Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(5): 8261–8278.
DOI: 10.3934/mbe.2023360
Received: 13 January 2023
Revised: 14 February 2023
Accepted: 21 February 2023
Published: 28 February 2023

Research article

A multitask optimization algorithm based on elite individual transfer

Yutao Lai, Hongyan Chen and Fangqing Gu*

School of Mathematics and Statistics, Guangdong University of Technology, Guangzhou, China

* **Correspondence:** Email: fqgu@gdut.edu.cn.

Abstract: Evolutionary multitasking algorithms aim to solve several optimization tasks simultaneously, and they can improve the efficiency of various tasks evolution through the knowledge transfer between different optimization tasks. Evolutionary multitasking algorithms have been applied to various applications and achieved certain results. However, how to transfer knowledge between tasks is still a problem worthy of research. Aiming to improve the positive transfer between tasks and reduce the negative transfer, we propose a single-objective multitask optimization algorithm based on elite individual transfer, namely MSOET. In this paper, whether to execute knowledge transfer between tasks depends on a certain probability. Meanwhile, in order to enhance the effectiveness and the global search ability of the algorithm, the current population and the elite individual in the transfer population are further utilized as the learning sources to construct a Gaussian distribution model, and the offspring is generated by the Gaussian distribution model to achieve knowledge transfer between tasks. We compared the proposed MSOET with ten multitask optimization algorithms, and the experimental results verify the algorithm's excellent performance and strong robustness.

Keywords: multitask optimization; Gaussian distribution; knowledge transfer; evolutionary algorithm; evolutionary multitasking algorithms

1. Introduction

Multitask optimization (MTO) solves multiple optimization tasks simultaneously in the evolutionary algorithm to improve the performance of solving each task independently via intertask knowledge transfer [1, 2]. Without loss of generality, an MTO can be expressed as [3, 4]:

$$\{\mathbf{x}_1^*, \cdots, \mathbf{x}_K^*\} = \arg \min_{(\mathbf{x}_1, \cdots, \mathbf{x}_K)} \{f_1(\mathbf{x}_1), \cdots, f_K(\mathbf{x}_K)\} \tag{1.1}$$

where $f_k(\mathbf{x}_k)$ is the objective function of kth optimization task, and $\mathbf{x}_k = (x_1, x_2, ..., x_{D_k}) \in R^{D_k}$ is a D_k-dimensional decision variable in space R. In other words, each optimization task is to solve the minimum optimization problem of an objective function. Single-objective multitask optimization problem aims to solve multiple single-objective problems at the same time by sharing common features that are beneficial to make a faster convergence between related tasks, which greatly improves the efficiency of the algorithm [5–7]. Moreover, some difficult single-objective problems can also be solved effectively by employing the correlation between optimization tasks.

Many problems in the real world can be regarded as single-objective multitask optimization problems [8]. For instance, a multi-population-based evolutionary multitasking optimization algorithm is presented for dynamic flexible job-shop scheduling in [9]. A task-oriented knowledge-sharing strategy is presented and achieves good performance by maintaining the quality and diversity of individuals for corresponding tasks well. A multitasking harmony search algorithm is proposed for detecting high-order single nucleotide polymorphisms epistatic interactions in [10]. A unified coding strategy is adopted for multiple tasks. A multifactor genetic programming framework is presented in [11] for solving symbolic regression problems. A knowledge adoption-based evolutionary multitask algorithm is presented for the profitable tour problem in [12]. Evolutionary algorithms have been successfully applied to solve a variety of single- and multi-objective optimization problems [13, 14]. Because of the good application prospects of multitask optimization, researchers have focused on this problem and developed various efficient multitask optimization algorithms in recent years [15, 16].

Leveraging task correlation to improve search efficiency is a critical problem in evolutionary multitasking algorithms. Meanwhile, many pieces of research have been on enhancing the positive transfer and reducing the negative transfer of evolutionary multitasking algorithms. Through the above discussion, these algorithms can be roughly divided into three categories according to the way of task collaboration: domain adaptation, individual transfer and knowledge transfer-based algorithms [17].

Domain adaptation-based methods [18] map the search spaces of the source task and the target task into a unified search space to enhance the positive transfer of the evolutionary multitask algorithms. For instance, the multi-factor evolutionary optimization algorithm (MFEA) [19] is one of the most representative algorithms in this category that transfers knowledge between tasks through the unified space and transfers useful genetic information from one task to another. A novel evolutionary multitasking algorithm based on subspace alignment is proposed in [20]. Similarly, a mapping matrix obtained by subspace learning is used to transform the search space of the population and reduce the probability of negative knowledge transfer between tasks. Lately, Liu et al. [21] proposed a discriminative reconstruction network model that is used to transfer useful knowledge during the reproduction of offspring. It can enhance the quality of learned knowledge to promote positive transfer. However, realizing space alignment and mapping the search spaces of multiple tasks into a unified search space is still a very challenging problem. Meanwhile, the computing cost of the domain adaptation-based methods should be noticed.

Knowledge transfer-based algorithms share some parameters of prior distributions of hyperparameters [22–24] to realize knowledge transfer between optimization tasks, which are easy to execute. What knowledge, namely parameters, is transferable is a key issue for knowledge transfer-based algorithms [25]. Therefore, a new knowledge transfer strategy based on inter-task gene similarity is proposed in [26]. It can perform more fine-grained and accurate knowledge transfer. An adaptive scheme combining transfer learning with evolutionary methods is presented for expensive problems in [27]. It reuses knowledge by sharing a Gaussian process model. However, the above strategies are pretty time-consuming. Hence, a gradient-free evolutionary multitask algorithm (MTES) [28] is designed based on a multitasking gradient algorithm (MTGD) to overcome the defect. The evolutionary single-task strategy (ES) has an approximate gradient decline, and MTGD also has a fast decline rate. Therefore, MTES has a faster convergence rate. Paper [29] proposed a knowledge transfer strategy based on local distribution estimation (LEKT). However, these algorithms will likely get stuck in a local optimal solution when dealing with multimodal problems.

Individual transfer-based algorithms aim to transfer some solutions from source tasks to target tasks to enhance the convergence speed of the target tasks [30]. For instance, a decision variable translation strategy and a decision variable shuffling strategy are proposed in [31] for solving expensive optimization problems. Recently, a self-adaptive multitask evolutionary algorithm is proposed in [32]. It adaptively adjusts the transfer rate of the population to reduce the harm of negative transfer and balance the diversity and convergence within the population. In paper [33], two multitasking optimization algorithms of Multifactorial Optimization (MFO), popular particle swarm optimization (MFPSO) and differential evolution search (MFDE), are proposed to explore the versatility of MFO in multitasking optimization. There are also some algorithms in this category proposed to solve multi-objective problems, and a transfer rank is defined in [34], which quantifies the priority of transferred solutions to improve the probability of a positive transfer. Transferred solutions are selected from the neighbors of solutions that achieved the positive transfer in [35], and an incremental learning method is used to select the transferred solutions in [30]. An effective transferred solutions selection mechanism is essential for enhancing the convergence characteristic of their target tasks. Most individual transfer-based algorithms randomly select individuals to be transferred, and the main focus is on when to transfer and how many individuals are to be transferred. There is a lack of research on which individuals are to be transferred. Also, reducing the notorious negative transfer of these algorithms should also be treated cautiously.

