仿射 Affine Transformation-Enhanced Multifactorial Optimization for Heterogeneous Problems 异构问题

Xiaoming Xue, Kai Zhang, *Member, IEEE*, Kay Chen Tan, *Fellow, IEEE*, Liang Feng, Jian Wang, *Member, IEEE*, Guodong Chen, Xinggang Zhao, Liming Zhang, and Jun Yao

***Abstract*—Evolutionary multitasking (EMT) is a newly emerging research topic in the community of evolutionary computation, which aims to improve the convergence characteristic across multiple distinct optimization tasks simultaneously by triggering knowledge transfer among them. Unfortunately, most of the existing EMT algorithms are only capable of boosting the optimization performance for homogeneous problems which explicitly share the same (or similar) fitness landscapes. Seldom efforts have been devoted to generalize the EMT for solving heterogeneous problems. A few preliminary studies employ domain adaptation techniques to enhance the transferability between two distinct tasks. However, almost all of these methods encounter a severe issue which is the so-called degradation of intertask mapping. Keeping this in mind, a novel rank loss function for acquiring a superior intertask mapping is proposed in this article. In particular, with an evolutionary-path-based representation model for optimization instance, an analytical solution of affine transformation for bridging the gap between two distinct problems is mathematically derived from the proposed rank loss function. It is worth mentioning that the proposed mapping-based transferability enhancement technique can be seamlessly embedded into an EMT paradigm. Finally, the efficacy of our proposed method against several state-of-the-art EMTs is verified experimentally on a number of synthetic multitasking and many-tasking benchmark problems, as well as a practical case study.**

***Index Terms*—Affine transformation, domain adaptation, evolutionary multitasking (EMT), heterogeneous problems, multifactorial optimization (MFO).**

Manuscript received December 23, 2019; revised May 13, 2020 and August 17, 2020; accepted October 27, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 51722406, Grant 52074340, and Grant 51874335; in part by the Shandong Provincial Natural Science Foundation under Grant JQ201808; in part by the Fundamental Research Funds for the Central Universities under Grant 18CX02097A; in part by the Major Scientific and Technological Projects of CNPC under Grant ZD2019-183-008; in part by the Science and Technology Support Plan for Youth Innovation of University in Shandong Province under Grant 2019KJH002; in part by the National Science and Technology Major Project of China under Grant 2016ZX05025001-006; and in part by 111 Project under Grant B08028. This article was recommended by Associate Editor M. Zhang. *(Corresponding author: Kai Zhang.)*

Xiaoming Xue, Kai Zhang, Guodong Chen, Xinggang Zhao, Liming Zhang, and Jun Yao are with the School of Petroleum Engineering, China University of Petroleum (East China), Qingdao 266580, China (e-mail: xminghsueh@gmail.com; zhangkai@upc.edu.cn; upcjellychen@163.com; xinggang_zhao@yeah.net; zhangliming@upc.edu.cn; yaojunhdpu@126.com).

Kay Chen Tan is with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: kaytan@cityu.edu.hk).

Liang Feng is with the College of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: liangf@cqu.edu.cn).

Jian Wang is with the College of Science, China University of Petroleum (East China), Qingdao 266580, China (e-mail: wangjiannl@upc.edu.cn).

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.2020.3036393.

Digital Object Identifier 10.1109/TCYB.2020.3036393

I. Introduction

INSPIRED by the principles of natural selection and genetics in the Darwinian theory of evolution [1], evolutionary algorithms (EAs) are population-based methods that mimic the biological evolution processes to explore promising regions of the given search space. Over the decades, EAs have gained increasing popularity in solving a variety of complex optimization problems due to their superior global searching ability and derivative-free characteristic, including continuous optimization [2]–[4], combinatorial optimization [5]–[8], multiobjective optimization [9]–[11], etc. The satellite layout design [12]; aerodynamic shape optimization [13]; fracture inversion [14], [15]; and job-shop scheduling [16] are just a few successful applications of EAs on various real-world optimization problems.

As a newly emerging branch of EAs, evolutionary multitasking (EMT) has attracted increasing research interests in the past three years, which aim to solve multiple distinct (but possibly similar) optimization instances simultaneously by exploiting the latent synergies between the tasks. In comparison with multitask learning [17]–[19] that has received numerous research achievements in the field of machine learning, fewer efforts have been devoted to transfer knowledge in the context of optimization problems. Multifactorial optimization (MFO) [20] is a precursory paradigm for realizing the EMT, which employs a single population to optimize multiple optimization tasks concurrently in a unified search space. Inspired from multifactorial inheritance [21], [22], the generation of offspring individuals in MFO is influenced by not only the genetic materials but also the cultural factors. Thus, the crossover between the individuals with different cultural factors allows for an implicit exchange of genetic materials from distinct tasks, which triggers the knowledge transfer between the component optimization instances. Recently, a multipopulation framework [23], [24] was proposed to realize the EMT paradigm, which adopts multiple populations to represent the distinct tasks and enable the knowledge transfer between the populations by migrating high-quality individuals.

Multifactorial EA (MFEA) [20] is a pioneering implementation of the MFO paradigm, which enables the knowledge

transfer between tasks via two approaches, namely: 1) assortative mating and 2) vertical cultural transmission. Over the last three years, MFEA has been successfully extended to solve various problems, including discrete [20], combinatorial [25], multiobjective [26], high-dimensional [27], and even computationally expensive [28] optimization problems. All these algorithms employ a prespecified scalar parameter *rmp* to govern the transfer intensity during the optimization process. However, without any prior knowledge about intertask similarity, a prespecified *rmp* for solving unrelated optimization instances concurrently can potentially lead to performance slowdowns. This phenomenon is also referred to as the "negative transfer" in much of the literature [29]–[32]. Recently, an improved version of the MFEA, called MFEA-II [33], was proposed to learn the scalar parameter *rmp* online based on the similarities between distinct tasks. A low value of *rmp* will be estimated for two tasks that explicitly possess discrepant landscapes. It is noted that the explicit similarity between two tasks is a precondition for all of the above MFEAs to make the positive knowledge transfer that is taken into effect. However, a number of studies in the domain of machine learning have demonstrated that such an explicit similarity is a sufficient but not necessary condition for triggering the positive knowledge transfer between the source–target tasks [34]–[36]. The transferability between two distinct tasks can be effectively enhanced with a proper domain adaptation technique [37]–[39].

Linearized-domain adaptation enhanced MFEA (LDA-MFEA) [38] is the first application of the domain adaptation technique in solving heterogeneous multitasking optimization problems. A linear mapping between the source–target instances is conducted by using a number of global representative samples. Subsequently, Feng *et al.* [40] proposed an EMT paradigm with explicit transfer of genetic materials. A single-layer autoencoder with a biased term is employed to learn the mapping between the source–target instances in an offline fashion. However, this offline manner neglects all the evaluated individuals in the optimization phase which may be of great value for refining intertask mapping. Besides, the pairwise learning fashion adopted by the above methods may incur an issue, called chaotic matching, which dramatically degrades the intertask mapping. This phenomenon will be discussed in detail in Section II. Recently, a generalized MFEA (G-MFEA) [28] was proposed to solve heterogeneous problems with different optimums and different numbers of decision variables, which consists of two key strategies, namely: 1) the decision variable translation strategy and 2) decision variable shuffling strategy. The proposed translation strategy can be regarded as an incomplete affine transformation with only the translation term. To address the above issues, a novel affine-transformation-based-domain adaptation technique is proposed in this work, which is mathematically derived from probabilistic representation models of the source and target tasks. The main contribution of this article is summarized as follows.

1) A novel rank loss function for obtaining a superior transformation between the source–target instances is proposed in this work, which aims to minimize the mismatch between the rank-based landscapes of two tasks.
2) An evolutionary-path-based probabilistic representation model is proposed to represent the optimization instances. The representation is updated online during the optimization process, which can be employed to act as an approximation of an optimization instance for building the intertask mapping. With the proposed representation model, the threat of chaotic matching between the source–target domains can be effectively avoided.
3) In particular, using the proposed representation model, a closed-form solution of affine transformation for bridging the gap between the source–target instances is mathematically derived from the proposed rank loss function.

The remainder of this article is organized as follows. Section II introduces several basic notions in MFO and briefly reviews the related work. In particular, the degradation of intertask mapping caused by chaotic matching is introduced and analyzed at the end of Section II. The details of the proposed representation method and domain adaptation technique for multitasking optimization problems are described in Section III. The experimental results of the proposed method against several state-of-the-art EMT algorithms are presented and analyzed in Section IV. Finally, Section V summarizes and concludes this article.

II. Preliminaries

A. Evolutionary Multitasking

EMT aims to solve multiple distinct optimization tasks simultaneously with the help of knowledge transfer. A multitasking optimization model can be formulized as follows:

$$\begin{cases} \mathbf{x}_1^* = \min f_1(x_1), & x_1 \in \Omega_1 \\ \mathbf{x}_2^* = \min f_2(x_2), & x_2 \in \Omega_2 \\ \cdots \\ \mathbf{x}_K^* = \min f_K(x_K), & x_K \in \Omega_K \end{cases} \tag{1}$$

where $\mathbf{x}_i^*$ is the optimal solution of task i (also denoted as T_i), f_i represents the objective function of T_i, Ω_i denotes the decision space of task i, and K is the total number of tasks. By triggering the solution transfer between the optimization tasks via implicit parallel of the population-based search engine, EMT optimizes the multiple problems in tandem so that the latent synergies between the tasks can be effectively utilized to significantly boost the overall optimization performance.

MFO is a proactive EMT paradigm where multiple optimization tasks are solved simultaneously by a single population [20]. Each individual in the population is associated with a factor used to represent a specific task. Under this multifactorial environment, each task f_i can be viewed as a factor influencing the evolution of the individuals. In the MFO framework, all individuals are encoded into a unified search space to cooperatively seek optimums of the multiple tasks. Each individual is decoded into its corresponding task-specific decision space to evaluate the fitness. Several key definitions associated with the individuals in MFO are as follows.

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

Definition 2: The factorial rank of p_i on T_j is the index of p_i in the list of individuals sorted in an ascending order of f_j, which is given by r_j^i.

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

Definition 4: The scalar fitness of p_i is defined as a normalized fitness based on its best rank over all tasks, which is given by $\varphi_i = 1/\text{min}_{j\in\{1,2,\ldots,K\}} r_j^i$.

Herein, the skill factor is used to represent the cultural trait, which can be inherited from the parental population in MFO. The scalar fitness enable evaluating the performance of the individuals from different tasks.

