation strength, $\sigma_i^0 = 1$. If the relative improvement in i -th class is greater, the greater mutation strength can be obtained. Moreover, the mutation strength of each class is independent of each other and can be adaptively controlled.

Selection strategy of transferred solutions

Since the valuable knowledge selection strategy [19] has an excellent performance in multiobjective EMT algorithms, the transferred strategy uses it to generate transferred solutions. Specifically, transferred solutions are randomly selected from each task in the first generation. When the generation $g > 1$, transferred solutions are selected according to the solutions which achieved the positive transfer in generation $g-1$. A transferred solution achieves positive transfer if it is nondominated in its target task. Afterwards, in the original task of this positive transferred solution, its several closest solutions will be the transferred solutions since these solutions are more likely to achieve positive transfer. Therefore, the transferred solutions are selected near the positive transferred solutions, and these solutions are more valuable.

The details of the selection strategy of transferred solutions are given in Algorithm 2. The transferred solutions are selected according to the solutions which achieved the positive transfer in generation $g > 1$. If there is no positive transferred solution in $g-1$ generation, the nondominated solutions (line 2) are selected as the transferred solutions in g generation. Otherwise, the transferred solutions (line 4) are

selected near the positive transferred solutions in g generation.

Algorithm 2 Selection strategy of transferred solutions

Input: g generation population $\mathbf{P}_g = \{\mathbf{x}_1^g, \mathbf{x}_2^g, \dots, \mathbf{x}_N^g\}$, number of positive

transferred solutions η in g generation, positive transferred solutions

$\{t_1^{g-1}, t_2^{g-1}, \dots, t_\eta^{g-1}\}$ in $g-1$ generation, population size N , number

of transferred solutions λ .

Output: transferred population $\mathbf{T}_g = \{t_1^g, t_2^g, \dots, t_\lambda^g\}$.

1: **If** $\eta = 0$ **then**

2: λ nondominated solutions in the population \mathbf{P}_g is selected as transferred

3: population $\mathbf{T}_g = \{t_1^g, t_2^g, \dots, t_\lambda^g\}$;

4: **Else**

5: Calculate the Euclidean distances between N solutions in population \mathbf{P}_g

6: and η positive transferred solutions in $g-1$ generation, and λ solutions

7: with the smallest distance in population \mathbf{P}_g are selected as transferred

8: population $\mathbf{T}_g = \{t_1^g, t_2^g, \dots, t_\lambda^g\}$;

9: **End If**

Generation predictive transferred solutions

The predictive transferred solutions in each class are generated based on the prediction directions and mutation strengths. Meanwhile, to improve the performance, a Gaussian perturbation related to the mutation strength is introduced. To be specific, the predictive transferred solution \hat{t}_j^g in i th class is calculated as follows:

$$\hat{t}_j^g = t_j^g + \sigma_i^g \hat{\Delta} \mathbf{x}_i^g + \boldsymbol{\gamma}^g, \quad (11)$$

where t_j^g is the i -th transferred solutions in the g generation; σ_i^g and $\hat{\Delta} \mathbf{x}_i^g$ are the prediction direction and mutation strength in the g generation of i -th class, respectively; $\boldsymbol{\gamma}^g$ is a Gaussian perturbation, and $\boldsymbol{\gamma}^g \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}^g)$. The calculation method of $\boldsymbol{\sigma}^g$ is

$$\boldsymbol{\sigma}^g = \frac{1}{M} \sum_{i=1}^M \left| \hat{\Delta} \mathbf{x}_i^g \right|, \quad (12)$$

where M is the number of classes; $\hat{\Delta} \mathbf{x}_i^g$ is the mutation strength in the g generation of i -th class.

Algorithm 3 presents the framework of transfer strategy based on multidirectional prediction method. First, two successive generations of representative points are calculated using adaptive clustering method (lines 1–2). Second, the prediction direction and mutation strength of each class are

Algorithm 3 The transfer strategy based on multidirectional prediction method

Input: $g - 1$ generation population $P_{g-1} = \{x_1^{g-1}, x_2^{g-1}, \dots, x_N^{g-1}\}$ and corresponding function values $\{F(x_1^{g-1}), F(x_2^{g-1}), \dots, F(x_N^{g-1})\}$, g generation population $P_g = \{x_1^g, x_2^g, \dots, x_N^g\}$ and corresponding function values $\{F(x_1^{g-1}), F(x_2^{g-1}), \dots, F(x_N^{g-1})\}$, transferred population $T_g = \{t_1^g, t_2^g, \dots, t_\lambda^g\}$, size of population N , number of transferred solutions λ , number of classes M .

Output: predictive transferred population $\hat{T}_g = \{\hat{t}_1^g, \hat{t}_2^g, \dots, \hat{t}_\lambda^g\}$.

- 1: Obtain M class centers in the $g - 1$ generation and g generation, and the populations P_{g-1} and P_g are divided into M classes by (2) and (3);
- 2: Calculate M class representative points $\bar{X}^{g-1} = \{\bar{x}_1^{g-1}, \bar{x}_2^{g-1}, \dots, \bar{x}_M^{g-1}\}$ and $\bar{X}^g = \{\bar{x}_1^g, \bar{x}_2^g, \dots, \bar{x}_M^g\}$ in the $g - 1$ generation and g generation by (4);
- 3: **For** $i = 1$ to M **do**
- 4: Calculate the prediction direction $\hat{\Delta} x_i^g$ of the i th class by (8);
- 5: Calculate the mutation strength σ_i^g of the i th class by (10);
- 6: **End For**
- 7: Obtain the prediction directions $\{\hat{\Delta} x_1^g, \hat{\Delta} x_2^g, \dots, \hat{\Delta} x_M^g\}$ and mutation strengths $\{\sigma_1^g, \sigma_2^g, \dots, \sigma_M^g\}$;
- 8: **For** $j = 1$ to λ **do**
- 9: Compute the j th predictive transferred solution \hat{t}_j^g by (11);
- 10: **End For**
- 11: Obtain λ predictive transferred solutions $\{\hat{t}_1^g, \hat{t}_2^g, \dots, \hat{t}_\lambda^g\}$ to form the predictive transferred population \hat{T}_g .

calculated (lines 4–5). Finally, the predictive transferred solutions are obtained by combining the prediction directions with mutation strengths (line 9).

Basic workflow of the proposed method

The basic workflow of transfer strategy based on multidirectional prediction method is shown in Fig. 5. In this paper, multidirectional prediction method is combined with the selection strategy of transferred solutions to generate the predictive transferred solutions as useful information to transfer from one task to another task. Multidirectional prediction method can make use of the evolutionary directions of the population. The transferred solutions are valuable solutions selected by selection strategy. Therefore, the predictive transferred solutions make use of both the information of evolutionary directions and transferred solutions, and they are used as useful information. The generation process of predictive transferred solutions is as follows. First, the multidirectional prediction method uses the binary clustering method and weighted sum method to generate representative points. Then the prediction directions and mutation strengths are generated according to the representative points. Second, a selection strategy of transferred solutions is utilized

to obtain the transferred solutions. Thirdly, the predictive transferred solutions are calculated by transferred solutions, prediction directions, mutation strengths, and Gaussian perturbation. Finally, the predictive transferred solutions are transferred between tasks as transfer knowledge.