In addition, to improve the effectiveness of the knowledge transfer, an evolutionary multitasking optimization algorithm based on a multi-source knowledge transfer mechanism (EMaTOMKT) [36] uses adaptive mating probability (AMP) strategy to determine the probability of knowledge transfer and uses task selection (MTS) strategy based on maximum mean discrepancy (MMD) [29] to select multiple similar tasks as learning sources. [37, 38] proposed a new evolutionary multitasking algorithm with two-stage adaptive knowledge transfer based on population distribution. A novel method of extracting and transferring knowledge is proposed to reduce the probability of generating negative transfer. A task selection strategy [39] based on credit assignment is to select a task for knowledge transfer by leveraging the feedback from the transferred solutions across tasks. In MaTEA [40], an adaptive mechanism is used to select the auxiliary tasks to achieve knowledge transfer between tasks. This adaptive mechanism is realized by dynamically measuring the similarity between tasks and analyzing the effectiveness of knowledge transfer between tasks.

Aiming to execute more effective and robust knowledge transfer between various optimization tasks, we propose a single-objective multitask optimization based on elite individual transfer, namely, MSOET, in this paper. Considering the effectiveness of the algorithm EMaTOMKT [36] in solving the single-objective many-task problem, this paper quotes the idea of using the parameters of the Gaussian distribution model to generate offspring to achieve knowledge transfer. We borrow this idea to enhance the global search ability of the algorithm. However, unlike how EMaTOMKT directly employs the clustering technique and the estimation of distribution algorithms to generate the offspring, the core of the MSOET is selecting the appropriate auxiliary tasks and transferring the common features of the elite population selected from auxiliary tasks. To improve the positive transfer probability between optimization tasks, in this paper, only the individuals in the transfer population which are better than the optimal individuals in the target population are selected for knowledge transfer. The main contributions of this paper are as follows.

- Select the individuals in the transfer population for transfer: Select the individuals which are better than the target tasks for transfer, which can improve the efficiency of knowledge transfer.
- The elite individuals transferred from the auxiliary tasks are combined with the current population to build a Gaussian distribution model, which can guide the population to converge faster and help to enhance the robustness of the algorithm, to produce offspring individuals.

The rest of this paper is organized as follows. Section 2 introduces the proposed algorithm framework. In Section 3, we experimentally compare MSOET with ten evolutionary multitask optimization algorithms on nine test instances. Finally, Section 4 concludes this paper.

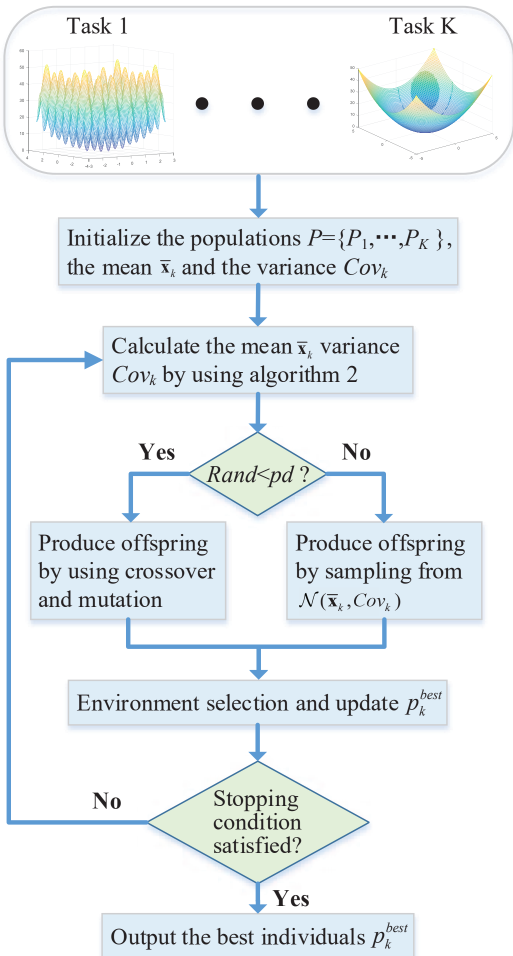


Figure 1. The flowchart of the proposed multitask optimization algorithm.

2. The proposed multitask optimization based on elite individual transfer

In this section, we first give the overall framework of the proposed MSOET algorithm and then present a detailed description of the elite knowledge transfer strategy. Finally, we provide an analysis of the proposed algorithm.

2.1. The framework of the proposed algorithm

First, different tasks may have different search spaces. The search space of all tasks is mapped to a unified search space according to Eq (2.1).

$$y_{ki} = \frac{x_{ki} - L_{ki}}{U_{ki} - L_{ki}}, k = 1, \cdots, K, i = 1, \cdots, D_k \tag{2.1}$$

where x_{ki} is the value of solution $\mathbf{x}$ on the i-th dimension of kth task, and U_{ki} and L_{ki} are the upper and lower bounds on i-th dimension of k-th task, respectively. y_{ki} is the value mapped from x_{ki} to the unified search space. After such a transformation, the search space of all tasks is unified to $[0, 1]^D$, where $D = \max\limits_{1 \le k \le K} \{D_k\}$. Thus, without loss of generality, we assume that all tasks have the same search space $X = [0, 1]^D$. Moreover, a skill factor is introduced for each individual to facilitate the assignment of fitness values and the comparison of individuals [19]. The skill factor τ_i of p_i is one of the component tasks, where p_i achieves the best rank among all tasks, that is,

$$\tau_i = \arg \min_{1 \le k \le K} \{r_{ik}\}, \tag{2.2}$$

where r_{ik} is the factorial rank, which is the index of p_i in the list of population members that are sorted in ascending order according to the value of the objective function f_k of the kth task.

The framework of the proposed MSOET is given in Algorithm 1. The flowchart of this algorithm is shown in Figure 1. First, we randomly initialize the population $P = \{P_1, \cdots, P_K\}$ with a size of N in the unified search space X and set the initial generation $t = 0$. The skill factors of individuals are assigned with a random number $k \in \{1, \cdots, K\}$, and the total number of individuals with skill factors of k is N. Record the optimal individual of the current population as p_k^{best}, $k = 1, \cdots, K$. The optimal individual of single-objective optimization refers to the individual with the minimum function value in the population. When the termination condition is not satisfied, the probability model of the subpopulation is constructed by Algorithm 2.

Then, choose different ways to produce offspring according to a certain probability pd (the value of pd can be found in the parameter setting part of the experiment). If $rand < pd$, for solution p_k^j, two offspring o_1 and o_2 are generated by using crossover and mutation on p_k^j and p_k^r which are randomly chosen from P_k. Otherwise, o_1 and o_2 are sampled from a Gaussian distribution model $\mathcal{N}(\overline{x}_k, Cov_k)$, where $\overline{x}_k$ is the mean, and Cov_k is the variance of the Gaussian distribution model obtained by Algorithm 2. $rand$ is a uniform random number in $[0, 1]$. In this way, knowledge transfer between tasks can be realized. The skill factors of o_1 and o_2 inherit that of their parents, and then evaluate offspring o_1 and o_2 in terms of the corresponding task. Population P and offspring Q are merged, and the best N individuals are selected as the next generation population. Finally, update the optimal individual p_k^{best} of the population. The optimal individual p_k is found. If the function value of p_k is smaller than that of p_k^{best}, which means that p_k is better than p_k^{best}, then replace P_k^{best} with p_k. Otherwise, p_k^{best} remains the same, $k = 1, \cdots, K$.

