A Multifactorial Optimization Framework Based on Adaptive Intertask Coordinate System

Zedong Tang, Maoguo Gong, *Senior Member, IEEE*, Yue Wu, *Member, IEEE*, A. K. Qin, *Senior Member, IEEE*, and Kay Chen Tan, *Fellow, IEEE*

***Abstract*—The searching ability of the population-based search algorithms strongly relies on the coordinate system on which they are implemented. However, the widely used coordinate systems in the existing multifactorial optimization (MFO) algorithms are still fixed and might not be suitable for various function landscapes with differential modalities, rotations, and dimensions; thus, the intertask knowledge transfer might not be efficient. Therefore, this article proposes a novel intertask knowledge transfer strategy for MFOs implemented upon an active coordinate system that is established on a common subspace of two search spaces. The proper coordinate system might identify some common modality in a proper subspace to some extent. In this article, to seek the intermediate subspace, we innovatively introduce the geodesic flow that starts from a subspace, reaching another subspace in unit time. A low-dimension intermediate subspace is drawn from a uniform distribution defined on the geodesic flow, and the corresponding coordinate system is given. The intertask trial generation method is applied to the individuals by first projecting them on the low-dimension subspace, which reveals the important invariant features of the multiple function landscapes. Since intermediate subspace is generated from the major eigenvectors of tasks' spaces, this model turns out to be intrinsically regularized by neglecting the minor and small eigenvalues. Therefore, the transfer strategy can alleviate the influence of noise led by redundant dimensions. The proposed method exhibits promising performance in the experiments.**

***Index Terms*—Coordinate system adaption, evolutionary multitasking, multifactorial optimization (MFO), multitask optimization.**

Manuscript received March 8, 2020; revised July 13, 2020 and October 4, 2020; accepted December 1, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 62036006; in part by the National Key Research and Development Program of China under Grant 2017YFB0802200; in part by the Key Research and Development Program of Shaanxi Province under Grant 2018ZDXM-GY-045; and in part by the Australian Research Council under Grant LP170100416, Grant LP180100114, and Grant DP200102611. This article was recommended by Associate Editor H. Ishibuchi. *(Corresponding author: Maoguo Gong.)*

Zedong Tang is with the Academy of Advanced Interdisciplinary Research, Xidian University, Xi'an 710071, China (e-mail: omegatangzd@gmail.com).

Maoguo Gong is with the School of Electronic and Engineering, Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an 710071, China (e-mail: gong@ieee.org).

Yue Wu is with the School of Computer Science and Technology, Xidian University, Xi'an 710071, China (e-mail: ywu@xidian.edu.cn).

A. K. Qin is with the Department of Computer Science and Software Engineering, Swinburne University of Technology, Hawthorn, VIC 3122, Australia (e-mail: kqin@swin.edu.au).

Kay Chen Tan is with the Department of Computer Science, City University of Hong Kong, Hong Kong, China, and also with the City University of Hong Kong, Shenzhen Research Institute, Shenzhen 518057, China (e-mail: kaytan@cityu.edu.hk).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TCYB.2020.3043509.

Digital Object Identifier 10.1109/TCYB.2020.3043509

I. Introduction

PROBLEMS seldom exist in separate tasks. Human beings can easily tackle multiple relative tasks at the same time by capturing the related characteristics of the tasks and utilizing the previous knowledge [1]. Thereby, the optimization tasks can benefit from the exchange of the shared problem-solving experience of multiple tasks. Many multitask schemes invoke a number of distinct types of domains. The roles of tasks in the multitask scheme are symmetric, namely, some tasks can be the targets of transferred knowledge, while the other tasks are the sources and vice versa. The source tasks are usually assumed to have underlying complementaries for the target tasks, although the distinction between the tasks is supposed to be considered. The main objective of multitasking is to efficiently utilize the shared problem-solving experience to attain good performance on every task [2]. In optimization, underlying complementaries among the different tasks can lie in either the distribution of solutions or the search directions, in some cases, both of them [3]–[5]. Hence, it is important to properly deal with the knowledge transfer between different domains. It is worth noting that we have no *a priori* concerning the explicit relationship of the tasks [6] in the black-box optimization. Some works [5], [7] analyzed task complementarity and gave some analytical views on the benefit of the intertask knowledge transfer in multitask optimization.

Evolutionary algorithms (EAs) [6]–[8] have achieved great success in solving various single-task optimization problems (STOs) in the field of science and technology, for instance, continuous optimization [9]–[12], combinatorial optimization [13]–[15], and constrained optimization [16]–[18], etc. However, when related tasks are demanded, it is wise to use the sharing problem-solving experience of them to accelerate the optimization and avoid searches from scratch [3], [4]. Techniques for dealing with multiple optimization tasks (MTOs) simultaneously by utilizing the redundancy of the population of EAs have been studied under the name of evolutionary multitasking. The main topic of evolutionary multitasking is how to find and leverage the underlying complementary of tasks. It is worth noting that MTO might not be good at tackling tasks which are entirely opposite.

Several works aim to derive the domain information by elaborating implicit individual generating schemes [19] or transforming the individuals via explicit models [20]. There are three typical transfer schemes to tackle the distinction among search spaces of different tasks in the

evolutionary multitasking, namely, the unified representation scheme [6], centering and random shuffling strategy [19], and the explicit autoencoding method [20]. The unified representation scheme [6] intuitively assumes the alleles are inherently aligned, and the chromosomes from different tasks perform crossover directly on a normalized search space. The centering and randomly aligned strategy [19] assumes further that the alleles are not aligned yet, but every allele in the chromosomes for a task can be randomly paired with an allele in the chromosomes for another task. Particularly, this type of combinational problems is also challenging. Therefore, the performance of this method is not stable enough. The autoencoding method [20] can map the solutions in a search space to another search space explicitly by a linear autoencoding model, but it actually forces modeling the distribution information of individuals, not the search direction. Generally, when solving the diverse problems of real-world applications, tasks have implicit complementary. Algorithms should extract adaptively the features of multiple function landscapes and dig the underlying similarity of them. Thus, an efficient intertask knowledge transfer that can handle different function landscape modalities and rotations is still desirable.

The eigencoordinate systems [21], [22] used in the population-based algorithms have demonstrated their advantage in identifying the modality and rotation of function landscapes and significantly enhance the searching ability of those algorithms. But seldom has the attempt been made to implement the intertask knowledge transfer method upon the eigencoordinate systems. In this work, our objective is to seek a coordinate system upon which the intertask knowledge transfer strategy is implemented with low drift, which can extract both distributional information and search direction information. Establishing the coordinate system in the multifactorial environment, which can identify the common features of both function landscapes, faces two challenges. First, the feasible intertask knowledge transfer method should tackle tasks with different dimensions. Particularly, if the solutions have the same dimension, the chromosomes of different tasks can synthesize freely without misleading transfer. But an eigencoordinate system synthesizing solutions with different dimensions is not considered yet in the literature, especially, when the distribution information is encompassed. Second, it is difficult to find a rational common coordinate system that can roughly reflect the distributional information of solutions concerning different tasks.

In order to deal with the first challenge, we innovatively extract the features of function landscapes in the common low-dimension subspace and establish the corresponding coordinate system in the subspace. We first state that if the active coordinate systems are calculated according to subpopulations belonging to two tasks can be placed into a geodesic flow on the Grassmann manifold, these two tasks are similar. Therefore, to tackle the second challenge, a geodesic flow with one parameter is built on the Grassmann manifold, by assuming that the change between two tasks' subspaces is continuous. By specifying the parameter of geodesic flow, an intermediate subspace and corresponding coordinate system are obtained which can extract the common feature of two function landscapes. Moreover, we propose an adaptive framework that tunes the coordinate system in every generation and incorporates the intertask knowledge transfer strategies occurring in the intermediate subspaces and original spaces in a random way. The benefits of the proposed method are twofold. On one hand, the solutions from tasks with different dimensions can synthesize to generate the offspring in the intermediate subspaces. The coordinate systems can identify the common modality of function landscapes. On the other hand, since the coordinate system is established by taking the most important components of search spaces, the model is to some extent regularized such that it is robust enough to withstand the shock of the noise. In short, the intertask eigencoordinate system can improve the performance of MTO and make the intertask evolutionary operations insensitive to rotation of function landscapes.

In summary, our main contributions are listed as follows.

1) A geodesic flow of search subspaces, which can build a curve on the Grassmann manifold, is proposed to exploit the intrinsic low-dimensional structures in the solutions of different tasks.
2) A random distribution is established on the geodesic flow from which the intrinsic subspace can be drawn, such that the solutions of both tasks share the most common features in this subspace.
3) The intertask knowledge transfer strategies occur in the original spaces, and the intermediate subspaces are incorporated to generate offspring randomly.
4) empirical studies are conducted on the comprehensive benchmark problems.

The remainder of this article is originated as follows. Section II introduces some preliminaries on evolutionary multitasking and Grassmann manifolds. Section III describes the proposed method in detail. In Section IV, the performance of the proposed method is verified on comprehensive benchmark problems of both synthesized tasks and reinforce learning tasks. The work is concluded in Section V.

II. Preliminary

A. Evolutionary Multitasking

Multitasking optimization [8] aims to improve the performance of EAs in solving MTOs simultaneously by leveraging problem-solving experience among them. In a multitask setting with K tasks, the kth task, denoted by T_k, has a search space X_k with an objective function f_k. Such the purpose of a multitasking instance is to seek a set of solutions $\{\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_K^*\}$ such that

$$\mathbf{x}_k^* = \text{argmin}_{\mathbf{x}\in X_k} f_k(\mathbf{x}_k) \tag{1}$$

where $\mathbf{x}_k^*$ is the sufficiently optimal solution of T_k. Evolutionary multitasking is to search multiple decision spaces of multiple problems via implicit parallel of population-based optimization algorithms.

Multifactorial optimization (MFO) is a novel evolutionary multitasking paradigm where the MTOs are performed simultaneously and each task f_i is viewed as a factor influencing

TABLE I
DIFFERENCES AMONG THREE TRANSFER SCHEMES

	Unified Representation	Centering & Random Shuffling	Autoencoding	Our
Fitting Function Landscape	×	×	×	√
Common Space Dimension	Full	Full	Full	Partial
Transfer Strategy	Allele Alignement	Random Assignment Decision Variable Translation	Linear Mapping	Basis Vector Alignment Subspace Sampling

the evolution of the individuals in the K-factorial environment [6]. In MFO, all individuals are first encoded into a unified representation space. Therefore, a single population of individuals can navigate the unified representation space with a unified performance criterion and seek for optimums of multiple tasks in a single run. Several definitions associated with the individuals are shown as follows.

Definition 1: The factorial cost of individual p_i on task T_j is the objective value f_j of potential solution p_i noted as ψ_j^i.

