Applied Soft Computing 107 (2021) 107399

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Multiobjective multifactorial immune algorithm for multiobjective multitask optimization problems

Zhiwei Xu [a], Kai Zhang [a,b,*]

[a] School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, China
[b] Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System, Wuhan 430065, China

ARTICLE INFO

Article history:
Received 29 January 2020
Received in revised form 28 February 2021
Accepted 2 April 2021
Available online 16 April 2021

Keywords:
Evolutionary multitasking
Multiobjective immune algorithm
Multiobjective optimization
Evolutionary algorithm

ABSTRACT

Inspired by human brains' ability to solve multiple tasks simultaneously, evolutionary multitasking is proposed to improve the overall efficiency of optimizing multiple tasks simultaneously by reusing the learned knowledge. The immune algorithm is inspired by the biological immune system that has been proven to be effective in many practical multiobjective optimization problems, with efficient convergence and search efficiency. In this paper, a novel multiobjective multifactorial immune algorithm is proposed with a novel information transfer method to solve multiobjective multitask optimization problems. For each task, information advantageous for this task will be transferred from the others to accelerate convergence through the proposed information transfer method. Finally, the proposed algorithm is compared with the state-of-the-art multiobjective evolutionary multitasking algorithms and the classic multiobjective evolutionary algorithms. The experimental results on the classical multiobjective multitask and the multiobjective many-task test suites demonstrate that the proposed algorithm provides very promising performances.

1. Introduction

Many scientific and engineering applications require the simultaneous optimization of multiple often conflicting objectives. They are called multiobjective optimization problems (MOPs), without loss of generality, suppose the problem is a minimization problem, a MOP can be indicated as follows:

$$\operatorname*{Min}_{x\in\Omega} \mathrm{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}), \dots, f_M(\mathbf{x})) \tag{1}$$

where $\mathbf{x} = (x_1, x_2, x_3, \dots, x_D) \epsilon \Omega$ is a decision vector with D dimensions, Ω is the feasible region of the decision space, mapping function F: $\Omega \rightarrow R^M$ defines M objective functions and R^M is the objective space. As the objectives may conflict with each other, it is usually impossible to find a solution that can meet the optimal conditions of all the objectives simultaneously. Thus, the definition of Pareto optimal is adopted to find out the best trade-offs among all the objectives. Given two decision vectors $\mathbf{x}$ and $\mathbf{y}$, if $\forall i \in \{1, 2, 3 \dots, M\}$ $f_i(\mathbf{x}) \le f_i(\mathbf{y})$ and $\exists j \in \{1, 2, 3 \dots M\}$ $f_j(\mathbf{x}) < f_j(\mathbf{y})$, $\mathbf{x}$ is said to Pareto dominate $\mathbf{y}$ denoted as $\mathbf{x} \prec \mathbf{y}$. If a solution $\mathbf{x}^*$ is not to be dominated by any other solutions in Ω, $\mathbf{x}^*$ is called a Pareto optimal solution or a nondominated solution. The set of all the Pareto optimal solutions is credited as the Pareto optimal set (PS). The set of vectors of objective values corresponding to PS is called Pareto optimal front (PF) [1].

Multiobjective evolutionary algorithms (MOEAs) have been widely used to solve MOPs because of their advantage of using only a small amount of unique dominance relationships and the population-based searching mechanism to obtain multiple Pareto optimal solutions in a single run [2–8]. The state-of-the-art and most famous MOEAs include the nondominated sorting genetic algorithm II (NSGA-II) [9], strength Pareto evolutionary algorithm (SPEA2) [10], MOEA based on decomposition (MOEA/D) [11]. These MOEAs are designed to solve one MOP once at a time and when faced with a new problem, the population must be reinitialized. But many MOPs in the real-world are related and the knowledge gained from solving a problem can provide insights for solving similar problems.

Inspired by the human brain's ability to process multiple tasks simultaneously [12], Gupta et al. [13] proposed a new paradigm in the evolutionary computation field namely multitask optimization (MTO) to solve multiple optimization problems simultaneously. Unlike conventional evolutionary algorithms (EAs), which aim to find the optimal solution for a single optimization problem, evolutionary multitasking (EMT) algorithms aim to improve solution quality and overall convergence rate of each task by utilizing potentially similar information among simultaneous optimized tasks [14]. EMT algorithms have received increasing attention, and various of them have been proposed to solve many complex problems including MOPs [15–28]. For

* Corresponding author.
E-mail address: zhangkai@wust.edu.cn (K. Zhang).

https://doi.org/10.1016/j.asoc.2021.107399
1568-4946/

example, Gupta et al. [13] presented a multifactorial evolutionary algorithm (MFEA) and proposed a multifactorial evolutionary framework including the mechanisms of assortative mating and vertical cultural transmission to empower the conventional EAs to solve MTO problems. Subsequently, the multiobjective multifactorial evolutionary algorithm (MOMFEA) [22] was proposed, which inherited the theories proposed by the MFEA and introduced the NSGA-II into the multifactorial framework to solve the multiobjective multitask optimization problem (MOMTO). Feng et al. [24] proposed an EMT algorithm with explicit genetic transfer across tasks by autoencoding (EMT-EGT) to effectively exploit multiple biases provided by various evolutionary operators to improve search performance. With more and more EMT algorithms being proposed, multiobjective many-task optimization (MOMaTO) has been paid much attention in the MOMTO area. Chen et al. [26] proposed a many-task evolutionary algorithm (MaTEA) with the adaptive selection mechanism based on the rewards of knowledge transfer to pick out the most appropriate assisted task from many candidate tasks for multiobjective many-task optimization problems.

However, most existing EMT algorithms perform well when solving two or three tasks simultaneously, but when solving the many-task problems the performance is not satisfactory. Negative knowledge transfer and low computational efficiency are the primary causes. In the state-of-the-art multifactorial framework, the information transfer occurs at a fixed probability. When solving many-task optimization problems, a large amount of useless information will disrupt the optimization processes of the target task. The EMT-EGT [24] implements the explicit information transfer through autoencoder. Although it has the ability to reduce the occurrence of negative transfer, it will cause unbearable calculation time as the number of the objective functions increases. The MaTEA [26] only selects the tasks possessing a similar distribution of solutions for transferring, the computing of the similarity of the distribution between every two tasks among many tasks will introduce a huge amount of additional calculation amount. In addition, the PSs of most practical problems are heterogeneous and the similarity of different task solutions cannot be a splendid guide for information transfer.

In this paper, a novel multiobjective multifactorial immune algorithm (MOMFIA) is proposed to solve MOMTO and MOMaTO problems. The proposed MOMFIA applied a novel multi-population framework and a novel information transfer method based on the dimensional information of solutions (DIS). The proposed multi-population framework can evenly distribute individuals to different subpopulations, each of which maintains an independent task module, can evolve independently, but is also equipped to transfer their knowledge when necessary. Compared with the traditional mixed population evolution mode, the multi-population framework can avoid the negative transfer caused by other unsuitable tasks and is conducive to analyzing the characteristics of different tasks to select the most suitable task for knowledge transfer. The proposed information transfer method based on DIS can construct a transfer population to provide favorable transfer information to the target task according to the iterative trend of the clustering center of nondominated solutions in each dimension of the population. The most important feature of this approach is that it not only relies on the knowledge provided by a single task but also selects information beneficial to the convergence of the target task among a massive number of tasks. Compared with the traditional method of selecting a single task for information transfer, this method can learn from more information sources and reduce the extra calculations caused by the evaluation of the similarity of different tasks. It is efficient when solving MOMaTO problems. To verify the effectiveness of the proposed MOMFIA, comprehensive experiments are conducted on the MOMTO benchmark test suite [27], and the MOMaTO CEC2019 competition benchmark test suite [28]. The proposed MOMFIA has achieved obvious advantages in comparison with other MOEAs.

The main contributions of this paper are highlighted as follows:

(1) A novel information transfer mechanism based on the dimensional information of solutions is proposed to realize efficient utilization and transfer among different tasks.

(2) A novel multiobjective evolutionary multitasking algorithm framework based on multi-population is proposed to improve the efficiency of source task selection and knowledge transfer between tasks.

(3) The proposed MOMFIA is the first attempt to introduce an artificial immune algorithm into the field of multiobjective multitask optimization. The excellent convergence ability of the immune algorithm brings strong momentum to the optimization.

(4) To assess the performance of the proposed MOMFIA, experiments are conducted on the classical MOMTO and MOMaTO test suites, the proposed MOMFIA is compared with two state-of-the-art MOEAs, such as NSGA-II and NNIA, and the well-known multiobjective EMT algorithm MOMFEA. The experimental results demonstrate that the proposed MOMFIA is superior to other comparison algorithms.

The remaining of this paper is constructed as follows. Section 2 reviews the concept of MTO and the notion of the multiobjective immune algorithm. Section 3 describes the proposed MOMFIA and the proposed information transfer mechanism based on DIS. Section 4 presents comprehensive experiments on both MOMTO and MOMaTO benchmark test suites to prove the superiority of the proposed MOMFIA. Finally, Section 5 summarizes this work and suggests some future directions.

2. Related work

2.1. Multitask optimization

Generally, according to the number of objective functions contained in a single problem, current mainstream optimization problems can be classified as MOPs and single objective optimization problems (SOPs) two categories. Inspired by the human cognitive ability to multitask processing, the knowledge gained from problem-solving can inspire optimization of related problems [13]. MTO is committed to implementing the evolutionary search on multiple optimization tasks simultaneously to improve convergence by seamlessly transferring knowledge between different problems.

Even though the MOP needs to optimize multiple objective functions simultaneously, its purpose is ultimately to solve one single problem, from the perspective of EMT, the MOEA is ultimately a single-task algorithm. However, the EMT algorithm is committed to simultaneously solving multiple unrelated problems, which may be SOPs, MOPs, or mixtures, and the decision space of each problem is heterogeneous. Both the SOP and the MOP can be considered as the special case of the MTO paradigm when there is only one task. And when a MTO problem contains one or more MOPs, it is called a MOMTO problem. Fig. 1 shows a schematic diagram of the MTO paradigm, in which all tasks to be optimized will be regarded as the input, and after the optimization of the EMT algorithm, MTO paradigm will output the optimal solutions for each task separately.