Algorithm 1: The Pseudocode of MSOET

input :
1. The population size N;
2. The mating probability pb.

output: The best individuals p_k^{best}, $k = 1, \cdots, K$.

◇ Initialize and evaluate the population $P = \{P_1, \cdots, P_K\}$ in the unified search space, $t = 0$;
◇ Initialize p_k^{best} with minimum function values in populations P_k, $k = 1, \cdots, K$;
while *the stopping criterion is unsatisfied* **do**
    for $k = 1$ *to* K **do**
        $(\overline{\mathbf{x}}_k, Cov_k) = SKT(P, k) \leftarrow$ **Algorithm 2**
        for *j=1 to* $N/2$ **do**
            if $rand < pd$ **then**
                ◇ Select the j-th individual p_k^j from P_k as the first parent;
                ◇ Randomly select an individual p_k^r from P_k as another parent;
                ◇ Two offspring o_1 and o_2 are produced with crossover and mutation on p_k^j and p_k^r.
            else
                ◇ Two offspring o_1 and o_2 are produced by sampling from $\mathcal{N}(\overline{\mathbf{x}}_k, Cov_k)$.
            end
            ◇ Evaluate o_1 and o_2 and let them inherit the skill factors of their parents;
            ◇ Put o_1 and o_2 into offspring population $O_i = O_i \cup o_1 \cup o_2$.
        end
        ◇ Choose the best N individual from $P_k \cup O_k$ to form the next population;
        ◇ Find the best individual p_k in $P_k \cup O_k$;
        if $f(p_k) < f(p_k^{best})$ **then**
            $p_k^{best} = p_k \leftarrow$ **Update** p_k^{best}
        end
    end
    ◇ Set $t = t + 1$;
end

2.2. *Knowledge transfer strategy based on elite individual transferred*

The selection of the transferring individuals is very important to improve the efficiency of the algorithm and avoid negative transfer in multitask optimization. Therefore, we propose a knowledge transfer strategy based on elite individual transfer in this paper. Algorithm 2 is the main innovation of this paper, and the flow chart of Algorithm 2 is shown in Figure 2.

First of all, we find out the population of the source task of each task, which is used for transferring. We calculate the Kendall correlation coefficient of the factorial rank of the population member. The Kendall correlation coefficient is one of the three most famous correlation coefficients in statistics, and it has been applied to a wide range of applications. Hence, we adopt the Kendall correlation coefficient

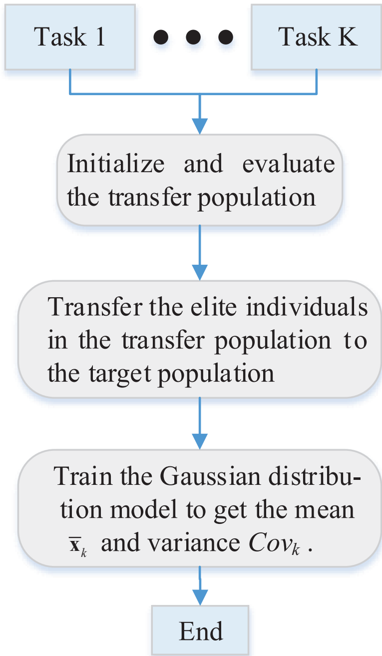


Figure 2. The flowchart of the knowledge transfer strategy based on elite individual transferred.

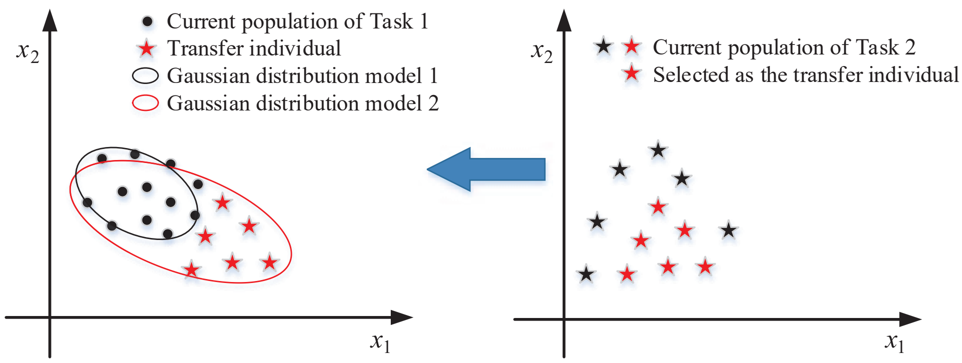


Figure 3. Performance analysis of elite knowledge transfer.

to measure the similarity between tasks.

$$\rho_{kl} = \frac{1}{2}\frac{C-D}{N(N-1)}, \tag{2.3}$$

where C is the number of concordant pairs, and D is the number of discordant pairs. A pair of solutions (p_i, p_j) is a concordant pair with respect to task k and task l, if $(r_{ik} < r_{jk}, r_{il} < r_{jl})$ or $(r_{ik} > r_{jk}, r_{il} > r_{jl})$. Otherwise, (p_i, p_j) is a discordant pair. Then, the source task of k-th task is given as

$$l^* = arg \max_{1 \le l \le K, l \ne k} \{\rho_{kl}\}. \tag{2.4}$$

Then, the transferring population of k-th task is $P_k^T = P_{l^*}$, and set the skill factor of P_k^T to k. Then, evaluate population P_k^T and all the individuals in P_k^T which are better than the current population P_k are put into the set S. That is, the individuals in P_k^T whose objective function value is smaller than the smallest function of P_k are put into the set S. A new set $P_k \cup S$ is obtained by merging the current population P_k with the individuals in S. A Gaussian distribution model is constructed by the merged population $P \cup S$. The mean $\bar{\mathbf{x}}_k$ and variance Cov_k of the Gaussian distribution model can be obtained by Eqs (2.5) and (2.6).

$$\bar{\mathbf{x}}_k = \frac{1}{|P_k \cup S|} \sum_{\mathbf{x} \in P_k \cup S} \mathbf{x} \tag{2.5}$$

$$Cov_i = \frac{1}{|P_k \cup S| - 1} \sum_{\mathbf{x} \in P_k \cup S} (\mathbf{x} - \bar{\mathbf{x}}_k)(\mathbf{x} - \bar{\mathbf{x}}_k)^T \tag{2.6}$$

where $|P_k \cup S|$ is the number of individuals in $P_k \cup S$. By doing so, the good individuals in P_k^T can help to improve the evolution of the population and enhance the global search ability of the algorithm. Through targeted selection, the offspring sampled from the Gaussian distribution composed of elite individuals and the current population are more likely to have higher quality.

Algorithm 2: Strategies for Knowledge Transfer

input :

1. The current population P;
2. Task k.

output: The mean $\bar{\mathbf{x}}_k$ and variance Cov_k of the Gaussian distribution model of the task k.

◇ Calculate the Kendall correlation coefficient ρ_{kl} by Eq (2.3).

◇ Find the transfer population P_k^T for task k: $P_k^T = P_{l^*}$ with $l^* = arg \max_{1 \le l \le K, l \ne k} \{\rho_{kl}\}$.

◇ Set the skill factor of the individual in P_k^T to k.

◇ Evaluate the population P_k^T.

◇ Find out the individuals in P_k^T that are better than the current population P_k and put them into S.

◇ Apply Eqs (2.5) and (2.6) to individual x in $P \cup S$ to get $\bar{\mathbf{x}}_k$ and Cov_k, respectively.

