Information Sciences 596 (2022) 236–263

Contents lists available at ScienceDirect

Information Sciences

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

A novel membrane-inspired evolutionary framework for multi-objective multi-task optimization problems

Zhiwei Xu [a], Kai Zhang [a,b], Juanjuan He [a,b,*], Xiaoming Liu [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 30 June 2021
Received in revised form 26 January 2022
Accepted 3 March 2022
Available online 8 March 2022

Keywords:
Membrane-inspired evolutionary algorithm
Membrane computing
Evolutionary multitasking
Multi-objective multi-task optimization
Multi-objective evolutionary algorithm

ABSTRACT

In recent years, many different membrane-inspired evolutionary algorithms have been proposed to solve various complex optimization problems. Considering membrane systems' powerful computing performance and parallel capability, it has outstanding potential in solving multi-task optimization problems. However, there is no research to explore the performance of membrane-inspired evolutionary algorithms in solving multi-task optimization problems. In this paper, a novel membrane-inspired evolutionary framework with a hybrid dynamic membrane structure is proposed to solve the multi-objective multi-task optimization problems. First, a novel membrane-inspired two-stage evolution strategy algorithm is proposed as the algorithm in the membrane to improve the convergence of the algorithm and the diversity of multisets. Second, the information molecule concentration vector is proposed to reduce negative information transfer. The information molecule concentration vector is inspired by the binding process of information molecules and receptors and can control the information transfer probability adaptively. Finally, comprehensive experimental results show that the proposed algorithm performs better than most advanced multi-objective evolutionary multitasking algorithms.

1. Introduction

Many scientific and engineering applications require simultaneous optimization of multiple conflicting objective functions. These problems are collectively referred to as multi-objective optimization problems (MOPs). Without loss of generality, assuming the problem is a minimization problem, a MOP can be expressed as

$$\min_{x\in\Omega} \mathrm{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}), \cdots, f_M(\mathbf{x})) \tag{1}$$

where $\mathbf{x} = (x_1, x_2, x_3, \cdots, x_D) \epsilon\, \Omega$ is a D-dimensional decision vector, Ω represents the feasible region in the decision space, and $\mathrm{F}(\mathbf{x})$ denotes a set of objective functions. Due to the contradictory objective functions, it is impossible to search out a single solution to optimize all objective functions. Thus, the Pareto optimal concept is proposed to seek out a set of optimal solutions that can tradeoff objective functions. Given two decision vectors $\mathbf{x}$ and $\mathbf{y}$, if $\forall i \in \{1, 2, 3 \cdots, M\}\ f_i(\mathbf{x}) \le f_i(\mathbf{y})$ and $\exists j \in \{1, 2, 3 \cdots M\}\ f_j(\mathbf{x}) < f_j(\mathbf{y})$, then $\mathbf{y}$ is Pareto dominated by $\mathbf{x}$, expressed as $\mathbf{x} \prec \mathbf{y}$. If no solution can dominate the solution $\mathbf{x}^*$ in the population, then $\mathbf{x}^*$ is a non-dominated solution or a Pareto optimal solution. The set of all non-dominated solutions is Pareto optimal set (PS), and its corresponding mapping in the objective space is Pareto optimal front (PF) [1].

* Corresponding author at: School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, China.
E-mail address: hejuanjuan@wust.edu.cn (J. He).

https://doi.org/10.1016/j.ins.2022.03.020
0020-0255/

Profit from the efficient search mode based on population, multi-objective evolutionary algorithms (MOEAs) can search out the PF with great convergence and distribution in a single run and have been widely used to solve MOPs. MOEAs can be divided into three main categories, Pareto dominance-based algorithms [2,3], algorithms based on indicators [4,5], and decomposition-based algorithms [6,7]. These MOEAs are designed to solve one MOP at a time, and when faced with a new problem, the population needs to be reinitialized. However, in the real world, many MOPs are related, and the knowledge gained from solving one task can be reused by related tasks and bring positive effects.