Main framework

The main framework of EMT-MPM is summarized in Algorithm 4. First, the initial population of the K tasks is generated in the unified search space [7] for effectively transferring knowledge across tasks. The unified space is $[0, 1]^D$, where $D = \max\{D_i\}$ and D_i is the decision space dimensionality of T_i . Then the initial populations of the K tasks are generated, and each solution in the K populations is evaluated by the corresponding optimization task (lines 1-2). Second, the predictive transferred solutions of the K tasks are selected through the transfer strategy based on multidirectional prediction method (line 5). Suppose L_k and U_k are the lower and upper boundaries of T_k , respectively. The formula of transferring a solution \hat{t}_j^g from T_i to T_k , denoted as \hat{t}'_j^g , is shown as follows:

$$\hat{t}'_j^g = \frac{\hat{t}_j^g - L_i}{U_i - L_i}(U_k - L_k) + L_k, \tag{13}$$

where L_i and U_i are the lower and upper boundaries of T_i , respectively. When $D_k \neq D_i$, this formula can be modified according to the strategy proposed in [15]. To further improve the performance of knowledge transfer, the predictive transferred solution \hat{t}'_j^g is multiplied by a distributed factor ξ with a probability p to obtain the new predictive transferred solution \hat{t}''_j^g (lines 7–9) and is defined as follows:

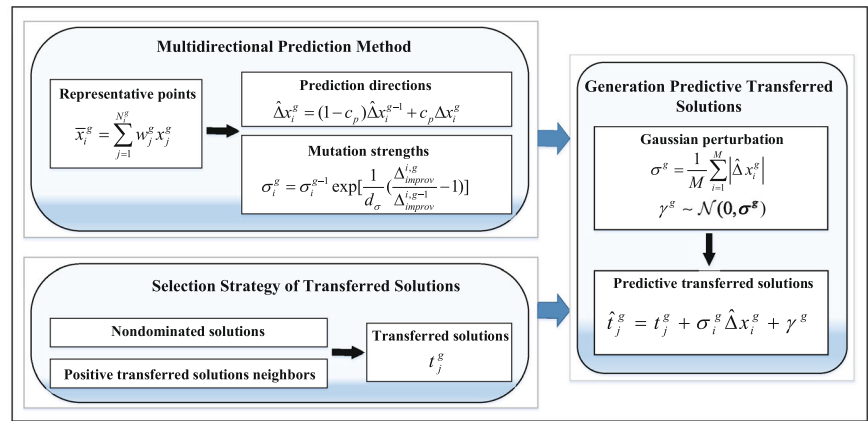
$$\hat{t}''_j^g = \xi \hat{t}'_j^g, \tag{14}$$

where ξ is a distributed factor, and $\xi \sim U(0, 2)$. By the distributed factor, the new predictive transferred solution can explore a large range and find the promising regions. Finally, the environmental selection is implemented on population $P_i^g \cup \hat{T}''_g \cup O_i^g$ to select N best solutions, and the new populations P_i^{g+1} are generated (line 14).

Complexity analysis

For the convenience of the analysis, it is assumed that the number of tasks in Algorithm 4 is set to K . The population size for each task is set to N . The number of decision variables and objectives are set to n and m , respectively. λ is the number of transferred solutions. M is the number of representative points. From lines 1–2 in Algorithm 4, the population initialization and the evaluation of solutions in K tasks are

Fig. 5 The workflow of transfer strategy based on multidirectional prediction method



Algorithm 4 Framework of EMT-MPM

Input: K multiobjective optimization tasks $T_1, T_2, \dots, T_i, \dots, T_K$, population

size N , number of transferred solutions λ .

Output: nondominated solutions of K optimization tasks.

- 1: Initialize K populations $P_1^0, P_2^0, \dots, P_i^0, \dots, P_K^0$;
- 2: Evaluate each solution in $P_1^0, P_2^0, \dots, P_i^0, \dots, P_K^0$;
- 3: **While** (termination criterion is not met) **do**
- 4: **For** $i = 1$ to K **do**
- 5: Obtain predictive transferred population $\hat{T}_g = \{\hat{t}_1^g, \hat{t}_2^g, \dots, \hat{t}_\lambda^g\}$ via
- 6: **Algorithm 4**;
- 7: **For** $j = 1$ to λ **do**
- 8: **If** $\text{rand}(0, 1) \leq p$ **then**
- 9: Generate new predictive transferred solution $t_j^{n,g}$ by (13) and (14);
- 10: **End If**
- 11: **End For**
- 12: Obtain new predictive transferred population $\hat{T}_g'' = \{t_1^{n,g}, t_2^{n,g}, \dots, t_\lambda^{n,g}\}$;
- 13: Generate offsprings O_i^g by P_i^g and \hat{T}_g'' ;
- 14: Evaluate population $P_i^g \cup \hat{T}_g'' \cup O_i^g$;
- 15: Select N fittest solutions from $P_i^g \cup \hat{T}_g'' \cup O_i^g$ to generate next generation
- 16: population P_i^{g+1} ;
- 17: **End For**
- 18: **End While**

performed, and the computational complexity of this process is $O(KN^2m)$. In line 5 of Algorithm 4, the predictive transferred populations of K tasks are obtained by Algorithm 3. First, M representative points in K tasks are generated, and its computational complexity is $O(KNM)$. Second, the computational complexity of M prediction directions in K tasks is $O(KNM)$. Thirdly, the computational complexity of the mutation strengths and the predictive transferred solutions are $O(KNm^2)$ and $O(K\lambda)$, respectively. From lines 12–14, the algorithm mainly performs nondominated sorting and environment selection, and its computational complexity is $O(KN^2m)$. In summary, the worst computational complex-

ity of the proposed algorithm is $O(KN^2m)$ or $O(KNM)$ or $O(KNm^2)$, whichever is larger.

Remarks

For the transfer strategy based on multidirectional prediction method, four comments are made.

1. Compared with the multidirectional prediction method in dynamic multiobjective optimization [26], the proposed multidirectional prediction method has the following four differences. First, the proposed method introduces the weighted sum method to calculate the representative points, and these representative points have better objective function values. This is because the solution with the better objective function value has greater weight. Second, when calculating the prediction directions, the proposed method makes use of representative points of two successive generations and historical prediction direction at the same time. In comparison with the multidirectional prediction method in dynamic multiobjective optimization, the proposed method can obtain the promising prediction directions. Thirdly, the multidirectional prediction method in dynamic multiobjective optimization uses a fixed mutation strength. By contrast, the proposed method can adaptively control the mutation strength according to the improvement degree of each class. Fourth, the predictive transferred solutions generated by the proposed method are to help the target task as transfer knowledge. However, the prediction solutions generated by the multidirectional prediction method in dynamic multiobjective optimization are used as a new initial population.
2. The binary clustering method is used to generate M class centers. Compared with the traditional K -means clustering method, the binary clustering method only needs to reclassify a corresponding class after getting the new class center, but does not need to classify all the individuals. This greatly reduces the computational complexity in the

process of generating the class center point. Moreover, when generating the prediction direction, the weight of the current and historical directions can be flexibly controlled by the parameter c_p .