2.3. Performance analysis of the knowledge transfer based on elite individual transferred

Figure 3 shows the elite knowledge transfer of the proposed MSOET algorithm. The solid black circles on the left of Figure 3 are the current population of Task 1, and the red stars are the elite individual which transferred from Task 2 to Task 1. The black ellipse is the Gaussian distribution model trained by the current population, and the red ellipse is the Gaussian distribution model trained by the current population and elite individuals. The black and red models are recorded as Gaussian distribution models 1 and 2, respectively. The solid black circle on the right of Figure 3 is the current population of Task 2, and the solid red circle is the individual selected for transfer.

From the comparison of Gaussian distribution models 1 and 2, it can be seen that model 2, which is trained by elite individuals transferred from Task 2 and the current population, can produce offspring with better convergence than model 1, so as to accelerate the evolution of the population and improve the performance of the algorithm.

3. Experimental studies

3.1. The compared algorithms

We compared the proposed MSOET algorithm with the following ten most advanced evolutionary algorithms. The code for these algorithms is available on the official platform MTO-Platform at https://github.com/intLyc/MTO-Platform.

1) AT-MFEA [41] employs domain adaptation technique for solving heterogeneous problems and proposes a novel rank loss function for acquiring a superior intertask mapping.
2) IMEA [42] is an evolutionary multitasking optimization framework based on the island model. Information about the individuals on one island is transferred to another island through an inter-island crossover.
3) LDA-MFEA [43] proposes a linearized domain adaptation strategy that transforms the search space of a simple task into a search space similar to its constitutive complex task.
4) MFDE [33] proposes a multifactorial optimization based on differential evolution and presents a mating approach for multitask optimization in MFDE.
5) MFEA [19] represents one of the most widely used implementation paradigms of the evolutionary multitasking optimization algorithms. An implicit genetic transfer strategy presented in this paper can accelerate convergence for a variety of complex optimization problems in a multitasking environment.
6) MFEA-AKT [25] proposes a new MFEA with adaptive knowledge transfer, in which the crossover operator employed for knowledge transfer is self-adapted based on the information collected along the evolutionary search process.
7) MFPSO [33] proposes a multifactorial optimization algorithm embarking with particle swarm optimization, and a mating approach is designed for multitask optimization in MFPSO.
8) MTEA-AD [44] is a multitask evolutionary algorithm based on anomaly detection. The anomaly detection model identifies candidate-transferred individuals that can effectively reduce the risk of negative transfer.
9) MTEA-SaO [45] adaptively selects a best-fitting solver for each task and enables knowledge transfer using the implicit similarities between tasks.
10) SOO algorithm is that two tasks evolve separately without communication between tasks.

3.2. Test problems

The test suite [46] is used to evaluate the performance of the proposed MSOET algorithm. This test suite consists of 9 test questions, each containing 2 tasks, and each task is a single-objective optimization problem. The properties of these nine test questions are shown in Table 1. Nine test questions are designed according to the global optimal degree of intersection and the similarity of the fitness landscape. According to the global optimal intersection degree, it can be divided into the complete intersection (CI), partial intersection (PI) and no intersection (NI). According to the similarity of fitness landscape between tasks, it can be divided into high similarity (HS), medium similarity (MS), and low similarity (LS). According to this classification method, nine combinations can be formed and designed, as shown in Table 1.

Table 1. Properties of single-objective multitask benchmarks.

Problem	Task NO.	Properties	D	Decision space
CIHS	T1	Griwank	50	[-100,100]
	T2	Rastrigin	50	[-50,50]
CIMS	T1	Ackley	50	[-50,50]
	T2	Rastrigin	50	[-50,50]
CILS	T1	Ackley	50	[-50,50]
	T2	Schwefel	50	[-500,500]
PIHS	T1	Rastrigin	50	[-500,500]
	T2	Sphere	50	[-50,50]
PIMS	T1	Ackley	50	[-50,50]
	T2	Rosenbrock	50	[-50,50]
PILS	T1	Ackley	50	[-50,50]
	T2	Weierstrass	50	[-0.5,0.5]
NIHS	T1	Rosenbrock	50	[-50,50]
	T2	Rastrigin	50	[-50,50]
NIMS	T1	Griwank	50	[-100,100]
	T2	Weierstrass	50	[-0.5,0.5]
NILS	T1	Rastrigin	50	[-50,50]
	T2	Schwefel	50	[-500,500]

3.3. Parameter settings

- Population Size N: The population size N of all algorithms is set to 100.
- Termination Condition: The maximum generation of all algorithms is 1000.
- Hybrid Variation Parameter: SBX [47] is used as the crossover operator, and polynomial mutation is used as the mutation operator. Among them, η_n is set to 2 in SBX, and the variation parameter of polynomial η_m is set to 5.
- Number of Runs: All algorithm runs 30 times independently for each test question.
- Private Parameter: The probability pd is set to 0.3 in Algorithm 1.
- The results of the compared algorithms are implemented with the official platform MTO-Platform* on MATLAB2020b. The algorithm-specific parameters use their default values.
- Statistical Test: The Wilcoxon rank-sum test with a level of 0.05 is used to analyze the significance of the performance between algorithms. The symbols "†", "§" and "‡" indicate that the proposed MSOET algorithm is inferior, equal and better than the contrasted algorithm, respectively. In addition, in order to compare the algorithms intuitively, the algorithms that perform best in the test problems were marked in bold.

*https://github.com/intLyc/MTO-Platform

3.4. Analysis of experimental results

Table 2. The average function value and significance analysis obtained by the compared algorithms over 30 independent runs on complete intersection test problems.

Algorithms	CIHS-T1	CIHS-T2	CIMS-T1	CIMS-T2	CILS-T1	CILS-T2	(†,§,‡)
MSOET	**2.89E-06**	**3.29E+00**	**8.88E-16**	**0.00E+00**	**2.01E+01**	4.19E+03	(—)
SOO	1.70E-01†	3.32E+02†	3.43E+00†	3.08E+02†	2.11E+01†	**2.93E+03**‡	(5,0,1)
AT-MFEA	4.63E-02†	4.93E+01†	1.52E-01†	6.88E+00†	2.12E+01†	2.44E+03‡	(5,0,1)
IMEA	9.62E-01†	3.51E+02†	4.57E+00†	3.56E+02†	2.12E+01†	4.46E+03†	(6,0,0)
LDA-MFEA	1.03E+00†	3.47E+02†	6.40E+00†	4.03E+02†	2.06E+01†	5.57E+03†	(6,0,0)
MFDE	5.26E-01†	3.64E+02†	2.75E+00†	3.55E+02†	2.12E+01†	1.43E+04†	(6,0,0)
MFEA	8.33E-01†	2.73E+02†	5.84E+00†	3.16E+02†	2.02E+01§	4.45E+03†	(5,1,0)
MFEA-AKT	5.69E-01†	2.71E+02†	6.67E+00†	3.51E+02†	2.03E+01†	4.39E+03†	(6,0,0)
MFPSO	1.14E+00†	4.54E+02†	8.24E+00†	5.07E+02†	2.12E+01†	1.31E+04†	(6,0,0)
MTEA-AD	9.24E-01†	2.96E+02†	4.47E+00†	3.12E+02†	2.10E+01†	4.47E+03†	(6,0,0)
MTEA-SaO	1.08E-01†	2.74E+02†	1.36E+00†	2.49E+02†	2.12E+01†	4.34E+03†	(6,0,0)

Table 3. The average function value and significance analysis obtained by the compared algorithms over 30 independent runs on partial intersection test problems.