Definition 2: The factorial rank of p_i on T_j is the rank index of p_i in the sorted objective value list given by r_j^i.

Definition 3: The skill factor is defined by the index of the task which an individual is assigned to. The skill factor of p_i is given by $\tau_i = \text{argmin}_{j\in\{1,2,\ldots,K\}} r_j^i$.

Definition 4: The scalar fitness of p_i is the inverse of r_{ϕ_i} given by $\theta_i = 1/\min_{j\in\{1,\ldots,K\}} r_j^i$.

Herein, the skill factor is regarded as the cultural trait that is inherited from the parents. In general, the scalar fitness is employed as the unified performance criterion.

MFEA is a popular implementation of MFO, which incorporates GAs into the MFO paradigm [6]. MFEA features assortative mating and selective imitation to transfer the information among different tasks. Specifically, assortative mating is employed to generate the offspring. It guarantees that the individuals associated with the same task have a high probability of mating, while the individuals associated with different tasks mate at a smaller probability. This procedure is controlled by a random mating parameter, denoted by *rmp*. Selective imitation simulates the phenomenon that the phenotype of offspring can inherit the phenotype of parents. It helps MFEA assign the skill factor of offspring. In the selective imitation strategy, an individual is evaluated on only one task indicated by its skill factor for reducing computational costs.

Currently, many works on methodology and real-world applications have been investigated since the primary framework of MFO was proposed in [6]. Some existing works focused on improving the efficiency of MFEA by tuning the modules of the original framework. Bali *et al.* [23] designed a linear domain adaptation and obtained a high-order representative space to translate the individuals belonging to different tasks. Wen and Ting [24] analyzed the searching behavior of MFO and proposed a novel strategy to tradeoff the intertask and inner domain crossover. Toward robust and efficient multitask optimization performance, Zhou *et al.* [25] proposed a new MFEA with adaptive knowledge transfer (MFEA-AKT) where the crossover operator employed for knowledge transfer is self-adaptively selected based on the information collected during the population evolving process. Gupta *et al.* [8] adapted MFEA in the field of multiobjective optimization by employing the nondomination rank as the performance criteria. Feng *et al.* [26] incorporated the swarm intelligence method, like particle swarm optimization and differential EAs, into MTO. Wen and Ting [27] incorporated the MFEA paradigm into genetic programming, leading to the MFGP algorithm for learning the ensemble of classifiers. There are other works focusing on developing the many-task optimization algorithms. Feng *et al.* [28] proposed a novel evolutionary multitasking algorithm to optimize multiple vehicle routing problems with occasional drivers (VRPHTOs) simultaneously with a single population. Feng *et al.* [29] proposed an explicit evolutionary multitasking algorithm (EEMTA) for vehicle routing problems (CVRP) consisting of a weighted l-norm-regularized learning process for capturing the transfer mapping, and a solution-based knowledge transfer process across VRPs. Liaw and Ting [30] made an early attempt to solve many-task optimization by simulating the evolution of biocoenosis through symbiosis. To address the drawback of their previous work that lacks the ability of seeking the tasks to transfer the complementing information to, Liaw and Ting [31] designed a novel adaptive information transfer for many-task optimization. Chen *et al.* [32] proposed a novel evolutionary framework for many-task optimization, where a composite complementary measure combining the similarity and the accumulated rewards of knowledge transfer is utilized. Besides, Liu *et al.* [33] proposed a surrogate-assisted multitask optimization framework, where the population is split into multiple subpopulations and every subpopulation focuses on solving a task. Each subpopulation utilizes a surrogate-assisted approach of the Gaussian process to search the decision space. MFEA-II proposed in [7] is a novel method, which enables online learning and exploitation of similarities between distinct tasks to minimize the tendency of harmful interactions across tasks. MO-MFEA-II proposed in [34] is a variant of the above method for multitask multiobjective optimization. It learns intertask relationships of the multiobjective optimization tasks based on overlaps in the probabilistic search distributions and adapts the extent of genetic transfer.

There are mainly three transfer schemes in the evolutionary multitasking in the literature. These methods are described under the names of unified representation scheme, centering and random shuffling strategy, and explicitly autoencoding method [19], [20], [23]. The distinctions among them are summarized in Table I. The essence in those methods is to construct a common space such that the solutions from different tasks can be properly represented and synthesized to generate offspring. The performance of the specific intertask

knowledge transfer strategy relies on the common space where it is realized. These intertask knowledge transfer strategies in three aspects are depicted in Table I. First, it is considered whether the basis of space (i.e., the coordinate system) is suitable for the function landscapes. Second, knowledge transfer can occur in the full space (without loss of information and even noise), or in the low-dimension subspace (with the essential information reserved), which have a significant influence on the performance of intertask knowledge transfer strategies. Third, the inductive bias must be made to successfully search the necessary parameters in the intertask knowledge transfer process. The corresponding characteristics of the intertask knowledge transfer strategies are summarized in Table I.

B. Works Related to Eigencoordinate System

In [35], it is reported that the fixed coordinate system commonly used in the population-based search algorithms might fail to identify the modality of the function landscapes or even a single function landscape at different evolution stages. Thus, the search is not necessarily efficient by the evolution operations only actualized in the fixed coordinate systems. Therefore, many researchers are turning to the eigencoordinate system, which can efficiently adapt the function landscapes and enhance the search ability of the evolution operations. The general formulation of the evolution operators in the eigen coordinate systems can be described as

$$\begin{aligned}\mathbf{r} &= \mathbf{B}\left(\sum_{i=1}^{m}\alpha\left(\mathbf{B}^T\mathbf{y}_i\right)+\sum_{i=1}^{n}\mathbf{W}_i\left(\mathbf{B}^T\mathbf{z}_i\right)\right)\\ &= \sum_{i=1}^{m}\alpha_i\mathbf{y}_i+\sum_{i=1}^{n}\mathbf{B}\mathbf{W}_i\mathbf{B}^T\mathbf{z}_i \end{aligned} \tag{2}$$

where r represents the resultant vector. The existing works related to the eigencoordinate system are clustered into two categories based on the method by which the evolution operations are actualized. In the first category, the evolution operations are only conducted in the eigencoordinate system. A typical algorithm in this category is CMAES [36] developed based on the evolution strategy (ES). In CMAES, the offspring are generated by the following equation:

$$\begin{aligned}\mathbf{x}_i &= \mathbf{m}+\sigma N(0,C)\\ &= \mathbf{m}+\mathbf{B}\mathbf{D}\mathbf{B}^T N(0,I),\quad i=1,2,\ldots,\lambda\end{aligned} \tag{3}$$

where $\mathbf{m}$ is the mean vector of search distribution at generation t, σ is the search step size, $\mathbf{C}$ is the covariance matrix, $\mathbf{B}$ is the eigenvectors of $\mathbf{C}$, and $\mathbf{D}$ is a diagonal matrix. $N(0,\mathbf{C})$ and $N(0,\mathbf{I})$ represent the multivariate Gaussian distribution with zero mean and covariance matrix $\mathbf{C}$ and a standard multivariate normal distribution, respectively. By closely looking at (3), it is worthwhile to note that CMAES only samples in the eigencoordinate system. In CMAES, the eigencoordinate system is generated from the covariance matrix C which is computed by the rank-μ and rand-one update methods [36]. Since CMAES can adaptively extract the features of the function landscapes, it improves the search performance of the typical ES. Another typical algorithm in this category is AE_{CMA} [37], which is an adaptive encoding mechanism. AE_{CMA} can be applied to ES and estimation of distribution algorithm (EDA) [38]. In the second category, the evolution operations are implemented in both the Eigen and original coordinate systems. DE/eig [22], CoBiDE [21] and ACoS [35] are three typical algorithms in this category, in which the evolution operations generate the offspring population in either Eigen or original coordinate systems. The above-mentioned works suggest that the combination of the Eigen coordinate system and the original coordinate system can usually outperform the usage of a consistent coordinate system in the entire search process.

The success of usage of the eigencoordinate system in the conventional population-based algorithms motivates us to improve the performance of MTO by conducting the intertask knowledge transfer upon it and make the intertask evolutionary operations insensitive to rotation of function landscapes. However, seldom has the works attempted to implement intertask knowledge transfer in the eigencoordinate system. The challenge is lying on seeking a proper coordinate system from function landscapes with different modalities and rotations, in which the intertask information can be transferred efficiently. In this article, we use the Grassmann manifold as the mathematical tools to find the intertask coordinate system. A brief introduction on the Grassmann manifold is included in Section SI in supplementary materials.

III. Proposed Method

A. Motivation

To make the knowledge sharing across tasks more robust to the rotation of function landscapes, we proposed to implement the intertask knowledge transfer method on the active coordinate system. The motivation of GFMFDE comes from three aspects as follows.

1) When the distribution of the solutions is no longer parallel to the coordinate basis, the offspring are generated by the search operations in the original coordinate systems will be drifted due to the modality and rotation of the function landscapes. The coordinate system plays a significant role in the performance of STO algorithms. But, there is no attempt to embark the study in the realm of MTO. It is an interesting topic to employ the active coordinate systems suitable in the design of intertask knowledge transfer process.
2) In evolutionary multitasking, the function landscapes of different tasks are particularly not aligned such that the direct synthesis of these target vectors from multiple tasks leads to a relatively large drift on the search direction as depicted in Fig. 1. However, how to establish a proper coordinate system across two nonaligned coordinate system possessed by different tasks remains an unexplored area.
3) As introduced in the previous section, some researchers have realized the importance of integrating the active and original coordinate systems when designing the search operations.

To address the above issues, we first establish a coordinate system for every task. Our objective is to seek a suitable active coordinate system whose basis is consistent with the basis of

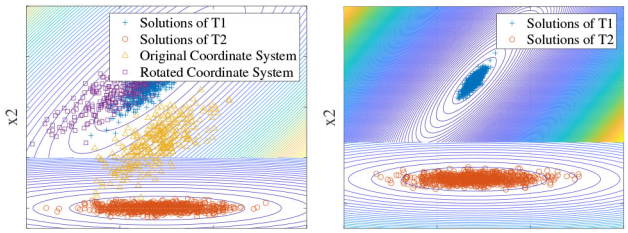


Fig. 1. Contours of three ellipsoidal functions. $T1$: Nonrotated ellipsoidal function. $T2$: Rotated ellipsoidal function (rotate 45°). Crosses and circles represent the solutions of $T1$ and $T2$. (a) Offspring generated in original coordinate system (denoted as triangles) and the proposed coordinate systems (denoted as squares). The offspring generated in original coordinate system are perturbed wildly beyond the search direction. (b) When the tasks have different scale of function landscapes, the large-scale coordinate system may dominate the small-scale one.

coordinate systems of two tasks. The active coordinate system should fit both function landscapes, such that the intertask transfer of low drift can be implemented. Also, from the view of geometrical intuition, one can think about two points (or two coordinate systems in the quotient space) in the Euclidean space and consider finding another point such that the total distance from it to two points is minimized. Such set of points satisfied above characteristic is practically the line segment whose ends are two points given at the beginning. This inspires us that once the "line" between two given subspaces is well defined, constructing the active coordinate system, whose basis is aligned with (closest to) two subspaces, is possible. So the key to seeking such active coordinate system is to establish the manifold of coordinate systems whose local areas are analogous to the Euclidean space. For this purpose, we have to recur to geodesic path, a counterpart of line in the Grassmann manifold. The analysis and inference of linear systems is feasible in the aid of geodesic path (parameterized geodesic path is called geodesic flow hereinafter). To efficiently make the inference on the linear spaces, we eventually fall back upon the mathematical tool introduced in [39].