3. HV indicator is introduced into MTO to define the improvement degree of solution. Then the algorithm adaptively controls the mutation strength by the improvement degree of each class. Moreover, different mutation strengths are set for different classes, which ensures that each class can flexibly control according to its own evolutionary state.
4. Each class generates the predictive transferred solutions by the prediction directions, mutation strengths, and Gaussian perturbation. Then the predictive transferred solutions generated in each class can transfer to promising regions. In addition, Gaussian perturbation mainly guarantees the validity and diversity of the predictive transferred solutions.

Experimental studies

In this section, NSGA-II [31], MO-MFEA [7], EMEA [18], MOMFEA-SADE [16], EMT/ET [19], MO-MFEA-II [9], TMO-MFEA [10], EMTIL [20], and EMT-PD [32] are used to compare with EMT-MPM on two MTO test suits. Moreover, the performance of EMT-MPM is analyzed in depth. Finally, the following points will be demonstrated.

1. EMT-MPM can significantly outperform NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, MO-MFEA-II, TMO-MFEA, EMTIL, and EMT-PD.
2. The transfer strategy based on multidirectional prediction method in EMT-MPM can transfer the solutions to the promising regions and significantly improve the performance of the algorithm.

Test problems and performance metric

The two test suits are used to demonstrate the performance of EMT-MPM. The CEC 2017 MTO benchmarks [33] are used as the first test suit which contains nine MTO problems. The second test suit named CPLX contains ten complex benchmark problems, which is presented for WCCI 2020 Competition on Evolutionary Multitasking Optimization [34].

In this paper, the inverted generational distance (IGD) [35] is used to evaluate the performance of all comparison algorithms. IGD measures the average of the minimum distance between the solutions of the real PF and the approximate solution set obtained by the algorithm. Therefore, calculating IGD requires obtaining a set of solutions that are uniformly sampled on the real PF in advance. Let P^* be a set of solu-

tion sets uniformly sampled on the real PF, and S is the PF approximate solution set obtained by the comparison algorithms. The IGD is calculated as follows:

$$IGD(S, P^*) = \frac{\sum_{x \in P^*} \text{dist}(x, S)}{|P^*|}, \quad (15)$$

where $\text{dist}(x, S)$ is the Euclidean distance between solution $x \in P^*$ and the nearest solution on S ; $|P^*|$ is the cardinality of the set P^* . The smaller the IGD value, the better the convergence and diversity of S .

Parameter settings

The parameter settings of all algorithms are listed below.

1. *Population Size*: MO-MFEA, MOMFEA-SADE, MO-MFEA-II, TMO-MFEA, and EMT-PD are set to 200, and each population for EMEA, EMT/ET, EMT-MPM, and EMTIL is 100. To be fair, the population size of NSGA-II is 100.
2. *Maximum number of function evaluations*: NSGA-II is set to 50000, and all other algorithms are set to 100000.
3. *Independent number of runs*: All algorithms are set to 20.
4. *Random mating probability (rmp)*: MO-MFEA, MOMFEA-SADE, and EMT-PD are set to 0.3. MO-MFEA-II is set as either 0.9, 0.6, or 0.3. TMO-MFEA is set to 1 and 0.3 for diversity variable and convergence variable, respectively.
5. *Operator of generating offspring*: For NSGA-II, MO-MFEA, EMEA, EMT/ET, MO-MFEA-II, TMO-MFEA, EMTIL, EMT-PD, and EMT-MPM, simulated binary crossover (SBX) [36] and polynomial mutation (PM) [37] are adopted. The parameter settings of SBX and PM for the seven algorithms are $p_c = 0.9$, $p_m = 1/D_{\max}$, $\eta_c = 20$, and $\eta_m = 15$, where D_{\max} is the largest dimension of all tasks. For MOMFEA-SADE, DE operators are used. The crossover rate CR is set to 0.6, and the scale factor F takes 0.5.
6. *Number (λ) of transferred solutions*: For EMEA, EMT/ET, EMTIL, and EMT-MPM, λ is set to 10, 8, 10, and 8, respectively.
7. *Other parameters*: $M = 6$, $c_p = 0.7$, $d_\sigma = 2$, and $p = 0.5$, since they can obtain the best performance on the experiments.

Results and statistical analysis

Comparison of NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, MO-MFEA-II, TMO-MFEA, EMTIL, EMT-PD, and EMT-MPM on the first test suit

The first test suit is utilized to verify the performance of EMT-MPM. The experimental results obtained by all algo-

Table 1 Average value and standard deviation of IGD for NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, and EMT-MPM on the first test suit

Problems	NSGA-II	MO-MFEA	EMEA	MOMFEA-SADE	EMT/ET	EMT-MPM
CIHS-T1	5.20E-01(4.25E-01)+	6.73E-02(1.22E-02)+	1.19E-03(1.22E-04)+	1.03E-03(6.83E-05)+	2.16E-04(1.42E-05)+	1.62E-04(1.27E-05)
CIHS-T2	3.19E-01(2.92E-02)+	2.14E-01(1.82E-02)+	1.52E-03(6.39E-04)+	8.21E-03(6.38E-04)+	2.00E-04(1.55E-04)+	1.83E-04(1.21E-04)
CIMS-T1	2.36E+00(2.69E+00)+	1.60E+00(2.23E+00)+	5.03E+00(1.46E+00)+	4.42E-04(8.97E-05)+	2.28E-04(1.56E-04)+	1.65E-04(1.07E-04)
CIMS-T2	1.31E+00(1.21E+00)+	2.96E-01(3.29E-01)+	3.42E-01(2.32E-01)+	1.90E-03(2.22E-04)+	4.13E-04(2.60E-04)+	1.93E-04(5.82E-05)
CILS-T1	1.65E+01(4.28E+00)+	5.01E-02(1.25E-02)+	1.38E-03(1.46E-04)+	2.00E-03(2.01E-04)+	1.93E-04(8.46E-06)	1.97E-04(9.01E-06)
CILS-T2	6.62E-03(9.46E-04)+	3.33E-03(3.50E-04)+	1.82E-03(2.03E-04)+	8.21E-04(6.19E-05)+	1.94E-04(1.28E-05)+	1.62E-04(8.93E-06)
PIHS-T1	2.08E-01(3.16E-02)+	1.31E+00(3.84E-01)+	2.04E-02(3.59E-03)+	1.66E-03(3.86E-04)+	1.86E-04(7.41E-06)+	1.64E-04(6.31E-06)
PIHS-T2	1.50E+01(5.08E+00)+	3.20E+01(4.93E+00)+	5.77E-01(2.54E-01)+	6.72E-01(4.36E-04)+	3.31E-04(6.07E-04)+	3.18E-04(5.38E-04)
PIMS-T1	3.17E-01(1.30E-01)+	1.17E-01(5.97E-02)+	5.83E-02(1.76E-02)+	2.43E-01(6.78E-01)+	4.32E-04(5.70E-04)+	2.57E-04(2.73E-04)
PIMS-T2	5.02E+02(1.27E+01)+	4.71E+02(9.21E+01)+	5.66E+01(6.49E+00)+	1.84E+01(2.70E+01)+	2.05E-04(1.55E-05)+	1.92E-04(1.28E-05)
PILS-T1	1.00E-02(3.31E-03)+	2.38E-02(6.31E-03)+	1.13E-02(3.85E-03)+	3.92E-03(3.50E-03)+	1.84E-04(8.34E-06)+	1.76E-04(7.91E-06)
PILS-T2	2.20E+01(8.74E-02)+	2.52E+00(2.43E-01)+	1.80E+00(1.51E-02)+	6.40E+00(4.04E+00)+	1.38E-03(2.35E-03)+	8.63E-04(1.05E-03)
NIHS-T1	3.65E+02(5.01E+01)+	8.14E+01(6.14E+01)+	4.73E+01(1.71E-01)+	4.66E+01(4.04E-01)+	1.48E+00(8.56E-03)+	1.47E+00(7.54E-03)
NIHS-T2	1.38E-01(2.63E-02)+	8.55E-02(2.48E-02)+	1.52E-03(5.08E-04)+	1.49E-03(3.10E-04)+	1.88E-04(8.08E-06)+	1.64E-04(5.38E-06)
NIMS-T1	1.74E+01(1.12E+01)+	3.79E+01(3.21E+01)+	1.64E+01(1.36E-01)+	1.85E+01(1.95E+01)+	1.55E-01(4.64E-03)+	1.49E-01(4.31E-03)
NIMS-T2	5.94E+00(5.36E+00)+	4.05E+00(6.21E+00)+	5.22E-04(1.36E-04)+	5.03E-02(7.01E-02)+	1.60E-04(8.95E-06)+	1.53E-04(8.72E-06)
NILS-T1	2.01E-02(5.04E-03)+	3.07E-02(8.33E-03)+	3.37E-02(9.04E-03)+	2.14E-02(6.90E-03)+	7.04E-04(2.62E-05)	7.52E-04(2.81E-05)
NILS-T2	2.04E+01(3.12E-02)+	2.05E+01(8.64E-02)+	2.00E+01(8.93E+00)+	1.99E+01(7.61E-05)+	4.88E-04(1.51E-03)+	1.83E-04(6.82E-04)
+	18	18	18	18	16	/
≈	0	0	0	0	0	/
-	0	0	0	0	2	/