Algorithms	PIHS-T1	PIHS-T2	PIMS-T1	PIMS-T2	PILS-T1	PILS-T2	(†,§,‡)
MSOET	**5.96E+01**	**3.37E-26**	**3.74E-14**	**5.71E+01**	**8.88E-16**	**2.44E-03**	(—)
SOO	3.06E+02†	1.82E+00†	3.33E+00†	1.06E+03†	3.34E+00†	1.24E+01†	(6,0,0)
AT-MFEA	3.87E+02†	1.34E-01†	2.71E-01†	1.20E+02†	3.43E-01†	2.28E-01†	(6,0,0)
IMEA	4.34E+02†	1.14E+02†	4.93E+00†	1.32E+04†	5.49E+00†	5.26E+00†	(6,0,0)
LDA-MFEA	7.48E+02†	3.55E+02†	6.64E+00†	5.90E+04†	1.90E+01†	1.46E+01†	(6,0,0)
MFDE	4.46E+02†	1.59E+01†	3.05E+00†	1.55E+03†	1.39E+01†	8.03E+00†	(6,0,0)
MFEA	6.59E+02†	8.79E+01†	5.10E+00†	9.71E+03†	1.96E+01†	1.91E+01†	(6,0,0)
MFEA-AKT	5.94E+02†	3.56E+01†	4.20E+00†	1.50E+03†	5.98E+00†	7.13E+00†	(6,0,0)
MFPSO	7.77E+02†	3.24E+03†	6.13E+00†	2.32E+04†	9.61E+00†	7.77E+00†	(6,0,0)
MTEA-AD	4.53E+02†	1.28E+02†	4.69E+00†	9.90E+03†	5.65E+00†	5.29E+00†	(6,0,0)
MTEA-SaO	3.43E+02†	1.71E+00†	1.58E+00†	4.82E+02†	2.53E+00†	2.00E+00†	(6,0,0)

Table 4. The average function value and significance analysis obtained the compared algorithms over 30 independent runs on no intersection test problems.

Algorithms	NIHS-T1	NIHS-T2	NIMS-T1	NIMS-T2	NILS-T1	NILS-T2	(†,§,‡)
MSOET	**4.77E+01**	**7.15E-02**	**1.08E-04**	**9.28E-03**	**6.13E+01**	3.82E+03	(—)
SOO	9.32E+02†	3.22E+02†	1.59E-01†	3.76E+01†	3.35E+02†	3.03E+03‡	(5,0,1)
AT-MFEA	1.56E+02†	1.95E+02†	5.63E-02†	3.78E+00†	3.91E+02†	**2.33E+03**‡	(5,0,1)
IMEA	1.77E+04†	4.23E+02†	9.81E-01†	2.55E+01†	4.42E+02†	4.44E+03†	(6,0,0)
LDA-MFEA	5.21E+04†	4.65E+02†	1.11E+00†	2.79E+01†	9.14E+02†	5.97E+03†	(6,0,0)
MFDE	1.38E+03†	3.79E+02†	5.49E-01†	1.15E+01†	4.49E+02†	1.42E+04†	(6,0,0)
MFEA	1.08E+04†	3.70E+02†	9.35E-01†	2.80E+01†	7.60E+02†	4.46E+03†	(6,0,0)
MFEA-AKT	2.24E+03†	3.31E+02†	6.80E-01†	2.19E+01†	7.10E+02†	4.34E+03†	(6,0,0)
MFPSO	1.02E+05†	5.03E+02†	1.44E+00†	2.25E+01†	1.25E+03†	1.31E+04†	(6,0,0)
MTEA-AD	1.31E+04†	3.81E+02†	9.94E-01†	2.40E+01†	4.42E+02†	4.59E+03†	(6,0,0)
MTEA-SaO	3.62E+02†	2.75E+02†	1.60E-01†	1.15E+01†	3.04E+02†	4.64E+03†	(6,0,0)

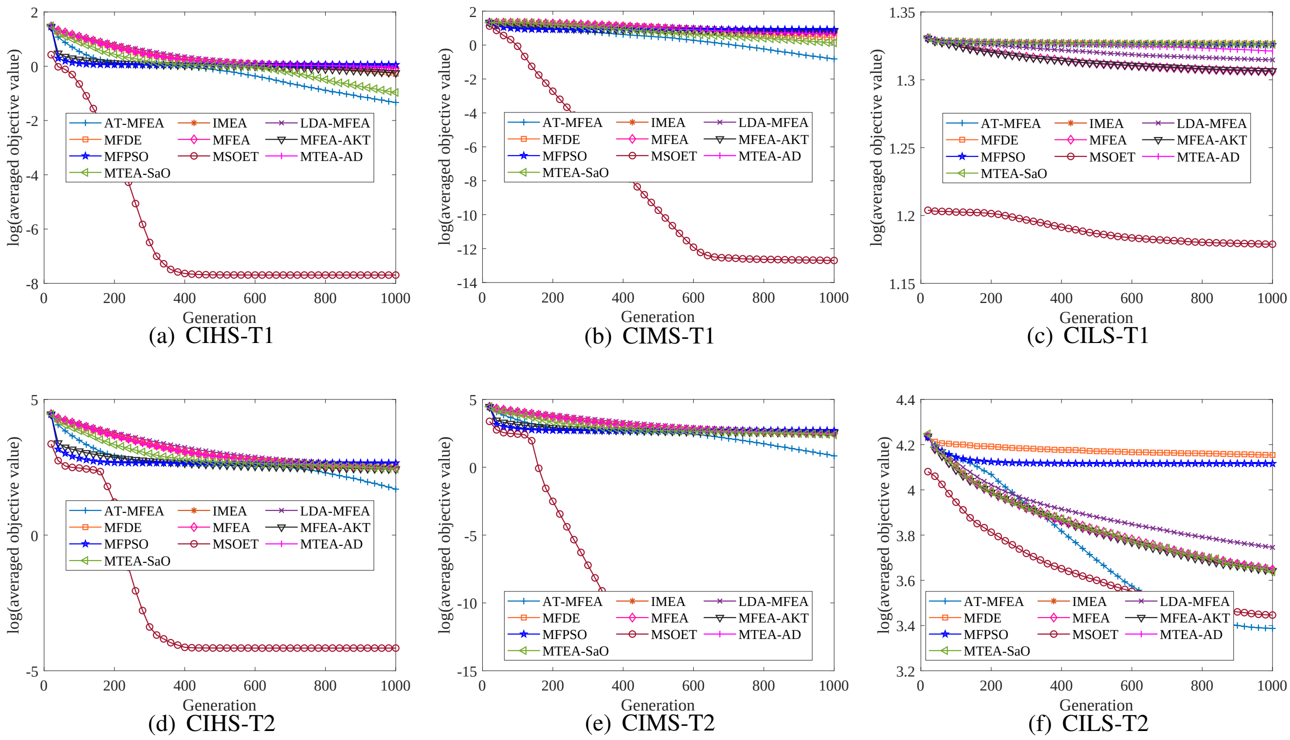
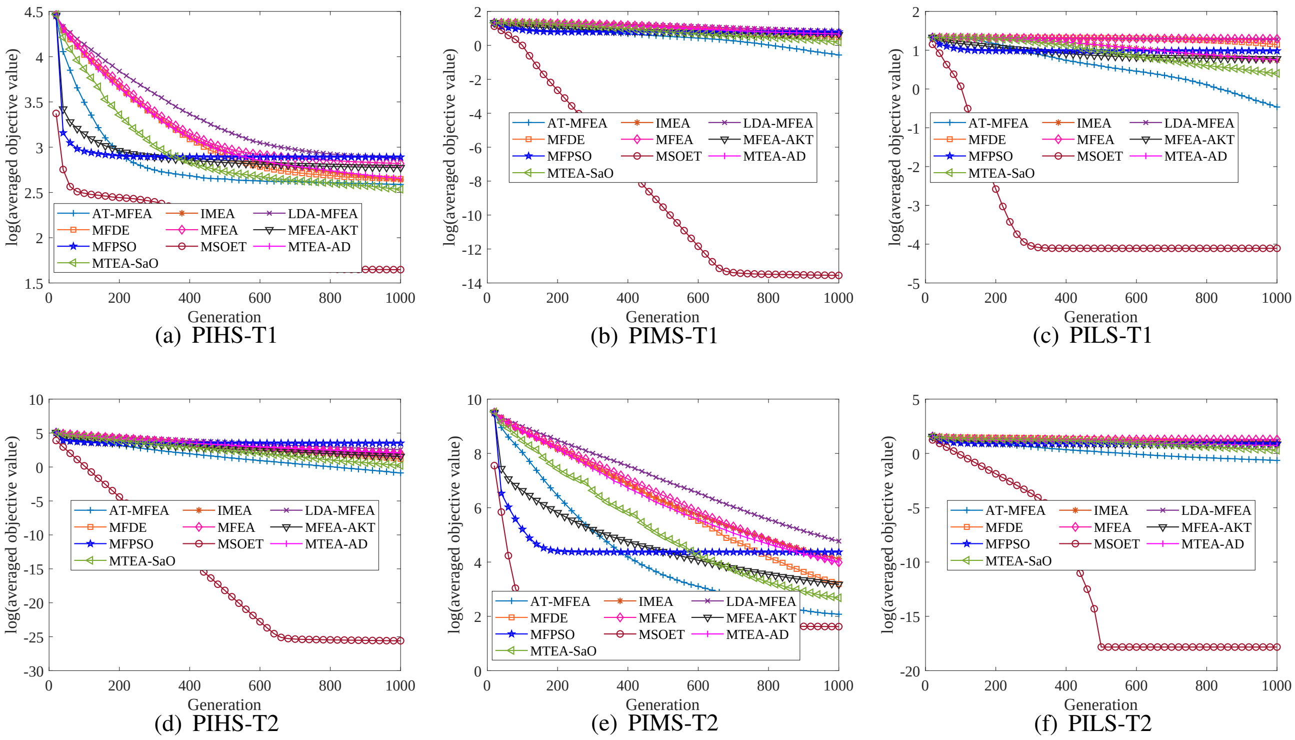