Without loss of generality, assume that the K problems to be solved simultaneously are all minimization problems, Eq. (2) is the formal expression of the MTO paradigm, and the optimal solution of the jth task T_j $(j = 1, 2, \ldots K)$ is represented as $\mathbf{x}_j^*$.

$$\{\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_K^*\}$$

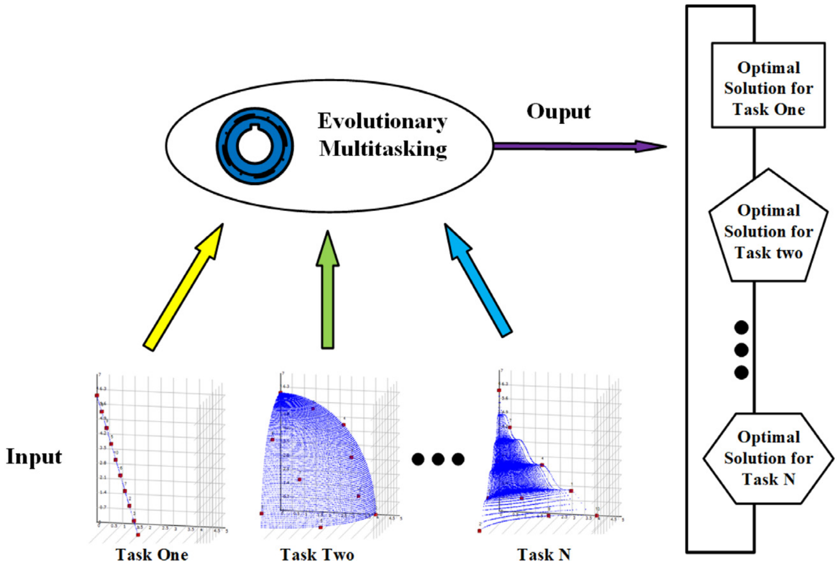

Fig. 1. Illustration of multitask optimization.

$$= \{\text{argmin}T_1(\mathbf{x}_1), \text{argmin}T_2(\mathbf{x}_2), \ldots, \text{argmin}T_K(\mathbf{x}_K)\} \tag{2}$$

Inspired by the multifactorial inheritance theory [29,30], the well-known multifactorial framework [13] is proposed to solve the MTO problems. In the multifactorial framework, each task is regarded as a cultural bias that can directly influence the evolution of offspring. The information exchange between tasks can be carried out by the hybridization of individuals with different cultural biases [31]. To execute information exchange between heterogeneous decision space, the multifactorial framework proposes the unified decision space theory. To guarantee the efficiency and intensity of information exchange, the assortative mating and the vertical cultural transmission mechanisms are proposed respectively. Unified decision space theory, assortative mating and vertical cultural transmission mechanisms are the three major constructing ideas of the multifactorial framework. The multifactorial framework is the earliest proposed instructive paradigm for solving MTO problems and is followed by almost the whole subsequent EMT algorithms [18–21].

The unified decision space theory is dedicated to providing the same number of dimensions of the decision variables and the same upper and lower bounds of decision variables in each dimension of different tasks. The number of dimensions of the unified decision space is the max number of dimensions of decision variables of all the tasks, and the upper and lower bounds of the unified space are generally set to 0 and 1 respectively. When performing population initialization, all individuals will be encoded into the unified decision space. And when an individual requires to be evaluated in a specific task, it will be scaled to the decision space of the corresponding problem, which namely decoding.

The assortative mating theory believes that the mating should be performed between the individuals from the same cultural environment. During the population initialization phase, each individual is assigned a skill factor to mark the task which it is adept at. To improve the evaluation efficiency, each individual will only be evaluated on the task corresponding to the skill factor. In the process of population evolution and selection, the skill factor which determines the current cultural environment of the individual will be redistributed. The assortative mating theory only allows the hybridization of individuals with the same skill factors to guarantee the excellent genes can inherit preferentially in their cultural environment, while the hybridization with different skill factors needs to satisfy the random mating probability (*rmp*), an artificial threshold that is designed to control the information migration intensity.

The vertical cultural transmission mechanism is the concrete manifestation of Darwinism in the multicultural environment that offspring should inherit their skill factors from their parents. Specifically, the offspring obtained by the crossover operator inherits the skill factor from its parents with the same probability, and the offspring produced by the mutation operator will inherit the skill factor of its only parent.

2.2. Multiobjective immune algorithm

The artificial immune system is mainly based on the information processing mechanism of the biological immune system to solve complex optimization problems [32]. In the artificial immune system area, the objective function is likened to the antigen, the candidate solution is considered the antibody, and the fitness value of a candidate solution is seen as the affinity which indicates the ability of the antibody to bind to the antigen.

With the rise of the artificial immune system, the multiobjective immune algorithm (MOIA) has been proposed to solve MOPs [32]. Specifically, the characteristic of MOIAs is the clonal selection principle that only a small proportion of individuals with fine convergence and diversity are proliferated to produce multiple clones [33–35]. Then, each clone is evolved by recombination and hypermutation to become a better individual. In this way, preponderant individuals take up more evolutionary resources, which helps speed up convergence. Therefore, MOIAs have competitive advantage in population diversity and convergence speed compared with other MOEAs [36–42]. Multiobjective immune algorithm with nondominated neighbor-based selection (NNIA) [34] is the first MOIA with real value encoding. In each generation, NNIA selects the solutions with superior crowding distance in the PF as the parents of the clone operator. According to the characteristics of the immune system, MOIAs can be categorized into three classes: the MOIAs based on the advanced clonal selection method, the MOIAs based on immune network methods, and the hybrid MOIAs.

The first kind of MOIAs is designed based on the clonal selection method [43], and the clone operator is applied to produce the copies of antibodies that have the highest affinity values. Yoo and Hajela [44] proposed the first related work on MOIAs and introduced the concept of antibody and antigen affinity into the MOPs for the first time. Jiao et al. [45] proposed the immune dominance clonal multiobjective algorithm which presented the antibody–antibody affinity concept to reflect the similarity between antibodies to guide the algorithm to focus more on the area that candidate solutions sparsely distributed. Coello et al. [46] presented a multiobjective immune system algorithm that not only inherited the clonal resource allocation method based on the fitness value but also introduced an adaptive grid method to improve the diversity of the population.

The second type of MOIAs employs immune network theory to evolve the population and hold the diversity of the population. Freschi et al. [47] proposed an artificial immune network based on a novel double-loop mechanism for MOPs. The inner loop of the algorithm is utilized for global search, and the outer loop is applied to remove individuals with high similarity to maintain the diversity of the population. Gao et al. [48] came up with a weight-based MOIA that a random weighted sum method was introduced to evaluate the fitness and a novel truncation method was proposed to eliminate the highly similar individuals to ensure population diversity.

The last type of MOIAs is to embed another algorithm into the immune system framework to improve the search ability of the algorithm. Lin et al. [49] proposed a novel hybrid MOIA by introducing the adaptive differential evolution as the search operator to enhance the robustness of the algorithm in solving various complex MOPs. Wong et al. [50] presented a novel hybrid immune algorithm that integrated uniform crossover, nondominated sorting, and multi-point mutation to solve the constrained MOPs. Recently, Lin et al. [40] came up with a novel hybrid evolutionary immune algorithm that the entire clone population is divided into different subpopulations, and each subpopulation is evolved by different evolutionary operators to improve the entire search ability.

The proposed MOMFIA follows the antigen–antibody affinity theory and the principle of clone selection, so it belongs to the first type of MOIAs. The proposed MOMFIA is the specific realization of the immune algorithm under the multifactorial framework to solve the MOMTO problems.

3. The proposed MOMFIA

The framework of MOMFIA is shown in Fig. 2, K represents the number of tasks to be optimized simultaneously. MOMFIA starts by initializing the population and setting some relevant parameters. After that, according to multifactorial theory, the entire population is divided into K subpopulations to solve different MOPs, and each subpopulation is used to solve a specific MOP. Then, the antibodies with the highest affinity in each subpopulation are selected in the active population respectively. Next, the DIS-based information transfer method is utilized to transfer the valid information from other tasks. That is, individuals from other tasks that are conducive to the convergence of the target task are selected to form the transfer population, and the clone population corresponding to the target task will mate not only with the nondominated population with the same skill factor but also with the individuals from transfer population to obtain advantageous genetic information. Then, the obtained offspring are mutated by the hypermutation operator. To make a fair comparison with other multiobjective EMT algorithms, for all the comparison algorithms, the simulated binary crossover (SBX) [51] operator is used as the default recombination operator and the polynomial mutation operator [52] is applied as the acquiescent hypermutation operator. Of course, if MOMFIA is dedicated to solving practical problems, in order to improve the efficiency of the algorithm, any other advanced mutation operator can be incorporated into the system. Finally, the elitist archive is used to collect all the nondominated antibodies.

The initialization process of the population is shown in **Algorithm 1**, all the decision variables are scaled to $[Uni_{low}, Uni_{up}]$ to form a unified decision space for facilitating knowledge transfer between multiple heterogeneous optimization tasks, where Uni_{low} and Uni_{up} represent the lower and the upper bounds of the unified decision space, respectively. All the individuals are evenly assigned to different tasks and distributed to the corresponding skill factor, and the individual is only evaluated at its own task to improve the evaluation efficiency. For each task, a separate elitist archive namely nondominated population is kept to store the nondominated antibodies of this subpopulation, and the crowding distance is calculated only on these nondominated individuals, reducing the amount of calculation.

Algorithm 1 The Pseudocode of Initialization.
Input: N: the total size of the entire population, K: the number of the optimization tasks, D: the number of dimensions of the unified search space.
Output: The initial population.
For each individual $\mathbf{x}_i$ in the total N individuals do
  For each dimension d of the $\mathbf{x}_i$ do
    $\mathbf{x}_i^d = Uni_{low} + random(Uni_{up} - Uni_{low})$
  End for
  Assign skill factor $\tau_i = mod(i, K) + 1$, and put $\mathbf{x}_i$ into τ_ith subpopulation.
  Evaluate the $\mathbf{x}_i$ for the task τ_i.
End for
For each subpopulation for the specific task k do
  Add the nondominated antibodies to the nondominated population P_N^k.
  Calculate the crowding distance for each antibody in P_N^k.
End for

3.1. Clonal selection strategy

The clonal selection strategy is inspired by the massive asexual reproduction and mitosis of antibody cells in immunology. The genes of the progeny cells are the same as those of the parent cells, which can enhance the binding of the antigen. The MOIAs place emphasis on the nondominated solutions and possess the elite archiving strategy that all the nondominated solutions in the current generation are preserved at the nondominated population P_N whose maximum capacity is the artificially set hyperparameter N_D. If the number of nondominated solutions in the current generation exceeds N_D, the N_D nondominated individuals with larger crowding distance are picked out. But in MOIAs, not all the nondominated solutions are eligible to be the templates for cloning but the solutions with the largest crowding distance in P_N are qualified and will be stored in the active population P_A whose maximum capacity is the artificially set hyperparameter N_A.