Following the categorization of transfer learning [41], we classify various EMT methods from a space-setting aspect in this article. If $\Omega_s \simeq \Omega_t$ and $f_s^r \simeq f_t^r$, where $\simeq$ represents a similar or equal relation and f^r denotes the rank-based fitness space, the scenario is called *homogeneous transfer optimization*. The problems encountered in this case are defined as *homogeneous problems*. The elite solutions from the source domain show competitive performance in the target domain. Thus, an explicit solution exchange between the problems can potentially lead to performance improvement. Otherwise, if $\Omega_s \neq \Omega_t$ or/and $f_s^r \neq f_t^r$, the scenario is referred to as *heterogeneous transfer optimization*. The problems encountered here are defined as *heterogeneous problems*. The explicit solution transfer without any transformation or modification may incur negative transfer due to the low explicit relevance between the source–target domains. In this case, a powerful domain adaptation method for enhancing the transferability is desirable. Next, various EMTs are categorized based on whether they employ a specific module for tackling the problem heterogeneity.

B. Homogenous Transfer Methods

Homogenous EMT algorithms trigger the knowledge transfer between two distinct tasks without any modification or transformation on the individuals (i.e., no specific module for tackling the problem heterogeneity is considered). Among all these algorithms, MFEA [20] is a pioneering implementation of MFO and has gained increasing research interests due to its simplicity, which incorporates the GA operators into the MFO paradigm. The pseudocode of a canonical MFEA is shown in Algorithm 1. The procedure employs a single population with $N \cdot K$ individuals to tackle multiple optimization tasks simultaneously in a unified search space. As shown in lines 2–4 of Algorithm 1, each of the K tasks is allocated with equal computational resources of N individuals in its subpopulation. There are two key features of MFEA, called assortative mating and selective imitation, that distinguish it from traditional EAs. The assortative mating mechanism allows not only the occurrence of the standard *intratask crossover* between parents from the same task but also the launch of the *intertask crossover* between distinct optimization instances, that is, possessing different skill factors. This simple feature triggers the knowledge transfer across tasks through crossover-based genetic exchanges. Lines 10–21 in Algorithm 1 show the

Algorithm 1: Implementation of a Canonical MFEA

```
1  Randomly sample N · K individuals in X to form initial population P(0);
2  for every individual p_i in P(0) do
3      Assign skill factor τ_i = mod(i, K) + 1, for the case of K tasks;
4      Evaluate p_i for task τ_i only;
5  Set t = 1;
6  while stopping conditions are not satisfied do
7      P(t) ←Select top N · K/2(≤ 50%) of generated individuals (based on scalar
       fitness φ);
8      Configure offspring populaion P_c(t) = ∅;
9      while offspring generated for each task < N do
10         Sample two individuals uniformly at random (without replacement): x_i
           and x_j from P(t);
11         if τ_i = τ_j then
12             [x_a, x_b] ← intratask crossover between x_i and x_j;
13             Assign offspring x_a and x_b with skill factor τ_i;
14         else if rand ≤ rmp then
15             [x_a, x_b] ← intertask crossover between x_i and x_j;
16             Each offspring is randomly assigned skill factor τ_i or τ_j;
17         else
18             [x_a] ← local variation (mutation) of x_i;
19             Assign offspring x_a skill factor τ_i;
20             [x_b] ← local variation (mutation) of x_j;
21             Assign offspring x_b skill factor τ_j;
22         Evaluate [x_a, x_b] for their assigned skill factors only;
23         P_c(t) = P_c(t) ∪ [x_a, x_b];
24     t = t + 1;
```

detailed implementation of the assortative mating. The intensity of knowledge transfer is controlled by a prespecified scalar parameter labeled as *random mating probability* (*rmp*). Line 14 in Algorithm 1 shows that the intertask crossover occurs under a probability specified by rmp $\in [0, 1]$, or els,e local variation comes into effect. Selective imitation is derived from the memetic concept of vertical cultural transmission, which aims to reduce the computational burden by evaluating an individual for their assigned task only. The computational realization of the selective imitation is shown in lines 4 and 22 in Algorithm 1.

As an emerging research topic, MFO has attracted increasing attention in the community of evolutionary computation. A number of EMT algorithms have been proposed in the last three years. In particular, Feng *et al.* [42] embedded the particle swarm optimization (PSO) and the differential evolution (DE) into the MFO framework. Gupta *et al.* [26] further extended the MFEA for solving multitasking multiobjective optimization problems. Zhong *et al.* [43] proposed multifactorial genetic programming for symbolic regression problems. Ma *et al.* [44] proposed a two-level transfer learning algorithm (TLTLA) to improve the search performance of MFEA further. It is noted that all these methods employ a prespecified scalar parameter *rmp* to govern the intensity of knowledge transfer. However, without any prior knowledge about the underlying intertask similarities, it is difficult to determine an optimal *rmp* in advance. Assuming the relatedness with a prespecified high *rmp* and optimizing multiple unrelated problems concurrently can potentially lead to performance slowdowns. Recently, to alleviate the threat of negative transfer, an improved version of the MFEA, called MFEA-II, with online learning of *rmp* was proposed [33]. A symmetric *RMP* matrix is estimated online by maximizing the log-likelihood of multiple mixture distribution representations. The magnitude of the estimated *rmp* can

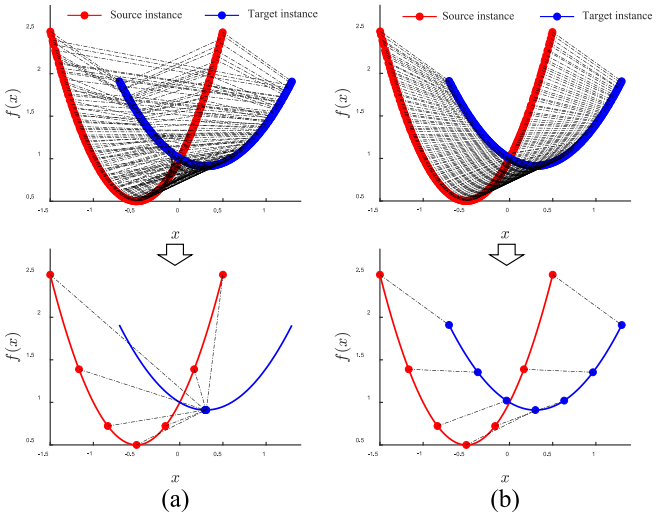


Fig. 1. Comparison of matching relationship and solution transfer between the chaotic matching and the modified matching. (a) Chaotic matching and its transfer results based on the learned mapping. (b) Modified mapping and its transfer results based on the learned mapping.

reflect the local similarity of two distinct optimization tasks. Therefore, the positive transfer between the problems with high degree of relatedness can be greatly boosted by estimating a relatively high *rmp*, while the negative transfer between the distinct problems with low similarity can be effectively curbed by estimating and employing a low *rmp* value.

C. Heterogeneous Transfer Methods

For heterogeneous problems which explicitly possess different decision (or/and fitness) spaces, the latent synergies can be uncovered and utilized to enhance the overall optimization performance as long as a suitable modification or transformation between the source and target instances can be conducted. This technique of bridging the gap between two distinct problems from different domains is also known as the heterogeneous domain adaptation (HDA) method [45], [46], which has gained tremendous research interests in the field of machine learning. However, seldom efforts have been devoted to solve heterogeneous multitasking optimization problems with domain adaptation in the community of evolutionary computation.

LDA-MFEA [38] is the first application of the domain adaptation technique in solving heterogeneous multitasking optimization problems. In order to facilitate the ordinal correlation (i.e., rank correlation) between the given two optimization instances [47], two matrices with evaluated solutions are ranked in ascending order according to their fitness values. Then, a linear transformation can be obtained using the ordinary linear least square method.

By employing an autoencoder, Feng *et al.* [40] proposed an EMT paradigm with explicit transfer of genetic materials. The autoencoder is used to learn a mapping between two problem domains. To improve the computational efficiency for training an autoencoder, the canonical autoencoder is simplified to a single layer mapping with a bias term. Theoretically, a single-layer autoencoder with a bias term can be regarded as an affine transformation with contractive transformation and translation transformation. It is noteworthy that the above pairwise learning procedures may encounter a severe issue, called chaotic matching. This phenomenon will be discussed in detail later.

Given two heterogeneous problems that explicitly possess different optima, Yin *et al.* [48] proposed to transfer direction vector (DV) information rather than individual in the MFEA framework. The heterogeneity can be alleviated to some extent using the proposed MFEA-DV method. Recently, a generalized MFEA (G-MFEA) [28] was proposed to solve heterogeneous multitasking optimization problems. Two key strategies, called decision variable translation and decision variable shuffling, were introduced to overcome the heterogeneity of distinct problems. The decision variable translation strategy aims to map all individuals into a common central location so that the solutions of all tasks can exchanged seamlessly in this calibrated point with no threats of negative transfer. Next, with the decision variable shuffling strategy, intertask crossover occurs in the common central region (i.e., with translated solutions). It can be seen that both MFEA-DV and G-MFEA implicitly employ a translation transformation between the source–target domains. Particularly, G-MFEA is accompanied with a decision variable shuffling module for tackling the dimensional heterogeneity.

D. Degradation of Intertask Mapping

To facilitate the rank correlation between the source–target instances, existing mapping-based transferability enhancement methods first sort the solutions according to the fitness values. Then, an intertask mapping is conducted by learning the training samples drawn from the sorted pairwise solutions. However, such pairwise learning suffers from a serious problem, called chaotic matching, which may result in a failed mapping. This phenomenon is called the degradation of intertask mapping in this study. To elaborate, we consider a multitasking problem with two 1-D convex functions herein. Thus, a univariate linear regression model is employed to learn an intertask mapping between the tasks, which is shown as follows:

$$x^t = ax^s + b \tag{2}$$

where x^t represents a target solution, x^s denotes a source solution, and $\boldsymbol{\theta} = [a, b]$ is a vector of mapping parameters.

Using the ordinary least squares method [49], the mapping parameters can be estimated as follows:

$$\begin{cases} \widehat{a} = \dfrac{\sum_{i=1}^{n}(x_i^s - \bar{x}^s)(x_i^t - \bar{x}^t)}{\sum_{i=1}^{n}(x_i^s - \bar{x}^s)^2} = \dfrac{n}{n-1} \cdot \dfrac{\mathrm{cov}(x_s, x_t)}{std(x^s)} \\ \widehat{b} = \bar{x}^t - \widehat{a}\bar{x}^s \end{cases} \tag{3}$$

where x_i^s denotes the ith sample of the source domain, $\bar{x}^s$ is the sample mean of the source domain, x_i^t is the ith sample of the target domain, $\bar{x}^t$ represents the sample mean of the target domain, and n is the number of samples.

The process of pairwise matching for estimating the mapping parameters is illustrated in the top of Fig. 1(a). The source and target solutions are paired as training samples according to their fitness ranks only. The dashed black lines are used to represent the matching relationship between the source–target

TABLE I
LEARNED PARAMETERS OF THE CHAOTIC MATCHING AND MODIFIED MATCHING WITH 100 TRAINING SAMPLES

Eatimates	Chaotic matching	Modified matching
a	0.0125	1.0001
b	0.3007	0.8002

samples. A great number of intersections among the matchings will impede the detection of linear correlation between the source–target instances (i.e., a low $\widehat{a}$), which may result in a failed mapping.