Inspired by multi-task learning and transfer learning in machine learning [8,9], Gupta et al. [10] proposed an optimization paradigm that utilizes the correlation between multiple optimization tasks to process these tasks simultaneously, called multi-task optimization (MTO). Unlike traditional MOEAs, evolutionary multitasking (EMT) algorithms are committed to finding the optimal solution (set) of all the problems at one time and improving the overall search speed by using the relevance of simultaneous optimization tasks. Because of its powerful search ability and the forward-looking theoretical viewpoint, MTO has become a research hotspot in the field of evolutionary computation [11–14].

Membrane computing is a distributed computing model inspired by the compartmentalized structures and functions of biological membranes in cells [15]. Membrane computing serves as the computational model's working mechanism by abstracting the chemical reactions between the membranes and the chemical reactions in the membrane regions. The membrane system, referred to as the P system, has strong parallelism to solve complex problems in polynomial time. And it has been proved to have the same computing power as the Turing general computing model [16,17]. Due to the P system's outstanding performance and natural parallel capability, the membrane system-inspired optimization algorithm (MIEA) is proposed, which combines the membrane structure, evolutionary rules, and computing mechanisms with the search principles of the meta-heuristic algorithm. The evolutionary algorithms nested in the membrane system structure are called algorithms in the membrane (AIMs). After twenty years of development, multitudinous MIEAs have been proposed to solve various complex optimization problems in the real-world [18–23].

Considering the membrane system has powerful computing performance and parallel capability, it can be envisaged that the outstanding potential of MIEAs when handling multitudinous tasks simultaneously. However, there is no research studying the performance of MIEAs in solving MTO problems. Therefore, this paper proposes a multi-objective multi-task evolutionary framework based on membrane system (EMT-MOMIEA) to solve the multi-objective multi-task optimization (MOMTO) problems. First, the overall algorithm is contained in a skin membrane, and each task is involved in a separate sub-membrane. The rewrite rules are introduced to evolve symbol objects to converge. The communication rules are adopted to exchange and reuse information between the membranes that solve different tasks. Second, a novel membrane system-based two-stage evolution strategy algorithm is proposed as AIM. In the first stage, a novel differential evolution strategy based on dynamic membrane structure (DMS-DES) is utilized to simultaneously perturb all decision variables to optimize linearly non-separable variables roughly. In the second stage, the precision controllable mutation-based evolution strategy (PCM-ES) is applied to mutate a single decision variable to optimize the linearly separable decision variables. Third, inspired by the binding process of information molecules and receptors during information exchange between cells, the information molecule concentration vector (IMCV) concept is proposed. The IMCV can dynamically adjust the information transfer probability and reduce negative information transfer. Finally, comprehensive empirical experiments are carried out on the classical, complex MOMTO test suite [43,44] and the multi-objective many-task optimization (MOMaTO) benchmark test suite [44] to verify the efficacy of EMT-MOMIEA. The experiment results clearly show that the proposed EMT-MOMIEA provides a competitive edge over the state-of-the-art EMT algorithms. Comparative experiments are carried out on the proposed operators to verify their effectiveness. Specifically, the main contributions can be summarized as follows:

(1) Inspired by the cell-like membrane system, a novel membrane-inspired evolutionary framework with a hybrid dynamic membrane structure is proposed to solve the MOMTO problems.
(2) To improve the convergence of the algorithm and the diversity of multisets, a novel membrane system-based two-stage evolution strategy algorithm is proposed as AIM.
(3) The IMCV concept is proposed to control the information transfer intensity between different tasks from the decision variable level to avoid the negative transfer.
(4) The performance of EMT-MOMIEA is investigated in various test suites and compared with the state-of-the-art EMT algorithms. Both strengths and weaknesses of the proposed method are discussed.