Then, the individuals in the P_A are allocated the clone resources that the individuals with larger crowding distances can have more offspring and cloned. The offspring obtained by cloning are saved to the clone population P_C for recombination and hypermutation. N_C indicating the size of the P_C is a hyperparameter set according to the scale of the problem. The mathematical model of proportional cloning is shown in formula (3).

$$P_C = \bigcup_{i=1}^{N_A} \left\{ h_i \bigotimes a_i \right\}, a_i \in P_A \tag{3}$$

where the operator $\bigotimes$ indicates the clone operator and the parameter h_i denotes the number of clones of each solution a_i in P_A. The value of h_i can be calculated by the mathematical formula (4).

$$h_i = \left\lceil N_C \times \frac{CD(a_i)}{\sum_{j=1}^{N_A} CD(a_j)} \right\rceil \tag{4}$$

where $CD(a_i)$ represents the crowding distance of the individual a_i. Because the crowding distance of the boundary solution is

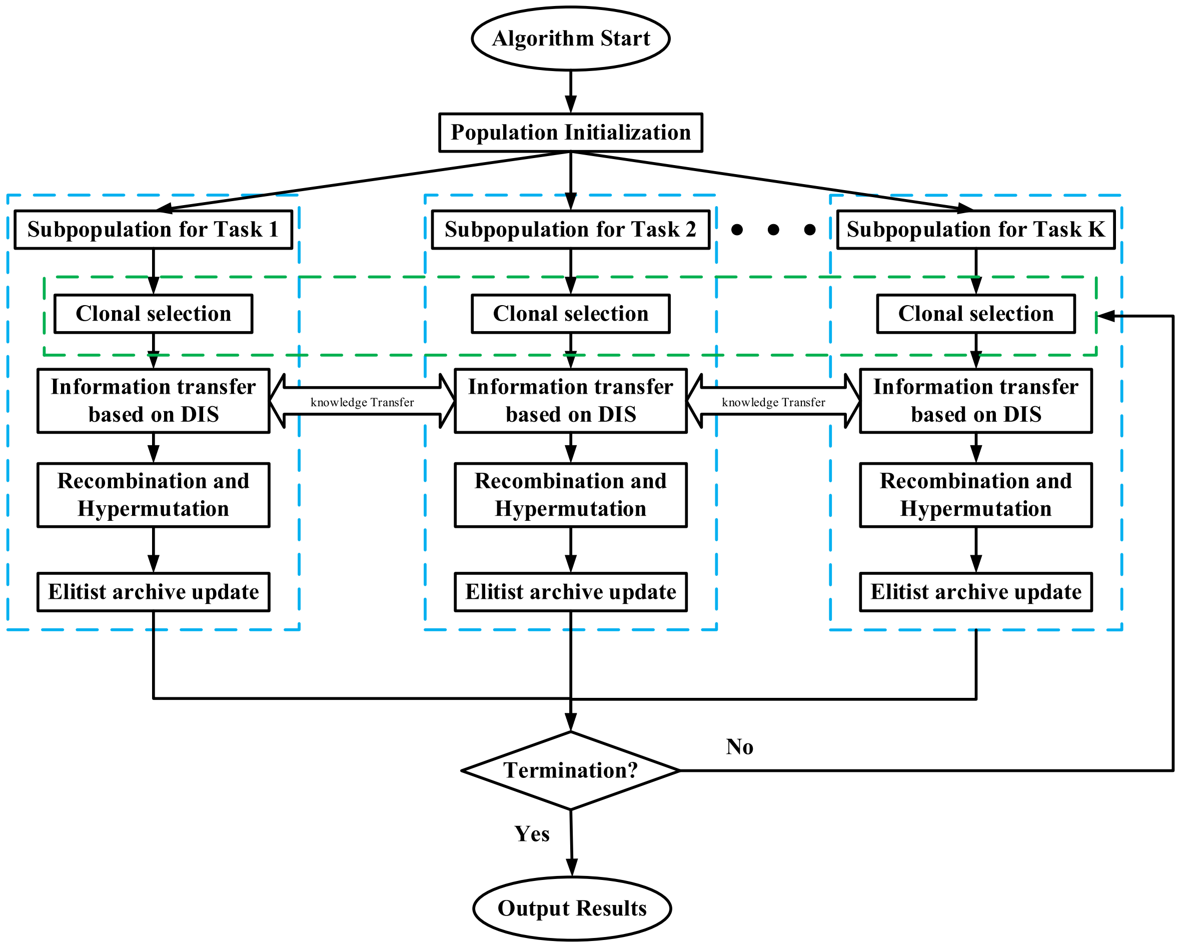

Fig. 2. The framework of proposed MOMFIA.

generally set to infinity, this will encounter difficulties in calculating the number of clones. Therefore, the crowding distance of the boundary solution is generally set to twice the maximum crowding distance of the non-boundary solution.

3.2. Information transfer based on DIS

In conventional MOMTO, the genetic information of individuals is usually exchanged by constantly utilizing the crossover operator such as SBX in the randomly selected dimensions of the two randomly selected individuals from two tasks [22]. When individuals from different tasks crossover, a local search is happening around the intersecting individuals, and the children will inherit the skill factor, thereby generating solutions that may be beneficial to the target task. However, the transfer strategy based on the random crossover will usually cause the trouble that the transfer information is not suitable for the target task. For example, there is a solution that is originally close to the true PS of the target task but steps away from the true PS after receiving information from other tasks. This situation is called the negative transfer of knowledge between tasks. In general, the method that randomly selecting individuals from different tasks to perform genetic information transfer will cause a negative transfer.

Fig. 3 shows the two dimensions of the decision space of a three-task single objective optimization problem to show the effective and the negative knowledge transfer among tasks, respectively. The offspring produced by the crossover of the target task and task one becomes far away from the optimal solution of the target task. The information from task one does not have a beneficial effect on the target task and negative knowledge transfer occurs. However, the offspring obtained by using the crossover operator between the target task and task two makes the solution of the target task closer to the optimal solution, and this kind of knowledge transfer is the ideal pursuit of multitasking optimization.

To avoid the negative knowledge transfer, this paper proposes a knowledge transfer strategy based on the dimensional information. According to the iterative trend of the characteristic solution in the target task, the proposed method selects tasks with similar iteration trends of feature points in other tasks to hybridize. Since the dimensions of all individuals are scaled to [0,1], the iterative trend of the characteristic solution is determined based on the Euclidean distance from the origin. To unambiguously reflect the location characteristics of the nondominated population, from the perspective of clustering, the center of the population cluster is selected as the characteristic solution of the nondominated population for a specific task. The information transfer process based on DIS is described as follows. First, the Euclidean distance of the characteristic solution from the origin point in the target task is recorded. Then, according to the variation trend of the distance of the characteristic solution from the origin point in the next-generation, other tasks that satisfy this trend are selected as the candidate transfer tasks. After that, the characteristic solutions of all candidate transfer tasks are extracted to form the transfer population for the target task. During the evolution process, by hybridizing the solution in the clone population of the target task and the individuals in the transfer population, this beneficial information can be transferred to the target population.

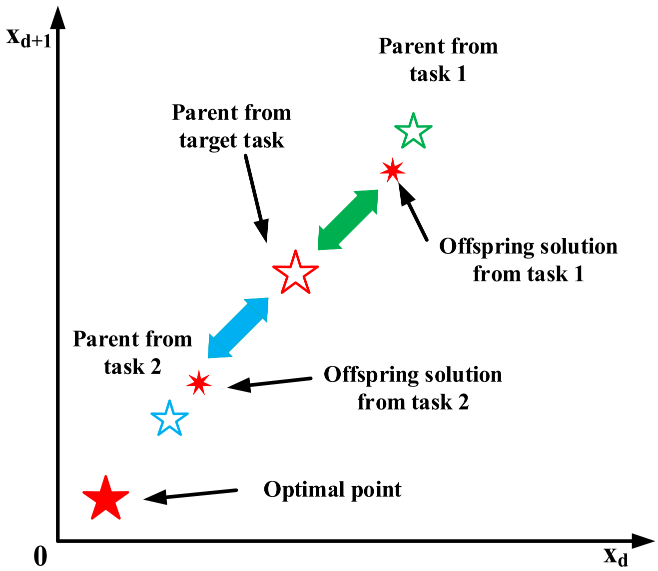

Fig. 3. Positive and negative knowledge transfer for the target task.

Fig. 4 shows the proposed dimensional information-based knowledge transfer process. In Fig. 4(a), the characteristic solution of the target task in *t+1* generation is closer to the origin than in the *t* generation considering *d* and *d+1* dimensions, indicating that the optimal solution in the *t+2* generation is more likely to be closer to the origin. Then, the solutions in the candidate transfer population that are closer to the origin than the target task in *t+1* generation are selected to hybridize. Similarly, in Fig. 4(b), suppose the characteristic solution of the target task in *t+1* generation is far away from the origin than in the *t* generation considering *d* and *d+1* dimensions. The solutions in the candidate transfer population that are farther from the origin will be selected. The pseudocode of establishing the transfer population for the target task is shown in **Algorithm 2**.

Algorithm 2 The pseudocode of establishing the transfer population for the target task.

Input: c_t^{target}: the characteristic solution of target task in *t* iteration, P_{CT}: the candidate transfer population, *ED*: the Euclidean distance of individual from the origin, *K*: the total number of the optimization tasks.
Output: P_T: the transfer population for the target task.
Collect all the characteristic solutions from the other *K-1* task into P_{CT}.
//Considering dimension *d* and *d+1* of unified search space.
If $ED\ (c_t^{target}) < ED\ (c_{t+1}^{target})$
  For each characteristic solution *cs* in the P_{CT} do
    If $ED(cs) > ED\ (c_{t+1}^{target})$
      P_T adds the *cs*.
    End if
  End for
End if
If $ED\ (c_t^{target}) > ED\ (c_{t+1}^{target})$
  For each characteristic solution *cs* in the P_{CT} do
    If $ED\ (cs) < ED\ (c_{t+1}^{target})$
      P_T adds the *cs*.
    End if
  End for
End if

To facilitate a concise and specific description of the proposed dimensional-based information transfer strategy, an SOP is taken as an example but not a MOP. The distance described is the Euclidean distance between the solution and the optimal point, not the distance between the solution and the PS in the decision variable space. This is because the purpose of MTO is to utilize the information of other tasks to promote the convergence of the target task but maintaining the diversity of the population is not the focus of MTO. Of course, our algorithm also considers diversity, but it depends on diversity maintenance ability of MOIAs itself, rather than setting a diversity enhancement strategy for MOMTO alone.