In this study, a matching fashion that obstructs the detection of intertask mapping is called *chaotic matching*. To address this issue, a simple yet effective way is to pair the source–target solutions by not only considering the rank correlation but also taking the topological consistency into account. For this 1-D case, a modified matching can be obtained by pairing the samples for left and right flanks of the quadratic functions independently, as illustrated in the top of Fig. 1(b). Detailed pairing procedures can be found in the supplementary material. Given 100 training samples, Table I lists the learned mapping parameters of the chaotic matching and the modified matching. We can see that the mapping obtained by the chaotic matching degrades into a failed form $\boldsymbol{\theta}_f = [0, \bar{x}^t]$. Such transformation tends to project diverse source solutions into the central position of the target domain, as shown in the bottom of Fig. 1(a). In contrast, the transformation offered by the modified matching is able to guarantee the correspondence in terms of solution quality between the source–targets domains, as shown in the bottom of Fig. 1(b). Unfortunately, it is difficult to generalize the above simple matching-modifying strategy into a high-dimensional case due to the lack of definition for high dimensional flanks. With this in mind, a representation-based learning mechanism for acquiring an intertask mapping is proposed in this study.

III. AFFINE TRANSFORMATION-ENHANCED MULTIFACTORIAL EVOLUTION

In this section, the proposed affine transformation-enhanced multifactorial evolution is presented in detail. First, a novel loss function for acquiring an intertasking mapping is formulized. Next, a progressional representation model acting as a surrogate is proposed. In particular, with the representation model, a closed-form solution of affine transformation for bridging the gap between two distinct problems is derived mathematically. Finally, the MFEA framework embedded with the proposed transferability enhancement technique is detailed.

A. Rank Loss Function

Herein, the objective functions of the source and target optimization instances are denoted as $f^s(\boldsymbol{x})$ and $f^t(\boldsymbol{x})$, respectively. The goal of the mapping-based multitasking optimization is to build a transformation between the source–target instances and utilize this transformation to transfer the solutions from the source instance to the target instance, and thus improve the optimization performance in solving the target instance. To facilitate the rank correlation, this transformation can be obtained by minimizing the mismatch between the rank-based landscape of the source instance and the rank-based landscape of the target instance with transferred source inputs. The obtained transformation guarantees that the elite solutions in the source instance also shows competitive fitness in the target instance with transferred form. A rank loss function for acquiring the transformation between the source and target instances is formulized as follows:

$$\phi^* = \min_{\phi \in \Phi} \int_{\boldsymbol{x}} \left\| \Re\left[f^s(\boldsymbol{x})\right] - \Re\left[f^t(\phi(\boldsymbol{x}))\right] \right\|^2 \tag{4}$$

where ϕ represents a transformation operation that is able to map a solution from the source domain into the target domain, Φ denotes a complete mapping space, and $\Re$ denotes a rank operation for turning the original fitness landscape into a rank-based landscape. The proposed loss function for acquiring the mapping relationship is applicable not only for heterogeneous problems but also for homogenous problems with the same (or similar) spaces. For homogenous problems, the mapping obtained using (4) is the equivalent transformation [i.e., $\boldsymbol{x}$ is identical to $\phi(\boldsymbol{x})$], thus the elite solutions from the source instance can be directly transferred to the target instance without any modification or transformation.

Solving (4) directly requires sufficient exploitation of the decision spaces due to the integration operation, which is practically infeasible. A practical alternative is to adopt a surrogate (i.e., the representation model) of the original fitness function in (4), as the representation is independent of the real function evaluation. Hence, the loss function can be turned into the following form:

$$\phi^* = \min_{\phi \in \Phi} \int_{\boldsymbol{x}} \left\| \Re\left[\hat{f}^s(\boldsymbol{x})\right] - \Re\left[\hat{f}^t(\phi(\boldsymbol{x}))\right] \right\|^2 \tag{5}$$

where $\hat{f}^s(\cdot)$ and $\hat{f}^t(\cdot)$ denote the representation models of the source and target tasks, respectively.

B. Optimization Instance Representation

Given a black-box optimization problem, only the evaluated individuals in different generations can be utilized to build a representation model. Herein, the evaluated individuals from the initial population to the current population are called *evolutionary path*. A probability density function of the candidate solutions for representing the optimization instance can be obtained by assimilating the data from the evolutionary path, which is formulized as follows:

$$\widehat{p}_t(\boldsymbol{x}) = \mathfrak{T}\left(\boldsymbol{x}; \mathrm{Pop}_1, \mathrm{Pop}_2, \ldots, \mathrm{Pop}_t\right) \tag{6}$$

where $\widehat{p}_t(\boldsymbol{x})$ represents an estimated distribution of the optimization instance which has been evolved for totally t generations by a population-based optimizer; $\mathfrak{T}(\cdot)$ denotes an operation that estimates a specified probability distribution using the data from the evolutionary path.

In particular, the distribution for the instance representation can be estimated globally or locally, which is expressed as follows, respectively:

$$\widehat{p}_t^g(\boldsymbol{x}) = \mathfrak{T}\left(\boldsymbol{x}; \mathrm{Pop}_1 \cup \mathrm{Pop}_2 \cup \cdots \cup \mathrm{Pop}_t\right) \tag{7}$$

$$\widehat{p}_t^l(\boldsymbol{x}) = \mathfrak{T}\left(\boldsymbol{x}; \mathrm{Pop}_t\right) \tag{8}$$

where $\widehat{p}_t^g(\boldsymbol{x})$ is the global representation with equal contribution assigned to each generation and $\widehat{p}_t^l(\boldsymbol{x})$ denotes the local representation.

The global representation gives equal importance to the generational populations, while the local representation only takes the current population into account. To retain the full information, a complete global representation requires a large amount of storage space as the number of evaluated individuals increases, which makes it computationally resource consuming. In contrast, the local representation is computationally efficient but very sensitive to the current population update. Therefore, a novel progressional representation is proposed in this article, which is able to assimilate the evaluated individuals in a more reasonable and flexible fashion. A progressional representation of the optimization instance is defined as

$$\begin{cases} \widehat{p}_1^p(\boldsymbol{x}) = \mathfrak{T}\big(\boldsymbol{x}; \mathrm{Pop}_1\big) \\ \widehat{p}_t^p(\boldsymbol{x}) = \alpha \widehat{p}_{t-1}^p(\boldsymbol{x}) + (1-\alpha)\mathfrak{T}\big(\boldsymbol{x}; \mathrm{Pop}_t\big), \quad t > 1 \end{cases} \tag{9}$$

where $\widehat{p}_t^p(\boldsymbol{x})$ is the progressional representation of the optimization instance which has been evolved for totally t generations and $0 \leq \alpha \leq 1$ denotes a preference coefficient about the previous population used to build the representation.

The above recursive expression can be rewritten as follows:

$$\begin{aligned} \widehat{p}_t^p(\boldsymbol{x}) &= \sum_{k=1}^{t} \alpha^{t-k}(1-\alpha)\mathfrak{T}\big(\boldsymbol{x}; \mathrm{Pop}_k\big) \\ &= \sum_{k=1}^{t} w^k \widehat{p}_k^l(\boldsymbol{x}) \end{aligned} \tag{10}$$

where w^k represents the weight of kth population in estimating the progressional representation and $\widehat{p}_k^l(\boldsymbol{x})$ denotes the estimated distribution of the local representation at generation k.

Equation (10) indicates that the progressional representation can be expressed in a decreasing weighted form (i.e., $w^{k-1} < w^k$) with respect to multiple generational local representations. In other words, the newly undated population is more important as compared to the previous population in building the progressional representation. Thus, the proposed representation model can be seen as a type of global representation with biased contributions. The difference of the contributions can be easily adjusted by changing the preference coefficient. In addition, benefiting from the recursive form in (9), the proposed model has the virtue of only requiring the storage of previous representation model (i.e., $\widehat{p}_{t-1}^p$). In this study, the independent multivariate Gaussian distribution is employed to estimate an representative probability density distribution for each population due to its sound mathematical properties, which is denoted as $\widehat{p}^l(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$, $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_n)$, $\boldsymbol{\sigma}^2 = (\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)$, where n represents the dimensionality, μ_j denotes the mean of the jth dimension, and σ_j^2 is the variance of the jth dimension. Hence, a local representation of the optimization instance at generation t is calculated as follows:

$$\mu_j^t = \frac{1}{N}\sum_{i=1}^{N} x_{ij}^t \tag{11}$$

$$\sigma_j^{t2} = \frac{1}{N-1}\sum_{i=1}^{N}\Big(x_{ij}^t - \mu_j^t\Big)^2 \tag{12}$$

where x_{ij}^t denotes the value of the jth dimension of the ith individual in the population at generation t, and N is the population size.

By substituting the local representation in (11) and (12) into (10), one can easily arrive at the progressional representation of the optimization instance, which is as follows:

$$\widehat{p}_t^p(\boldsymbol{x}) \sim \mathcal{N}\Big(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\sigma}}^2\Big) \tag{13}$$

where

$$\hat{\boldsymbol{\mu}} = \big(\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_n\big) \tag{14}$$

$$\hat{\mu}_j = (1-\alpha)\frac{1}{N}\sum_{k=1}^{t}\alpha^{t-k}\sum_{i=1}^{N} x_{ij}^k \tag{15}$$

$$\hat{\boldsymbol{\sigma}}^2 = \Big(\hat{\sigma}_1^2, \hat{\sigma}_2^2, \ldots, \hat{\sigma}_n^2\Big) \tag{16}$$

$$\hat{\sigma}_j^2 = (1-\alpha)\frac{1}{N-1}\sum_{k=1}^{t}\alpha^{t-k}\sum_{i=1}^{N}\Big(x_{ij}^k - \mu_j^k\Big)^2. \tag{17}$$

Now, with the proposed representation model, the rank loss function in (5) is given as follows:

$$\phi^* = \min_{\phi\in\Phi}\int_{\boldsymbol{x}} \big\|\Re\big[p^s(\boldsymbol{x})\big] - \Re\big[p^t(\phi(\boldsymbol{x}))\big]\big\|^2 \tag{18}$$

where $p^s(\cdot)$ and $p^t(\cdot)$ denote the Gaussian progressional representation models of the source and target tasks, respectively.