The remaining sections are organized as follows. Section 2 conducts a comprehensive survey of related works. Section 3 details the proposed EMT-MOMIEA. Section 4 provides the extensive experimental results to verify the effectiveness of EMT-MOMIEA. Finally, section 5 summarizes the paper and outline future research directions.

2. Related work

2.1. Membrane-inspired evolutionary algorithms

MIEA is a novel heuristic algorithm developed by combining the P system's membrane structure, evolutionary rules, and computational mechanism with the search principle of the evolutionary algorithm. Inspired by the cell-like P system, the

hierarchical structure MIEAs have been widely used to solve real-world problems due to their diversified structure and rich computing rules [18,19,35–40]. The proposed EMT-MOMIEA is also based on the hierarchical structure.

Inspired by the structures and functions of living cells, the essential components of a cell-like P system include membrane structure, reaction rules, and multisets. As shown in Fig. 1, the membrane structure is the hierarchical arrangement of membranes. The outermost membrane is called the skin membrane, which separates the P system from the external environment. The membrane that does not contain the sub-membrane is called the elementary membrane. Each membrane defines a region. For an elementary membrane, the region is the space it has. While for a non-elementary membrane, the region refers to the space between itself and the membrane it directly contains. The region includes multisets and reaction rules. The multiset is the collection of symbol objects which abstract the elements in living cells. Reaction rules simulate the chemical reactions and substances flow in biological membranes. The evolution of symbol objects can be realized by executing reaction rules in the region. When the reactions in all regions have been executed, the multiset is output to the environment or the designated membrane as the final result.

According to the cell membrane structure, the hierarchical MIEA can be classified as nest membrane structure (NMS), one-level membrane structure (OLMS), hybrid membrane structure (HMS), and dynamic membrane structure (DMS). In NMS, each membrane only contains one sub-membrane, the membrane is nested with each other, and the innermost membrane contains the final optimal solution [18,19]. In the evolution process, the intermediate membrane continuously transmits the superior solution to the nearby inner layer and the inferior solution to the adjacent outer layer through the communication rule. Each region assembles a unique AIM that can independently apply diverse rewrite rules to evolve objects.

Concerning OLMS, there are only m elementary membranes in the skin membrane, and each membrane can have unique AIM and reaction rules. The reaction rules mainly include rewrite rules and communication rules. The communication rule sends the fittest object from each elementary membrane as the local optimum to the skin membrane. Then, the fittest object in the skin membrane is as the global optimum transmitted back to each elementary membrane. Therefore, the OLMS can balance exploration and exploitation and is more efficient than the NMS. The OLMS has been combined with various AIMs and applied to various practical and benchmark problems [35,36].

HMS and DMS have more flexible membrane structures and diverse reaction rules. HMS [37,38] is the hybridization of NMS and OLMS that multiple HMS are compacted in a single OLMS structure. DMS [39,40] can dynamically adjust membrane structure through division rule and dissolution rule in the evolutionary process, contributing to the algorithm adopting disparate operators in different evolution stages to balance exploration and exploitation. Through the division rule, the membrane can be divided into several sub-membranes. The dissolution rule can dissolve the membrane to disperse all the objects to the outer membrane and redistribute the object resource.

2.2. Evolutionary multitasking

Current optimization algorithms often start the search at the ground zero knowledge state and suppose that the optimized problems are independent. After dealing with a problem, the optimizer's ability to solve similar problems will not be improved. However, many problems are interrelated and have similar characteristics. For example, Rastrigin function and Sphere function have the same global optimal point, located at ($x_1 = 0, x_2 = 0$). Rastrigin function possesses multiple local minima and is hard to find the global optimum, but the Sphere function has only one minimum value and is easier to solve,