3.3. Recombination and hypermutation

The proposed MOMFIA uses the same recombination and hypermutation operators as the contrast MOMTO algorithms for a fair comparison to highlight the advantages of MOIAs and the efficiency of the proposed information transfer method for solving MOMTO problems. The proposed MOMFIA utilizes the SBX [51] as the crossover operator for recombination and the polynomial mutation [52] as the mutation operator for hypermutation. However, since the transfer population is selected based on the information of characteristic solutions on *d* and *d+1* dimensions, SBX and polynomial mutation will only be performed on the corresponding dimension at each time. $P_C = (c_1, c_2, \ldots, c_{|P_C|})$ denotes the clone population from applying proportional cloning to the active population P_A, P_U represents the union population of the P_A and the transfer population P_T. Then the recombination operator R on the clone population P_C is defined as formula (5).

$$R(c_1 + c_2 + \cdots + c_{|P_C|}) = crossover(c_1, P_U) + crossover(c_2, P_U) + \cdots + crossover(c_{|P_C|}, P_U) \quad (5)$$

where *crossover* $(c_i,\ P_U)$, $i = \{1,\ 2,\ ,\ |\ P_C|\}$ denotes an individual randomly selected from the P_U is hybridized with the clone individual c_i. After that, hypermutation is performed on the result of the recombination operator. If $P_R = (r_1, r_2, \ldots, r_{|P_R|})$ is the clone population after performing recombination process, then the offspring population after hypermutation is expressed as formula (6).

$$M(r_1+r_2+\cdots+r_{|P_R|}) = mutate(r_1)+mutate(r_2)+\cdots+mutate(r_{|P_R|}) \quad (6)$$

3.4. Elitist archive update

After recombination and hypermutation, the beneficial information transferred from other tasks has been passed into the evolved clone population. Then, for each task, after calculating the objective function values of all the individuals in the evolved clone population according to their own corresponding skill factors, these individuals are merged with the current nondominated population of the task to generate a new population P_Q. Afterward, nondominated sorting is performed to P_Q to generate a new generation of nondominated population. If the termination condition is met, the final nondominated population is output as the result, otherwise, the clonal selection operator is executed for the nondominated population and starts a new cycle. Note that unlike other MOEAs, MOIA is based on elitism strategy and archive mechanism, so the number of solutions in the nondominated population and the active population is not fixed, but only the upper limits of these populations are determined. The key point is that the MOIA updates the archive constantly and only selects the nondominated solutions as the parents to ensure very efficient convergence performance just utilizing a small population. This is the greatest strength of MOMFIA compared to other MOMTO algorithms.

3.5. The complete MOMFIA

Algorithm 3 summarizes the MOMFIA framework. First, all the individuals are initialized randomly in the unified decision space and equally assigned to K tasks, and each subpopulation is assigned a skill factor corresponding to the task, and the individuals are evaluated by the objective functions corresponding

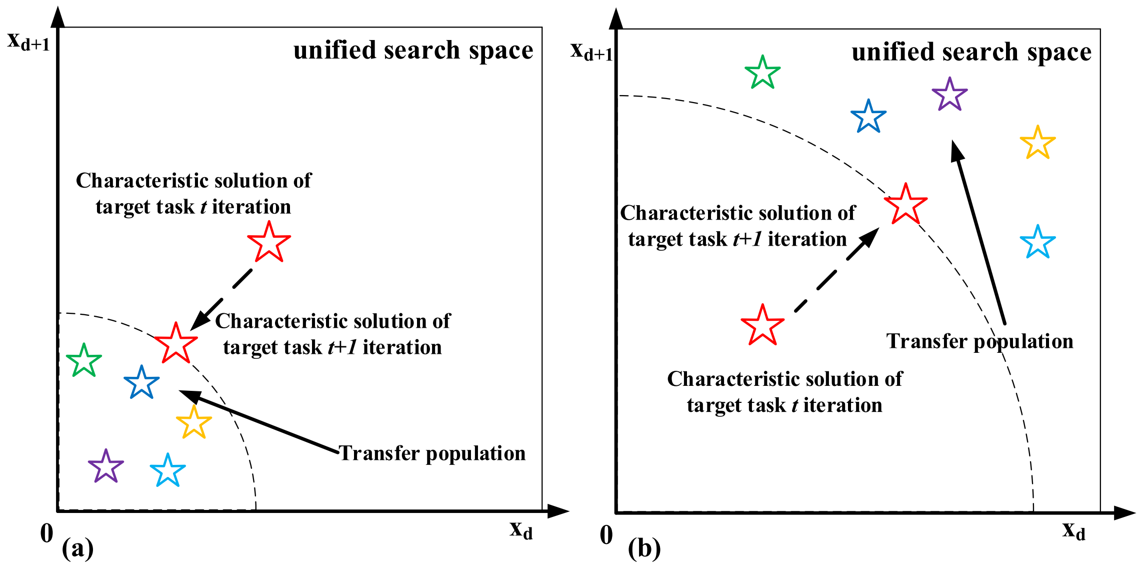

Fig. 4. The proposed transfer strategy based on DIS. (a) The situation that the characteristic solution gradually approaches the origin. (b) The situation that the characteristic solution is gradually away from the origin.

to the skill factors. The detailed description of initialization is shown in **Algorithm 1**. Next, perform nondominated sorting to each subpopulation separately, save nondominated solutions into the nondominated population P_N, and select individuals with larger crowding distance in P_N to store as active solutions in the active population P_A. The proportional cloning is executed to the P_A according to the crowding distance of the solution as shown in formula (3). Then, the characteristic solutions with the same iterative trend are selected according to the DIS from other tasks to form the transfer population P_T, as shown in **Algorithm 2.** Then, the clone population P_C of the target task is recombined according to the formula (5) and is hypermutated according to the formula (6). After that, the evolved clone population P_C' is evaluated according to the objective functions corresponding to the skill factor and then merged with the nondominated population P_N to generate union population P_Q. Finally, the archive P_N is updated by nondominated sorting performed to the P_Q. If the program termination condition is reached, the final nondominated population P_N is output. Otherwise, continue to the next iteration.

Algorithm 3 The Pseudocode of MOMFIA.
Input: K: the number of the optimization tasks.
Output: P_N: the final nondominated population.
1. Initialize the population according to the **Algorithm 1**.
2. **While** not reach the maximum number of evaluations **do**
3. **For** task k in K tasks do
4. Update the archive active population P_A^k.
5. Execute the proportional cloning according to formula (3) and generate the clone population P_C^k.
6. Obtain beneficial information from other tasks and construct the transfer population P_T^k according to **Algorithm 2**.
7. Union P_A^k and P_T^k to form P_U^k.
8. Recombinate and hypermutate the clone population P_C^k with P_U^k according to formula (5) and formula (6) to get the evolved clone population P_C^k'.
9. Evaluate individuals in P_C^k' according to their own skill factors.
10. Union P_C^k' and the current nondominated population P_N^k to form the Q_k.
11. Perform nondominated sorting of Q_k to obtain the new P_N^k.
12. **End for**
13. **End while**

3.6. Complexity analysis

In this section, the computational complexity of one generation of the proposed MOMFIA is discussed. Suppose d denotes the total number of the dimensions of the unified decision space, N_D indicates the population size of the nondominated population, N_C expresses the population size of the clone population, m indicates the total number of the objective functions. In the environment selection, the time complexities of the nondominated sorting is $O(m(N_D + N_C)^2)$, the calculating of the crowding distance is $O(m(N_D + N_C)log(N_D + N_C))$ and the updating of the archive active population is $O(mN_D logN_D)$, respectively. The time complexity of the information transfer based on DIS is $O(dN_D)$. The time complexity of the cloning is $O(N_C)$, and the time complexity of the recombination and hypermutation is $O(dN_C)$. Overall, the total computational complexity of MOMFIA is $O(m(N_D + N_C)^2) + O(m(N_D + N_C)log(N_D + N_C)) + O(mN_D logN_D) + O(dN_D) + O(N_C) + O(dN_C)$. According to the operation rules of symbol O, the time complexity of the proposed MOMFIA can be simplified as $O(m(N_D + N_C)^2)$.

4. Experiments

In this section, the proposed MOMFIA is compared with the classic MOEA NSGA-II and the representative MOIA NNIA and the well-known multiobjective EMT algorithm MOMFEA. The performance of MOMFIA is synthetically evaluated by the classical MOMTO and the CEC2019 MOMaTO test suites.

4.1. Test suites introduction

The performances of the proposed MOMFIA and the comparison algorithms are evaluated by the classical MOMTO benchmark test suite [27] and the MOMaTO benchmark test suite namely MATP presented recently in the CEC2019 many-task optimization competition [28].

MTO believes that the improvement in the overall search efficiency of the EMT algorithms brought about by the information transfer is mainly affected by the global optimum intersection degree and similarity of the fitness landscapes of the simultaneously optimized tasks. It is worth noting that there is not a single global optimal solution in MOP. And the necessary and sufficient condition for obtaining Pareto optimal solutions on MOP is that the constructor function *q(x)* takes the global optimum [27].

Therefore, in the MOMTO problems, the intersection degree of the global optimums refers to that of the global optimal solutions of the MOPs' constructor functions *q(x)*s. If the global

Table 1
Parameters setting for the experiments.