Table 2 Average value and standard deviation of IGD for MO-MFEA-II, TMO-MFEA, EMTIL, EMT-PD, and EMT-MPM on the first test suit

Problems	MO-MFEA-II	TMO-MFEA	EMTIL	EMT-PD	EMT-MPM
CIHS-T1	7.30E-02(1.60E-02)+	8.06E-03(4.21E-05)+	2.10E-02(5.14E-03)+	9.87E-04(1.62E-03)+	1.62E-04(1.27E-05)
CIHS-T2	2.40E-01(4.68E-02)+	9.19E-02(1.84E-02)+	1.10E-01(1.77E-02)+	2.18E-02(5.83E-03)+	1.83E-04(1.21E-04)
CIMS-T1	2.15E+00(2.31E+00)+	4.18E+00(3.50E+02)+	1.38E+00(2.08E+00)+	6.93E+00(4.93E+00)+	1.65E-04(1.07E-04)
CIMS-T2	2.80E-01(2.34E-01)+	5.12E-03(5.50E-02)+	2.84E-01(2.49E-01)+	2.86E-03(3.89E-03)+	1.93E-04(5.82E-05)
CILS-T1	6.61E-02(1.87E-02)+	2.57E-02(5.70E-02)+	8.07E-03(1.31E-03)+	3.05E-03(7.31E-04)+	1.97E-04(9.01E-06)
CILS-T2	3.12E-03(3.96E-04)+	2.35E-03(2.59E-04)+	4.17E-03(3.73E-04)+	9.71E-04(2.13E-04)+	1.62E-04(8.93E-06)
PIHS-T1	1.90E-01(5.17E-02)+	3.54E-03(6.03E-04)+	2.73E-02(2.44E-02)+	1.52E-01(3.05E-02)+	1.64E-04(6.31E-06)
PIHS-T2	1.35E+01(1.95E+00)+	1.85E-01(6.32E-02)+	2.96E+00(9.72E-01)+	1.34E+01(1.27E+00)+	3.18E-04(5.38E-04)
PIMS-T1	8.08E-02(1.91E-02)+	1.26E+00(3.99E-02)+	5.13E-03(3.40E-04)+	1.36E-02(8.34E-03)+	2.57E-04(2.73E-04)
PIMS-T2	4.69E+02(1.06E+02)+	5.06E+02(3.24E+02)+	5.04E+01(2.69E+01)+	3.42E+01(1.06E+00)+	1.92E-04(1.28E-05)
PILS-T1	1.58E-02(6.12E-03)+	2.08E-02(4.29E-04)+	1.10E+00(4.44E-16)+	1.59E-02(3.76E-04)+	1.76E-04(7.91E-06)
PILS-T2	2.02E+01(5.07E-02)+	2.23E+00(2.34E-01)+	1.55E+01(8.19E+00)+	1.75E+01(7.42E+00)+	8.63E-04(1.05E-03)
NIHS-T1	6.67E+01(7.78E+00)+	4.64E+01(2.12E-01)+	3.56E+00(1.40E+00)+	5.62E-02(6.24E-04)-	1.47E+00(7.54E-03)
NIHS-T2	1.11E-01(3.17E-02)+	5.65E-02(3.46E+02)+	3.20E-02(9.36E-03)+	1.67E-03(7.35E-04)+	1.64E-04(5.38E-06)
NIMS-T1	6.58E+01(2.66E+01)+	1.75E+01(3.18E-01)+	4.06E+01(3.12E+01)+	7.34E+00(5.81E-01)+	1.49E-01(4.31E-03)
NIMS-T2	1.29E+00(1.70E+00)+	3.71E-03(3.30E-02)+	7.09E-01(8.59E-01)+	9.93E-04(2.74E-05)+	1.53E-04(8.72E-06)
NILS-T1	3.11E-02(1.59E-02)+	3.92E-02(1.08E-02)+	1.25E-01(3.67E-03)+	6.82E-03(1.83E-04)+	7.52E-04(2.81E-05)
NILS-T2	2.04E+01(6.60E-02)+	2.10E+01(1.43E+00)+	3.22E-01(7.06E-02)+	7.18E-02(4.05E-02)+	1.83E-04(6.82E-04)
+	18	18	18	17	/
≈	0	0	0	0	/
-	0	0	0	1	/

rithms over 20 runs are shown in Tables 1 and 2, respectively. The symbols “+”, “≈”, and “−” represent that EMT-MPM is better, similar or worse than the compared algorithms, respectively. In addition, the best result of every task is marked in bold.

Table 1 gives the average value and standard deviation of IGD for NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, and EMT-MPM on the first test suit. The performance of NSGA-II is significantly worse than the five multiobjective EMT algorithms, which shows the effectiveness of knowledge transfer between different tasks. Moreover, the performance of EMT-MPM is significantly superior to MO-MFEA, EMEA, MOMFEA-SADE, and EMT/ET, which benefits from the proposed multidirectional prediction method. In addition, Table 2 shows the average value and standard deviation of IGD for MO-MFEA-II, TMO-MFEA, EMTIL, EMT-PD, and EMT-MPM on the first test suit. As shown in Table 2, EMT-MPM surpasses MO-MFEA-II, TMO-MFEA, EMTIL, and EMT-PD on 18, 18, 18, and 17 tasks, respectively. This indicates that the proposed multidirectional prediction method can obtain more promising transferred solutions. Furthermore, this paper conducts Friedman’s test [38] on six compared algorithms based on the first test suit to IGD. Table 3 presents the test results that EMT-MPM has the best ranking to IGD, sequentially followed by EMT/ET, MOMFEA-SADE, EMT-PD, EMEA, EMTIL, TMO-MFEA, MO-MFEA-II, MO-MFEA, and NSGA-II.