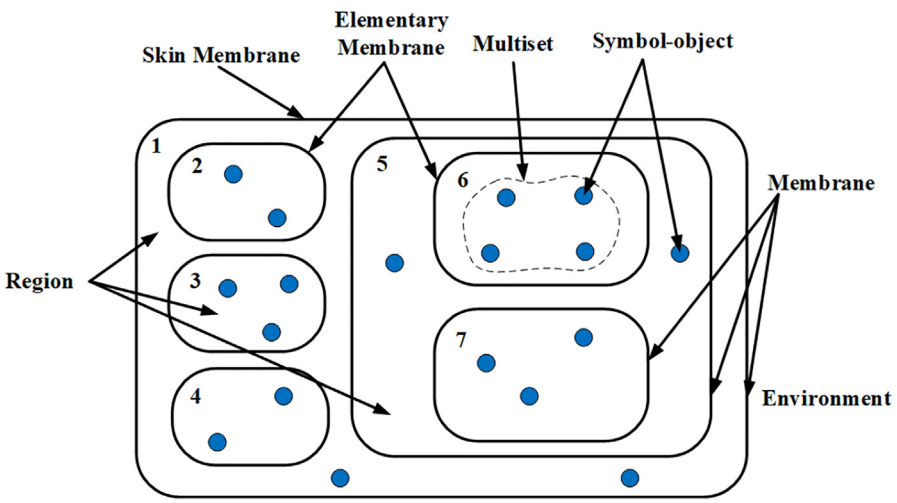

Fig. 1. The basic structure of the membrane systems.

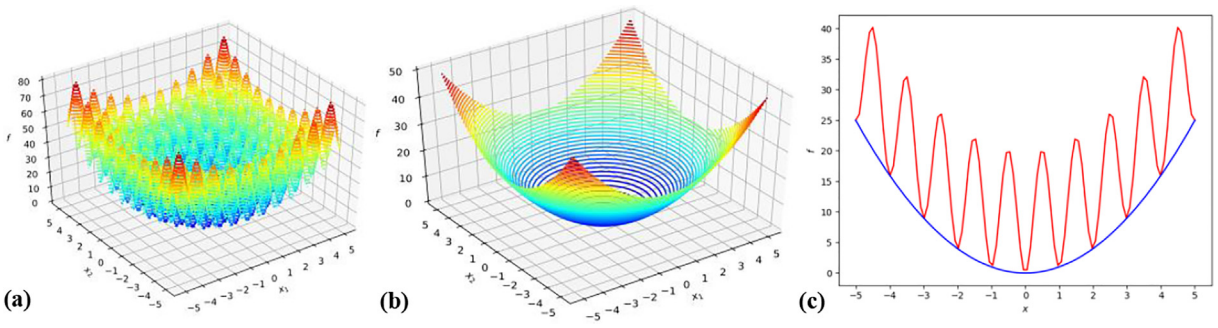

Fig. 2. Schematic diagram of Sphere function and Rastrigin function. (a) Contour map with Rastrigin function of 2D decision variables, (b) Contour map of Sphere function with 2D decision variables, (c) Function graph of Rastrigin function, and Sphere function with 1D decision variable.

as shown in Fig. 2**.** If the population information when optimizing the Sphere can be reused in the process of optimizing the Rastrigin function, it can accelerate the Rastrigin function's optimization.

Inspired by multi-task learning and transfer learning in machine learning [8,9], the EMT theory is proposed that the knowledge obtained from one problem can stimulate related issues. The EMT theory aims to accelerate the simultaneous optimization of multiple problems by reusing similar information to improve the overall convergence performance. The paradigm of optimizing multiple tasks simultaneously is called MTO. The source of helpful information is called the source task, and the task of reusing information is referred to as the target task.

Without loss of generality, assuming K simultaneous optimization tasks are all minimization problems, MTO's formal representation is shown as Eq. (2). $\mathbf{x}_j^*$ represents the optimum solution (set) of the jth task T_j (j = 1, 2, ...K).

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

In the problem paradigm, the task T_j can be a single objective optimization problem (SOP) or a MOP. If an MTO problem contains at least one MOP, it can be called a MOMTO problem.