Parameter	MOMFIA	MOMFEA	NSGA-II	NNIA
Population size for all the tasks in the classical MOMTO test suite	200	200	200	200
Population size for all the tasks in the MOMaTO test suite	500	500	2500	2500
Population size of active population for all tasks in the classical MOMTO test suite	80	–	–	80
Population size of clone population for all tasks in the classical MOMTO test suite	200	–	–	200
Population size of active population for all tasks in the MOMaTO test suite	100	–	–	1000
Population size of clone population for all tasks in the MOMaTO test suite	500	–	–	2500
Maximum number of evaluations for all the tasks in the classical MOMTO test suite	200,000	200,000	200,000	200,000
Maximum number of evaluations for all the tasks in the MOMaTO test suite	500,000	500,000	500,000	500,000
Random mating probability	–	0.3	0.3	-
Mutation probability	$1/D$	$1/D$	$1/D$	$1/D$
Crossover probability	0.9	0.9	0.9	0.9
Distribution index of mutation	20	20	20	20
Distribution index of crossover	20	20	20	20

optimums of two tasks are the same in all dimensions, namely complete intersection (CI), and the information transfer between tasks can bring the greatest improvement to the overall convergence performance. If the global optimal solutions are different in all dimensions, called the no intersection (NI). The information transfer can hardly contribute any gain to the overall convergence but will increase the difficulty of optimization and cause negative transfer. The other cases are called partial intersection (PI). The similarity of the fitness landscapes refers to the similarity of the objective function values corresponding to the same decision vector on $q(x)$s attached to different tasks. The more similar the fitness landscapes are, the more the knowledge learned from one task can assist to optimize another task. To calculate the similarities between the fitness landscapes of the constructor $q(x)$s, 1,000,000 points are randomly sampled in the unified decision space, then the Spearman's rank correlation coefficients between $q(x)$s are calculated as the similarities. 1/3 and 2/3 will be used as the cutoff values for high similarity (HS), medium similarity (MS), and low similarity (LS) respectively. According to the above two classification tactics, nine continuous MOMTO benchmark sub-problems are proposed, each sub-problem is composed of two MOP tasks. The detailed description of the classical MTO test suite is shown in reference [27].

In the MOMaTO benchmark test suite MATP which has six sub-suites, each sub-suite contains 50 tasks that have different PSs caused by various shift vectors and diverse rotation matrixes, and each task is a two-objective optimization problem. The MATP test set can effectively evaluate the information transfer mechanism of multiobjective EMT algorithms. With the increase in the number of simultaneous optimization tasks, the difficulty of information transfer greatly increases. If the information from other tasks is not effectively screened, invalid information will consume computing resources and evaluation times of the target task as noise and even hinder the algorithm's convergence on the target task. Detailed information of the MATP can be found in the literature [28].

4.2. Compared algorithms

The proposed MOMFIA will be compared performance with the well-known multiobjective EMT algorithm MOMFEA, the classical single-task MOEAs, NSGA-II and NNIA. The MOMFEA [22] introduced EMT theory into the MOPs for the first time and is the originator of the MOMTO. The MOMFEA inherits the definitions of the skill factor and the scalar fitness in MFEA, expands the concept of the factorial rank, and embeds the NSGA-II [9] to be the evolutionary operator to execute nondominated sorting and calculate the crowding distance for solving multiple MOPs simultaneously. Limited speaking, MOMFEA can be regarded as an improved NSGA-II using multitasking theory for solving MOMTO problems. By comparing the performances of MOMFEA and NSGA-II in solving the same problem, it can prove the advantages of the multiobjective EMT algorithms in solving the MOMTO problems, so NSGA-II will also be adopted for a comparison algorithm. NNIA [34] is the first real-valued encoding MOIA and contributes a lot to the later MOIAs. The clonal selection operator in the proposed MOMFIA also inherits from the NNIA, so NNIA is also adopted as one of the comparison algorithms.

4.3. Parameter settings

For a fair comparison, in the classical MOMTO test suite [27], the population sizes of the multiobjective EMT algorithms MOMFEA and MOMFIA are both set as 200, and the maximum evaluation times are both set as 200,000, but for the conventional single-task MOEAs NSGA-II and NNIA, the population size for each task is set as 100 and the maximum number of evaluations

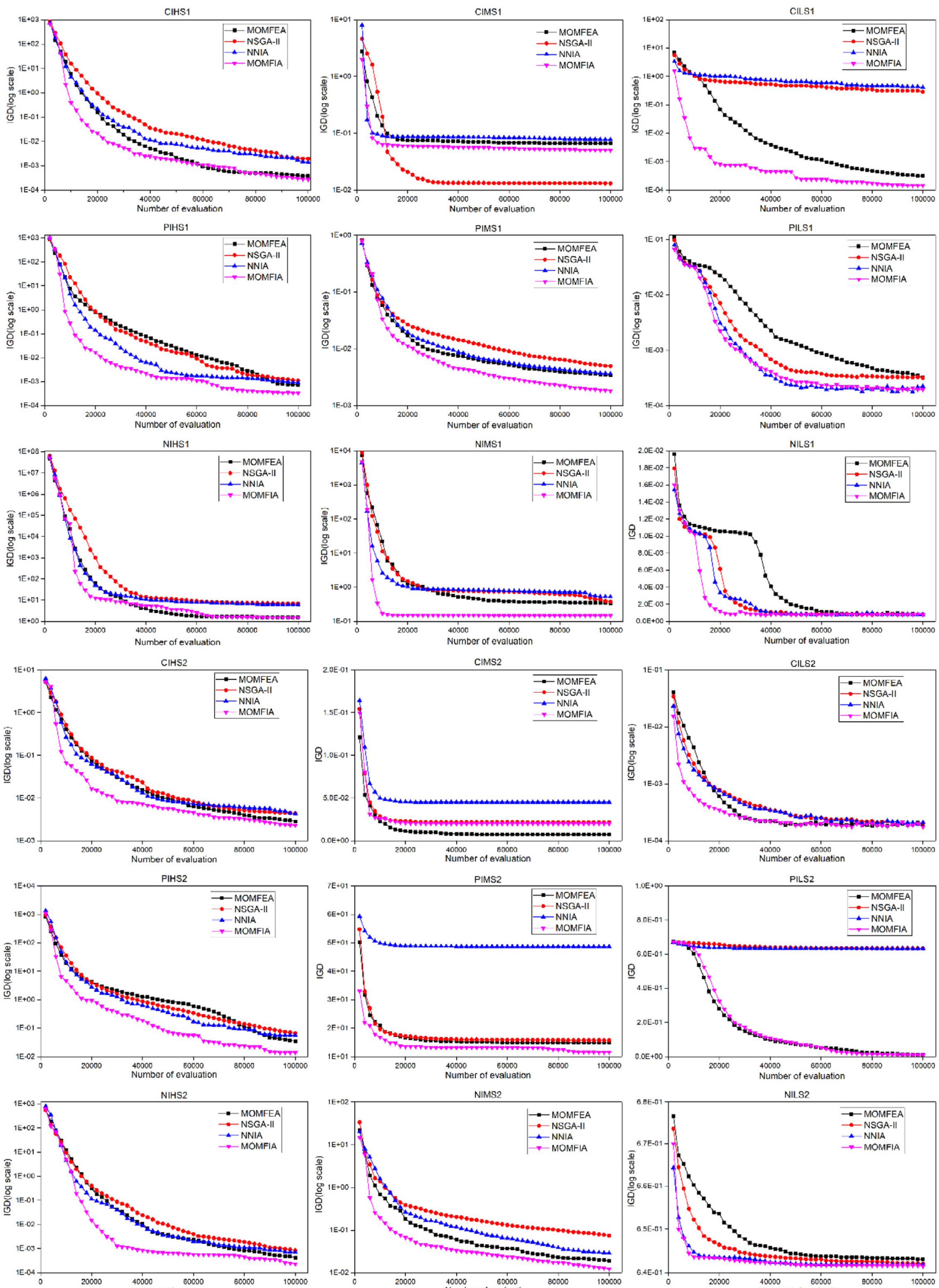

Fig. 5. The average IGD with the number of evaluations for MOMFEA, NSGA-II, NNIA, and MOMFIA on the MOMTO benchmark test suite.

for each task is set as 100,000. In the MOMaTO test suite [28], the total population sizes of the multiobjective EMT algorithms MOMFEA and MOMFIA are both set as 500, but for the conventional single-task MOEAs NSGA-II and NNIA, the population size for each task is set as 50. The maximum number of fitness evaluations for a single task is set as 10000. The settings of other parameters are shown in Table 1.

4.4. Experimental results

The average IGD values and standard deviations obtained by each algorithm 20 independent runs at the classical MOMTO benchmark test suite are shown in Table 2. The best result of each sub-problem is emphasized in bold. In addition, the Wilcoxon

Table 2
The average and standard deviation of the IGD obtained by MOMFIA, MOMFEA, NSGA-II and NNIA on the classical MOMTO benchmark suite for 20 independent runs. The better average IGD values are highlighted in bold.

Problem	Task	MOMFIA	MOMFEA	NSGA-II	NNIA
CIHS	T1	**2.745E−4** (9.695E−5)	3.741E−4 (-) (7.072E−5)	1.972E−3 (-) (4.081E−4)	1.071E−3 (-) (3.338E−4)
	T2	**2.301E−3** (2.908E−4)	2.826E−3 (-) (3.004E−4)	4.362E−3 (-) (5.512E−4)	4.347E−3 (-) (9.167E−4)
CIMS	T1	**5.758E−2** (6.957E−2)	6.592E−2 (-) (6.361E−2)	1.316E−1 (-) (7.207E−2)	7.774E−2 (-) (6.197E−2)
	T2	2.468E−2 (8.268E−3)	**7.265E−3** (+) (9.944E−3)	2.153E−2 (+) (2.071E−2)	4.967E−2 (-) (3.532E−2)
CILS	T1	**1.457E−4** (1.013E−5)	3.105E−4 (-) (3.805E−5)	2.862E−1 (-) (8.001E−2)	4.257E−1 (-) (8.480E−2)
	T2	**1.747E−4** (1.004E−5)	1.936E−4 (-) (5.311E−6)	2.041E−4 (-) (8.709E−6)	2.071E−4 (-) (7.255E−6)
PIHS	T1	**3.329E−4** (4.620E−5)	7.214E−4 (-) (4.746E−4)	1.108E−3 (-) (1.788E−4)	9.642E−4 (-) (1.880E−4)
	T2	**1.354E−2** (5.574E−3)	3.526E−2 (-) (1.161E−2)	6.638E−2 (-) (2.161E−2)	5.665E−2 (-) (1.801E−2)
PIMS	T1	**1.764E−3** (4.862E−4)	3.415E−3 (-) (1.232E−3)	4.966E−3 (-) (1.879E−3)	3.832E−3 (-) (1.614E−3)
	T2	**1.154E1** (3.443E0)	1.484E1 (-) (3.982E0)	1.578E1 (-) (1.741E0)	4.982E1 (-) (1.638E1)
PILS	T1	**1.966E−4** (1.031E−4)	3.225E−4 (-) (9.422E−5)	3.167E−4 (-) (1.01E−4)	2.228E−4 (-) (1.008E−4)
	T2	**1.016E−2** (3.248E−3)	1.143E−2 (-) (2.686E−3)	6.346E−1 (-) (4.256E−4)	6.338E−1 (-) (6.044E−4)
NIHS	T1	**1.527E0** (1.072E−2)	1.543E0 (-) (8.461E−3)	6.897E0 (-) (2.289E0)	5.957E0 (-) (1.281E0)
	T2	**2.359E−4** (2.888E−5)	4.265E−4 (-) (7.567E−5)	8.419E−4 (-) (2.192E−4)	7.108E−4 (-) (1.332E−4)
NIMS	T1	**1.480E−1** (1.311E−1)	3.401E−1 (-) (2.538E−1)	3.771E−1 (-) (3.246E−1)	5.245E−1 (-) (3.339E−1)
	T2	**1.247E−2** (3.634E−3)	1.909E−2 (-) (1.795E−2)	7.479E−2 (-) (5.472E−2)	2.886E−2 (-) (1.223E−2)
NILS	T1	**7.573E−4** (4.169E−5)	8.445E−4 (-) (5.709E−5)	8.248E−4 (-) (5.333E−5)	7.815E−4 (-) (3.235E−5)
	T2	**6.415E−1** (1.037E−4)	6.431E−1 (-) (3.498E−4)	6.421E−1 (-) (2.826E−4)	6.418E−1 (-) (2.117E−4)