When establishing the low-dimension subspace, in the process of intertask knowledge sharing, minor components are eliminated to reduce the influence of noise. At the same time, it is computationally efficient to construct the middle subspace which is similar to the two subspaces. However, each task still uses the original space-based evolution mechanism in the intratask search process, so the subpopulations of each task can reserve the task-specific distribution information without loss of accuracy. Besides, to strike the balance of the fidelity and robustness in knowledge sharing across tasks, we propose to combine the active coordinate system and original coordinate system in the design of the intertask evolutionary mechanism.

B. Construction of the Intermediate Subspace

1) Geodesic Flow: For simplicity, we first consider the simple case where there are two tasks T_0 and T_1. Both of them are associated with a D-dimension unified representative space, where D is the maximum dimension of decision spaces of T_0 and T_1. To make the distribution information derived from our method more sufficient, the cumulative population of two generations are archived, forming the history of the search process. The history size is double the population size, namely, $2NP$. The history of subpopulations possessed by the respective tasks are denoted by $P_{(t0)}$ and $P_{(t1)}$.

Algorithm 1 PCA

Input: X input matrix, d the desired number of principle components.
Output: A basis of the subspace.
1: Compute the covariance matrix by $C = X^T X$.
2: Calculate the eigen vectors U and eigen values $diag(\Lambda)$ by $C = U^T \Lambda U$.
3: Selected the first d largest eigen values $\{\lambda_m, m = 1, \cdots, d\}$ from $diag(\Lambda)$.
4: Stack vertically the eigen vectors corresponding to $\{\lambda_m\}$ from U.
5: Output the stacked matrix as A.

To establish the subspaces that can identify the essential features of the function landscapes, we first calculate the d-dimension subspaces $A_{(t0)}$ and $A_{(t1)}$ of the search spaces from the history of T_0 and T_1, respectively, where $d < D$. Since DE uses the algebraic operations to generate the next offspring, it actually takes the linear coordinate system in the local area of the search space. When solving the nonlinear optimization problems, DE takes the iterative scheme. In each iteration, a local coordinate system of local area of the search space is employed, and new positions are predicted upon it. Thus, we can establish the active coordinate system via a linear dimension reduction method, the principle component analysis (PCA) [40], which can capture the components where the population exhibits large variances.

The PCA as shown in Algorithm 1 is employed to seek the subspace

$$A^{(t0)} = \mathrm{PCA}\left(P^{(t0)}, d\right), \quad A^{(t1)} = \mathrm{PCA}\left(P^{(t1)}, d\right) \tag{4}$$

where $A_{(t0)}$ and $A_{(t1)}$ are the orthogonal matrices $\mathbb{R}^{N\times d}$, which represent, respectively, the subspaces of the original search spaces possessed by T_0 and T_1. The goal of the proposed method is to conduct the evolutionary mechanism in the intermediate subspace of $A_{(t0)}$ and $A_{(t1)}$. So the first job is to compute the intermediate subspace according to two specific subspaces. Since only the first d principle components are employed to establish the coordinate system of the intermediate subspace, this model is intrinsically regularized. Herein, d is a parameter that controls the dimensions of the subspaces. Too small d will lead to a loss of population information in the knowledge transfer, while too large d will lead to the excessive perturbation in the generated intertask offspring population because noise can be reserved in the intermediate subspace. As it is important to balance the fidelity and the robustness of the intertask knowledge sharing, we conduct an empirical experiment to analyze the sensitivity of the performance to the parameter d in the experimental section.

By using the generalized singular value decomposition (SVD) to efficiently compute the intermediate subspace of two search spaces, we need to compute the null space of A_0. The null space of a matrix A is defined as the set of solutions

to the equation $A\boldsymbol{x} = 0$. In a word, if A, which is a $D \times d$ matrix, contains the orthogonal base vectors of a d-dimension subspace, the base vectors of null space, represented by a $D\times(D-d)$ matrix denoted as N_0, are orthogonal to all columns of A. Therefore, all columns of A and N_0 form a basis of the D-dimension space. The null space of the tasks' subspaces A_0 and A_1 is solved according to the following:

$$A_{(t0)}^T \mathbf{x} = \mathbf{0}. \tag{5}$$

We can solve the above equation, then the null space of space $A_{(t0)}$ is given by the basis of solution space, denoted as $N_{(t0)}$, which is specified by a N-by-(N-d) matrix in $\mathbb{R}^{N\times(N-d)}$. Based on the source subspace and the target subspace, the geodesic flow between them is established via the method introduced in [39]. The key to establishing the geodesic flow is to compute the principle angles of two subspaces. Then, we utilize the principle angles to parameterize the geodesic flow.

The principle angles $0 \le \theta_1 \le \theta_2 \le \cdots \le \theta_d \le \pi/2$ between $\mathbf{A}_{(t0)}$ and $\mathbf{A}_{(t1)}$ are defined by

$$\begin{aligned} \cos\theta_m = \max_{\boldsymbol{u}_m^{(1)}\in \mathrm{span}(A_{(t0)})} \max_{\boldsymbol{v}_m\in \mathrm{span}(A_{(t1)})} & \boldsymbol{u}_m^{(1)T}\boldsymbol{v}_m \\ \text{subject to} \quad & \boldsymbol{u}_m^{(1)T}\boldsymbol{u}_m^{(1)} = 1, \quad \boldsymbol{v}_m^T\boldsymbol{v}_m = 1 \\ & \boldsymbol{u}_m^{(0)T}\boldsymbol{u}_n^{(0)} = 1, \quad \boldsymbol{v}_m^T\boldsymbol{v}_n = 0 \\ & (n = 1, \ldots, m-1). \end{aligned} \tag{6}$$

We solve the principle angles via SVD described as follows:

$$\begin{aligned} A_{(t0)}^T A_{(t1)} &= U_1 C V^T \\ N_{(t0)}^T A_{(t1)} &= -U_2 S V^T \\ \text{s.t.} \quad C^2 + S^2 &= I \end{aligned} \tag{7}$$

where $U_1 = \{\boldsymbol{u}_m^{(1)}\}$, $U_2 = \{\boldsymbol{u}_m^{(2)}\}$ and $V = \{\boldsymbol{v}_m\}$, $m = 1, \ldots, d$ are orthogonal matrices. C and S are the diagonal matrices containing the cosines and sines of $\theta_m (m = 1, \ldots, d)$, respectively. It is worthwhile to note that the elements of the diagonal of C are the cosine of the principle angles of two subspaces, while the ones of S are the sine of the principle angles. The principle angles between the subspaces can be calculated according to the diagonal of the singular value matrices C and S, namely, $\theta_m = \arccos(C_{m,m}) = \arcsin(S_{m,m})$.

2) Sampling the Subspace: The curve on the manifold of the subspaces passing through the subspaces of two tasks, denoted as geodesic path, plays an important role in the construction of the intermediate subspace. Such curve is parameterized by a real number ranged in [0, 1], denoted as $\Phi(t)$. When $t = 0$, the subspace $\Phi(0)$ represents the subspace $\mathbf{A}_{(t0)}$. When $t = 1$, $\Phi(1)$ is, respectively, the subspace of the second task. Once the parameterized curve is constructed, it is anticipated that a probability distribution can be defined upon the random variable t ranged in [0, 1] where each real number represents a subspace. An intermediate subspace is drawn from the distribution by randomly sampling a certain value of the random variable t. The proper setting of the subspaces' distribution may need more extend studies. But, due to no *a prior* knowledge on the distribution of the subspaces between two search spaces, we assume that the subspaces on the curve are drawn from a uniform distribution for simplicity and computational efficiency.

Given a random variable t, we can obtain an intermediate subspace based on the precomputed U_1, U_2, and V and the corresponding principle angles θ_m, $m = 1, \ldots, D$. Specifically, the principle angles between the intermediate subspace and the initial subspace are given as $\hat{\theta}_m = t\theta_m$. We use the notations $C(t)$ and $S(t)$ to denote the diagonal matrices (singular value matrices) with a parameter t. $C(t)$ and $S(t)$ are obtained by computing the following formulation:

$$\begin{aligned} C_{m,m}(t) &= \cos(\hat{\theta}_m) = \cos(t\theta_m) \\ S_{m,m}(t) &= \sin(\hat{\theta}_m) = \sin(t\theta_m) \\ C_{m,n}(t) &= 0, \ \ S_{m,n}(t) = 0, \ \ (n \ne m) \end{aligned} \tag{8}$$

$$(m, n = 1, \ldots, d). \tag{9}$$

The projector matrix containing the orthogonal base vectors of the active coordinate system is then calculated as follows:

$$\begin{aligned} B_{(t0,t1)} &= \Phi(t)|_{t\sim U(0,1)} \\ &= A_{(t0)}U_1 C(t) - N_{(t0)}U_2 S(t)\big|_{t\sim U(0,1)}. \end{aligned} \tag{10}$$

It is not hard to verify that $\Phi(0)$ and $\Phi(1)$ equals $A_{(t0)}V$ and $A_{(t1)}U_1$, respectively. Although, $A_{(t0)}V$ and $A_{(t0)}$ have different forms, the only different of them is the order of the column vectors. Thus, $A_{(t0)}V$ and $A_{(t0)}$ are equivalent. Similarly, $A_{(t1)}U_1$ and $A_{(t1)}$ are also equivalent. Moreover, the coordinate system $B_{(t1,t0)}$ can be also obtained via (10) due to the symmetry of this equation on the interval [0, 1].

C. Knowledge Transfer in the Intermediate Coordinate System

For implementing the *intertask knowledge transfer process*, two operations in the active coordinate system are designed herein, called *centering mutation* and *bias mutation*. Then, to conduct a diverse intertask knowledge transfer, the proposed method adjusts mutation operation implemented in the active and original coordinate systems in a random way. For clarity, by following the previous section, let P_0 and P_1 represent the solutions independently sampled from the search spaces of the optimization tasks T_0 and T_1, respectively.