C. Affine Transformation for Source–Target Instances

The complete mapping space Φ in the loss function contains all possible transformations which can be categorized as linear transformation and nonlinear transformation as a whole. For simplicity, only the linear transformation is considered in this study, which is also referred to as *affine transformation*. A transferred solution for the target instance can be obtained using the affine transformation on the source instance, which is given as follows:

$$\tilde{\boldsymbol{x}} = \phi_a(\boldsymbol{x}; \boldsymbol{\theta}) = \boldsymbol{x}A + \boldsymbol{b} \tag{19}$$

where $\tilde{\boldsymbol{x}}_{1\times n}$ denotes a transferred solution for the target instance, $\phi_a(\cdot)$ represents the affine transformation, $\boldsymbol{x}_{1\times n}$ is a solution from the source instance, $\boldsymbol{\theta} = [A, \boldsymbol{b}]$ denotes the parameters of the affine transformation, $A_{n\times n}$ is a contractive transformation, and $\boldsymbol{b}_{1\times n}$ represents a translation transformation.

By substituting the affine transformation into the representation-based loss function, (18) is expressed as follows:

$$\boldsymbol{\theta}^* = \min_{\boldsymbol{\theta}}\int_{\boldsymbol{x}} \big\|\Re\big[p^s(\boldsymbol{x})\big] - \Re\big[p^t(\phi_a(\boldsymbol{x}; \boldsymbol{\theta}))\big]\big\|^2 \tag{20}$$

Now, the goal is to configure an optimal affine transformation by minimizing the mismatch between the source and target (with transferred inputs) representations in the rank space. Before we move on, two pivotal traits of the Gaussian distribution are listed as follows.

1) The Gaussian distribution is analytically trackable.
2) A multiplier δ does not change the rank landscape of a Gaussian distribution (i.e., $\Re[\delta \times \mathcal{N}(\cdot)] = \Re[\mathcal{N}(\cdot)]$).

To solve (20), we try to let the rank landscape of target representation (with transferred inputs) be exactly the same as the rank landscape of source representation, by adjusting the mapping parameters of the affine transformation. In this case, the mismatch in (20) is zero and the corresponding configuration for the affine transformation is optimal. The above condition is satisfied if the following equation holds:

$$\Re\big[p^s(\boldsymbol{x})\big] = \Re\big[p^t(\phi_a(\boldsymbol{x};\boldsymbol{\theta}))\big],\ \boldsymbol{x} \in \Omega. \tag{21}$$

With the second trait of the Gaussian distribution, (21) is equivalent to the following equation:

$$\Re\big[p^s(\boldsymbol{x})\big] = \Re\big[\delta \times p^t(\phi_a(\boldsymbol{x};\boldsymbol{\theta}))\big],\ \boldsymbol{x} \in \Omega \tag{22}$$

where δ is an unknown multiplier used to compensate the difference of the function value landscape between the source and target representations.

Considering the continuity and the differentiability of the Gaussian distribution, (22) is converted into the following form:

$$p^s(\boldsymbol{x}) = \delta \times p^t(\phi_a(\boldsymbol{x};\boldsymbol{\theta})). \tag{23}$$

There are two major benefits to converting the original minimizing rank loss function into an equivalent algebraic form in (23). On the one hand, solving the original minimization task directly is computationally time-consuming as compared to solving the algebraic equation. On the other hand, the algebraic equation possesses an analytical solution. To solve the algebraic equation, the probability distributions in (23) are expanded further as follows:

$$p^s(\boldsymbol{x}) = \frac{1}{(2\pi)^{n/2}|\Sigma_s|^{1/2}} \times \mathbb{E}\big(\boldsymbol{x}, \boldsymbol{\mu}_s, \Sigma_s\big) \tag{24}$$

$$p^t(\tilde{\boldsymbol{x}}) = \frac{1}{(2\pi)^{n/2}|\Sigma_t|^{1/2}} \times \mathbb{E}\big(\boldsymbol{x}A + \boldsymbol{b}, \boldsymbol{\mu}_t, \Sigma_t\big) \tag{25}$$

where

$$\mathbb{E}(\boldsymbol{x}, \boldsymbol{\mu}, \Sigma) = \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})^\top\right)$$

where Σ_s and Σ_t denote the covariance matrices of the source and target representation models, respectively.

By applying the method of completing square for (23), we have

$$\delta = \frac{|\Sigma_t|^{1/2}}{|\Sigma_s|^{1/2}} \tag{26}$$

$$\boldsymbol{\mu}_s = \big(\boldsymbol{\mu}_t - b\big)A^{-1} \tag{27}$$

$$\Sigma_s^{-1} = A\Sigma_t^{-1}A^\top. \tag{28}$$

With the Cholesky decomposition, (28) is rewritten as follows:

$$L_s L_s^\top = A L_t L_t^\top A^\top \tag{29}$$

where L_s and L_t denote the lower triangular matrices obtained from the Cholesky decomposition on the source and target inverse covariance matrices, respectively.

Finally, the transformation parameters A and $\boldsymbol{b}$ are obtained as follows[1]:

$$\begin{cases} A = L_s L_t^{-1} \\ \boldsymbol{b} = \boldsymbol{\mu}_t - \boldsymbol{\mu}_s A. \end{cases} \tag{30}$$

Now, given the evolutionary paths of two optimization instances, the mapping-based transfer can be triggered using the proposed progressional representation and the affine transformation. In particular, here, we apply the proposed method on the 1-D case in Section II. The parameter a is estimated as follows:

$$a = \frac{\sqrt{D(x_s)}}{\sqrt{D(x_t)}} = \frac{0.3332}{0.3331} = 1.0003$$

where $D(\cdot)$ denotes the variance function.

The biased term b is estimated as follows:

$$b = \bar{x}^t - a\bar{x}^s = 0.2999 - 1.0003 \times (-0.5001) = 0.8002.$$

We can see that the proposed method successfully identified the intertask mapping on the 1-D problem without the need for modifying the matching relationship. Most importantly, the proposed representation-based mapping technique can be easily generalized to a high-dimensional case. The threat of chaotic matching can be exquisitely avoided. In addition, unlike the mapping method developed by Feng which adopts an offline learning manner, the proposed progressional representation-based mapping is updated online during the optimization process. To elaborate further, a more complex multitasking problem with two 1-D nonconvex component functions is considered here. A shifted Rastrigin's function is set as the source optimization instance, while a shifted Ackley's function is set as the target instance. Evolutionary paths with three generations of these two component problems are shown in Fig. 2(a). A canonical EA (CEA) with simulated binary crossover (SBX) and polynomial mutation (PM) is employed as the search engine. We can see that the individuals of the two tasks gradually converge to their promising regions with high fitness values as the generation increases. Next, two representation models for the source–target instances are constructed using the proposed progressional representation method and the evolutionary paths, as shown in Fig. 2(b). Based on (30), an affine transformation can be analytically derived from the representation models. Finally, as shown in Fig. 2(c), a number of elite solutions from the source instance are transferred to the target instance with the help of the obtained mapping. The dashed black lines are used to represent the correspondences between the source solutions and the transferred solutions. We can see that the potential region of the target instance is explored effectively with the transferred solutions. The chaotic matching is no longer a threat for implementing the mapping-based knowledge transfer between two distinct problems. Similarly, a reverse mapping can be conducted to transfer the solutions from the target domain to the source domain in a multitasking environment. To better illustrate the superiority of the proposed method as compared to a number of state-of-the-art mapping techniques, a

[1] A more detailed derivation is provided in Section S-I of the supplementary material.

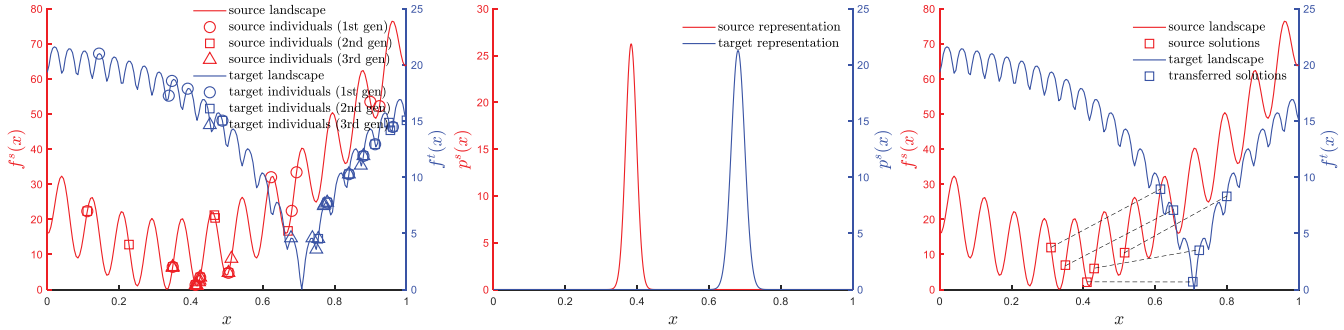


Fig. 2. Illustrative example about building and utilizing the proposed representation-based affine transformation. (a) Evolutionary paths of the source and target instances. (b) Representation models drawn from the evolutionary paths. (c) Building an affine transformation and transferring the source solutions using the obtained mapping.

Algorithm 2: Mapping-Based Intertask Crossover

1 **Input:** Two individuals x_i and x_j from two distinct tasks and the corresponding evolutionary paths E^i and E^j;
2 **Output:** Two offsprings x_a and x_b;
3 $\left[p^i(\boldsymbol{x}), p^j(\boldsymbol{x})\right]$ ←progressional representation conditioned on E^i and E^j;
4 $[A_{ij}, \boldsymbol{b}_{ij}]$ ←affine transformation derived from source $p^i(\boldsymbol{x})$ and target $p^j(\boldsymbol{x})$;
5 $[A_{ji}, \boldsymbol{b}_{ji}]$ ←affine transformation derived from source $p^j(\boldsymbol{x})$ and target $p^i(\boldsymbol{x})$;
6 $x'_i \leftarrow x_i A_{ij} + \boldsymbol{b}_{ij}$;
7 $x'_j \leftarrow x_j A_{ji} + \boldsymbol{b}_{ji}$;
8 x_a ←crossover between x'_i and x_j;
9 x_b ←crossover between x_i and x'_j;

detailed comparison with regard to the mapping behavior and the algorithmic complexity is provided in the supplementary material.[2]

D. Affine Transformation-Enhanced Multifactorial Evolution

The proposed transformation-based transferability enhancement technique can be seamlessly embedded into the canonical MFEA framework by replacing the intertask crossover in line 15 of Algorithm 1 with a mapping-based manner. Algorithm 2 shows the detailed implementation of the mapping-based intertask crossover, which serves as an offspring generator based on two parental individuals with different skill factors (i.e., distinct tasks). The procedure is divided into following four steps.