rank sum test at the 95% confidence level is applied for the experimental results to compare the proposed MOMFIA with other comparison algorithms, the significantly better and significantly worse using + and - to represent respectively.

As shown in Table 2, from the average IGD values, MOMFIA is superior to the state-of-the-art MOIA NNIA in all the sub-problems in the classical MOMTO test suite. Compared to the classic MOEA NSGA-II, MOMFIA performs better in 17 out of the 18 sub-problems. And MOMFIA surpasses the well-known multiobjective EMT algorithm MOMFEA in 17 out of 18 sub-problems. The above statistical results prove the competitiveness and potential of MOMFIA in solving MOMTO problems. It is worth emphasizing that in most low-to-medium similarity test sub-problems such as NILS, NIMS, PILS, PIMS, and CILS, the performance of MOMFIA is better than that of MOMFEA, which indicates that MOMFIA can overcome the negative transfer caused by the information exchange with the dissimilar tasks and maintain the advantages gained in the target task. It mainly due to that the proposed DIS-based information transfer strategy is more purposeful and is able to effectively screen the transferred knowledge, can transfer effective information in other tasks, and thus avoids negative transfer. Since MOMFEA applies the random crossover mechanism as the transfer knowledge method between tasks, it cannot handle the interference to the target task caused by the negative transfer. Multiobjective EMT algorithms are essentially an improvement of MOEA by using information transfer strategies, so they will still be more or less affected by the original MOEA. Comparing NNIA and NSGA-II, NNIA performs poorly on CIMS-T2, CILS, PIMS-T2, and NIMS-T1. Except that MOMFIA is worse than MOMFEA in CIMS-T2, MOMFIA performs better than MOMFEA in other test problems. This indirectly proves that the proposed MOMFIA uses the mechanism of information transfer to learn effective information from other tasks, and overcomes the shortcomings of immune algorithms in optimizing CILS, PIMS-T2, and NIMS-T1 problems.

Fig. 5 presents the iterative curves of the average IGD of the proposed MOMFIA, MOMFEA, NSGA-II, and NNIA after 20 independent runs on the classical MOMTO benchmark test suite. It should be noted that the recording starts from the 2000*th* evaluations, not from the initialization of the population to show the changes in IGD more clearly. It can be seen that the proposed MOMFIA has good convergence ability and can obtain relatively low IGD values at an early stage. This is due to the fact that the immune algorithm can generate a large number of clones of non-dominated solutions in the iteration to perform a more detailed local search around them. At the later stage of the iteration, the IGD of many algorithms tends to a stable state, which indicates that the solutions in the population are nondominated with each

Table 3
The average and standard deviation of the IGD obtained by MOMFIA, MOMFEA, NSGA-II and NNIA on the MOMaTO benchmark test suite for 20 independent runs. The better average IGD values are highlighted in bold.

Task	MOMFIA	MOMFEA	NSGA-II	NNIA
T1	**2.274E−1** (5.389E−2)	2.244E1 (-) (4.882E0)	3.527E0 (-) (9.596E−1)	9.201E−1 (-) (2.191E−1)
T2	**2.231E−1** (7.213E−2)	2.611E1 (-) (4.047E0)	3.973E0 (-) (1.072E0)	8.638E−1 (-) (2.254E−1)
T3	**2.257E−1** (7.096E−2)	2.105E1 (-) (5.706E0)	3.512E0 (-) (1.171E0)	1.135E0 (-) (2.247E−1)
T4	**2.487E−1** (7.224E−2)	1.761E1 (-) (2.872E0)	4.009E0 (-) (1.043E0)	8.887E−1 (-) (2.129E−1)
T5	**2.787E−1** (7.591E−2)	2.567E1 (-) (4.753E0)	3.711E0 (-) (7.204E−1)	7.626E−1 (-) (2.148E−1)
T6	**2.615E−1** (4.989E−2)	2.326E1 (-) (3.234E0)	3.962E0 (-) (5.735E−1)	1.033E0 (-) (3.243E−1)
T7	**2.384E−1** (4.785E−2)	1.599E1 (-) (3.062E0)	2.834E0 (-) (5.978E−1)	9.091E−1 (-) (2.856E−1)
T8	**2.436E−1** (8.447E−2)	2.038E1 (-) (1.614E0)	3.103E0 (-) (5.798E−1)	1.103E0 (-) (3.989E−1)
T9	**2.598E−1** (7.412E−2)	1.627E1 (-) (5.054E0)	3.766E0 (-) (1.055E0)	8.114E−1 (-) (2.371E−1)
T10	**2.496E−1** (6.966E−2)	2.084E1 (-) (5.824E0)	3.714E0 (-) (7.560E−1)	9.499E−1 (-) (3.488E−1)
T11	**2.125E−1** (3.725E−2)	1.823E1 (-) (3.114E0)	3.275E0 (-) (9.053E−1)	9.832E−1 (-) (3.694E−1)
T12	**2.423E−1** (7.989E−2)	1.792E1 (-) (2.801E0)	3.717E0 (-) (7.634E−1)	9.332E−1 (-) (2.919E−1)
T13	**2.178E−1** (5.679E−2)	1.602E1 (-) (2.962E0)	3.763E0 (-) (9.369E−1)	8.234E−1 (-) (1.812E−1)
T14	**2.348E−1** (6.948E−2)	1.715E1 (-) (1.981E0)	3.432E0 (-) (6.908E−1)	9.103E−1 (-) (2.642E−1)
T15	**2.414E−1** (7.915E−2)	2.693E1 (-) (1.135E1)	3.620E0 (-) (9.981E−1)	9.389E−1 (-) (2.352E−1)
T16	**2.681E−1** (9.327E−2)	1.842E1 (-) (6.057E0)	3.723E0 (-) (7.426E−1)	1.075E0 (-) (3.202E−1)
T17	**2.251E−1** (5.832E−2)	2.545E1 (-) (3.618E0)	3.243E0 (-) (7.174E−1)	7.739E−1 (-) (1.981E−1)
T18	**2.649E−1** (5.919E−2)	1.984E1 (-) (6.158E0)	3.382E0 (-) (6.117E−1)	8.656E−1 (-) (3.543E−1)
T19	**2.126E−1** (8.912E−2)	2.625E1 (-) (5.563E0)	3.667E0 (-) (8.006E−1)	1.018E0 (-) (2.344E−1)
T20	**2.517E−1** (6.949E−2)	1.917E1 (-) (5.036E0)	3.825E0 (-) (1.261E0)	8.397E−1 (-) (1.974E−1)
T21	**2.579E−1** (1.050E−1)	1.617E1 (-) (3.179E0)	3.288E0 (-) (8.473E−1)	9.436E−1 (-) (3.157E−1)
T22	**2.226E−1** (5.685E−2)	2.229E1 (-) (3.923E0)	3.725E0 (-) (1.092E0)	7.525E−1 (-) (1.363E−1)
T23	**2.249E−1** (4.906E−2)	1.832E1 (-) (4.887E0)	3.434E0 (-) (7.831E−1)	8.203E−1 (-) (1.883E−1)
T24	**2.521E−1** (9.256E−2)	2.113E1 (-) (2.667E0)	3.335E0 (-) (1.085E0)	9.397E−1 (-) (3.349E−1)
T25	**2.368E−1** (4.237E−2)	1.924E1 (-) (5.692E0)	3.591E0 (-) (1.218E0)	9.493E−1 (-) (3.263E−1)
T26	**2.458E−1** (5.191E−2)	1.741E1 (-) (3.092E0)	3.298E0 (-) (8.234E−1)	8.807E−1 (-) (1.902E−1)
T27	**2.352E−1** (3.659E−2)	2.191E1 (-) (3.488E0)	3.408E0 (-) (4.727E−1)	8.889E−1 (-) (1.345E−1)
T28	**2.736E−1** (9.808E−2)	2.233E1 (-) (4.696E0)	3.323E0 (-) (6.420E−1)	9.338E−1 (-) (2.395E−1)
T29	**2.501E−1** (8.628E−2)	1.759E1 (-) (3.011E0)	4.215E0 (-) (1.198E0)	9.855E−1 (-) (3.222E−1)

(*continued on next page*)

Table 3 (*continued*).