Figure 4. Plot of the average convergence trend of the compared algorithms over 30 independent runs on CIHS, CIMS and CILS.



(a) PIHS-T1 (b) PIMS-T1 (c) PILS-T1

(d) PIHS-T2 (e) PIMS-T2 (f) PILS-T2

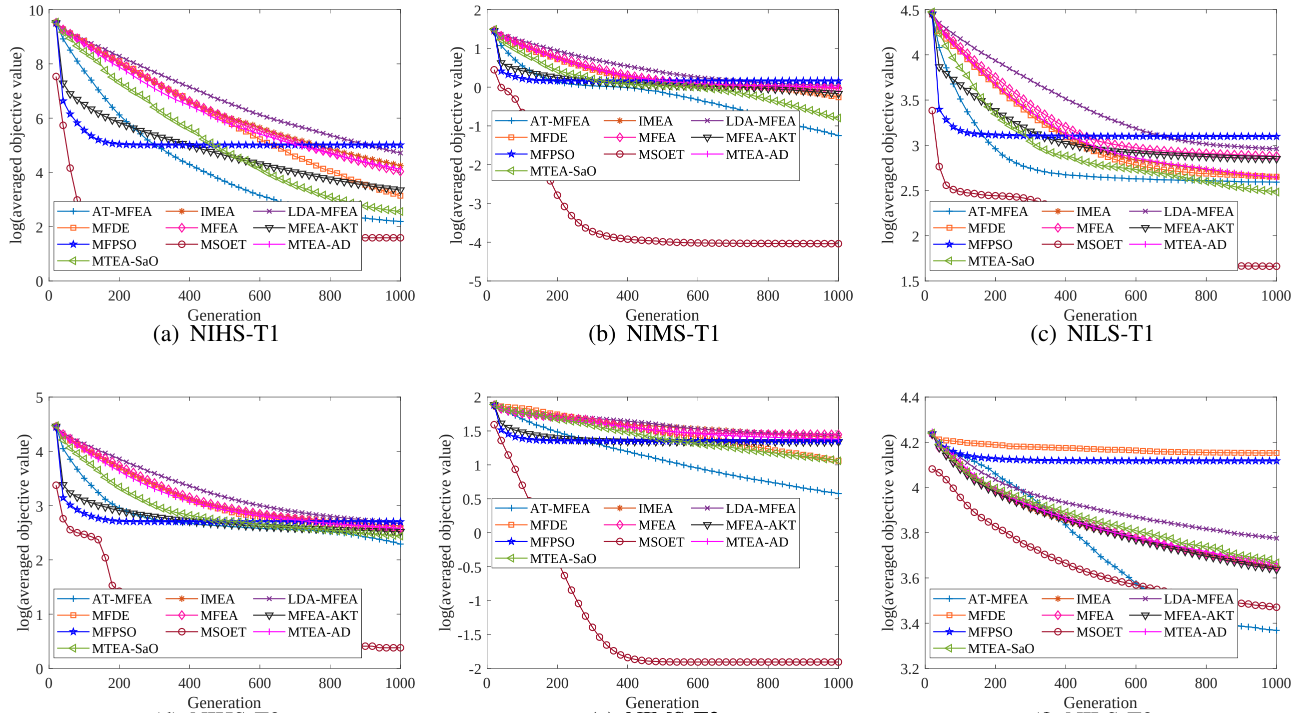
Figure 5. Plot of the average convergence trend of the compared algorithms over 30 independent runs on PIHS, PIMS and PILS.

Tables 2–4 show the average function values obtained by the proposed MSOET algorithm and the compared algorithms over 30 independent runs on the test problems. For the sake of intuition, the algorithm that performs best on the test problem is represented with a gray background. The last column of these tables shows the results of the significance analysis of these algorithms. Figures 4–6 plot the average convergence trend of the compared algorithms over 30 independent runs on the test problems. These tables and figures show that the proposed MSOET algorithm performs best in 16 of the 18 tasks. This proves the effectiveness of the proposed MSOET algorithm. The number of "†" is far more than that of "‡", which also shows that the proposed algorithm has advantages over other algorithms.

This phenomenon is because the proposed MSOET algorithm makes use of the cooperation between tasks to transfer the high-quality individuals of another task to the target task and work with the current population to build a model for generating offspring. Using this model can produce high-quality offspring, accelerate the evolution of the population, and improve the algorithm's performance.