1) *Representation of Optimization Instance:* Given two parental individuals x_i and x_j with different skill factors, their corresponding evolutionary paths are denoted as E^i and E^j. The representations of the two tasks can be easily conducted using the proposed progressional representation method which does not require the full access to all evaluated individuals. The procedure is executed in a recursive manner as shown in (9). This computationally resource-saving trait allows the proposed representation method to solve large-scale problems efficiently.
2) *Learning of Intertask Mappings:* The intertask mappings can be conducted using the established representations. Here, the tasks associated with the two individuals are denoted as T_i and T_j, respectively. An affine transformation $\boldsymbol{\theta}_{ij} = [A_{ij}, \boldsymbol{b}_{ij}]$ for transferring the solutions from T_i to T_j can be derived from (30) by setting the source–target representations $p^s(\boldsymbol{x})$ and $p^t(\boldsymbol{x})$ as $p^i(\boldsymbol{x})$ and $p^j(\boldsymbol{x})$, respectively. Conversely, the reverse mapping $\boldsymbol{\theta}_{ji} = [A_{ji}, \boldsymbol{b}_{ji}]$ can be obtained by swapping the positions of T_i and T_j.
3) *Solution Transfer:* Subsequently, a transformed solution x'_i for T_j is obtained by mapping the original individual x_i into a transferred space with the help of the acquired affine transformation $\boldsymbol{\theta}_{ij}$. The transformation of x_j is processed in the same manner, which is shown in line 7 of Algorithm 2.
4) *Crossover:* Finally, one specific crossover operator can be employed to generate the offspring individuals. Instead of directly executing the intertask crossover, x_i and x_j undergo the crossover operation with the corresponding transformed individuals x'_j and x'_i, respectively. Thus, the negative transfer can be effectively alleviated.

In the intertask crossover module of the MFEA framework, skill factors for generated offsprings are assigned randomly. In this study, this nondirectional assignment strategy of skill factors is retained in the proposed affine transformation-enhanced MFEA (AT-MFEA) method. The nondirectional assignment strategy allows the generated offsprings to be evolved in not only the source domain but also the target domain, which gives equal preference in solving homogeneous and heterogeneous problems.

This feature makes AT-MFEA more robust in optimizing the black-box problems with no prior knowledge about the intertask similarity. It is noted that the only difference between the canonical MFEA (i.e., Algorithm 1) and the proposed AT-MFEA is the intertask crossover in line 15 of Algorithm 1. The implementation of AT-MFEA can be easily obtained by replacing line 15 of Algorithm 1 with the mapping-based intertask crossover (i.e., Algorithm 2). Therefore, for the sake of less redundancy, the complete implementation of AT-MFEA is not supplied here.

IV. Experimental Study

In this section, a number of numerical experiments are conducted to verify the efficacy of the proposed AT-MFEA method.[3] For rigorous analysis, the optimization performance

[2]See Section S-II of the supplementary material accompanying this article.

[3]The MATLAB implementation of the AT-MFEA method can be accessed from the following link: https://github.com/XmingHsueh/ATMFEA.

TABLE II
DETAILED PROPERTIES OF THE SINGLE-OBJECTIVE 2-TASK BENCHMARK PROBLEMS AND ITS MODIFIED VERSION

Category	Task	Search Space	Landscape	x_u^*	$\widehat{x}_u^*$	R_s
CI+HS	Griewank (T1)	$D = 50, x \in [-100, 100]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_1 \sim U^D\,(0.25, 0.75)$	1.0000
	Rastrigin (T2)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	x_1	
CI+MS	Ackley (T1)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_2 \sim U^D\,(0.25, 0.75)$	0.2261
	Rastrigin (T2)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	x_2	
CI+LS	Ackley (T1)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.92, 0.92, ..., 0.92]^D$	$x_3 \sim U^D\,(0.25, 0.75)$	0.0002
	Schwefel (T2)	$D = 50, x \in [-500, 500]^D$	multimodal, separable	$[0.92, 0.92, ..., 0.92]^D$	x_3	
PI+HS	Rastrigin (T1)	$D = 50, x \in [-500, 500]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_4 \sim U^D\,(0.25, 0.75)$	0.8670
	Sphere (T2)	$D = 50, x \in [-50, 50]^D$	unimodal, separable	$[\underbrace{0.5, 0.5, ...}_{D/2}, \underbrace{0.6, 0.6}_{D/2}]^D$	$x_4\,(1 : D/2) \cup U^{D/2}\,(0.25, 0.75)$	
PI+MS	Ackley (T1)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[\underbrace{0.5, 0.5, ...}_{D/2}, \underbrace{0.51, 0.51}_{D/2}]^D$	$x_5 \sim U^D\,(0.25, 0.75)$	0.2154
	Rosenbrock (T2)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.51, 0.51, ..., 0.51]^D$	$U^{D/2}\,(0.25, 0.75) \cup x_5\,(D/2 + 1 : D)$	
PI+LS	Ackley (T1)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_6 \sim U^D\,(0.25, 0.75)$	0.0725
	Weierstrass (T2)	$D = 25, x \in [-0.5, 0.5]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_6\,(1 : D)$	
NI+HS	Rosenbrock (T1)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.51, 0.51, ..., 0.51]^D$	$x_7^1 \sim U^D\,(0.25, 0.75)$	0.9434
	Rastrigin (T2)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_7^2 \sim U^D\,(0.25, 0.75)$	
NI+MS	Griewank (T1)	$D = 50, x \in [-100, 100]^D$	multimodal, nonseparable	$[0.55, 0.55, ..., 0.55]^D$	$x_8^1 \sim U^D\,(0.25, 0.75)$	0.3669
	Weierstrass (T2)	$D = 50, x \in [-0.5, 0.5]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_8^2 \sim U^D\,(0.25, 0.75)$	
NI+LS	Rastrigin (T1)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$[0.5, 0.5, ..., 0.5]^D$	$x_9^1 \sim U^D\,(0.25, 0.75)$	0.0016
	Schwefel (T2)	$D = 50, x \in [-500, 500]^D$	multimodal, separable	$[0.92, 0.92, ..., 0.92]^D$	$x_9^2 \sim U^D\,(0.25, 0.75)$	

of the proposed AT-MFEA is compared against a number of state-of-the-art multitasking solvers, ranging from homogenous to heterogenous transfer algorithms. First, a single-objective multitasking benchmark suite with 9 problems [47] is employed herein. It is noted that the benchmark is modified in this study in consideration of fair comparison between the selected solvers. Then, a more challenging heterogeneous many-tasking problem and a practical optimization case from the petroleum industry are considered.

A. Experimental Configuration

The multitasking population size N is kept the same for all algorithms in solving a specific problem. In this work, we only employ the mapping components of the heterogenous transfer methods and embed them into the MFEA framework to verify their efficacy against the proposed mapping technique. Three variants of MFEA associated with linearized-domain adaptation, autoencoder, and translation-shuffling-based mapping are denoted as LDA-MFEA, AE-MFEA, and G-MFEA, respectively. For the sake of fairness, the algorithmic structure and parameter settings of the above variants are set to be consistent with the AT-MFEA except for the module of mapping-based intertask crossover. The general algorithmic settings for all problems are outlined as follows.

1) Unified representation space with range $[0, 1]^{D_{\text{unified}}}$.
2) Evolutionary operators for all algorithms:
   a) SBX crossover with probability $p_c = 1$ and distribution index $\eta_c = 15$;
   b) PM with probability $p_m = 1/d$ and distribution index $\eta_m = 15$.
3) For the single-objective multitasking benchmark suite:
   a) Population size (N): 100;
   b) Maximum function evaluations: 100 000.
4) For the many-tasking case study:
   a) Population size (N): 200;
   b) Maximum function evaluations: 200 000.
5) For the practical optimization case:
   a) Population size (N): 30;
   b) Maximum function evaluations: 600;
6) No separate local search steps are performed.
7) *rmp* is set to 0.3 for MFEA, LDA-MFEA, AE-MFEA, G-MFEA, and AT-MFEA.

The population size is divided equally for the component tasks in the CEA with no knowledge transfer. The parameters settings of LDA-MFEA, AE-MFEA, and G-MFEA for building the intertask mappings are consistent with the original reports, while the related settings of AT-MFEA are summarized as follows.

1) The probabilistic model for the local representation: independent multivariate normal distribution.
2) The preference coefficient α is set to 0.5.

B. Single-Objective Multitasking Benchmark Problems

Based on a technical report on the EMT optimization, nine single-objective two-task benchmark problems are employed in this study. The detailed properties about the benchmark is provided in Table II. It can be seen that the benchmark suite comprises of a wide variety of test problems which are built by pairing classical single-objective functions, such as *Ackley*, *Rosenbrock*, *Rastrigin*, etc. Each problem is featured in terms

TABLE III
AVERAGE OBJECTIVE VALUE AND STANDARD DEVIATION OBTAINED BY CEA, MFEA, MFEA-II, LDA-MFEA, AE-MFEA, G-MFEA, AND AT-MFEA ON THE MODIFIED SINGLE-OBJECTIVE 2-TASK BENCHMARK SUITE. THE HIGHLIGHTED ENTRIES ARE SIGNIFICANTLY BETTER (THE WILCOXON RANK-SUM TEST WITH THE HOLM p-VALUE CORRECTION, $\alpha = 0.05$)