Task	MOMFIA	MOMFEA	NSGA-II	NNIA
T30	**2.136E−1**	2.191E1 (-)	3.466E0 (-)	9.289E−1 (-)
	(5.174E−2)	(4.005E0)	(7.701E−1)	(3.779E−1)
T31	**2.534E−1**	2.299E1 (-)	3.158E0 (-)	9.261E−1 (-)
	(5.827E−2)	(5.945E0)	(1.093E0)	(3.029E−1)
T32	**2.333E−1**	2.639E1 (-)	3.329E0 (-)	1.109E0 (-)
	(6.921E−2)	(5.464E0)	(7.609E−1)	(3.825E−1)
T33	**2.54E−1**	2.039E1 (-)	3.262E0 (-)	1.006E0 (-)
	(7.045E−2)	(3.409E0)	(8.666E−1)	(2.655E−1)
T34	**2.388E−1**	2.302E1 (-)	3.496E0 (-)	8.394E−1 (-)
	(6.824E−2)	(5.073E0)	(1.004E0)	(3.339E−1)
T35	**2.647E−1**	2.189E1 (-)	3.601E0 (-)	8.438E−1 (-)
	(6.423E−2)	(3.326E0)	(3.986E−1)	(2.665E−1)
T36	**1.969E−1**	2.149E1 (-)	3.323E0 (-)	8.089E−1 (-)
	(4.807E−2)	(6.735E0)	(5.565E−1)	(2.364E−1)
T37	**2.382E−1**	1.824E1 (-)	3.705E0 (-)	1.028E0 (-)
	(6.917E−2)	(5.334E0)	(9.545E−1)	(3.213E−1)
T38	**2.435E−1**	2.007E1 (-)	3.133E0 (-)	8.626E−1 (-)
	(3.504E−2)	(4.448E0)	(7.443E−1)	(2.279E−1)
T39	**2.629E−1**	2.014E1 (-)	3.408E0 (-)	8.503E−1 (-)
	(7.344E−2)	(2.413E0)	(5.108E−1)	(1.826E−1)
T40	**2.461E−1**	2.627E2 (-)	3.272E0 (-)	8.156E−1 (-)
	(5.247E−2)	(7.632E1)	(7.952E−1)	(3.218E−1)
T41	**2.213E−1**	2.027E1 (-)	3.481E0 (-)	9.725E−1 (-)
	(4.601E−2)	(3.849E0)	(7.556E−1)	(2.428E−1)
T42	**2.816E−1**	1.989E1 (-)	3.666E0 (-)	9.964E−1 (-)
	(9.769E−2)	(1.844E0)	(8.947E−1)	(1.696E−1)
T43	**2.562E−1**	1.928E1 (-)	3.776E0 (-)	1.117E0 (-)
	(5.819E−2)	(2.246E0)	(6.265E−1)	(3.643E−1)
T44	**2.378E−1**	2.026E1 (-)	3.634E0 (-)	8.958E−1 (-)
	(5.082E−2)	(3.643E0)	(8.074E−1)	(3.627E−1)
T45	**2.439E−1**	2.578E1 (-)	3.679E0 (-)	9.939E−1 (-)
	(7.754E−2)	(5.856E0)	(8.61E−1)	(1.833E−1)
T46	**2.448E−1**	2.091E1 (-)	3.625E0 (-)	1.035E0 (-)
	(5.219E−2)	(6.478E0)	(5.984E−1)	(3.108E−1)
T47	**2.857E−1**	2.298E1 (-)	3.053E0 (-)	8.534E−1 (-)
	(6.626E−2)	(4.574E0)	(5.894E−1)	(2.842E−1)
T48	**2.224E−1**	1.659E1 (-)	3.361E0 (-)	9.740E−1 (-)
	(4.807E−2)	(3.491E0)	(5.402E−1)	(2.695E−1)
T49	**2.272E−1**	2.518E1 (-)	3.495E0 (-)	8.614E−1 (-)
	(8.514E−2)	(6.206E0)	(7.725E−1)	(1.526E−1)
T50	**2.676E−1**	2.232E1 (-)	3.722E0 (-)	8.868E−1 (-)
	(9.157E−2)	(3.817E0)	(6.725E−1)	(2.429E−1)

other, and it is difficult to find a superior solution is able to replace the nondominated solution in the population through evolution. MOMFIA focuses on the area with the largest crowding distance in the non-dominated solution. Even at the later stage of the iteration, MOMFIA can still improve the diversity of the population by continuously homogenizing the nondominated solutions which lead IGD to decline continuously. In all the test problems, MOMFIA always on top in terms of IGD value within 20,000 evaluations. Besides, the proposed MOMFIA converges faster than MOMFEA NSGA-II, and NNIA in most problems.

In this paper, the performance of the proposed MOMFIA is compared with other algorithms in the first subset of the MATP test suite, namely MATP-1. The average IGD values and standard deviations obtained by each algorithm 20 independent runs at the MOMaTO benchmark test suite are shown in Table 3. The best result of each sub-problem is emphasized in bold. In addition, the Wilcoxon rank sum test at the 95% confidence level is applied for the experimental results to compare the proposed MOMFIA with other comparison algorithms, the significantly better and significantly worse using + and - to represent respectively.

As shown in Table 3, from the perspective of the average IGD value, MOMFIA shows obvious advantages in all the tasks in the MOMaTO benchmark test problem MATP-1. Compared with the state-of-the-art MOEA NSGA-II, NNIA, and the conventional multiobjective EMT algorithm MOMFEA. In the four comparison algorithms, MOMFEA performs significantly worse in MOMaTO problems. This is because when the number of tasks increases, individuals randomly selected from other tasks may not be suitable for the target task, and the information they carry does not have a beneficial effect on the target task. The hybridizations of individuals from different tasks do not make the target task better but waste valuable evaluation times. There is an obvious negative transfer situation here, that is, the algorithm under the EMT framework is not as effective as the basic single-task MOEA. For example, MOMFEA which is the improved NSGA-II for evolutionary multitasking is obviously inferior to NSGA-II here. The proposed MOMFIA has achieved better results in the processing of MOMaTO, which indicates that the proposed DIS-based information transfer strategy can effectively screen effective information for the target task, effectively reduce negative transfer, and promote the convergence of target task to the true PF.

5. Conclusion

In this paper, a novel dimensional information based multi-objective multifactorial algorithm MOMFIA is proposed to solve MOMTO problems. For each task, the clone population selects individuals from the other tasks that have similar iteration trends of feature points to hybridize. The proposed information transfer method can effectively reduce the negative information transfer and improve the efficiency of knowledge transfer compared with the conventional randomly selection based implicit transfer. The MOMFIA can obtain well-converged and well-diversified PFs for different tasks simultaneously. The performance is compared against several different state-of-the-art approaches including MOMFEA, NNIA, and NSGA-II on the classical MOMTO and the MOMaTO benchmark test suites. The experimental results show that the proposed MOMFIA provides better IGD measures on nine MOMTO problems and MOMaTO benchmark problems. Nevertheless, there are still some issues that can be further studied and improved in future work. For example, on each dimension of the decision variable, the DIS-based information transfer method will select the appropriate solutions for information transfer. With the increase of variable dimensions, this method will introduce a large amount of computation. When solving large-scale optimization problems, the dimensionality reduction and refinement of decision variables methods can be applied to the information transfer method to improve the efficiency of the algorithm. In addition, considering the complex optimization problems, the algorithm can be improved into the hybrid MOIA, and a variety of evolutionary operators should be introduced to enhance the search capability.

CRediT authorship contribution statement

Zhiwei Xu: Data curation, Writing - original draft, Software, Validation, Writing - review & editing. **Kai Zhang:** Conceptualization, Methodology, Supervision, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, 2001.

[2] D. Gong, Y. Han, J. Sun, A novel hybrid multi-objective artificial bee colony algorithm for blocking lot-streaming flow shop scheduling problems, Knowl.-Based Syst. 148 (2018) 115–130, http://dx.doi.org/10.1016/j.knosys.2018.02.029.

[3] D. Gong, B. Xu, Y. Zhang, Y. Guo, S. Yang, A similarity-based cooperative co-evolutionary algorithm for dynamic interval multiobjective optimization problems, IEEE Trans. Evol. Comput. 24 (2020) 142–156, http://dx.doi.org/10.1109/TEVC.2019.2912204.

[4] B. Xu, Y. Zhang, D. Gong, Y. Guo, M. Rong, Environment sensitivity-based cooperative co-evolutionary algorithms for dynamic multi-objective optimization, IEEE/ACM Trans. Comput. Biol. Bioinform. 15 (2018) 1877–1890, http://dx.doi.org/10.1109/TCBB.2017.2652453.

[5] D. Gong, J. Sun, X. Ji, Evolutionary algorithms with preference polyhedron for interval multi-objective optimization problems, Inform. Sci. 233 (2013) 141–161, http://dx.doi.org/10.1016/j.ins.2013.01.020.

[6] Y. Zhang, D. Gong, Z. Ding, Handling multi-objective optimization problems with a multi-swarm cooperative particle swarm optimizer, Expert Syst. Appl. 38 (2011) 13933–13941, http://dx.doi.org/10.1016/j.eswa.2011.04.200.

[7] J. Del Ser, E. Osaba, D. Molina, X.-S. Yang, S. Salcedo-Sanz, D. Camacho, S. Das, P.N. Suganthan, C.A. Coello Coello, F. Herrera, Bio-inspired computation: Where we stand and what's next, Swarm and Evol. Comput. 48 (2019) 220–250, http://dx.doi.org/10.1016/j.swevo.2019.04.008.

[8] K. Zhang, Z. Xu, S. Xie, G.G. Yen, Evolution strategy-based many-objective evolutionary algorithm through vector equilibrium, IEEE Trans. Cybern. (2020) 1–13, http://dx.doi.org/10.1109/TCYB.2019.2960039.

[9] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197, http://dx.doi.org/10.1109/4235.996017.

[10] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the Strength ParetoEvolutionary Algorithm, Technical Report 103, Computer Engineering andNetworks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 2001.

[11] Qingfu. Zhang, Hui. Li, MOEA/D: a multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (2007) 712–731, http://dx.doi.org/10.1109/TEVC.2007.892759.

[12] R. Chandra, Y.-S. Ong, C.-K. Goh, Co-evolutionary multi-task learning with predictive recurrence for multi-step chaotic time series prediction, Neurocomputing 243 (2017) 21–34, http://dx.doi.org/10.1016/j.neucom.2017.02.065.

[13] A. Gupta, Y. Ong, L. Feng, Multifactorial evolution: Toward evolutionary multitasking, IEEE Trans. Evol. Comput. 20 (2016) 343–357, http://dx.doi.org/10.1109/TEVC.2015.2458037.

[14] M. Gong, Z. Tang, H. Li, J. Zhang, Evolutionary multitasking with dynamic resource allocating strategy, IEEE Trans. Evol. Comput. 23 (2019) 858–869, http://dx.doi.org/10.1109/TEVC.2019.2893614.