First, the composite subpopulation for each individuals is created by merging the subpopulations of target task and source task. Considering an individual $\boldsymbol{x}_i$ in population P, the subpopulation of target task P_0 consists of the individuals belonging to the same task as $\boldsymbol{x}_i$. The source task is randomly selected from K tasks. The corresponding subpopulation P_1 contains the individuals assigned to source task. The composite subpopulation Q is the intersect set of P_0 and P_1. For simplicity, we denote the projector matrix as $B_{(t0,t1)}$ by B.

Next, we consider the *centering mutation*. To extract the information of the modality and rotation of function objectives, the individuals are centered by subtracting the mean of the subpopulation corresponding to the tasks and only the variances of individuals are considered. The proposed centering mutation strategy first project the centered parents into the intermediate subspace, which is achieved by applying the linear mapping B. Mathematically, the projected centered coordinates of any solution vector $\boldsymbol{x}$ are $\boldsymbol{y} = B^T(\boldsymbol{x} - \bar{\boldsymbol{x}})$, where $\bar{\boldsymbol{x}}$ is the cluster center of the solutions assigned to the task which $\boldsymbol{x}$ belongs to.

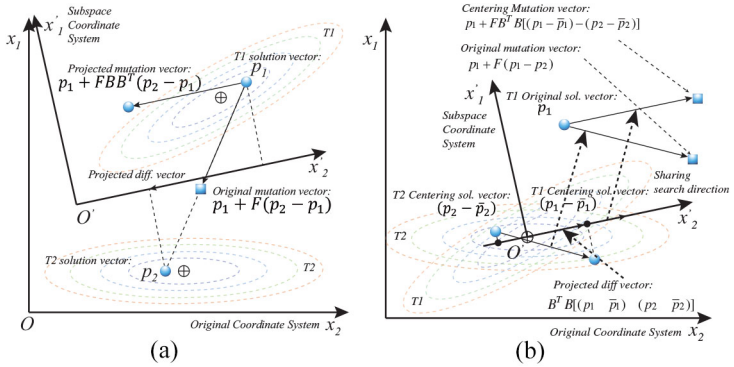


Fig. 2. DE operations work in different coordinate systems. Generation region is shown in the term of coordinate systems. The active coordinate system is relatively consistent with the coordinate systems of multiple function landscapes. The intertask trial vector generation strategies implemented in this coordinate systems can generate offspring combining the information of multiple function landscapes. (a) Bias mutation. (b) Centering mutation.

Herein, three unique individuals $\boldsymbol{x}_{rm}, m = 1, 2, 3$ are randomly selected from the composite population Q. By using the above notations, the trial vector generation of the centering mutation can be derived as follows:

$$\begin{aligned}\boldsymbol{u}_i &= \boldsymbol{x}_{r1} + F \cdot BR\big(\boldsymbol{y}_{r2} - \boldsymbol{y}_{r3}\big) \\ &= \boldsymbol{x}_{r1} + F \cdot BR\big(B^T(\boldsymbol{x}_{r2} - \bar{\boldsymbol{x}}_{r2}) - B^T(\boldsymbol{x}_{r3} - \bar{\boldsymbol{x}}_{r3})\big) \\ &= \boldsymbol{x}_{r1} + F \cdot BRB^T((\boldsymbol{x}_{r2} - \bar{\boldsymbol{x}}_{r2}) - (\boldsymbol{x}_{r2} - \bar{\boldsymbol{x}}_{r2})) \qquad (11)\end{aligned}$$

where R is a diagonal matrix whose diagonal is composed of the random numbers drawn from the uniform distribution in the range of [0, 1]. The trial vector $\boldsymbol{c}_i$ is created by the crossover the mutation vector $\boldsymbol{u}_i$ and the target vector $\boldsymbol{x}_i$.

Then, the intertask *bias mutation* where biases of individuals are considered is given by the following equation:

$$\begin{aligned}\boldsymbol{u}_i &= \boldsymbol{x}_{r1} + F \cdot BR\big(B^T\boldsymbol{x}_{r2} - B^T\boldsymbol{x}_{r3}\big) \qquad (12) \\ &= \boldsymbol{x}_{r1} + F \cdot BRB^T(\boldsymbol{x}_{r2} - \boldsymbol{x}_{r3}) \qquad (13)\end{aligned}$$

where the matrix R has the same definition as the one in (11). The trial vector $\boldsymbol{c}_i$ is created by the crossover of mutation vector $\boldsymbol{u}_i$ and target vector $\boldsymbol{x}_i$. This intertask trial vector generation strategy considers the bias of the individuals from different tasks. If the solutions of two tasks are close to each other in the unified representation space, the bias intertask transfer benefits solving of the tasks. So it is promising to consider simultaneously the complementaries of both search directions and the bias among the solutions of different tasks. Fig. 2 providers a graphical visualization of the proposed intertask offspring strategies.

In some problems where the tasks are similar to each other, the offspring generation strategy upon the original space provides a fast convergence than that upon the subspace, because it can reserve the entire information of population. Thereby, we also encompass the original intertask mutation operation analogous to the DE/rand/1 mutation operation. This intertask mutation operation is defined as follows:

$$\boldsymbol{u}_i = \boldsymbol{x}_{r1} + F \cdot (\boldsymbol{x}_{r2} - \boldsymbol{x}_{r3}) \qquad (14)$$

where $\boldsymbol{x}_{r1}$, $\boldsymbol{x}_{r2}$, and $\boldsymbol{x}_3$ are selected from the composite subpopulation Q, and $\boldsymbol{x}_i$ is selected from the target task. The trial vector $\boldsymbol{c}_i$ is created by the crossover of mutation vector $\boldsymbol{u}_i$ and target vector $\boldsymbol{x}_i$.

In the *intratask knowledge transfer process*, to slow down the convergence speed of the and encourage the exploration of region around the promise solutions, a DE/current-to-center/1 mutation operation is employed in the proposed method as an intratask trial vector generation strategy. This mutation operation is modified from the commonly used DE/current-to-best/1 mutation operation

$$\boldsymbol{u}_i = \boldsymbol{x}_{r1} + F \cdot (\boldsymbol{x}_{r2} - \boldsymbol{x}_{r3}) + F \cdot (\boldsymbol{x}_{\text{cen}} - \boldsymbol{x}_i) \qquad (15)$$

where $\boldsymbol{x}_{\text{cen}}$ denotes the center of the corresponding task's solutions, and $\boldsymbol{x}_{rm}, m = 1, 2, 3$ are selected from the solutions of the task that $\boldsymbol{x}_i$ belongs to.

It is worthwhile to note that $\boldsymbol{x}_{r1}$, $\boldsymbol{x}_{r2}$, and $\boldsymbol{x}_{r3}$ are sampled from the same task, which is different from (14). After mutation, the trial vector $\boldsymbol{c}_i$ is formed by a binomial crossover operation of $\boldsymbol{u}_i$ and target vector $\boldsymbol{x}_i$

$$c_{im} = \begin{cases} u_{im}, & \text{if rand()} \leq CR \text{ or } m == m_r \\ x_{im}, & \text{otherwise} \end{cases} \qquad (16)$$

where rand() is a uniform random number ranged in [0, 1] and m_r is a random integer ranged in [1, D]. Crossover probability CR roughly infers the fraction of vector components inherited from the mutation vector.

D. Framework

By combining the above components, the proposed method is given as follows. The basic structure follows the overall framework of multifactorial differential evolution as shown in Algorithm 2. The main modified part is the intertask transfer strategy. The proposed intertask trial vector generation strategy replaces the original intertask transfer. As described in the above section, the modified intertask trial vector generation strategy first establishes an intermediate subspace such that the principle axes of two distinct decision spaces can be aligned as well as possible. By this means, the individuals from different tasks can share more complementaries on the new subspace with the distant axes are neglected. Thus, the efficiency of the proposed method comes down to the intertask trial vector generation strategy on the intermediate subspace.

At the beginning of the evolution process, the subspaces where the individuals are located are established by applying the PCA in line 9. Then, the principle angles and the corresponding pairwise closest principle base vectors are computed by the method described in Section III-B in lines 10–12. The trial vector generation strategy is applied to the low-dimension representations of solutions in lines 19 and 20. There are some previous works [22], [35] that reported the efficiency of the combined reproductive operations on diverse coordinate systems. They are incorporated to strike the balance of loss of information and the robustness in line 19.

Finally, we analyze the computational complexity of the proposed knowledge transfer strategy. Most of additional computational efforts are spent in calculating the active coordinate system. The general time complexities of PCA and computing active coordinate system are $O(D^2d)$ and $O(D^2d)$, respectively. The matrix multiplication costs $O(Dd^2)$. The totally time complexity of calculating the active coordinate system is $O(2D^2d + Dd^2)$.

Algorithm 2 Main Framework of GFMDE

1: Initialize the population P_0.
2: $g = 0$.
3: **repeat**
4: Attach P_g in the history of population.
5: **if** the history size is greater than $2 * NP$ **then**
6: Remove the elder individuals from the history.
7: **end if**
8: Divide the population into subpopulations each of which possessed by one task.
9: Establish the subspaces for K tasks via **Algorithm 1**.
10: **for** each pair tasks $(t0, t1)$ **do**
11: Sample an active coordinate system $B_{(t0,t1)}$ from the history according to Section III-B.
12: **end for**
13: **for** $i = 1$ to NP **do**
14: **if** $rand() < rmp$ **then** {Inter-task mutation begins.}
15: Assign $t0$ with the task possessing $\boldsymbol{x}_i$.
16: Select a task $t1$ different from $t0$ randomly.
17: Form the composite subpopulation by merging the individuals belonging to $t0$ and $t1$.
18: **if** $rand() < 0.5$ **then**
19: Select a mutation operation from the candidate operations described in Section III-C.
20: Conduct the selected operation to generate the ith offspring.
21: **else**
22: Conduct the selected operation in the original coordinate system to generate the ith offspring.
23: **end if**
24: **else** {Intra-task mutation begins.}
25: Conduct the canonical mutation operation to generate the ith offspring.
26: **end if**
27: **end for**
28: Evaluate the offspring population.
29: Execute the selection operation to update P_g.
30: $g = g + 1$.
31: **until** Terminal condition holds.
32: Select the best individuals from final population and output $\{\boldsymbol{x}_1^*, \boldsymbol{x}_2^*, ..., \boldsymbol{x}_K^*\}$.

IV. Empirical Study

We assess the efficiency of the proposed method on comprehensive benchmarks. First, to confirm that the success of the proposed method is attributed to the novel intertask knowledge transfer, we compare the proposed MFDE with the canonical MFDE. Also, an ablation experiment is conducted to verify the efficacy of the proposed components. Next, the influence of the intermediate subspace dimension is empirically analyzed. To further demonstrate the superior performance of the proposed method, it races with several typical intertask transfer schemes on the same racing track. Eventually, the proposed method is compared with the currently proposed MTO and single-objective algorithms for an overview of this realm. Finally, an experiment of the proposed method on reinforce learning tasks are performed.