Inspired by the multifactorial inheritance theory, Gupta *et al.* [10] first proposed a general EMT framework, namely multifactorial evolutionary algorithm (MFEA). Each task is regarded as an independent environmental influence bias that affects the evolution of offspring. In a multifactorial environment, the individual has different fitness in different environments, and the index of the fittest environment is marked as the skill factor. Following Darwinism, the MFEA paradigm proposed two strategies, vertical cultural transmission, and assortative mating, to realize information transfer in the multifactorial environment. The vertical cultural transmission theory states that the offspring should inherit its parent's adept task, and the individual will only be evaluated on the task according to its skill factor. Specifically, if the offspring has only one parent, it directly inherits the parent's skill factor. Otherwise, it inherits the skill factor from either parent with equal probability. The assortative mating theory is proposed to control the information transfer intensity across tasks. It points out that only individuals with the same skill factor can mate freely. In contrast, individuals with different skill factors must satisfy a probability threshold, namely random mating probability (*rmp*), which indicates the information transfer intensity across tasks.

However, the information transfer is not necessarily effective, and negative transfer often occurs. The negative transfer means that the solution generated after the target task receives the information is inferior to the offspring without the transfer. Fig. 3 shows the performance on test problems [43] after information transfer [10] by simulated binary crossover operator (SBX) every 50 generations. As shown in Fig. 3(a), for the PIHS1 problem, the information obtained through SBX from PIHS2 is valuable that can make PIHS1 converge quickly within 200 generations, which means a positive transfer. For the PILS2 problem, the information from PILS1 interferes with PILS2′s convergence, which indicates a negative transfer, as shown in Fig. 3(b).

Current EMT algorithms inherit the MFEA framework and focus on improving search efficiency and avoiding negative information transfer. They can be classified into five categories, including methods for improving knowledge transfer strategies [24,25], approaches based on allocating search resources [26,27], methods for search space mapping [28,29], methods based on selecting optimal source tasks [30,31], and improving search strategies approaches [32–34]. The proposed EMT-MOMIEA put forward the information transfer method based on IMCV to improve knowledge transfer efficiency. A membrane system-based two-stage evolution strategy algorithm is also proposed as AIM from the aspect of enhancing search efficiency.

3. Proposed algorithm

This section introduces the proposed multi-objective multi-task evolutionary framework based on membrane system (EMT-MOMIEA), the proposed two-stage membrane system-based evolution strategy algorithm, and the proposed IMCV-based information transfer strategy.

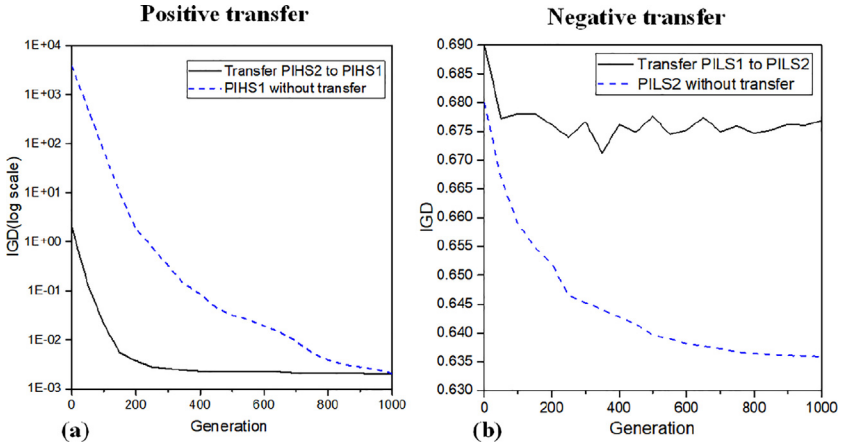

Fig. 3. Illustration of positive transfer and negative transfer between tasks.