4. Conclusions

In this paper, we have proposed a new evolutionary single-objective multitasking algorithm based on elite knowledge transfer (MSOET), which can solve multiple single-objective optimization tasks at the same time. MSOET combined the excellent individuals of the transfer population with the current population to construct a Gaussian distribution model and used the calculated mean and variance to generate offspring to achieve knowledge transfer between tasks. The excellent individual can effectively and robustly improve the performance of the algorithm. From the analysis of the experimental part, we can see that compared with the existing ten advanced evolutionary single-objective multitasking algorithms, MSOET has a better convergence effect. We expect to execute more effective and robust knowledge transfer between various optimization tasks, even low-related optimization tasks, in our future works. Meanwhile, we will explore how to execute knowledge transfer on the population level instead of the individual level.



(a) NIHS-T1 (b) NIMS-T1 (c) NILS-T1

(d) NIHS-T2 (e) NIMS-T2 (f) NILS-T2

Figure 6. Plot of the average convergence trend of the compared algorithms over 30 independent runs on NIHS, NIMS and NILS.

Acknowledgments

This research was funded in part by the National Natural Science Foundation of China (62172110), in part by the Natural Science Foundation of Guangdong Province (2021A1515011839, 2022A1515010130), and in part by the Programme of Science and Technology of Guangdong Province (2021A0505110004).

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. K. K. Bali, Y. S. Ong, A. Gupta, P. S. Tan, Multifactorial evolutionary algorithm with online transfer parameter estimation: MFEA-II, *IEEE Trans. Evol. Comput.*, **24** (2020), 69–83. https://doi.org/10.1109/tevc.2019.2906927
2. R. Liaw, C. Ting, Evolutionary many-tasking based on biocoenosis through symbiosis: A framework and benchmark problems, in *2017 IEEE Congress on Evolutionary Computation, CEC 2017, Donostia*, (2017), 2266–2273. https://doi.org/10.1109/cec.2017.7969579
3. X. Ma, J. Yin, A. Zhu, X. Li, Y. Yu, L. Wang, et al., Enhanced multifactorial evolutionary algorithm with meme helper-tasks, *IEEE Trans. Cybern.*, **52** (2022), 7837–7851. https://doi.org/10.1109/tcyb.2021.3050516
4. H. L. Liu, F. Gu, Y. M. Cheung, S. Xie, J. Zhang, On solving WCDMA network planning using iterative power control scheme and evolutionary multiobjective algorithm, *IEEE Comput. Intell. Mag.*, **9** (2014), 44–52. https://doi.org/10.1109/mci.2013.2291690
5. J. Yi, W. Zhang, J. Bai, W. Zhou, L. Yao, Multifactorial evolutionary algorithm based on improved dynamical decomposition for many-objective optimization problems, *IEEE Trans. Evol. Comput.*, **26** (2022), 334–348. https://doi.org/10.1109/tevc.2021.3135691
6. L. Chen, H. L. Liu, K. C. Tan, Y. M. Cheung, Y. Wang, Evolutionary many-objective algorithm using decomposition-based dominance relationship, *IEEE Trans. Cybern.*, **49** (2019), 4129–4139. https://doi.org/10.1109/TCYB.2018.2859171
7. H. L. Liu, L. Chen, Q. Zhang, K. Deb, Evolutionary many-objective algorithm using decomposition-based dominance relationship, *IEEE Trans. Evol. Comput.*, **22** (2018), 433–448. https://doi.org/10.1109/TEVC.2017.2725902
8. Q. Peng, Y. M. Cheung, X. You, Y. Y. Tang, A hybrid of local and global saliencies for detecting image salient region and appearance, *IEEE Trans. Syst. Man Cybern. Syst.*, **47** (2017), 86–97. https://doi.org/10.1109/TSMC.2016.2564922
9. F. Zhang, Y. Mei, S. Nguyen, M. Zhang, Multitask multiobjective genetic programming for automated scheduling heuristic learning in dynamic flexible job-shop scheduling, *IEEE Trans. Cybern.*, **2022** (2022), 1–14. https://doi.org/10.1109/tcyb.2022.3196887
10. S. Tuo, C. Li, F. Liu, A. Li, L. He, Z. W. Geem, et al., MTHSA-DHEI: multitasking harmony search algorithm for detecting high-order SNP epistatic interactions, *Complex Intell. Syst.*, **9** (2023), 637–658. https://doi.org/10.1007/s40747-022-00813-7
11. J. Zhong, L. Feng, W. Cai, Y. S. Ong, Multifactorial genetic programming for symbolic regression problems, *IEEE Trans. Syst. Man Cybern. Syst.*, **50** (2020), 4492–4505. https://doi.org/10.1109/tsmc.2018.2853719
12. S. Handoko, H. Lau, A. Gupta, Y. Ong, H. Kim, P. Tan, Solving multi-vehicle profitable tour problem via knowledge adoption in evolutionary bi-level programming, in *2022 IEEE Congress on Evolutionary Computation (CEC)*, (2015), 2713–2720. https://doi.org/10.1109/cec.2015.7257225
13. F. Gu, H. L. Liu, Y. M. Cheung, M. Zheng, A rough-to-fine evolutionary multiobjective optimization algorithm, *IEEE Trans. Cybern.*, **52** (2022), 13472–13485. https://doi.org/10.1109/tcyb.2021.3081357

14. R. Geng, R. Ji, S. Zi, Research on task allocation of UAV cluster based on particle swarm quantization algorithm, *Math. Biosci. Eng.*, **20** (2022), 18–33. https://doi.org/10.3934/mbe.2023002

15. A. Gupta, J. Mańdziuk, Y. S. Ong, Evolutionary multitasking in bi-level optimization, *Complex Intell. Syst.*, **1** (2015), 83–95. https://doi.org/10.1007/s40747-016-0011-y

16. M. Xu, Y. Zheng, Y. S. Ong, Z. Zhu, X. Ma, A multifactorial differential evolution with hybrid global and local search strategies, in *2022 IEEE Congress on Evolutionary Computation (CEC)*, (2022), 1–9. https://doi.org/10.1109/cec55065.2022.9870335

17. S. J. Pan, Q. Yang, A survey on transfer learning, *IEEE Trans. Knowl. Data Eng.*, **22** (2010), 1345–1359. https://doi.org/10.5220/0006396700170027

18. L. Feng, L. Zhou, J. Zhong, A. Gupta, Y. S. Ong, K. C. Tan, et al., Evolutionary multitasking via explicit autoencoding, *IEEE Trans. Cybern.*, **49** (2019), 3457–3470. https://doi.org/10.1109/tcyb.2018.2845361

19. A. Gupta, Y. S. Ong, L. Feng, Multifactorial evolution: Toward evolutionary multitasking, *IEEE Trans. Evol. Comput.*, **20** (2016), 343–357. https://doi.org/10.1109/tevc.2015.2458037

20. Z. Liang, H. Dong, C. Liu, W. Liang, Z. Zhu, Evolutionary multitasking for multiobjective optimization with subspace alignment and adaptive differential evolution, *IEEE Trans. Cybern.*, **52** (2022), 2096–2109. https://doi.org/10.1109/tcyb.2020.2980888

21. S. Liu, Q. Lin, L. Feng, K. C. Wong, K. C. Tan, Evolutionary multitasking for large-scale multiobjective optimization, *IEEE Trans. Evol. Comput.*, **2022** (2022), 1–15. https://doi.org/10.1109/tevc.2022.3166482