A. Experimental Setup

The proposed method is compared with following algorithms. All the parameters in the corresponding algorithms take the settings as in the literature. For a fair comparison, all parameters are fixed in entire experiments. The detailed parameter settings are listed in Table SII in supplementary materials. All methods run 30 times on each benchmark for eliminating the influence of random factors, and the mean values and standard deviation are recorded.

1) Our Method: The intertask knowledge strategy generates the trial vector by first projecting the tasks' solutions into the intermediate subspace. The parameters are set by referring to the settings of MFDE [26].

2) Variants of MFDE: Three variants are encompassed into the experiments. The original MFDE[1] uses the differential vector to transfer the perturbation. MFDE with centering and random shuffle (MFDE-RS) employs the intertask transfer strategy mentioned in [19], and the basis structure MFDE is reserved. MFDE-AE replaces the intertask knowledge transfer strategy by a linear mapping [20]. SREMTO [41] uses a self-adaptation process to tune the intensity of knowledge transfer in MFEA as per the relatedness between tasks. It employs a hybrid operation-based DE and GA in the intertask knowledge transfer process.

3) Other MTO Algorithms: The current proposed methods and a single-objective differential EA are included. MFEA [6],[2] the first attempt of evolutionary multitasking, has been verified to be efficient on varieties of scientific and engineering problems. MFEA2 [7] is an improved MFEA with an active intertask transfer probability, which can avoid the inefficient trial-and-error search for an appropriate parameter setting. MFPSO [26] incorporates PSO into the MFO framework.

4) Other Single-Objective Algorithms: DE[3] is the backbone of our method, which has been employed to tackle various

[1]The source codes of MFDE, MFDE with explicit autoencoding (MFDE-AE) and MFPSO are available in http://www.bdsc.site/websites/MTO/index.html.

[2]The source codes of MFEA and MFEA-II are available in https://www.ntu.edu.sg/home/asysong/home.html.

[3]The source code is available in https://www3.ntu.edu.sg/home/epnsugan/.

TABLE II
EFFICIENCY VERIFICATION BY COMPARING GFMFDE WITH MFDE. *MEAN* AND *STD* REPRESENT THE AVERAGED BEST FUNCTION VALUE AND STANDARD DEVIATION OF THE BENCHMARK PROBLEMS S10-18 OVER 30 INDEPENDENT RUNS. THE BEST ONES ARE TYPED IN *BOLD*. THE WILCOXON'S RANK SUM TEST WITH 95% CONFIDENCE LEVEL IS PERFORMED ON THE EXPERIMENTAL RESULTS

Pro.		GFMFDE		MFDE		p-value	
		T1	T2	T1	T2	T1	T2
S10	mean	$\mathbf{0.0000E+00}$	$\mathbf{0.0000E+00}$	$7.2190E-14$	$9.3152E-11$	0.0004	0.0004
	std	*0.00E+00*	*0.00E+00*	*7.58E-14*	*9.45E-11*		
S11	mean	$\mathbf{1.1529E-13}$	$\mathbf{0.0000E+00}$	$3.3178E-08$	$6.1326E-13$	0.0004	0.0004
	std	*2.56E-13*	*0.00E+00*	*1.00E-08*	*5.64E-13*		
S12	mean	$\mathbf{1.0520E-08}$	$\mathbf{6.3638E-04}$	$2.1174E+01$	$1.1444E+04$	0.0004	0.0004
	std	*1.43E-08*	*0.00E+00*	*4.02E-02*	*5.13E+02*		
S13	mean	$\mathbf{8.3702E+01}$	$\mathbf{1.1807E-19}$	$9.2391E+01$	$1.3549E-13$	0.5877	0.0004
	std	*2.31E+01*	*2.64E-19*	*6.86E+01*	*1.05E-13*		
S14	mean	$\mathbf{5.8233E-11}$	$6.7236E+01$	$1.8383E-07$	$\mathbf{5.0168E+01}$	0.0004	0.6886
	std	*1.30E-10*	*2.28E+01*	*5.93E-08*	*1.23E+01*		
S15	mean	$\mathbf{9.8884E-11}$	$\mathbf{1.0051E-11}$	$7.0465E-01$	$5.0225E-02$	0.0004	0.0101
	std	*2.21E-10*	*2.25E-11*	*3.86E+00*	*2.74E-01*		
S16	mean	$\mathbf{4.2817E+01}$	$2.5869E+00$	$5.2315E+01$	$\mathbf{2.4211E+00}$	0.0008	0.2480
	std	*1.71E+00*	*4.75E+00*	*1.76E+01*	*6.11E+00*		
S17	mean	$\mathbf{1.0165E-13}$	$3.4240E+00$	$1.3575E-09$	$\mathbf{5.6820E-02}$	0.0004	0.0004
	std	*2.27E-13*	*9.88E-01*	*9.42E-10*	*1.70E-01*		
S18	mean	**8.8423E+01**	$\mathbf{6.3638E-04}$	$1.0549E+02$	$2.1447E+03$	0.0152	0.0004
	std	*1.35E+01*	*1.63E-12*	*9.51E+01*	*5.68E+02*		

complicate and difficult problems due to its simplicity and efficiency. ACoSDE [35] is a variant of DE based on an adaptive coordinate system. Also, we compare GFMFDE with various adaptive parameter DEs, such as JADE [42], L-SHADE [43], and L-MPEDE [44].

The benchmark problems used in the experiments are composed of three different benchmark suits. The detailed description is in Table SI in supplementary materials. First, we design the benchmark suite consisting of 9 two-task problems based on the computational complexity of tasks. These problems are combinations of functions with different complexity degrees and dimensions. The second benchmark suit includes the CEC'2017 benchmark problems in [45], each of which is consisted of two tasks with different Spearman's rank correlation and the degree of intersection between the search spaces. The third suite used in the experiments is the CEC'2019 benchmark problems that contain ten much more complex benchmark problems and the details of them available in the website.[4] The fourth benchmarks are reinforce learning tasks for balancing two poles in different environments. We will open the source of these benchmarks.

B. Efficiency Analysis

First, to showcase the improved performance of the proposed method attributed to our intertask knowledge transfer strategy, the original version of MFDE is viewed as baseline, where the intertask mutation is implemented in the original coordinate system. The experimental results are reported in Table SIII, and parts of results on CEC'2017 is listed in Table II.

As shown in Table SIII, against MFDE, GFMFDE obtains good results on most of the benchmarks, no matter different-dimension, nonintersected or complex problems. Our method can tackle the rotated function landscapes and transfer the information the population conveys in the intermediate subspaces. It is seen that on problems S19–28 composed of problems with rotated function landscapes and enormous local optimal valleys, GFMFDE can outperform MFDE, which can be attributed to the low-dimension intertask transfer strategy.

Next, we investigated the performance of two novel mutations, centering and bias mutations. To this end, three variants of GFMFDE are performed. The first method, GFMFDE combines two proposed mutations during the search process. The variant, GFMFDE with centering mutation, only employs the centering mutation. The third variant, GFMFDE with bias mutation, only turns on the bias mutation. Entire results are reported in Table SIV in supplementary materials, and the results on CEC'2017 are listed in Table III.

According to the results shown in Table SIV, two mutations in the active coordinate system have their own advantages. By observing the data in Tables II and III together, compared with MFDE, these two mutations can achieve an improved overall performance on problems S10–S18. Compared with bias mutation, centering mutation performs well on problems with noninsert and low-similar benchmarks, such as S12, S15, and S18. In contrast, bias mutation achieves a better performance than centering mutation on problems with benchmarks with full-intersection optimums. The results of GFMFDE in Table III confirm that the embedding strategy of centering and bias mutations can outperform the methods using only one of them.

C. Coefficient Sensitivity Analyses

In the proposed knowledge transfer strategy, the dimension of the intermediate subspace is an important user-defined parameter on which the search performance might rely. To achieve the balance of information reservation and the consistency of coordinate systems, this parameter and is studied by an empirical experiment.

[4]Available in http://www.bdsc.site/websites/MTO/index.html.

TABLE III
ABLATION EXPERIMENT BY COMPARING GFMFDE WITH TWO VARIANTS OF GFMFDE, WHICH ONLY USE CENTERING MUTATION AND BIAS MUTATION, RESPECTIVELY. *MEAN* AND *STD* REPRESENTS THE AVERAGED BEST FUNCTION VALUE AND STANDARD DEVIATION OF THE BENCHMARK PROBLEMS S10-18 OVER 30 INDEPENDENT RUNS. THE BEST ONES ARE TYPED IN *BOLD*

Pro.		GFMFDE		Centering Mutation		Bias Mutation	
		T1	T2	T1	T2	T1	T2
S10	mean	**0.00E+00**	**0.00E+00**	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	std	*0.00E+00*	*0.00E+00*	*0.00E+00*	*0.00E+00*	*0.00E+00*	*0.00E+00*
S11	mean	1.15E-13	**0.00E+00**	**1.00E-13**	0.00E+00	1.95E-12	0.00E+00
	std	*2.56E-13*	*0.00E+00*	*2.18E-13*	*0.00E+00*	*4.35E-12*	*0.00E+00*
S12	mean	1.05E-08	**6.36E-04**	**2.28E-09**	6.36E-04	2.28E-07	6.36E-04
	std	*1.43E-08*	*0.00E+00*	*3.91E-09*	*6.51E-12*	*1.68E-07*	*0.00E+00*
S13	mean	**8.37E+01**	**1.18E-19**	9.36E+01	3.04E-19	9.75E+01	2.05E-17
	std	*2.31E+01*	*2.64E-19*	*2.11E+01*	*6.79E-19*	*1.25E+01*	*4.58E-17*
S14	mean	**5.82E-11**	**6.72E+01**	8.49E-11	8.39E+01	5.98E-11	6.77E+01
	std	*1.30E-10*	*2.28E+01*	*1.90E-10*	*9.57E-01*	*1.34E-10*	*2.26E+01*
S15	mean	9.89E-11	1.01E-11	**3.41E-11**	**0.00E+00**	8.48E-11	0.00E+00
	std	*2.21E-10*	*2.25E-11*	*1.90E-10*	*0.00E+00*	*7.62E-11*	*0.00E+00*
S16	mean	4.28E+01	2.59E+00	4.29E+01	**0.00E+00**	**4.19E+01**	5.66E+00
	std	*1.71E+00*	*4.75E+00*	*1.17E+00*	*0.00E+00*	*2.66E+00*	*8.66E+00*
S17	mean	**1.02E-13**	3.42E+00	3.23E-13	3.38E+00	1.19E-13	**2.47E+00**
	std	*2.27E-13*	*9.88E-01*	*7.22E-13*	*7.18E-01*	*2.66E-13*	*1.54E+00*
S18	mean	**8.84E+01**	**6.36E-04**	9.62E+01	6.36E-04	1.17E+02	6.36E-04
	std	*1.35E+01*	*1.63E-12*	*1.46E+01*	*4.88E-12*	*1.38E+01*	*2.44E-11*

TABLE IV
AVERAGED RANKS OF GFMFDE, MFDE-RS, AND MFDE-AE IN TERMS OF THE AVERAGED BEST FUNCTION OBJECTIVE VALUES

Algorithm	Task 1	Task 2
GFMFDE	**1.2143**	**1.1786**
MFDE-RS	2.3929	2.4286
MFDE-AE	2.3929	2.3929

The results shown in Fig. 3 suggest that a middle-dimension subspace is particularly enough for the problems with distinct function landscapes, for example, S11–S15. For problems with high similarity and the same optimum, like S10, relatively high-dimension subspace is appropriated. In this case, the subpopulations communicating with a high-fidelity channel can serve as a large group to search a wide region. But GFMFDE with middle-dimension subspace does not suffer much loss in search performance compared with that with full-dimension space. Another interesting phenomenon is that GFMFDE with the 1-D subspace can also provide comparative search performance on S11–S15 compared with those with higher dimension subspace. This is mainly because a rough direction extracted from the subpopulation can bias properly the subpopulation of another task, playing a role on sharing the information between multiple tasks. We take 50% dimensions of full space in our method because under this parameter setting the method can achieve a sufficient performance on most benchmark problems.