3.1. The framework of EMT-MOMIEA

The main framework of EMT-MOMIEA is summarized in **Algorithm 1**. First, according to **Algorithm 2**, initialize the membrane structure and multiset in line 1. In **Algorithm 2**, first, the skin membrane is initialized and the division rule is called to generate K sub-membranes. Each sub-membrane is used to optimize an independent task. The division rule is shown in Eq. (3), where n is the number of sub-membranes, $[]_0$ represents the skin membrane, and $[]_i$ represents the ith sub-membrane. Then, the division rule is utilized in each sub-membrane to generate an inner membrane to implement the two-stage evolution strategy algorithm. Each task is allocated the same computing resources, and N symbol objects are equally distributed to K sub-membranes. For a single symbol object $\mathbf{x}_i$ in the kth sub-membrane, initialize it randomly between the upper and lower bounds of the kth task, assign it to skill factor k, and evaluate it on the kth task.

Division rule:

$$[]_0 \rightarrow [[]_1, []_2, []_3, \ldots []_i \ldots, []_{n-1}, []_n]_0 \tag{3}$$

Algorithm 1
The overall framework of EMT-MOMIEA

Input: N: the total number of symbol objects, K: the number of tasks.
Output: the final multisets.
1. Initialize the membrane structure and the symbol objects according to **Algorithm 2**.
2. **For each** kth inner membrane in K inner membranes
3. ***DMS-DES*** (kth sub-multiset) as **Algorithm 3**.
4. **End for**
5. Call ***Dissolution rule*** to release all the inner membranes.
6. **For** kth sub-membrane in K sub-membranes
7. $j = random(K), j \neq k$.
8. Initialize the information molecule concentration vector $\overrightarrow{imcv}^K = \overrightarrow{0.5}$.
9. **End for**
10. **While** the stop criterion is not met **do**
11. **For** kth sub-membrane in K sub-membranes
12. $\overrightarrow{ngc} = \overrightarrow{0}$, $\overrightarrow{ngt} = \overrightarrow{0}$, $\overrightarrow{nsc} = \overrightarrow{0}$, $\overrightarrow{nst} = \overrightarrow{0}$.
13 NS = ***Obtain the non-dominated solution set*** (kth sub-multiset) according to **Algorithm 4**.
14. **For each** symbol object **x** in kth sub-multiset
15. **For each** d dimension
16. **If** $d \leq DimMin$ **then**
17. **o**, $Istransfer$ = ***Crossover with communication rule based on IMCV*** (**x**, j) as **Algorithm 6**.
18. **End if**

19. **o** = ***PCM-ES*** (**o**) according to **Algorithm 5**.
20. ***Update the influence factor of IMCV*** according to **Algorithm 7**.
21. **End for**
22. **End for**
23. ***Update the IMCV*** according to **Algorithm 8**.
24. **End for**
25. **End while**
26. Call ***Dissolution rule*** to release all the sub-membranes to the skin membrane.

After initializing the membrane structure, the DMS-DES is applied to evolve the multiset in each inner membrane according to **Algorithm 3** in line 3. The dissolution rule is used to dissolve the membrane structure and release all the multisets into the sub-membranes to proceed to the next evolution stage. The dissolution rule is shown in formula (4). After the ith membrane structure is dissolved, the symbol objects are released into the adjacent outer membrane containing the ith membrane.

Dissolution rule:

$$[[\mathbf{x}]_i]_{outter} \rightarrow [\mathbf{x}]_{outter} \tag{4}$$

Then, a random task j is selected from K tasks as the source task for one sub-membrane, where $j \neq k$. Afterward, initialize the information molecule concentration vector $\overrightarrow{imc\nu}$ of the kth task as a $DimMin$ dimensional all-zero vector where $DimMin$ = min {$Dimj$, $Dimk$}. Next, traverse and evolve all sub-membranes until the stop condition is reached. In the kth sub-membrane, initialize the influence factors of $\overrightarrow{imc\nu}$ as $DimMin$ dimensional vectors, which are $\overrightarrow{ngc}$, $\overrightarrow{ngt}$, $\overrightarrow{nsc}$ and $\overrightarrow{nst}$ in line 12. $\overrightarrow{ngc}$ and $\overrightarrow{ngt}$ denote the number of offspring generated by conventional crossover and generated by transfer crossover, respectively. $\overrightarrow{nsc}$ and $\overrightarrow{nst}$ represent the number of offspring superior to parents generated by conventional crossover and generated by transfer crossover, respectively. Afterward, the non-dominated solution set in the current kth sub-multiset is obtained by **Algorithm 4** in line 13. Next, each symbol object **x** in the sub-multiset evolves by the crossover with communication rule based on IMCV as shown in **Algorithm 6** in line 17. Then, symbol object **x** is evolved using PCM-ES according to **Algorithm 5** in line 19. After a symbol object has evolved, the influence factors of IMCV are updated according to **Algorithm 7** in line 20. After all the individuals in the kth sub-multiset have been evolved, update the IMCV according to **Algorithm 8** in line 23. Finally, after all the evolution processes are over, call the ***Dissolution rule*** to release all the sub-membranes to the skin membrane. The flowchart of the proposed EMT-MOMIEA is indicated in Fig. 4, and the membrane structure of the proposed EMT-MOMIEA is shown in Fig. 5. The details of the proposed two-stage evolution strategy algorithm based on the membrane system and the proposed information transfer rule based on IMCV are described in the following subsections.

Algorithm 2
The pseudocode of Initialization.

Input: N: the total number of the symbol objects, K: the total number of the tasks.
Output: The initial skin membrane and K sub-membranes with initial multisets.
1. Create the skin membrane.
2. Call the ***Division rule*** to create K sub-membranes.
3. **For each** kth sub-membrane in the K sub-membranes
4. Call the ***Division rule*** to create one inner membrane.
5. Assign $\lfloor N/K \rfloor$ symbol objects to the kth sub-membrane.
6. **For each** symbol object $\mathbf{x}_i$ in the kth sub-membrane
7. **For each** dimension d of the $\mathbf{x}_i$ do
8. $\mathbf{x}_i^d = Low_k^d + random(Up_k^d - Low_k^d)$
9. **End for**
10. Assign skill factor k to $\mathbf{x}_i$.
11. Evaluate the $\mathbf{x}_i$ for the task k.
12. **End for**
13. **End for**