22. Y. Cai, D. Peng, S. Fu, H. Tian, Multitasking differential evolution with difference vector sharing mechanism, in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, (2019), 3039–3046. https://doi.org/10.1109/ssci44817.2019.9002698

23. X. Chen, Y. Huang, W. Zhou, L. Feng, Evolutionary multitasking via artificial neural networks, in *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, (2021), 1545–1552. https://doi.org/10.1109/smc52423.2021.9659031

24. S. Huang, J. Zhong, W. J. Yu, Surrogate-assisted evolutionary framework with adaptive knowledge transfer for multi-task optimization, *IEEE Trans. Emerging Topics Comput.*, **9** (2021), 1930–1944. https://doi.org/10.1109/tetc.2019.2945775

25. L. Zhou, L. Feng, K. C. Tan, J. Zhong, Z. Zhu, K. Liu, et al., Toward adaptive knowledge transfer in multifactorial evolutionary computation, *IEEE Trans. Cybern.*, **51** (2021), 2563–2576. https://doi.org/10.1109/tcyb.2020.2974100

26. X. Ma, Y. Zheng, Z. Zhu, X. Li, L. Wang, Y. Qi, et al., Improving evolutionary multitasking optimization by leveraging inter-task gene similarity and mirror transformation, *IEEE Comput. Intell. Mag.*, **16** (2021), 38–53. https://doi.org/10.1109/mci.2021.3108311

27. A. T. W. Min, Y. S. Ong, A. Gupta, C. K. Goh, Multiproblem surrogates: Transfer evolutionary multiobjective optimization of computationally expensive problems, *IEEE Trans. Evol. Comput.*, **23** (2019), 15–28. https://doi.org/10.1109/tevc.2017.2783441

28. L. Bai, W. Lin, A. Gupta, Y. S. Ong, From multitask gradient descent to gradient-free evolutionary multitasking: A proof of faster convergence, *IEEE Trans. Cybern.*, **52** (2022), 8561–8573. https://doi.org/10.1109/tcyb.2021.3052509

29. A. Gretton, K. M. Borgwardt, M. J. Rasch, A. Smola, B. Schölkopf, A. Smola, A kernel two-sample test, *J. Mach. Learn. Res.*, **13** (2012), 723–773.

30. J. Lin, H. L. Liu, B. Xue, M. Zhang, F. Gu, Multiobjective multitasking optimization based on incremental learning, *IEEE Trans. Evol. Comput.*, **24** (2020), 824–838. https://doi.org/10.1109/tevc.2022.3147568

31. J. Ding, C. Yang, Y. Jin, T. Chai, Generalized multitasking for evolutionary optimization of expensive problems, *IEEE Trans. Evol. Comput.*, **23** (2019), 44–58. https://doi.org/10.1109/tevc.2017.2785351

32. J. Liang, L. Zhang, K. Yu, B. Qu, C. Yue, K. Qiao, A differential evolution based self-adaptive multi-task evolutionary algorithm, in *2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT)*, (2021), 150–156. https://doi.org/10.1109/acait53529.2021.9731139

33. L. Feng, W. Zhou, L. Zhou, S. W. Jiang, J. H. Zhong, B. S. Da, et al., An empirical study of multifactorial PSO and multifactorial DE, in *2017 IEEE Congress on Evolutionary Computation (CEC)*, (2017), 921–928. https://doi.org/10.1109/cec.2017.7969407

34. H. Chen, H. L. Liu, F. Gu, K. C. Tan, A multi-objective multitask optimization algorithm using transfer rank, *IEEE Trans. Evol. Comput.*, **2022** (2022), 1–15. https://doi.org/10.1109/TEVC.2022.3147568

35. J. Lin, H. L. Liu, K. C. Tan, F. Gu, An effective knowledge transfer approach for multiobjective multitasking optimization, *IEEE Trans. Cybern.*, **51** (2021), 3238–3248. https://doi.org/10.1109/tcyb.2020.2969025

36. Z. Liang, X. Xu, L. Liu, Y. Tu, Z. Zhu, Evolutionary many-task optimization based on multisource knowledge transfer, *IEEE Trans. Evol. Comput.*, **26** (2022), 319–333. https://doi.org/10.1109/tevc.2021.3101697

37. Z. Liang, W. Liang, Z. Wang, X. Ma, L. Liu, Z. Zhu, Multiobjective evolutionary multitasking with two-stage adaptive knowledge transfer based on population distribution, *IEEE Trans. Syst. Man Cybern. Syst.*, **52** (2022), 4457–4469. https://doi.org/10.1109/tsmc.2021.3096220

38. W. Zhang, X. Zhang, X. Hao, M. Gen, G. Zhang, W. Yang, Multi-stage hybrid evolutionary algorithm for multiobjective distributed fuzzy flow-shop scheduling problem, *Math. Biosci. Eng.*, **20** (2023), 4838–4864. https://doi.org/10.3934/mbe.2023224

39. Q. Shang, L. Zhang, L. Feng, Y. Hou, J. Zhong, A. Gupta, et al., A preliminary study of adaptive task selection in explicit evolutionary many-tasking, in *2019 IEEE Congress on Evolutionary Computation (CEC)*, (2019), 2153–2159. https://doi.org/10.1109/cec.2019.8789909

40. Y. Chen, J. Zhong, L. Feng, J. Zhang, An adaptive archive-based evolutionary framework for many-task optimization, *IEEE Trans. Emerging Topics Computat. Intell.*, **4** (2020), 369–384. https://doi.org/10.1109/tetci.2019.2916051

41. X. Xue, K. Zhang, K. C. Tan, L. Feng, J. Wang, G. Chen, et al., Affine transformation-enhanced multifactorial optimization for heterogeneous problems, *IEEE Trans. Cybern.*, **52** (2022), 6217–6231. https://doi.org/10.1109/tcyb.2020.3036393

42. R. Hashimoto, H. Ishibuchi, N. Masuyama, Y. Nojima, Analysis of evolutionary multi-tasking as an island model, in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, (2018), 1894–1897. https://doi.org/10.1145/3205651.3208228

43. K. K. Bali, A. Gupta, L. Feng, Y. S. Ong, T. P. Siew, Linearized domain adaptation in evolutionary multitasking, in *2017 IEEE Congress on Evolutionary Computation (CEC)*, (2017), 1295–1302. https://doi.org/10.1109/cec.2017.7969454

44. C. Wang, J. Liu, K. Wu, Z. Wu, Solving multi-task optimization problems with adaptive knowledge transfer via anomaly detection, *IEEE Trans. Evol. Comput.*, **26** (2022), 304–318. https://doi.org/10.1109/tevc.2021.3068157

45. Y. Li, W. Gong, S. Li, Multitasking optimization via an adaptive solver multitasking evolutionary framework, *Inf. Sci.*, **2022** (2022), 1–24. https://doi.org/10.1016/j.ins.2022.10.099

46. B. Da, Y. Ong, L. Feng, A. K. Qin, A. Gupta, Z. Zhu, et al., Evolutionary multitasking for single-objective continuous optimization: Benchmark problems, performance metric, and baseline results, preprint, arXiv:1706.03470. https://doi.org/10.48550/arXiv.1706.03470

47. R. B. Agrawal, K. Deb, R. B. Agrawal, Simulated binary crossover for continuous search space, *Complex Syst.*, **9** (1994), 115–148.