D. Comparison With Other Transfer Schemes

In the literature, there exist several reported transfer schemes for evolutionary multitasking as mentioned in Section II-A. In this part of the experiments, the proposed method is compared with the variants of MFDE, which employ different intertask knowledge transfer schemes and incorporate them into the basic structure of the original MFDE. The entire results are listed in Table SV in supplementary materials.

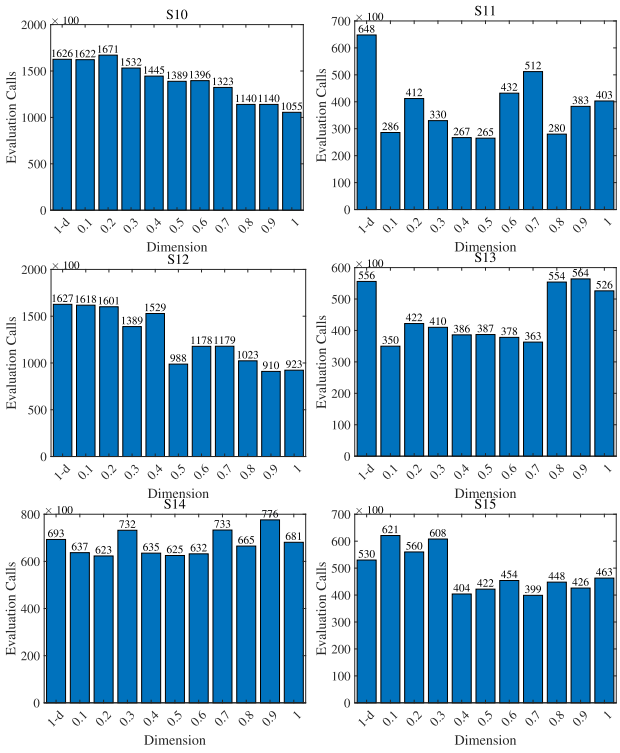


Fig. 3. Convergence Speeds of MFDE, MFDE-RS, MFDE-AE, and GFMFDE which have different intertask transfer methods on problems S10–S15 over 30 independent runs. The *y*-axis is the number of function evaluation calls and the *x*-axis is the ratio of the dimension of subspace to the dimension of full space. The number of evaluation calls on each benchmark is record after both objective function values are less than the averaged best objective function values.

As shown in rows 1–9 of Table SV, GFMFDE obtains relatively good results on the problems with identified solutions in the unified representation space, compared with MFDE-RS and MFDE-AE, for example, on problems S6–9 with different optimums or complex tasks. That is because the proposed method has the ability to transfer the significant features of

TABLE V
PERFORMANCE COMPARISON OF GFMFDE, MFEA2, MFEA, MFPSO, SREMTO, DE, AND ACoSDE ON TEST SUITS 10–18. THE MEAN AND STD VALUES OF OBJECTIVE VALUES AFTER MAXIMUM FES OVER 30 RUNS. THE BEST RESULTS ARE TYPED IN *BOLD*

Pro.		GFMFDE		MFEA2		MFEA		MFPSO		SREMTO	
		T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
S10	mean	**0.00E+00**	**0.00E+00**	9.07E-03	5.03E+01	5.51E-03	1.67E+01	1.07E+00	9.60E+02	8.25E-03	3.24E+01
	std	*0.00E+00*	*0.00E+00*	*8.50E-03*	*4.86E+01*	*9.11E-03*	*2.83E+01*	*4.72E+00*	*4.22E+03*	*7.95E-03*	*2.62E+01*
S11	mean	**2.66E-15**	**0.00E+00**	2.05E+00	7.08E+01	1.74E+00	5.73E+01	1.27E+00	1.21E+03	2.89E+00	5.01E+01
	std	*5.59E-15*	*0.00E+00*	*1.11E-01*	*1.88E+01*	*5.23E-01*	*2.69E+01*	*4.73E+00*	*5.28E+03*	*5.70E-01*	*1.92E+01*
S12	mean	**1.31E-12**	**6.36E-04**	1.71E+00	1.89E+03	1.72E+00	2.41E+03	7.63E+00	4.22E+03	3.86E+00	6.06E+03
	std	*4.43E-12*	*3.56E-12*	*3.36E-01*	*3.59E+02*	*3.52E-01*	*3.70E+02*	*8.69E+00*	*5.13E+03*	*5.81E-01*	*8.48E+02*
S13	mean	**6.91E+01**	9.35E-15	1.11E+02	3.25E-03	1.38E+02	1.02E-02	3.49E+02	4.56E+03	2.01E+02	**2.52E-16**
	std	*2.66E+01*	*2.78E-14*	*2.52E+01*	*1.31E-03*	*3.01E+01*	*3.24E-03*	*1.76E+02*	*3.28E+02*	*4.13E+01*	*4.87E-16*
S14	mean	**1.91E-01**	**4.10E+01**	1.47E+00	1.01E+02	1.68E+00	1.29E+02	4.29E+00	9.33E+05	2.22E+00	8.05E+01
	std	*4.79E-01*	*1.56E+00*	*8.65E-01*	*6.20E+01*	*4.90E-01*	*3.74E+01*	*2.80E+00*	*4.17E+06*	*4.05E-01*	*2.62E+01*
S15	mean	5.84E-01	**2.45E-01**	1.91E+00	2.35E+00	1.68E+00	2.20E+00	**2.87E-01**	3.53E-01	3.28E+00	2.62E+00
	std	*8.64E-01*	*4.25E-01*	*4.32E-01*	*1.65E-01*	*5.90E-01*	*6.35E-01*	*9.07E-01*	*9.16E-01*	*6.71E-01*	*6.56E-01*
S16	mean	**3.90E+01**	**6.27E+00**	1.35E+02	1.29E+02	1.06E+02	9.65E+01	7.01E+07	1.11E+03	7.33E+01	4.56E+01
	std	*1.79E+01*	*1.30E+01*	*3.03E+01*	*2.39E+01*	*2.77E+01*	*3.88E+01*	*3.13E+08*	*4.24E+03*	*3.49E+01*	*1.94E+01*
S17	mean	**1.20E-12**	**1.60E+00**	4.27E-03	1.93E+01	1.10E-02	1.66E+01	8.33E-01	1.60E+01	6.53E-03	1.24E+01
	std	*5.35E-12*	*1.06E+00*	*3.06E-03*	*2.05E+00*	*7.90E-03*	*3.51E+00*	*5.72E-01*	*9.36E+00*	*6.55E-03*	*1.51E+00*
S18	mean	**8.84E+01**	**6.36E-04**	1.11E+02	1.90E+03	1.12E+02	2.03E+03	2.31E+03	7.68E+03	2.10E+02	6.02E+03
	std	*6.59E+01*	*1.16E-11*	*1.53E+01*	*6.67E+02*	*1.95E+01*	*5.07E+02*	*3.76E+03*	*4.03E+03*	*4.67E+01*	*1.10E+03*

TABLE VI
AVERAGED RANKS OF GFMFDE, MFEA2, MFEA, MFPSO, SREMTO, DE, AND ACoSDE IN TERMS OF THE AVERAGED BEST FUNCTION OBJECTIVE VALUES

Algorithm	Task 1	Task 2
GFMFDE	**1.3571**	**1.6071**
MFEA2	3.3571	3.1429
MFEA	3.8214	3.3214
MFPSO	6.1071	4.0357
SREMTO	4.1786	4.0357
DE	3.3929	4.9643
ACOSDE	5.7857	4.5000

the function landscapes with different dimensions and modalities. Particularly, MFDE-RS and GFMFDE achieved similar performance on problems S1–3. It is because that they apply a linear combination to the decision variables before synthesizing the target vectors. Specifically, MFDE-RS reserved the diversity in the population by random combining the decision variables of two tasks, which is equivalent to multiplying target vector by an elementary matrix. Compared to MFDE-RS, GFMFDE first projects the solutions into the low-dimension subspace, then, synthesizing the intertask solutions to generate mutation vectors. Besides, we ensemble the centering and bias mutation operations in our scheme, so it can benefit from the characteristics of two operations that both directional and distributional information can be identified and propagated across multiple tasks. Although MFDE-AE has the ability to tackle the function landscapes with different dimensions, it is actually sensitive to noise because it relies on the order of individuals. Therefore, MFDE-AE performs better than MFDE-RS does on problems S4-6 but worse than GFMFDE.

As depicted in rows 10–28 of Table SV, overall, GFMFDE performs well on problems S10–18 compared with the other transfer schemes. Especially, on problems S16–18, MFDE-AE achieves better performance compared with MFDE-RS. That is because MFDE-AE is not sensitive to the bias between the function landscapes. Our method is superior MFDE-AE and MFDE-RS on these complex problems S16–18. It is mainly because our method can identify the common modality of the function landscapes and transfer the characteristics of the current population from one task to another. On the CEC2019 benchmarks of S19–S28, GFMFDE has exhibited the advantage in solving complex benchmark problems compared with MFDE-AE and MFDE-RS. GFMFDE performs better on 8 out of 10 problems compared with MFDE-RS. It is significantly superior to MFDE-AE on 8 out of 10. Table IV provides a summary and comparison of the transfer schemes in terms of the averaged rank of the objective function values.