In summary, the experimental results show that EMT-MPM has the best performance except for CILS-T1, NIHS-T1, and NILS-T1. This is because the transfer strategy based on multidirectional prediction method can transfer solutions to promising regions by prediction directions and mutation strengths. Moreover, the validity and diversity of predictive transferred solutions can be improved by Gaussian perturbation. Therefore, the predictive transferred solutions can guide the current task to accelerate evolution and improve the quality of the solutions.

Comparison of NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, and EMT-MPM on the second test suit

Table 4 shows the experiment results by six compared algorithms on the second test suit. As shown in Table 4, EMT-MPM surpasses NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, and EMT/ET on 20, 20, 18, 18, and 17 tasks, respectively. As a comparison, EMT-MPM is surpassed by NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, and EMT/ET on 0, 0, 2, 2, and 3 tasks, respectively. This indicates that EMT-MPM can achieve excellent performance on the second test suit. What’s more, Friedman’s test on six compared algorithms based on the second test suit to IGD is conducted. Table 5 shows that EMT-MPM obtains the best

Table 3 Results obtained for NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, MO-MFEA-II, TMO-MFEA, EMTIL, EMT-PD, and EMT-MPM by the Friedman’s test on the first test suit to IGD

Algorithms	Ranking
EMT-MPM	1.1667
EMT/ET	1.9444
MOMFEA-SADE	4.5000
EMT-PD	4.7778
EMEA	5.2222
EMTIL	6.5000
TMO-MFEA	6.6111
MO-MFEA-II	7.8056
MO-MFEA	7.9444
NSGA-II	8.5278

ranking, which indicates that the multidirectional prediction method is effective for MTO problems.

Analysis of the convergence trends of the algorithms

As shown in Fig. 6, the convergence trends of NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, and EMT-MPM on three representative problems of the first test suit is illustrated. The lines in Fig. 6 represent the average IGD values of the six compared algorithms. The convergence speed of NSGA-II is the slowest in most problems, which indicates that the transferred knowledge between tasks is useful. Moreover, the convergence speed of EMT-MPM is significantly faster than the other five compared algorithms, which illustrates that the proposed transfer strategy based on multidirectional prediction method can significantly improve the performance of the algorithm. This phenomenon also shows that EMT-MPM can transfer more valuable knowledge. In conclusion, the performance of EMT-MPM is significantly better than other algorithms. Therefore, the proposed multidirectional prediction method is an efficient method for MTO.

Effectiveness of the proposed method

The proposed transfer strategy based on multidirectional prediction method mainly includes prediction direction, mutation strength, and Gaussian perturbation. First, to verify the effectiveness of the prediction direction, only the natural gradient is used as the prediction direction. This algorithm is constructed, namely EMT-NG. Second, the mutation strength is set as 1 to verify its effectiveness, and the corresponding algorithm is defined as EMT-NMS. Thirdly, the Gaussian perturbation in the EMT-MPM algorithm is removed, which is defined as EMT-NGP. Finally, EMT-NG, EMT-NMS, EMT-NGP, and EMT-MPM are run 20 times on the first test suit.

Table 6 gives the average value and standard deviation of IGD values obtain by four multiobjective EMT algorithms.

Table 4 Average value and standard deviation of IGD for NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, and EMT-MPM on the second test suit

Problems	NSGA-II	MO-MFEA	EMEA	MOMFEA-SADE	EMT/ET	EMT-MPM
CPX1-T1	7.19E-03(4.88E-05)+	7.54E-03(1.78E-04)+	2.71E-04(4.68E-05)-	3.13E-04(7.13E-06)-	5.32E-04(1.73E-05)+	3.79E-04(2.01E-04)
CPX1-T2	7.62E-02(1.83E-02)+	4.56E-02(1.77E-02)+	4.50E-03(4.81E-04)+	1.99E-03(6.43E-04)+	1.87E-03(5.83E-04)+	1.42E-03(5.36E-04)
CPX2-T1	7.36E-03(2.59E-04)+	6.56E-03(2.35E-04)+	2.56E-04(2.63E-05)+	3.08E-04(7.62E-06)+	3.06E-03(2.39E-05)+	2.01E-04(1.38E-05)
CPX2-T2	2.14E-01(7.69E-02)+	2.91E-02(4.39E-03)+	1.19E-02(3.96E-03)+	5.56E-03(1.92E-03)+	1.32E-03(1.42E-03)+	1.18E-03(1.13E-03)
CPX3-T1	1.08E-01(3.08E-02)+	1.19E-01(2.14E-02)+	3.85E-03(2.83E-03)+	3.99E-03(5.29E-04)+	2.97E-03(1.72E-04)-	3.02E-03(1.95E-04)
CPX3-T2	3.69E-02(3.44E-03)+	3.17E-02(1.93E-03)+	1.50E-03(9.79E-04)+	2.67E-03(4.78E-04)+	1.47E-03(1.25E-03)+	1.25E-03(1.01E-03)
CPX4-T1	7.90E-02(6.85E-03)+	8.41E-02(3.90E-02)+	7.89E-03(4.51E-03)+	4.14E-03(5.82E-04)-	4.17E-03(4.63E-04)-	4.23E-03(3.37E-03)
CPX4-T2	3.01E-01(3.39E-02)+	6.94E-02(1.67E-02)+	8.37E-03(2.98E-03)+	4.57E-03(2.53E-04)+	4.42E-03(2.07E-03)+	3.98E-03(1.84E-03)
CPX5-T1	4.91E-02(1.22E-02)+	3.90E-02(6.32E-03)+	5.34E-03(1.67E-03)+	2.49E-03(2.41E-04)+	2.16E-03(4.18E-04)+	2.01E-03(3.81E-04)
CPX5-T2	4.44E-01(1.11E-01)+	3.52E-01(8.35E-02)+	1.79E-02(1.34E-03)+	4.26E-03(1.60E-04)+	4.13E-03(1.55E-04)+	2.51E-03(8.63E-05)
CPX6-T1	5.02E-02(1.14E-02)+	3.31E-02(3.57E-03)+	4.36E-03(3.75E-03)+	2.46E-03(2.55E-04)+	2.48E-03(2.86E-04)+	2.18E-03(2.31E-03)
CPX6-T2	1.27E-01(2.99E-02)+	9.31E-02(1.09E-02)+	5.80E-03(9.72E-04)+	4.24E-03(8.45E-04)+	5.63E-03(7.52E-04)+	4.19E-03(3.87E-04)
CPX7-T1	3.94E-02(4.79E-03)+	3.27E-02(2.97E-03)+	2.31E-03(5.38E-04)+	2.26E-03(3.48E-04)+	2.13E-03(2.26E-04)-	2.21E-03(2.83E-04)
CPX7-T2	3.91E-02(3.65E-03)+	3.43E-02(5.47E-03)+	2.13E-03(3.83E-03)+	1.67E-03(3.41E-04)+	1.62E-03(3.38E-04)+	1.48E-03(3.12E-04)
CPX8-T1	3.75E-02(4.66E-03)+	3.12E-02(2.51E-03)+	6.05E-03(2.38E-03)+	2.11E-03(3.40E-04)+	1.68E-03(5.42E-04)+	1.32E-03(4.38E-04)
CPX8-T2	2.96E-01(1.03E-01)+	2.33E-01(3.14E-02)+	8.24E-03(1.49E-03)+	4.86E-03(8.17E-04)+	4.77E-03(6.92E-04)+	4.52E-03(5.57E-04)
CPX9-T1	3.89E-01(8.05E-02)+	3.17E-01(9.60E-02)+	4.31E-03(3.82E-04)+	4.45E-03(1.76E-04)+	4.39E-03(1.84E-04)+	4.11E-03(1.27E-04)
CPX9-T2	1.33E-01(5.34E-02)+	8.83E-02(7.81E-03)+	4.48E-03(2.01E-03)+	4.71E-03(2.17E-04)+	3.99E-03(2.01E-04)+	2.86E-03(1.74E-04)
CPX10-T1	4.43E-01(1.87E-01)+	1.40E-01(3.52E-02)+	1.58E-02(7.29E-03)+	5.93E-03(2.85E-03)+	5.89E-03(2.61E-03)+	3.24E-03(1.27E-03)
CPX10-T2	2.84E-01(5.04E-02)+	1.70E-01(4.05E-02)+	2.31E-03(4.56E-04)-	1.50E-02(1.30E-03)+	1.23E-02(3.41E-04)+	1.19E-02(3.36E-03)
+	20	20	18	18	17	-
≈	0	0	0	0	0	-
-	0	0	2	2	3	-