[15] M.-Y. Cheng, A. Gupta, Y.-S. Ong, Z.-W. Ni, Coevolutionary multitasking for concurrent global optimization: With case studies in complex engineering design, Eng. Appl. Artif. Intell. 64 (2017) 13–24, http://dx.doi.org/10.1016/j.engappai.2017.05.008.

[16] J. Zhong, L. Feng, W. Cai, Y.-S. Ong, Multifactorial genetic programming for symbolic regression problems, IEEE Trans. Syst. Man Cybern, Syst. (2018) 1–14, http://dx.doi.org/10.1109/TSMC.2018.2853719.

[17] C. Yang, J. Ding, Y. Jin, C. Wang, T. Chai, Multitasking multiobjective evolutionary operational indices optimization of beneficiation processes, IEEE Trans. Automat. Sci. Eng. 16 (2019) 1046–1057, http://dx.doi.org/10.1109/TASE.2018.2865593.

[18] K.K. Bali, Y. Ong, A. Gupta, P.S. Tan, Multifactorial evolutionary algorithm with online transfer parameter estimation: MFEA-II, IEEE Trans. Evol. Comput. (2019) 1, http://dx.doi.org/10.1109/TEVC.2019.2906927.

[19] H.T.T. Binh, N.Q. Tuan, D.C.T. Long, A multi-objective multi-factorial evolutionary algorithm with reference-point-based approach, in: 2019 IEEE Congress on Evol. Comput. (CEC), 2019, pp. 2824–2831. doi:10.1109/CEC.2019.8790034.

[20] J. Yin, A. Zhu, Z. Zhu, Y. Yu, X. Ma, Multifactorial Evolutionary Algorithm Enhanced with Cross-task Search Direction, in: 2019 IEEE Congress on Evol. Comput. (CEC), 2019, pp. 2244–2251, doi:10.1109/CEC.2019.8789959.

[21] L. Zhou, L. Feng, K. Liu, C. Chen, S. Deng, T. Xiang, S. Jiang, Towards Effective Mutation for Knowledge Transfer in Multifactorial Differential Evolution, in: 2019 IEEE Congress on Evol. Comput. (CEC), 2019, pp. 1541–1547, doi:10.1109/CEC.2019.8790143.

[22] A. Gupta, Y. Ong, L. Feng, K.C. Tan, Multiobjective multifactorial optimization in evolutionary multitasking, IEEE Trans. Cybern. 47 (2017) 1652–1665, http://dx.doi.org/10.1109/TCYB.2016.2554622.

[23] Z. Liang, J. Zhang, L. Feng, Z. Zhu, A hybrid of genetic transform and hyper-rectangle search strategies for evolutionary multi-tasking, Expert Syst. Appl. 138 (2019) 112798, http://dx.doi.org/10.1016/j.eswa.2019.07.015.

[24] L. Feng, L. Zhou, J. Zhong, A. Gupta, Y. Ong, K. Tan, A.K. Qin, Evolutionary multitasking via explicit autoencoding, IEEE Trans. Cybern. 49 (2019) 3457–3470, http://dx.doi.org/10.1109/TCYB.2018.2845361.

[25] Y. Chen, J. Zhong, M. Tan, A fast memetic multi-objective differential evolution for multi-tasking optimization, in: 2018 IEEE Congress on Evolutionary Computation (CEC), IEEE, Rio de Janeiro, 2018, pp. 1–8, http://dx.doi.org/10.1109/CEC.2018.8477722.

[26] Y. Chen, J. Zhong, L. Feng, J. Zhang, An adaptive archive-based evolutionary framework for many-task optimization, IEEE Trans. Emerg. Top. Comput. Intell. (2019) 1–16, http://dx.doi.org/10.1109/TETCI.2019.2916051.

[27] Y. Yuan, Y.-S. Ong, L. Feng, A.K. Qin, A. Gupta, B. Da, Q. Zhang, K.C. Tan, Y. Jin, H. Ishibuchi, Evolutionary multitasking for multiobjective continuous optimization: Benchmark problems, performance metrics and baseline results, 2017, ArXiv:1706.02766.

[28] CEC 2019 Competition on Evolutionary Multi-task Optimization, (n.d.). http://www.bdsc.site/websites/MTO_competiton_2019/MTO_Competition_CEC_2019.html.

[29] J. Rice, C.R. Cloninger, T. Reich, Multifactorial inheritance with cultural transmission and assortative mating. I. Description and basic properties of the unitary models, Am. J. Hum. Genet. 30 (1978) 618–643.

[30] C.R. Cloninger, J. Rice, T. Reich, Multifactorial inheritance with cultural transmission and assortative mating. II. a general model of combined polygenic and cultural inheritance, Am. J. Hum. Genet. 31 (1979) 176–198.

[31] A. Gupta, Y. Ong, L. Feng, Insights on transfer optimization: Because experience is the best teacher, IEEE Trans. Emerg. Top. Comput. Intell. 2 (2018) 51–64, http://dx.doi.org/10.1109/TETCI.2017.2769104.

[32] C.A.C. Coello, N.C. Cortes, Solving multiobjective optimization problems using an artificial immune system, Genet. Program. Evol. Mach. 6 (2005) 163–190, http://dx.doi.org/10.1007/s10710-005-6164-x.

[33] R. Shang, L. Jiao, F. Liu, W. Ma, A novel immune clonal algorithm for MO problems, IEEE Trans. Evol. Comput. 16 (2012) 35–50, http://dx.doi.org/10.1109/TEVC.2010.2046328.

[34] M. Gong, L. Jiao, H. Du, L. Bo, Multiobjective immune algorithm with nondominated neighbor-based selection, Evol. Comput. 16 (2008) 225–255, http://dx.doi.org/10.1162/evco.2008.16.2.225.

[35] Q. Lin, J. Chen, A novel micro-population immune multiobjective optimization algorithm, Comput. Oper. Res. 40 (2013) 1590–1601, http://dx.doi.org/10.1016/j.cor.2011.11.011.

[36] Z.-H. Hu, A multiobjective immune algorithm based on a multiple-affinity model, European J. Oper. Res. 202 (2010) 60–72, http://dx.doi.org/10.1016/j.ejor.2009.05.016.

[37] L. Li, Q. Lin, S. Liu, D. Gong, C.A. Coello Coello, Z. Ming, A novel multi-objective immune algorithm with a decomposition-based clonal selection, Appl. Soft Comput. 81 (2019) 105490, http://dx.doi.org/10.1016/j.asoc.2019.105490.

[38] Y. Lei, M. Gong, J. Zhang, W. Li, L. Jiao, Resource allocation model and double-sphere crowding distance for evolutionary multi-objective optimization, European J. Oper. Res. 234 (2014) 197–208, http://dx.doi.org/10.1016/j.ejor.2013.09.007.
[39] Z. Liang, R. Song, Q. Lin, Z. Du, J. Chen, Z. Ming, J. Yu, A double-module immune algorithm for multi-objective optimization problems, Appl. Soft Comput. 35 (2015) 161–174, http://dx.doi.org/10.1016/j.asoc.2015.06.022.
[40] Q. Lin, J. Chen, Z.-H. Zhan, W.-N. Chen, C.A.C. Coello, Y. Yin, C.-M. Lin, J. Zhang, A hybrid evolutionary immune algorithm for multiobjective optimization problems, IEEE Trans. Evol. Comput. 20 (2016) 711–729, http://dx.doi.org/10.1109/TEVC.2015.2512930.
[41] Q. Lin, Y. Ma, J. Chen, Q. Zhu, C.A.C. Coello, K.-C. Wong, F. Chen, An adaptive immune-inspired multi-objective algorithm with multiple differential evolution strategies, Inf. Sci. 430–431 (2018) 46–64, http://dx.doi.org/10.1016/j.ins.2017.11.030.
[42] Q. Lin, Q. Zhu, N. Wang, P. Huang, W. Wang, J. Chen, Z. Ming, A multi-objective immune algorithm with dynamic population strategy, Swarm and Evol. Comput. 50 (2019) 100477, http://dx.doi.org/10.1016/j.swevo.2018.12.003.
[43] M. Heidelberger, The clonal selection theory of acquired immunity, Arch. Biochem. Biophys. 89 (1960) 151, http://dx.doi.org/10.1016/0003-9861(60)90028-X.
[44] J. Yoo, P. Hajela, Immune network simulations in multicriterion design, Struct. Optim. 18 (1999) 85–94, http://dx.doi.org/10.1007/BF01195983.
[45] L. Jiao, M. Gong, R. Shang, H. Du, B. Lu, Clonal selection with immune dominance and anergy based multiobjective optimization, in: E. Zitzler C.A. Coello Coello (Ed.), Evolutionary Multi-Criterion Optimization, Springer, Berlin, Heidelberg, 2005, pp. 474–489, http://dx.doi.org/10.1007/978-3-540-31880-4_33.
[46] C.A.C. Coello, N.C. Cortés, An Approach To Solve Multiobjective Optimization Problems Based on an Artificial Immune System, 2002.
[47] F. Freschi, M. Repetto, VIS: An artificial immune network for multi-objective optimization, Eng. Optim. 38 (2006) 975–996, http://dx.doi.org/10.1080/03052150600880706.
[48] J. Gao, J. Wang, WBMOAIS: A novel artificial immune system for multi-objective optimization, Comput. Oper. Res. 37 (2010) 50–61, http://dx.doi.org/10.1016/j.cor.2009.03.009.
[49] Q. Lin, Q. Zhu, P. Huang, J. Chen, Z. Ming, J. Yu, A novel hybrid multi-objective immune algorithm with adaptive differential evolution, Comput. Oper. Res. 62 (2015) 95–111, http://dx.doi.org/10.1016/j.cor.2015.04.003.
[50] E.Y.C. Wong, H.S.C. Yeung, H.Y.K. Lau, Immunity-based hybrid evolutionary algorithm for multi-objective optimization in global container repositioning, Eng. Appl. Artif. Intell. 22 (2009) 842–854, http://dx.doi.org/10.1016/j.engappai.2008.10.010.
[51] R.B. Agrawal, K. Deb, R.B. Agrawal, Simulated binary crossover for continuous search space, Complex Syst. 9 (2000) 115–148.
[52] K. Deb, D. Deb, Analysing mutation schemes for real-parameter genetic algorithms, Int. J. Artif. Intell. Soft Comput. 4 (2014) 1–28, http://dx.doi.org/10.1504/IJAISC.2014.059280.