E. Comparison on Other Multifactorial Optimization Algorithms and Single-objective Algorithms

For demonstrating the performance of the proposed method compared, it is compared with four methods proposed in the last years and variant versions of single-objective DE. In this section, we focus on comparing our method with the compared algorithms in terms of optimization accuracy and convergence speed.

The experimental results of CEC'2017 are listed in Table V, and entire results are shown in Table SVI in supplementary materials. According to the reported results in Tables SVI, GFMFDE outperforms MFEA on 24 out of 28 benchmark problems, while it is also superior to MFEA2 on 23 out of 28 problems, which is currently a state-of-the-art algorithm proposed in [7]. As depicted in Table SVI, our algorithm is good at solving the rotated objective functions and is significantly superior to MFEA2 and MFEA on all benchmark problems S10–18. Even compared with SREMTO which employs a hybrid search operation of DE and GA, GFMFDE can achieve a better performance on 21 out of 28 benchmarks. By employing the low-dimension coordinate system and composite intertask trial vector strategies, the proposed GFMFDE can improve the performance of solving complex problems by

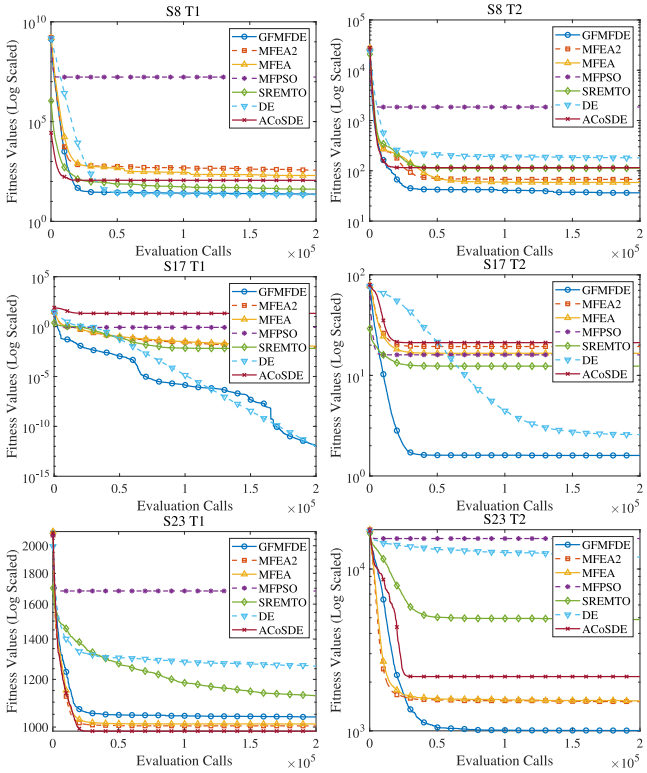


Fig. 4. Convergence trends of GFMFDE compared with the other multifactorial algorithms MFEA2, MFEA, MFPSO, and SREMTO, in addition, the single-objective DE and ACoSDE on benchmark problems S8, S17, and S23. The y-axis is the log-scale function values and the x-axis represents the number of function evaluations calls.

emphasizing the essential commonality and complementary of tasks. Thus, GFMFDE still performs well with a fixed intertask random mating probability.

According to the graphical demonstration of convergence trends in Fig. 4, DEs (GFMFDE and DE) and PSOs (MFPSO) have a considerable convergence speed on problems S10, S18, and S15, however, this may cause the permutation of population. Single DE performs well on unimodal and simple multimodal objective functions, such as S8T1 and S17T1, but suffers a significant reduction of performance on complex problems, such as S8T2 and S17T2. ACoSDE converges faster than DE, due to the adaptive coordinate system. While MFPSO converges fast at the begin of evolution, the diversity in population reduces quickly so that the accuracy of optimal solutions is relatively inferior. Our method can perform well on these problems since we employ an innovative mutation operation, called DE/current-to-center/2 to reserve the necessary diversity in the population. Table VI summarizes the overall performance of the compared algorithms. Generally, GFMFDE is ranked first in terms of the averaged ultimate objective function values.

When compared with the current parameter adaptive methods, JADE, L-SHADE, and L-MPEDE, our method can exhibit superior performance as shown in Table VII, remembering that GFMFDE only employs a vanilla DE/rand/1/bin with fixed parameters. This confirms the power of intertask knowledge transfer to assist the algorithm in solving the listed

TABLE VII
COMPARISON OF GFMFDE AND SINGLE-OBJECTIVE DIFFERENTIAL EVOLUTIONS

□ vs GFMFDE		T_1	T_2
JADE	+ (better)/- (worse)/≈ (no sig.)	4/19/5	4/21/3
L-SHADE	+ (better)/- (worse)/≈ (no sig.)	9/14/5	7/18/3
L-MPEDE	+ (better)/- (worse)/≈ (no sig.)	6/19/3	5/22/1
ACoSDE	+ (better)/- (worse)/≈ (no sig.)	2/19/6	0/24/4
DE	+ (better)/- (worse)/≈ (no sig.)	4/13/10	0/23/5

TABLE VIII
RESULTS OF TWO-TASK DOUBLE CART-POLE BALANCING PROBLEMS

Task	T_1, T_2	T_1, T_3	T_2, T_3	T_1, T_2, T_3
l_s(m)	(0.60, 0.65)	(0.60, 0.70)	(0.65, 0.70)	(0.6,0.65,0.7)
GFMFDE	90%, 90%	93%, 17%	40%, 20%	100%, 100%, 100%
MFEA-II	30%, 27%	30%, 7%	27%, 27%	47%, 37%, 33%
MFEA	35%, 25%	45%, 5%	10%, 5%	37%, 37%, 30%
SREMTO	36%, 36%	10%, 3%	13%, 7%	90%, 90%, 90%
MFDE	55%, 45%	40%, 10%	15%, 0%	100%, 100%, 90%
DE	35%, 5%	35%, 0%	5%, 0%	35%, 5%, 20%

28 benchmark problems. The detailed results are shown in Table SIII in supplementary materials.

F. Practical Application

In this part of experiment, we set up an experimental environment of simple reinforce learning referring to [7]. The goal of this experiment is to balance double poles connected to a cart via the hinges [46]. We showcase the efficacy of GFMFDE in tackling practical tasks.

In this experiment, we mainly focus on the Markovian case of double cart-pole balancing task, where the velocity of cart is given as input to the controller. Namely, the controller has totally six inputs, including the cart position x, the cart velocity $\dot{x}$, the rotation angle of longer pole θ_1, the rotation angle of shorter pole θ_2, the angular velocity of longer pole $\dot{\theta}_1$, the angular velocity of shorter pole $\dot{\theta}_2$. A simple neural network with six inputs, ten hidden units, and one output is used as the controller. We encode the weights of every controller into a individual with 71 real variables. One pole is fixed to 1.0 m, while the length of shorter one is sequentially set to 0.6, 0.65, 0.7 m. The other parameter settings and the environment settings are referred to [7] and [47].

According to the results in Table VIII, it is observed that the intertask knowledge transfer can help MFO methods solve the balancing tasks at a higher success rate compared with the single-task DE which only successes in the rate of 35% on simple task T_1, 5% on T_2, and fails on T_3. The proposed GFMFDE outperforms MFEA-II, MFEA, MFDE, MTO-AE, and SREMTO. Especially, in the problems consisted of simple and hard tasks, like (T_1, T_3), GFMFDE can conduct a positive transfer between two tasks and solve them with a high success rate. GFMFDE can solve T_3 by the chance of 20%, it also achieves a slightly improvement on T_1 with 93% success rate compared with 90% in the case of (T_1,T_2). When solving three-task problem (T_1, T_2, T_3), the performances of all MFOs on three tasks are improved. GFMFDE achieves a good performance on this problem, which highlights the efficiency of the proposed knowledge transfer strategy.

V. Conclusion

This work represented a first step toward the adaptive generation of the intertask active coordinate system where the solutions from the distinct but related tasks can share more commonalities. The negative transfer can be eased by eliminating the variance caused by the intertask trial vector generation within a fixed coordinate inconsistent with the principle components of the sampling solutions from distinct tasks. Specifically, an active low-dimension coordinate system was drawn from the subspace distribution established upon the one-parameter geodesic flow. Then, two intertask trial vector generation strategies implemented on this active coordinate system were discussed. In contrast to the fixed coordinate systems, the active coordinate systems generated by synthesizing the distributional information of the task-specific subpopulations can identify the common modality of different tasks' function landscapes. In addition, for the aims of making the intertask knowledge transfer more efficient, a framework to embed various coordinate systems (the eigencoordinate system or the original coordinate system) was proposed. A comprehensive numerical experiment was conducted to validate the efficacy of the proposed intertask knowledge transfer strategy compared with the state-of-the-art evolutionary multitasking methods.

In this work, the first attempt of intertask transfer in the corresponding subspace to alleviate the inconsistency among the principle components of the different tasks' solutions presents the effectiveness of this strategy. Although that, it is essential to study the efficiency and the effectiveness of applying this strategy on many-task problems and complicate real-world applications.

References

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Know. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[2] Y. Zhang and Q. Yang, "An overview of multi-task learning," *Nat. Sci. Rev.*, vol. 5, pp. 30–43, Jan. 2018.

[3] K. Swersky, J. Snoek, and R. P. Adams, "Multi-task Bayesian optimization," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2013, pp. 2004–2012.

[4] A. Gupta, Y.-S. Ong, and L. Feng, "Insights on transfer optimization: Because experience is the best teacher," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 1, pp. 51–64, Feb. 2018.

[5] A. Gupta,Y. S. Ong, B. Da, L. Feng, and S. D. Handoko "Landscape synergy in evolutionary multitasking," *Proc. IEEE Conf. Evol. Comput. (CEC)*, Vancouver, BC, Canada, 2016, pp. 3076–3083.

[6] A. Gupta, Y.-S. Ong, and L. Feng, "Multifactorial evolution: Toward evolutionary multitasking," *IEEE Trans. Evol. Comput.*, vol. 20, no. 3, pp. 343–357, Jun. 2016.

[7] K. K. Bali, Y.-S. Ong, A. Gupta, and P. S. Tan, "Multifactorial evolutionary algorithm with online transfer parameter estimation: MFEA-II," *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 69–83, Feb. 2020, doi: 10.1109/TEVC.2019.2906927.

[8] A. Gupta, Y.-S. Ong, L. Feng, and K.-C. Tan, "Multiobjective multifactorial optimization in evolutionary multitasking," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1652–1665, Jul. 2017.

[9] H. Maaranen, K. Miettinen, and A. Penttinen, "On initial populations of a genetic algorithm for continuous optimization problems," *J. Global Optimiz.*, vol. 37, no. 3, p. 405, Jul. 2006.

[10] D. K. Saxena, A. Sinha, J. A. Duro, and Q. Zhang, "Entropy-based termination criterion for multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 485–498, Aug. 2016.