Table 5 Results obtained for NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, and EMT-MPM by the Friedman’s test on the second test suit to IGD

Algorithms	Ranking
EMT-MPM	1.3572
EMT/ET	2.3027
MOMFEA-SADE	3.0001
EMEA	3.3533
MO-MFEA	5.1503
NSGA-II	5.8542

EMT-MPM surpasses EMT-NG, EMT-NMS, and EMT-NGP on 18, 16, and 16 tasks, respectively. In contrast, EMT-MPM is surpassed by EMT-NG, EMT-NMS, and EMT-NGP on 0, 2, and 2 tasks, respectively. Therefore, the experimental

results indicate that each part of the proposed multidirectional prediction method is effective.

Parameter sensitivity analysis

In EMT-MPM, M controls the number of representative points. To analyze the effect of M , the experiments are conducted on the first test suit. M is set as 4, 5, 6, 7, and 8, respectively. Table 7 shows the experimental results by EMT-MPM with different M . When $M = 6$, EMT-MPM has the best performance. In addition, EMT-MPM has the worst performance when $M = 4$. This is because too few representative points lead to too few promising directions. When $M > 6$, the performance of EMT-MPM decreases gradually.

Fig. 6 The convergence trends of NSGA-II, MO-MFEA, EMEA, MOMFEA-SADE, EMT/ET, and EMT-MPM over 20 times on three representative problems of the first test suit

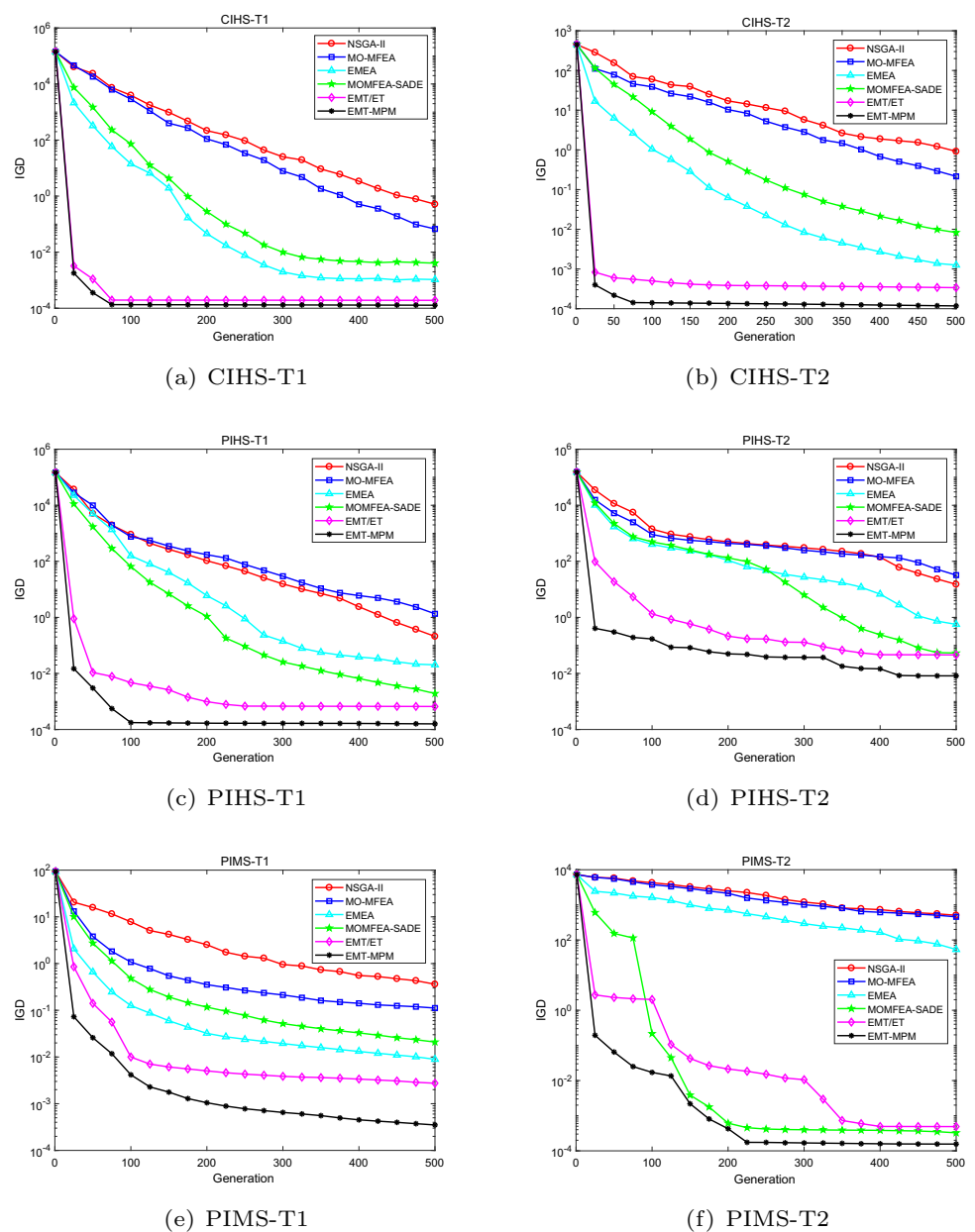


Table 6 Average value and standard deviation of IGD for EMT-NG, EMT-NMS, EMT-NGP, and EMT-MPM on the first test suit