Category	Task	CEA	MFEA	MFEA-II	LDA-MFEA	AE-MFEA	G-MFEA	AT-MFEA
CI+HS	T1	2.67e-2±1.22e-2	2.49e-2±1.45e-2	2.47e-2±1.29e-2	3.27e-2±1.32e-2	3.16e-2±9.5e-3	1.43e-2±6.70e-3	2.06e-2±9.00e-3
	T2	1.55e+2±3.01e+2	4.65e+1±2.16e+1	5.17e+1±2.80e+1	1.39e+2±2.88e+1	1.39e+2±3.34e+1	6.50e+1±2.69e+1	3.19e+1±1.62e+1
CI+MS	T1	1.77e+0±3.11e-1	1.94e+0±4.31e-1	2.09e+0±4.79e-1	1.95e+0±4.00e-1	1.79e+0±4.20e-1	1.93e+0±3.56e-1	1.31e+0±3.75e-1
	T2	1.37e+2±3.37e+1	5.50e+1±2.95e+1	8.94e+1±3.19e+1	1.38e+2±2.63e+1	1.40e+2±3.59e+1	5.53e+1±2.11e+1	4.18e+1±2.99e+1
CI+LS	T1	1.90e+0±3.84e-1	1.98e+0±2.80e-1	1.93e+0±5.13e-1	2.04e+0±3.02e-1	2.08e+0±3.03e-1	1.94e+0±3.19e-1	1.32e+0±5.53e-1
	T2	2.14e+3±5.03e+2	2.42e+3±4.74e+2	2.10e+3±4.48e+2	2.17e+3±6.11e+2	2.21e+3±4.32e+2	2.07e+3±5.55e+2	2.25e+3±3.91e+2
PI+HS	T1	1.50e+2±2.95e+1	1.32e+2±2.46e+1	1.34e+2±2.11e+1	1.52e+2±3.02e+1	1.51e+2±3.76e+1	1.40e+2±2.94e+1	1.28e+2±3.12e+1
	T2	5.30e-2±2.40e-2	6.68e-2±2.72e-2	5.00e-2±1.17e-2	7.32e-2±1.86e-2	6.98e-2±2.21e-2	6.26e-2±2.28e-2	2.82e-2±1.47e-2
PI+MS	T1	1.78e+0±5.20e-1	1.73e+0±3.15e-1	1.83e+0±2.69e-1	2.07e+0±4.48e-1	1.88e+0±3.66e-1	1.78e+0±3.59e-1	1.13e+0±5.61e-1
	T2	4.35e+2±1.07e+3	8.27e+2±1.40e+3	5.38e+2±1.09e+3	1.05e+3±1.73e+3	8.83e+1±4.59e+1	5.59e+2±1.12e+3	2.92e+2±5.25e+2
PI+LS	T1	1.83e+0±3.73e-1	1.89e+0±4.55e-1	1.96e+0±4.73e-1	1.97e+0±3.22e-1	1.81e+0±4.60e-1	1.92e+0±3.33e-1	1.45e+0±3.13e-1
	T2	1.22e+1±3.63e+0	2.53e+0±7.09e-1	2.56e+0±6.26e-1	1.17e+1±3.19e+0	8.31e+0±1.90e+0	2.48e+0±4.36e-1	1.79e+0±5.06e-1
NI+HS	T1	7.10e+2±1.21e+3	6.48e+2±1.18e+3	1.01e+3±1.53e+3	9.62e+2±1.31e+3	7.89e+1±5.25e+1	6.42e+2±1.13e+3	5.55e+2±8.02e+2
	T2	1.45e+2±3.25e+1	1.25e+2±3.17e+1	1.34e+2±3.39e+1	1.51e+2±3.40e+1	1.35e+2±3.37e+1	1.33e+2±2.29e+1	1.45e+2±3.05e+1
NI+MS	T1	2.96e-2±9.60e-3	3.46e-2±1.66e-2	2.42e-2±6.30e-3	3.66e-2±1.42e-2	3.19e-2±1.26e-2	2.93e-2±1.02e-2	2.19e-2±9.10e-3
	T2	2.77e+1±4.74e+0	3.57e+1±9.24e+0	2.89e+1±6.13e+0	3.01e+1±6.00e+0	2.11e+1±4.60e+0	3.50e+1±8.08e+0	2.36e+1±1.10e+1
NI+LS	T1	1.39e+2±2.57e+1	1.26e+2±1.81e+1	1.43e+2±4.01e+1	1.34e+2±2.57e+1	1.36e+2±2.82e+1	1.35e+2±3.00e+1	1.41e+2±3.36e+1
	T2	2.16e+3±3.91e+2	2.26e+3±4.45e+2	1.97e+3±4.53e+2	2.12e+3±4.81e+2	2.10e+3±4.72e+2	2.09e+3±3.96e+2	2.04e+3±4.25e+2

of the search space, fitness landscape, optimum in the unified space (x_u^*) and intertask similarity (R_s). In the original report, the intertask similarity is defined as Spearman's rank correlation coefficient between two tasks. For more details of the benchmark, interested readers can refer to [47].

As can be observed from Table II, the optimums x_u^* of the most of the problems are designed to locate in the center of the unified search space. However, this uniform design may incur misleading results in evaluating the performance of a mapping-based multitasking solver. A degraded mapping learned from the samples between the source–target instances tends to transform all the solutions from the source task into the central position of the samples from the target task. In particular, the degraded mapping with global sampling on the target instance will coincidently speed up the convergence of a multitasking optimizer in solving the problem with the central optimum. Therefore, the optimums of the original benchmark are modified here to facilitate a fair comparison between the solvers. In this study, the new optimums $\widehat{x}_u^*$ are sampled from a uniform distribution with a boundary $(0.25, 0.75)$ in the unified search space. The detailed design of the new optimums are shown in column 6 of the Table II.

1) Comparative Analysis of Optimization Performance: Table III summarizes the objective mean and standard deviation obtained by CEA, MFEA, MFEA-II, LDA-MFEA, AE-MFEA, G-MFEA, and AT-MFEA over 20 independent runs on the modified single-objective 2-task benchmark suite. Superior performance is highlighted in bold based on the Wilcoxon rank-sum test with 95% confidence level. A problem without any highlighted entry indicates that all the multitasking solvers show comparable performance in solving the problem. Overall, the proposed AT-MFEA shows superior performance over a wide range of problems as compared all other multitasking solvers, while CEA shows the worst performance. The results confirmed the efficacy of the solvers with knowledge transfer in solving multitasking optimization problems. In particular, AT-MFEA performs well on not only the homogeneous problems (i.e., complete intersection) but also the heterogenous problems (i.e., partial intersection or no intersection), which verified the effectiveness of the proposed domain adaptation method for a multitasking optimization solver.

In addition, we believe that as a general and effective mapping-based transferability enhancement technique, the proposed representation-based affine transformation can be embedded into other sophisticated EMT paradigms to further boost their performance.

Despite a good deal of successful performance improvements brought by AT-MFEA on the modified benchmark, it can be seen that there are also a few draws (e.g., $T2$ on the NI+HS problem) during the competition. The existing transferability enhancement methods are incapable of ensuring the performance improvement on various types of heterogeneous multitasking problems. Thus, a more sophisticated representation or an efficient mapping technique with a high learning capacity is desired to further uncover the latent similarities between heterogeneous problems, which will be the subject of our future work. In summary, the proposed AT-MFEA shows better generalizability and more robust improvement of optimization performance on a variety of multitasking problems as compared with a number of state-of-the-art multitasking solvers.

2) Comparative Analysis of Transfer's Reliability: To further investigate the transfer behaviors of the multiple multitasking solvers, the quality of transferred solutions in each

TABLE IV
SUMMARY OF PROPERTIES OF THE MANY-TASKING PROBLEM

Component Task	Search Space	Landscape	Global Optimum	Rotation
Sphere (T1)	$D = 50, x \in [-100, 100]^D$	unimodal, separable	$x_1 \sim U^D(0.25, 0.75)$	randomly generated orthogonal matrix
Griewank (T2)	$D = 50, x \in [-100, 100]^D$	multimodal, nonseparable	$x_2 \sim U^D(0.25, 0.75)$	
Rastrigin (T3)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$x_3 \sim U^D(0.25, 0.75)$	
Ackley (T4)	$D = 50, x \in [-50, 50]^D$	multimodal, nonseparable	$x_4 \sim U^D(0.25, 0.75)$	
Weierstrass (T5)	$D = 50, x \in [-0.5, 0.5]^D$	multimodal, nonseparable	$x_5 \sim U^D(0.25, 0.75)$	

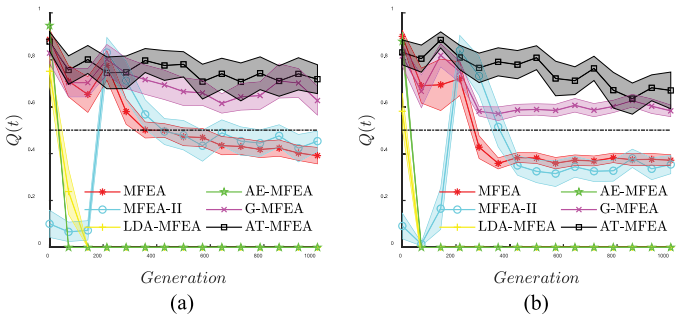


Fig. 3. Transfer performance of AT-MFEA against other EMTs on the PI+LS problem. (a) Task 1. (b) Task 2. The shaded area spans 1/3 standard deviation on either side of the mean.

generation are compared in details herein. Inspired by [50], a feedback-based metric for evaluating the quality of transferred solutions is proposed in this study, which is defined as follows:

$$Q_i^t = \frac{P_i^t + 1 - R_i^t}{P_i^t} \tag{31}$$

where $Q_i^t \in [0, 1]$ is a performance indicator for representing the quality of transferred solutions on the ith task at generation t, P_i^t denotes the total number of nontransferred individuals associated with ith task at generation t, and R_i^t represents the rank of transferred solutions with highest fitness among the individuals in the ith task at generation t.

It can be seen that the value of Q_i^t reflects the performance of transferred solutions on the target instance. In particular, the promising transferred individual shows dominated performance on the target problem if $Q_i^t = 1$, while $Q_i^t = 0$ indicates that there is no transferred individual performs better than the nontransferred solutions. Therefore, a reference value $Q = 0.5$ can be used to distinguish the positive effect and the negative effect of knowledge transfer.

With the proposed metric, the transfer performance of six EMT solvers on the PI+LS problem are analyzed in details here.

To present a clear illustration, the transfer performance (i.e., Q) of AT-MFEA against other multitasking solvers are shown in Fig. 3. A dashed black line is plotted to represent the reference level of the Q value. It can be seen from Fig. 3 that the proposed method achieves the highest Q value as compared to other algorithms. The quality of transferred solutions offered by MFEA decreases to a reference level as the generation increases, while the transfer performance of MFEA-II fluctuates a lot in the early stage and gradually remains stable around the reference level due to the adaptive estimation mechanism of *rmp*. As for the pairwise-learning-based transfer methods, Fig. 3 clearly shows that the LDA-MFEA and AE-MFEA show poor transfer performance on the two component tasks. LDA-MFEA and AE-MFEA are incapable of providing promising transferred solutions to the target domain due to the degradation of learned intertask mapping. It can be observed that G-MFEA shows competitive performance as compared to AT-MFEA on the first component task. However, the performance of G-MFEA in terms of transfer stability is significantly worse than AT-MFEA on the second task. This difference may be attributed to the incomplete representation or incomplete transformation adopted by G-MFEA. In summary, the quality of transferred solutions offered by AT-MFEA lies at a superior level across the entire optimization process in comparison with other state-of-the-art methods.

C. Many-Tasking Case Study

According to the configuration of the single-objective multitasking benchmark, five different commonly used single-objective continuous optimization functions, that is, "Sphere," "Griewank," "Rastrigin," "Ackley," and "Weierstrass," which are labeled as task 1 ($T1$)–task 5 ($T5$), respectively, are employed here to develop a many-tasking optimization problem. The detailed properties of these component optimization tasks, including search space, landscape, and global optimum, are summarized in Table IV. To increase the heterogeneity of the component problems, random shift and rotation (see Table IV) is performed on them independently. In such a scenario, without any task selection strategy, it is a challenging task for the EMT method to leverage the intertask synergies due to the strong intertask heterogeneity. The study of the task selection method is beyond the scope of this study, as such a strategy can be employed to cooperate with the proposed transferability enhancement method with ease using a feedback-based reward and punishment mechanism. Thus, the efficacy of CEA and six EMTs are compared on this heterogeneous many-tasking problem without any task selection module.