[11] S. Jiang and S. Yang, "Evolutionary dynamic multiobjective optimization: Benchmarks and algorithm comparisons," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 198–211, Jan. 2017.

[12] Y. Sun, B. Xue, M. Zhang, and G. G. Yen, "A new two-stage evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 748–761, Oct. 2019, doi: 10.1109/TEVC.2018.2882166.

[13] K. C. Tan, Y. H. Chew, and L. H. Lee, "A hybrid multiobjective evolutionary algorithm for solving vehicle routing problem with time windows," *Comput. Optim. Appl.*, vol. 34, no. 1, p. 115, Oct. 2005.

[14] Z. Zhu, J. Xiao, S. He, Z. Ji, and Y. Sun, "A multi-objective memetic algorithm based on locality-sensitive hashing for one-to-many-to-one dynamic pickup-and-delivery problem," *Inf. Sci.*, vol. 329, pp. 73–89, Feb. 2016.

[15] M. Mavrovouniotis, F. M. Muller, and S. Yang, "Ant colony optimization with local search for dynamic traveling salesman problems," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1743–1756, Jul. 2017.

[16] K. Tang, Y. Mei, and X. Yao, "Memetic algorithm with extended neighborhood search for capacitated arc routing problems," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 1151–1166, Oct. 2009.

[17] Y.-L. Li, Z.-H. Zhan, Y.-J. Gong, W.-N. Chen, J. Zhang, and Y. Li, "Differential evolution with an evolution path: A DEEP evolutionary algorithm," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 1798–1810, Sep. 2015.

[18] J. Wang, G. Liang, and J. Zhang, "Cooperative differential evolution framework for constrained multiobjective optimization," *IEEE Trans. Cybern.*, vol. 49, no. 6, pp. 2060–2072, Jun. 2019.

[19] J. Ding, C. Yang, Y. Jin, and T. Chai, "Generalized multitasking for evolutionary optimization of expensive problems," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 44–58, Feb. 2019.

[20] L. Feng *et al.*, "Evolutionary multitasking via explicit autoencoding," *IEEE Trans. Cybern.*, vol. 49, no. 9, pp. 3457–3470, Sep. 2019.

[21] Y. Wang, H.-X. Li, T. Huang, and L. Li, "Differential evolution based on covariance matrix learning and bimodal distribution parameter setting," *Appl. Soft Comput.*, vol. 18, pp. 232–247, May 2014.

[22] S.-M. Guo and C.-C. Yang, "Enhancing differential evolution utilizing eigenvector-based crossover operator," *IEEE Trans. Evol. Comput.*, vol. 19, no. 1, pp. 31–49, Feb. 2015.

[23] K. K. Bali, A. Gupta, L. Feng, Y. S. Ong, and T. P. Siew, "Linearized domain adaptation in evolutionary multitasking," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, San Sebastian, Spain, 2017, pp. 1295–1302.

[24] Y.-W. Wen and C.-K. Ting, "Parting ways and reallocating resources in evolutionary multitasking," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, San Sebastian, Spain, 2017, pp. 2404–2411.

[25] L. Zhou *et al.*, "Toward adaptive knowledge transfer in multifactorial evolutionary computation," *IEEE Trans. Cybern.*, early access, Mar. 6, 2020, doi: 10.1109/TCYB.2020.2974100.

[26] L. Feng *et al.*, "An empirical study of multifactorial PSO and multifactorial DE," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, San Sebastian, Spain, 2017, pp. 921–928.

[27] Y.-W. Wen and C.-K. Ting, "Learning ensemble of decision trees through multifactorial genetic programming," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, 2016, pp. 5293–5300.

[28] L. Feng *et al.*, "Solving generalized vehicle routing problem with occasional drivers via evolutionary multitasking," *IEEE Trans. Cybern.*, early access, Dec. 23, 2020, doi: 10.1109/TCYB.2019.2955599.

[29] L. Feng *et al.*, "Explicit evolutionary multitasking for combinatorial optimization: A case study on capacitated vehicle routing problem," *IEEE Trans. Cybern.*, early access, Mar. 4, 2020, doi: 10.1109/TCYB.2019.2962865.

[30] R. Liaw and C. Ting, "Evolutionary many-tasking based on biocoenosis through symbiosis: A framework and benchmark problems," in *Proc. IEEE Conf. Evol. Comput.*, San Sebastian, Spain, 2017, pp. 2266–2273.

[31] R.-T. Liaw and C.-K. Ting, "Evolutionary manytasking optimization based on symbiosis in biocoenosis," in *Proc. AAAI*, 2019, pp. 4295–4303.

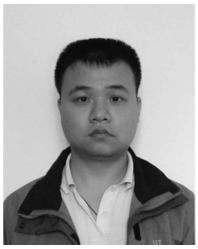
[32] Y. Chen, J. Zhong, L. Feng, and J. Zhang, "An adaptive archive-based evolutionary framework for many-task optimization," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 4, no. 3, pp. 369–384, Jun. 2020, doi: 10.1109/TETCI.2019.2916051.

[33] D. Liu, S. Huang, and J. Zhong, "Surrogate-assisted multi-tasking memetic algorithm," in *Proc. IEEE Conf. Evol. Comput.*, Rio de Janeiro, Brazil, 2018, pp. 1–8.

[34] K. K. Bali, A. Gupta, Y. Ong, and P. S. Tan, "Cognizant multitasking in multiobjective multifactorial evolution: MO-MFEA-II," *IEEE Trans. Cybern.*, early access, Apr. 23, 2020, doi: 10.1109/TCYB.2020.2981733.

[35] Z.-Z. Liu, Y. Wang, S. Yang, and K. Tang, "An adaptive framework to tune the coordinate systems in nature-inspired optimization algorithms," *IEEE Trans. Cybern.*, vol. 49, no. 4, pp. 1403–1416, Apr. 2019.

[36] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evol. Comput.*, vol. 9, no. 2, pp. 159–195, Jun. 2001.
[37] N. Hansen. (2009). *Variable Metrics in Evolutionary Computation*. [Online]. Available: https://www.lri.fr/hansen/hansen-habil-manu.pdf
[38] Q. Zhang, A. Zhou, and Y. Jin, "RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm," *IEEE Trans. Evol. Comput.*, vol. 12, no. 1, pp. 41–63, Feb. 2008.
[39] K. A. Gallivan, A. Srivastava, X. Liu, and P. Van Dooren, "Efficient algorithms for inferences on Grassmann manifolds," in *Proc. IEEE Workshop Stat. Signal Proc.*, 2003, pp. 1–4.
[40] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer-Verlag, 2006.
[41] X. Zheng, A. K. Qin, M. Gong, and D. Zhou, "Self-regulated evolutionary multitask optimization," *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 16–28, Feb. 2020.
[42] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.
[43] R. Tanabe and A. S. Fukunaga, "Improving the search performance of shade using linear population size reduction," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Beijing, China, 2014, pp. 1658–1665.
[44] G. Wu, R. Mallipeddi, P. Suganthan, R. Wang, and H. Chen, "Differential evolution with multi-population based ensemble of mutation strategies," *Inf. Sci.*, vol. 329, no. 1, pp. 329–345, Feb. 2016.
[45] B. Da *et al.*, "Evolutionary multitasking for single-objective continuous optimization: Benchmark problems, performance metrics and baseline results," 2017. [Online]. Available: arXiv:1706.03470.
[46] F. Gomez, J. Schmidhuber, and R. Miikkulainen, "Accelerated neural evolution through cooperatively coevolved synapses," *J. Mach. Learn. Res.*, vol. 9, pp. 937–965, May 2008.
[47] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evol. Comput.*, vol. 10, no. 2, pp. 99–127, Mar. 2002.



Zedong Tang received the B.Eng. and Ph.D. degrees from Xidian University, Xi'an, China, in 2014 and 2020, respectively.

He has been working as a Lecturer with Xidian University since 2020. His current research interests include computational intelligence and machine learning.

Maoguo Gong (Senior Member, IEEE) received the B.S. degree (First Class Hons.) in electronic engineering and the Ph.D. degree in electronic science and technology from Xidian University, Xi'an, China, in 2003 and 2009, respectively.

Since 2006, he has been a Teacher with Xidian University. In 2008 and 2010, he was promoted to Associate Professor and as a Full Professor, respectively, both with exceptive admission. His research interests are in the area of computational intelligence with applications to optimization, learning, data mining, and image understanding.

Dr. Gong received the prestigious National Program for the Support of Top-Notch Young Professionals from the Central Organization Department of China, the Excellent Young Scientist Foundation from the National Natural Science Foundation of China, and the New Century Excellent Talent in University from the Ministry of Education of China. He is an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.

Yue Wu (Member, IEEE) received the B.Eng. and Ph.D. degrees from Xidian University, Xi'an, China, in 2011 and 2016, respectively.

Since 2016, he has been a Teacher with Xidian University, where he is currently an Associate Professor. He has authored or coauthored more than 40 papers in refereed journals and proceedings. His research interests include computer vision and computational intelligence.

A. K. Qin (Senior Member, IEEE) received the B.Eng. degree from Southeast University, Nanjing, China, in 2001, and the Ph.D. degree from Nanyang Technology University, Singapore, in 2007.

From 2007 to 2017, he was with the University of Waterloo, Waterloo, ON, Canada; INRIA, Grenoble-Rhône-Alpes, France; and RMIT University, Melbourne, VIC, Australia. In 2017, he joined Swinburne University of Technology, Melbourne, as an Associate Professor, where he is currently the Director of Swinburne Intelligent Data Analytics Laboratory, the Program Lead of Swinburne Data Science Research Institute, and the Leader of Machine Learning and Intelligent Optimization Research Group. His major research interests include machine learning, evolutionary computation, computer vision, remote sensing, services computing, and pervasive computing.

Dr. Qin was a recipient of the 2012 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award and the Overall Best Paper Award at the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems in 2014. He is now the Vice-Chair of the IEEE Neural Networks Technical Committee and the IEEE Emergent Technologies Task Forces on "Collaborative Learning and Optimization" and "Multitask Learning and Multitask Optimization."

Kay Chen Tan (Fellow, IEEE) received the B.Eng. degree (Hons.) and the Ph.D. degree from the University of Glasgow, Glasgow, U.K., in 1994 and 1997, respectively.

He is a Full Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong, China, and also with the City University of Hong Kong, Shenzhen Research Institute, Shenzhen, China. He has published over 200 refereed articles and six books.

Prof. Tan is the Editor-in-Chief of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION. He was the Editor-in-Chief of the *IEEE Computational Intelligence Magazine* from 2010 to 2013, and currently serves on the Editorial Board Member of over 20 journals. He is an Elected Member of IEEE CIS AdCom from 2017 to 2019.