Problems	EMT-NG	EMT-NMS	EMT-NGP	EMT-MPM
CIHS-T1	2.03E−04(1.52E−05)	1.92E−04(1.41E−05)	1.77E−04(1.36E−05)	1.62E−04(1.27E−05)
CIHS-T2	2.14E−04(1.67E−05)	1.98E−04(1.47E−05)	1.93E−04(1.38E−05)	1.83E−04(1.21E−04)
CIMS-T1	1.79E−04(1.56E−04)	1.85E−04(1.62E−04)	1.81E−04(1.57E−04)	1.65E−04(1.07E−04)
CIMS-T2	3.72E−04(2.13E−04)	3.35E−04(1.09E−04)	3.41E−04(1.14E−04)	1.93E−04(5.82E−05)
CILS-T1	2.03E−04(9.77E−04)	2.07E−04(9.82E−04)	1.95E−04(8.59E−04)	1.97E−04(9.01E−06)
CILS-T2	1.75E−04(1.31E−04)	1.92E−04(1.23E−04)	1.83E−04(1.09E−04)	1.62E−04(8.93E−06)
PIHS-T1	1.85E−04(7.61E−06)	1.69E−04(6.83E−06)	1.77E−04(7.02E−06)	1.64E−04(6.31E−06)
PIHS-T2	3.23E−04(5.86E−04)	3.21E−04(5.72E−04)	3.28E−04(6.03E−04)	3.18E−04(5.38E−04)
PIMS-T1	2.81E−04(3.14E−04)	3.57E−04(4.27E−04)	3.62E−04(3.08E−05)	2.57E−04(2.73E−04)
PIMS-T2	2.14E−04(1.79E−05)	1.96E−04(1.43E−05)	2.01E−04(1.47E−05)	1.92E−04(1.28E−05)
PILS-T1	1.78E−04(7.54E−06)	1.73E−04(7.24E−06)	1.82E−04(8.44E−06)	1.76E−04(7.91E−06)
PILS-T2	1.35E−03(2.46E−03)	1.26E−03(1.89E−03)	1.17E−03(1.73E−03)	8.63E−04(1.05E−03)
NIHS-T1	1.48E+00(7.52E−03)	1.48E+00(7.31E−03)	1.48E+00(8.34E−03)	1.47E+00(7.54E−03)
NIHS-T2	1.96E−04(9.27E−06)	1.69E−04(5.88E−04)	1.78E−04(6.21E−06)	1.64E−04(5.38E−06)
NIMS-T1	1.52E−01(4.52E−03)	1.58E−01(5.02E−03)	1.43E−01(4.27E−03)	1.49E−01(4.31E−03)
NIMS-T2	1.63E−04(9.85E−06)	1.58E−04(8.83E−06)	1.64E−04(9.83E−06)	1.53E−04(8.72E−06)
MILS-T1	7.65E−04(2.69E−04)	7.24E−04(2.51E−04)	7.81E−04(3.09E−04)	7.52E−04(2.81E−04)
MILS-T2	3.04E−04(7.31E−04)	3.77E−04(8.92E−04)	4.02E−04(1.20E−03)	1.83E−04(6.82E−04)

Table 7 Average value and standard deviation of IGD for $M = 4$, $M = 5$, $M = 6$, $M = 7$, and $M = 8$ on the first test suit

Problems	$M = 4$	$M = 5$	$M = 6$	$M = 7$	$M = 8$
CIHS-T1	2.04E−04(1.47E−05)	1.80E−04(1.39E−05)	1.62E−04(1.27E−05)	1.76E−04(1.35E−05)	1.98E−04(1.41E−05)
CIHS-T2	1.98E−04(1.55E−04)	1.87E−04(1.42E−04)	1.83E−04(1.21E−04)	1.92E−04(1.46E−04)	1.94E−04(1.53E−04)
CIMS-T1	2.23E−04(1.54E−04)	1.82E−04(1.15E−04)	1.65E−04(1.07E−04)	1.78E−04(1.29E−04)	2.13E−04(1.50E−04)
CIMS-T2	4.08E−04(1.87E−04)	2.14E−04(7.82E−05)	1.93E−04(5.82E−05)	1.85E−04(5.49E−05)	3.16E−04(1.87E−04)
CILS-T1	2.04E−04(8.94E−06)	1.93E−04(7.63E−06)	1.97E−04(9.01E−06)	2.10E−04(9.31E−06)	2.13E−04(9.42E−06)
CILS-T2	1.85E−04(1.14E−05)	1.82E−04(1.07E−05)	1.62E−04(8.93E−06)	1.71E−04(9.48E−06)	1.86E−04(1.15E−05)
PIHS-T1	1.78E−04(7.31E−06)	1.67E−04(6.53E−06)	1.64E−04(6.31E−06)	1.71E−04(6.82E−06)	1.84E−04(7.39E−06)
PIHS-T2	3.19E−04(5.47E−04)	3.06E−04(4.81E−04)	3.18E−04(5.38E−04)	3.27E−04(5.81E−04)	3.30E−04(6.01E−04)
PIMS-T1	3.40E−04(4.16E−04)	2.75E−04(2.96E−04)	2.57E−04(2.73E−04)	3.37E−04(3.40E−04)	4.12E−04(4.08E−04)
PIMS-T2	2.06E−04(1.62E−05)	1.97E−04(1.45E−05)	1.92E−04(1.28E−05)	1.94E−04(1.38E−05)	2.01E−04(1.52E−05)
PILS-T1	1.87E−04(8.92E−06)	1.78E−04(8.06E−06)	1.76E−04(7.91E−06)	1.79E−04(8.02E−06)	1.83E−04(8.41E−06)
PILS-T2	1.26E−03(2.48E−03)	1.03E−03(2.14E−03)	8.63E−04(1.05E−03)	9.74E−04(1.76E−03)	9.75E−04(1.68E−03)
NIHS-T1	1.49E+00(9.93E−03)	1.48E+00(7.61E−03)	1.47E+00(7.54E−03)	1.48E+00(7.39E−03)	1.48E+00(8.24E−03)
NIHS-T2	1.85E−04(8.03E−06)	1.81E−04(7.63E−06)	1.64E−04(5.38E−06)	1.59E−04(4.82E−06)	1.73E−04(6.82E−06)
NIMS-T1	1.55E−01(4.69E−03)	1.50E−01(4.37E−03)	1.49E−01(4.31E−03)	1.52E−01(4.46E−03)	1.54E−01(4.57E−03)
NIMS-T2	1.60E−04(9.08E−06)	1.58E−04(8.73E−06)	1.53E−04(8.72E−06)	1.56E−04(8.83E−06)	1.61E−04(9.11E−06)
MILS-T1	7.65E−04(3.02E−04)	7.63E−04(2.98E−04)	7.52E−04(2.81E−04)	7.01E−04(2.37E−04)	7.48E−04(2.60E−04)
MILS-T2	4.73E−04(1.38E−03)	3.51E−04(9.24E−04)	1.83E−04(6.82E−04)	2.47E−04(8.42E−04)	3.95E−04(1.03E−03)

When the number of representative points is too large, there will be too few points in each class, resulting in inaccurate prediction directions.

Conclusion

In this paper, a multiobjective EMT algorithm based on multidirectional prediction method is presented. First, the binary clustering method is used to generate multiple classes and

corresponding representative points. Second, two successive generations of representative points are used to generate the prediction direction for each class. Then the mutation strength is adaptively controlled according to the improvement degree of each class. Thirdly, each class generates the predictive transferred solutions by the prediction directions and mutation strengths. Finally, EMT-MPM is compared with one multiobjective algorithm and four multiobjective EMT algorithms on two MTO test suits. Experimental results demonstrate that EMT-MPM achieves excellent performance.