The average convergence curves of CEA and six EMTs across 20 independent runs on the many-tasking problem are depicted in Fig. 4. On $T1$ and $T2$, it can be observed that the proposed AT-MFEA shows better convergence performance than the canonical single-task solver, while MFEA-II shows comparable performance as compared to CEA. In contrast, MFEA and the other three heterogeneous EMTs (i.e., LDA-MFEA, AE-MFEA, and G-MFEA) suffer from the negative transfer due to the problem heterogeneity. However, on $T3$, all

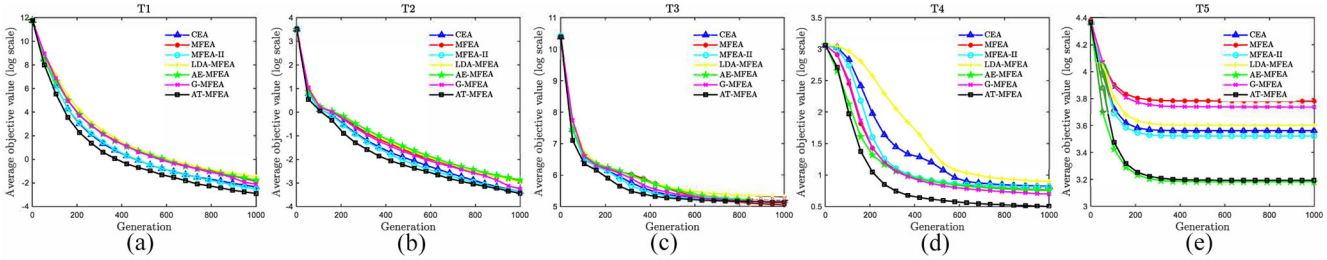


Fig. 4. Average convergence curves of CEA, MFEA, MFEA-II, LDA-MFEA, AE-MFEA, G-MFEA, and AT-MFEA over 20 independent runs on the many-tasking problem. (a) Task 1. (b) Task 2. (c) Task 3. (d) Task 4. (e) Task 5.

EMT solvers, including CEA show comparable performance. The knowledge transferred from the other tasks fails to accelerate the problem-solving efficiency on $T3$. Nevertheless, the knowledge transfer among the tasks does not have a negative impact. Notably, AT-MFEA achieves best convergence performance on $T4$. The superior performance of AT-MFEA and AE-MFEA together with the poor performance of LDA-MFEA and G-MFEA on $T5$ indicate that an incomplete mapping may be unable to bridge the gap between the heterogeneous problems. As all EMT solvers in this study possess the same algorithmic structures and search operators, the obtained results confirmed the efficacy of the proposed transferability enhancement technique for heterogeneous problems in evolutionary many-tasking.

D. Practical Case From the Petroleum Industry

One of the potential application scenarios of our proposed AT-MFEA method is expected to be the domain of the complex engineering design where a variety of models or products, often with a plethora of latent similarities, are required to be optimized simultaneously. The design cycle across multiple distinct problems can be effectively reduced by exploiting the latent synergies between them.

In particular, the above improvement is of significant value in real-world optimization problems which involves high-fidelity numerical simulations. As is well known, such simulations are often computationally time-consuming, which may take from several min to a few hours to complete a single run. To this end, the efficacy of AT-MFEA against other state-of-the-art EMTs is demonstrated on a real-world finite difference simulation-based multitasking optimization instance from the petroleum industry. Before we proceed the details of our experimental settings, a brief introduction about the well location optimization problem is presented.

Given a particular reservoir of interest, well location optimization aims to enhance the ultimate oil recovery substantially by adopting a set of optimal well locations. The optimization objective for each component problem is defined as the net present value (NPV) for measuring the economic benefit for the life-cycle production [51], which is given as follows:

$$\max_{x} \ \mathcal{NPV}(x, v_k, d_k) \tag{32}$$

where $\mathcal{NPV}(\cdot)$ is an NPV evaluation function that relies on the numerical simulation, x denotes the well locations, v_k is a high-dimensional feature vector used to characterize the kth reservoir model, d_k represents the development schema of the kth reservoir model.

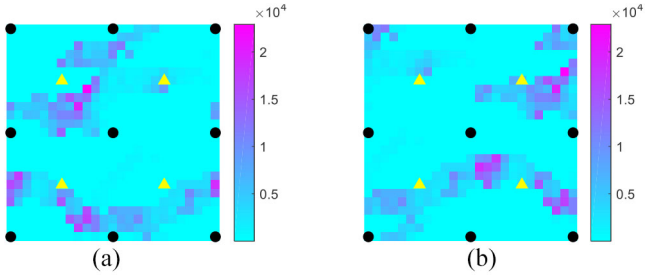


Fig. 5. Illustration of permeability distributions and well locations for the two reservoir models. (a) First reservoir model. (b) Second reservoir model. The production wells and injection wells are represented as black circles and yellow triangles, respectively.

In this demonstration, we consider the multitasking optimization of well locations corresponding to two different reservoir models that share some latent similarities. Two single-layer reservoir models with distinct permeability distributions are adopted as the component tasks, which are illustrated in Fig. 5. For the sake of brevity, the details of reservoir properties and the finite-difference-based reservoir numerical simulation are not reported herein, but can be found in [52] and [53]. Despite the clear distinctions characterized by different permeability realizations, there is also expected to exist some latent similarities between the two well location optimization problems. Such a scenario is really common in many real-world problems, which can be adopted as a testing platform for investigating the optimization performance of multiple EMTs. We can see from Fig. 5 that each component reservoir model contains nine producers and four injectors. In this work, the well locations of four injectors are optimized within their feasible regions to achieve higher NPV. The production wells are operated at constant bottom hole pressure of 1800 psi, while the injectors are operated at constant injection rates 1500 STB/D. The production life time is 1800 days.

Fig. 6 shows the NPV mean obtained by CEA, MFEA, MFEA-II, LDA-MFEA, AE-MFEA, G-MFEA, and AT-MFEA over 20 independent runs on the multitasking well location optimization problem. On the first model illustrated in Fig. 6(a), we can see that the proposed AT-MFEA method shows dominated convergence performance among seven EMT solvers, while the rest of MFEAs perform poorly in comparison with the CEA method due to the problem heterogeneity. Fig. 6(b)

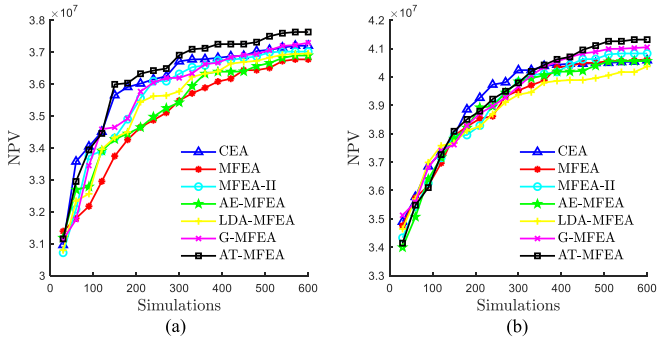


Fig. 6. Convergence curves of CEA and six MFEAs on the multitasking well location optimization problem. (a) First model. (b) Second model. The solid line with marks represent the average objective value across 20 independent runs.

shows that all EMT solvers show comparable convergence performance at the early optimization stage on the second model. Notably, AT-MFEA achieves higher NPV as compared to other methods when the computational resources run out. The promising outcome demonstrates the potential application of the affine-transformation-based domain adaptation for heterogeneous engineering design optimization problems.

V. Conclusion

In this article, we first classified various EMTs into two categories: 1) homogeneous transfer methods and 2) heterogeneous transfer methods. Apart from some intrinsic defects (e.g., incomplete transformation) of the existing heterogeneous transfer methods, an undiscovered but severe issue, called the degradation of intertask mapping, was proposed and discussed in this work. This phenomenon is caused by the chaotic matching between the source and target samples. To address the issues faced by the existing heterogeneous EMTs, two key techniques were proposed in this article, which are summarized as follows.

1) A novel rank loss function for acquiring an intertask mapping was proposed.
2) With a progressional Gaussian representation model, an analytical solution of affine transformation for bridging the gap between two distinct problems was mathematically derived.

The efficacy of the proposed transferability enhancement technique against a number of state-of-the-art methods under the canonical MFEA framework was confirmed on a modified single-objective 2-task benchmark suite and a heterogeneous many-tasking problem, as well as a practical case from the petroleum industry.

In future study, a more sophisticated representation model and a transformation with high learning capacity may be considered. In addition, we would like to generalize the proposed AT-MFEA to the multiobjective optimization and the expensive optimization.

References

[1] T. Back, U. Hammel, and H. Schwefel, "Evolutionary computation: Comments on the history and current state," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 3–17, Apr. 1997.

[2] H. Maaranen, K. Miettinen, and A. Penttinen, "On initial populations of a genetic algorithm for continuous optimization problems," *J. Global Optim.*, vol. 37, no. 3, pp. 405–436, 2007.

[3] X. Xue *et al.*, "A topology-based single-pool decomposition framework for large-scale global optimization," *Appl. Soft Comput.*, vol. 92, Jul. 2020, Art. no. 106295.

[4] Y. Sun, M. Kirley, and S. K. Halgamuge, "A recursive decomposition method for large scale continuous optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 5, pp. 647–661, Oct. 2018.

[5] K. C. Tan, T. H. Lee, 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, pp. 115–151, 2006.

[6] M. Gong, H. Li, E. Luo, J. Liu, and J. Liu, "A multiobjective cooperative coevolutionary algorithm for hyperspectral sparse unmixing," *IEEE Trans. Evol. Comput.*, vol. 21, no. 2, pp. 234–248, Apr. 2017.

[7] S. Nguyen, M. Zhang, M. Johnston, and K. C. Tan, "Automatic design of scheduling policies for dynamic multi-objective job shop scheduling via cooperative coevolution genetic programming," *IEEE Trans. Evol. Comput.*, vol. 18, no. 2, pp. 193–208, Apr. 2014.

[8] S. Nguyen, M. Zhang, and K. C. Tan, "Surrogate-assisted genetic programming with simplified models for automated design of dispatching rules," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2951–2965, Sep. 2017.

[9] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[10] L. Zhang *et al.*, "Cooperative artificial bee colony algorithm with multiple populations for interval multiobjective optimization problems," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 5, pp. 1052–1065, May 2019.

[11] X. Ma, Y. Yu, X. Li, Y. Qi, and Z. Zhu, "A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 24, no. 4, pp. 634–649, Aug. 2020.

[12] H. Teng, Y. Chen, W. Zeng, Y. Shi, and Q. Hu, "A dual-system variable-grain cooperative coevolutionary algorithm: Satellite-module layout design," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 438–455, Jun. 2010.

[13] Y. Ong, P. B. Nair, and K. Y. Lum, "Max–min surrogate-assisted evolutionary algorithm for robust design," *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 392–404, Aug. 2006.

[14] L. Zhang, C. Cui, X. Ma, Z. Sun, and K. Zhang, "A fractal discrete fracture network model for history matching of naturally fractured reservoirs," *Fractals*, vol. 27, no. 1, 2019, Art. no. 1940008.