For future work, some new prediction methods such as support vector regression predictor and adaptive multilevel prediction can be introduced to MTO. Then it is also an interesting idea to combine prediction method with subspace alignment technique. Moreover, it is very meaningful to extend the proposed EMT-MPM algorithm to the practical application.

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

References

- Zhong J, Wang T, Cheng L (2021) Collision-free path planning for welding manipulator via hybrid algorithm of deep reinforcement learning and inverse kinematics. *Complex Intell Syst* 1–14
- Gao K, Huang Y, Sadollah A, Wang L (2019) A review of energy-efficient scheduling in intelligent production systems. *Complex Intell Syst* 1–13
- Chen J, Li J, Guo Y, Li J (2021) Consensus control of mixed-order nonlinear multi-agent systems: framework and case study. *IEEE Trans Cybern*
- Dehkordi MR, Seifzadeh H, Beydoun G, Nadimi-Shahraki MH (2020) Success prediction of android applications in a novel repository using neural networks. *Complex Intell Syst* 6:573–590
- Gupta A, Mandziuk J, Ong YS (2015) Evolutionary multitasking in bi-level optimization. *Complex Intell Syst* 1(1–4):83–95
- Liu WL, Yang J, Zhong J, Wang S (2021) Genetic programming with separability detection for symbolic regression. *Complex Intell Syst* 7(3):1185–1194
- Gupta A, Ong YS, Feng L, Tan KC (2016) Multiobjective multifactorial optimization in evolutionary multitasking. *IEEE Trans Cybern* 47(7):1652–1665
- Gupta A, Ong YS, Feng L (2015) Multifactorial evolution: toward evolutionary multitasking. *IEEE Trans Evolut Comput* 20(3):343–357
- Bali KK, Gupta A, Ong YS, Tan PS (2020) Cognizant multitasking in multiobjective multifactorial evolution: MO-MFEA-II. *IEEE Trans Cybern* 51(4):1784–1796
- Yang C, Ding J, Tan KC, Jin Y (2017) Two-stage assortative mating for multi-objective multifactorial evolutionary optimization. In: *IEEE 56th annual conference on decision and control*, pp 76–81
- Zheng X, Qin AK, Gong M, Zhou D (2019) Self-regulated evolutionary multitask optimization. *IEEE Trans Evolut Comput* 24(1):16–28
- Zhou L, Feng L, Tan KC, Zhong J, Zhu Z, Liu K, Chen C (2020) Toward adaptive knowledge transfer in multifactorial evolutionary computation. *IEEE Trans Cybern*
- Bali KK, Gupta A, Feng L, Ong YS, Siew TP (2017) Linearized domain adaptation in evolutionary multitasking. In: *IEEE congress on evolutionary computation*, pp 1295–1302
- Feng L, Huang Y, Zhou L, Zhong J, Gupta A, Tang K, Tan KC (2020) Explicit evolutionary multitasking for combinatorial optimization: A case study on capacitated vehicle routing problem. *IEEE Trans Cybern*
- Ding J, Yang C, Jin Y, Chai T (2017) Generalized multitasking for evolutionary optimization of expensive problems. *IEEE Trans Evolut Comput* 23(1):44–58
- Liang Z, Dong H, Liu C, Liang W, Zhu Z (2020) Evolutionary multitasking for multiobjective optimization with subspace alignment and adaptive differential evolution. *IEEE Trans Cybern*
- Zhou L, Feng L, Gupta A, Ong YS (2021) Learnable evolutionary search across heterogeneous problems via kernelized autoencoding. *IEEE Trans Evolut Comput*
- Feng L, Zhou L, Zhong J, Gupta A, Ong YS, Tan KC, Qin AK (2018) Evolutionary multitasking via explicit autoencoding. *IEEE Trans Cybern* 49(9):3457–3470
- Lin J, Liu HL, Tan KC, Gu F (2020) An effective knowledge transfer approach for multiobjective multitasking optimization. *IEEE Trans Cybern*
- Lin J, Liu HL, Xue B, Zhang M, Gu F (2019) Multiobjective multitasking optimization based on incremental learning. *IEEE Trans Evolut Comput* 24(5):824–838
- Zhou A, Jin Y, Zhang Q, Sendhoff B, Tsang E (2007) Prediction-based population re-initialization for evolutionary dynamic multi-objective optimization. In: *International conference on evolutionary multi-criterion optimization*, pp 832–846
- Zhou A, Jin Y, Zhang Q (2013) A population prediction strategy for evolutionary dynamic multiobjective optimization. *IEEE Trans Cybern* 44(1):40–53
- Jiang S, Yang S (2017) A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization. *IEEE Trans Evolut Comput* 21(1):65–82
- Peng Z, Zheng J, Zou J, Liu M (2015) Novel prediction and memory strategies for dynamic multiobjective optimization. *Soft Comput* 19(9):2633–2653
- Wu Y, Jin Y, Liu X (2015) A directed search strategy for evolutionary dynamic multiobjective optimization. *Soft Comput* 19(11):3221–3235
- Rong M, Gong D, Zhang Y, Jin Y, Pedrycz W (2019) Multidirectional prediction approach for dynamic multiobjective optimization problems. *IEEE Trans Cybern* 49(9):3362–3374
- Hansen N (2006) The CMA evolution strategy: a comparing review, towards a new evolutionary computation, pp 75–102
- Li Z, Zhang Q (2017) A simple yet efficient evolution strategy for large-scale black-box optimization. *IEEE Trans Evolut Comput* 22(5):637–646
- Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. *Evolut Comput* 9(2):159–195
- Zhang X, Tian Y, Jin Y (2014) A knee point-driven evolutionary algorithm for many-objective optimization. *IEEE Trans Evolut Comput* 19(6):761–776

31. Deb K, Pratap A, Agarwal S, Meyarivan TAMT (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evolut Comput* 6(2):182–197
32. Liang Z, Liang W, Wang Z, Ma X, Liu L, Zhu Z (2021) Multiobjective evolutionary multitasking with two-stage adaptive knowledge transfer based on population distribution. *IEEE Trans Syst Man Cybern Syst*
33. Yuan Y, Ong YS, Feng L, Qin AK, Gupta A, Da B, Zhang Q, Tan KC, Jin Y, Ishibuchi H (2017) Evolutionary multitasking for multiobjective continuous optimization: Benchmark problems, performance metrics and baseline results. *arXiv preprint [arXiv:1706.02766](https://arxiv.org/abs/1706.02766)*
34. Feng L, Qin K, Gupta A, Yuan Y, Ong YS, Chi X (2019) IEEE CEC 2019 competition on evolutionary multi-task optimization (online). <http://cec2019.org/programs/competitions.html#cec02>
35. Van Veldhuizen DA, Lamont GB (1998) Multiobjective evolutionary algorithm research: a history and analysis. In: Technical report TR-98-03, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio
36. Deb K, Agrawal RB (1995) Simulated binary crossover for continuous search space. *Complex Syst* 9(2):115–148
37. Zitzler E, Kunzli S (2004) Indicator-based selection in multiobjective search. In: International conference on parallel problem solving from nature, pp 832–842
38. Friedman M (1937) The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Publ Am Stat Assoc* 32(200):675–701

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