[15] X. Ma *et al.*, "Multiscale-network structure inversion of fractured media based on a hierarchical-parameterization and data-driven evolutionary-optimization method," *SPE J.*, vol. 25, no. 5, pp. 2729–2748, 2020.

[16] S. M. K. Hasan, R. Sarker, D. Essam, and D. Cornforth, "Memetic algorithms for solving job-shop scheduling problems," *Memetic Comput.*, vol. 1, no. 1, pp. 69–83, 2009.

[17] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.

[18] S. Ozawa, A. Roy, and D. Roussinov, "A multitask learning model for online pattern recognition," *IEEE Trans. Neural Netw.*, vol. 20, no. 3, pp. 430–445, Mar. 2009.

[19] X. Tian, Y. Li, T. Liu, X. Wang, and D. Tao, "Eigenfunction-based multitask learning in a reproducing kernel hilbert space," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 6, pp. 1818–1830, Jun. 2019.

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

[21] J. Rice, C. R. Cloninger, and T. Reich, "Multifactorial inheritance with cultural transmission and assortative mating. I. Description and basic properties of the unitary models," *Amer. J. Hum. Genet.*, vol. 30, no. 6, pp. 618–643, 1978.

[22] C. R. Cloninger, J. Rice, and T. Reich, "Multifactorial inheritance with cultural transmission and assortative mating. II. A general model of combined polygenic and cultural inheritance," *Amer. J. Hum. Genet.*, vol. 31, no. 2, pp. 176–198, 1979.

[23] Z. Tang, M. Gong, F. Jiang, H. Li, and Y. Wu, "Multipopulation optimization for multitask optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Wellington, New Zealand, 2019, pp. 1906–1913.

[24] M. Gong, Z. Tang, H. Li, and J. Zhang, "Evolutionary multitasking with dynamic resource allocating strategy," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 858–869, Oct. 2019.

[25] L. Zhou, L. Feng, J. Zhong, Y. Ong, Z. Zhu, and E. Sha, "Evolutionary multitasking in combinatorial search spaces: A case study in capacitated vehicle routing problem," in *Proc. IEEE Symp. Series Comput. Intell. (SSCI)*, Athens, Greece, 2016, pp. 1–8.

[26] A. Gupta, Y. 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.

[27] Q. Chen, X. Ma, Z. Zhu, and Y. Sun, "Evolutionary multi-tasking single-objective optimization based on cooperative co-evolutionary memetic algorithm," in *Proc. 13th Int. Conf. Comput. Intell. Security (CIS)*, Hong Kong, China, 2017, pp. 197–201.

[28] 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.

[29] A. Gupta, Y. 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.

[30] L. Feng, Y. Ong, S. Jiang, and A. Gupta, "Autoencoding evolutionary search with learning across heterogeneous problems," *IEEE Trans. Evol. Comput.*, vol. 21, no. 5, pp. 760–772, Oct. 2017.

[31] B. Da, A. Gupta, and Y. Ong, "Curbing negative influences online for seamless transfer evolutionary optimization," *IEEE Trans. Cybern.*, vol. 49, no. 12, pp. 4365–4378, Dec. 2019.

[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.

[33] K. K. Bali, Y. 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.

[34] J. Zhang, W. Zhou, X. Chen, W. Yao, and L. Cao, "Multisource selective transfer framework in multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 24, no. 3, pp. 424–438, Jun. 2020.

[35] Y. Chen, S. Song, S. Li, L. Yang, and C. Wu, "Domain space transfer extreme learning machine for domain adaptation," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1909–1922, May 2019.

[36] Y. Luo, Y. Wen, and D. Tao, "Heterogeneous multitask metric learning across multiple domains," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 9, pp. 4051–4064, Sep. 2018.

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

[38] 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.

[39] F. Liu, G. Zhang, and J. Lu, "A novel fuzzy neural network for unsupervised domain adaptation in heterogeneous scenarios," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, New Orleans, LA, USA, 2019, pp. 1–6.

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

[41] F. Zhuang *et al.*, "A comprehensive survey on transfer learning," 2019. [Online]. Available: arXiv:1911.02685.

[42] 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.

[43] J. Zhong, L. Feng, W. Cai, and Y. Ong, "Multifactorial genetic programming for symbolic regression problems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 11, pp. 4492–4505, Nov. 2020.

[44] X. Ma, Q. Chen, Y. Yu, Y. Sun, and Z. Zhu, "A two-level transfer learning algorithm for evolutionary multitasking," *Front. Neurosci.*, vol. 13, p. 1408, Jan. 2020.

[45] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1134–1148, Jun. 2014.

[46] L. Zhou and L. Ma, "Extreme learning machine-based heterogeneous domain adaptation for classification of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 11, pp. 1781–1785, Nov. 2019.

[47] B. Da *et al.*, "Evolutionary multitasking for single-objective continuous optimization: Benchmark problems, performance metric, and baseline results," Dept. Comput. Sci., Nanyang Technol. Univ., Singapore, Rep., 2016.

[48] J. Yin, A. Zhu, Z. Zhu, Y. Yu, and X. Ma, "Multifactorial evolutionary algorithm enhanced with cross-task search direction," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Wellington, New Zealand, 2019, pp. 2244–2251.

[49] T. Hastie, J. Friedman, and R. Tibshirani, *The Elements of Statistical Learning*. Beijing, China: World Book, Inc., 2015.

[50] Q. Shang *et al.*, "A preliminary study of adaptive task selection in explicit evolutionary many-tasking," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Wellington, New Zealand, 2019, pp. 2153–2159.

[51] K. Zhang *et al.*, "Current status and prospect for the research and application of big data and intelligent optimization methods in oil-field development," *J. China Univ. Petrol. Ed. Nat. Sci.*, vol. 44, no. 4, pp. 28–38, 2020.

[52] G. Chen, K. Zhang, L. Zhang, X. Xue, and Y. Yang, "Global and local surrogate-model-assisted differential evolution for waterflooding production optimization," *SPE J.*, vol. 25, no. 01, pp. 105–118, 2019.

[53] D. W. Peaceman, "Fundamentals of numerical reservoir simulation," in *Developments in Petroleum Science*, vol. 1. New York, NY, USA: Elsevier Sci. Publ., 1977, pp. 18–23.

Xiaoming Xue received the B.S. degree in petroleum engineering and the M.Sc. degree in oil and gas field development engineering from the China University of Petroleum (East China), Qingdao, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the Department of Computer Science, City University of Hong Kong, Hong Kong.

His research interests include large-scale optimization, evolutionary transfer optimization, transfer learning, and their applications on engineering design optimization problems.

Kai Zhang (Member, IEEE) received the Ph.D. degree in petroleum engineering from the China University of Petroleum (East China), Qingdao, China, in 2008.

From June 2007 to May 2008, he studied with the University of Tulsa, Tulsa, OK, USA. He has been a Teacher with the China University of Petroleum (East China) since 2008. He teaches courses, including fluid flow in porous media and reservoir engineering. As a Project Leader, he has been in charge of three projects supported by the Natural Science Foundation of China, one project supported by the National Natural Science Foundation of Shandong Province, and 20 projects supported by SINOPEC, CNOOC, and CNPC. He has already published more than 60 papers. His research focuses on reservoir simulation, production optimization, history matching, and development of nonconventional reservoir.

Kay Chen Tan (Fellow, IEEE) received the B.Eng. degree (First Class Hons.) in electronics and electrical engineering 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. 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 as the editorial board member of over ten journals. He was an Elected Member of the IEEE CIS AdCom from 2017 to 2019.

Liang Feng received the Ph.D. degree in computer engineering from Nanyang Technological University, Singapore, in 2014.

He is currently a Professor with the College of Computer Science, Chongqing University, Chongqing, China. His research interests include computational and artificial intelligence, memetic computing, and big data optimization and learning, as well as transfer learning.

Dr. Feng's research work on evolutionary multitasking won the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award in 2019. He is an Associate Editor of the *IEEE Computational Intelligence Magazine*, *Memetic Computing*, and *Cognitive Computation*. He is also the Founding Chair of the IEEE CIS Intelligent Systems Applications Technical Committee Task Force on "Transfer Learning and Transfer Optimization."

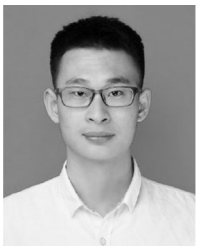
Jian Wang (Member, IEEE) received the Ph.D. degree in computational mathematics from the Dalian University of Technology, Dalian, China, in 2012.

He is currently an Associate Professor and serves as the Head of the Laboratory for Intelligent Information Processing with the College of Science, China University of Petroleum (East China), Qingdao, China. His current research interests include pattern recognition, supervised learning, regularization theory, and neural networks.

Dr. Wang was a recipient of the Natural Science Academic Achievement Award of Liaoning Province in 2012 and 2013, separately. In recent years, he has focused on investigating the deterministic (asymptotic) convergent behaviors of supervised learning schemes based on neural network models, and feature selection in high dimensional data based on regularization theory. He served as the Publication Chair of the 24th International Conference on Neural Information Processing. He is an Editor of the *Neural Computing and Applications*, an Associate Editor of the *Journal of Applied Computer Science Methods*, and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.

Guodong Chen received the Bachelor of Petroleum Engineering degree from the China University of Petroleum (East China), Qingdao, China, in 2018, where he is currently pursuing the postgraduation degree in oil and gas field development engineering.

His research interests include data-driven evolutionary optimization, high-dimensional expensive optimization, and oil reservoir production optimization.



Xinggang Zhao received the B.S. degree in engineering from the Chengdu University of Technology, Chengdu, China, in 2018. He is currently pursuing the M.S. degree with the China University of Petroleum (East China), Qingdao, China.

His current research interests include evolutionary optimization, machine learning, and reservoir production optimization.

Liming Zhang received the Ph.D. degree in petroleum engineering from the China University of Petroleum (East China), Qingdao, China, in 2009.

From August 2008 to September 2009, she studied with the University of Leeds, Leeds, U.K. She has been a Teacher with the China University of Petroleum (East China) since 2009. She teaches courses, including oil production engineering and multiphase flow theory and computation. She has already published more than 20 papers. Her research focuses on multiphase flow and separation, profile modification and water shutoff, and reservoir simulation.

Jun Yao received the Ph.D. degree in petroleum engineering from the China University of Petroleum (East China), Qingdao, China, in 2002.

He is currently a Vice President of the China University of Petroleum (East China), Qingdao, China, the Director of the Center of Multiphase Flow in Porous Media, a Committee Member of the Academic Degrees Committee of the State Council, a Chairman of Chinese Society for Fluid Flow in Porous Media, the Head of "Cheung Kong Scholars" Innovative Research Team by National Ministry of Education, and a Member of the New Century National Talent Project, Taishan Scholars Climbing Program of Shandong Province. He proposed a modern system of multiphase flow in porous media, micro-scale and large-scale, well testing interpretation, and oil-field production optimization. He has already published more than 200 papers.