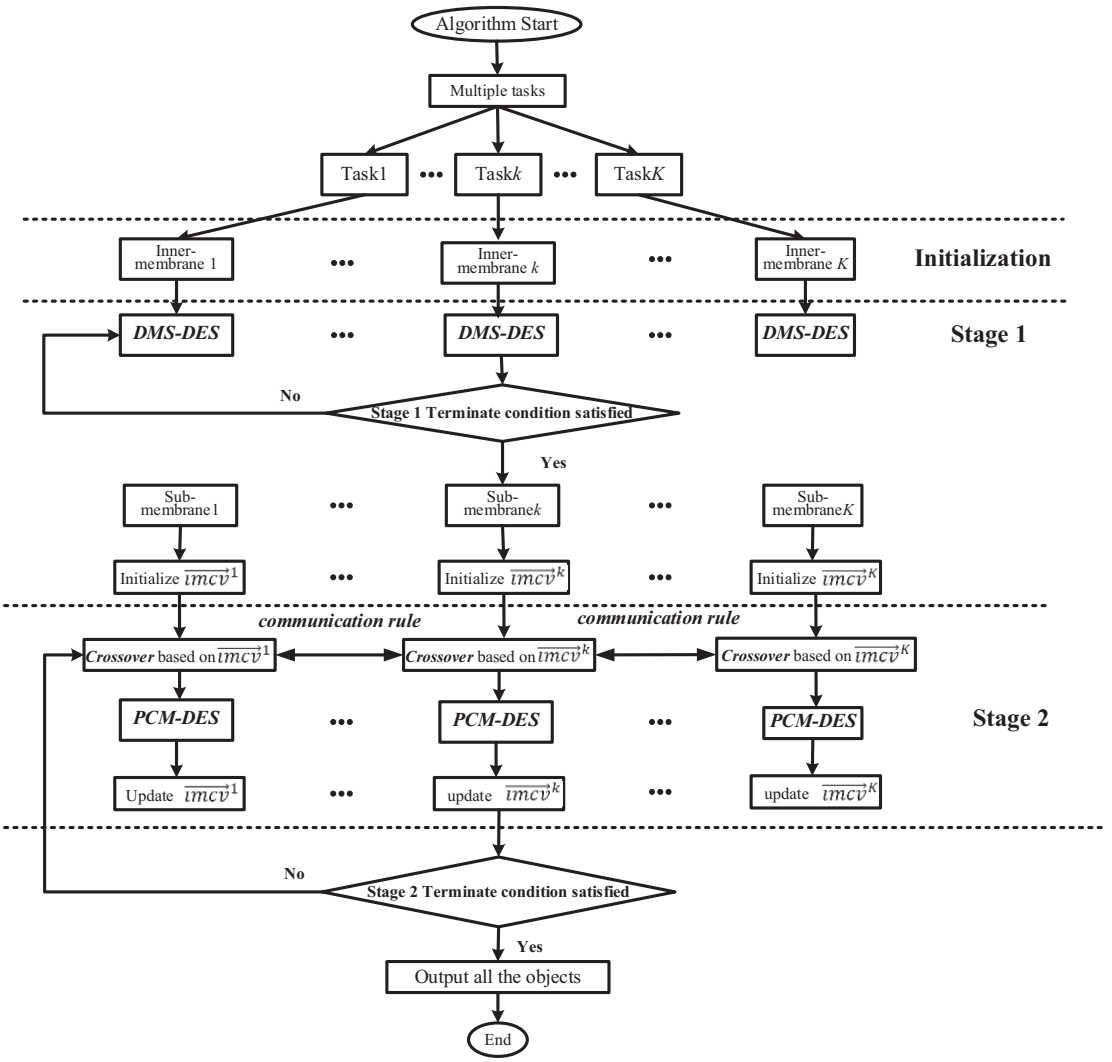

Fig. 4. The flowchart of the proposed EMT-MOMIEA.

3.2. Two-stage evolution strategy algorithm based on membrane system

Real-world problems often have parameter interactions which make decision variables linearly non-separable. The linearly non-separable means no D-1 dimensional hyperplane that can separate the D dimensional space. Problems with parameter interactions are always difficult to solve. For example, take one single-objective minimum optimization problem with two-dimensional decision variables, as shown in Eq. (5).

$$f_1(x_1, x_2) = {x_1}^2 + \lambda {x_2}^2 \tag{5}$$

Let λ be 2, Eq. (5) is a very simple separable problem, and the global optimal solution is $(x_1 = 0, x_2 = 0)$. But after introducing the rotation matrix $\begin{bmatrix} \alpha_1 & \alpha_2 \\ \alpha_3 & \alpha_4 \end{bmatrix}$ into Eq. (5), which $\begin{bmatrix} {x_1}' \\ {x_2}' \end{bmatrix} = \begin{bmatrix} \alpha_1 & \alpha_2 \\ \alpha_3 & \alpha_4 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, the problem becomes non-separable, as shown in Eq. (6).

$$f_2(x_1, x_2) = (\alpha_1 x_1 + \alpha_2 x_2)^2 + \lambda(\alpha_3 x_1 + \alpha_4 x_2)^2 = ({\alpha_1}^2 {x_1}^2 + {\alpha_2}^2 {x_2}^2 + 2\alpha_1\alpha_2 x_1 x_2) + \lambda({\alpha_3}^2 {x_1}^2 + {\alpha_4}^2 {x_2}^2 + 2\alpha_3\alpha_4 x_1 x_2)$$

$$= ({\alpha_1}^2 + \lambda{\alpha_3}^2){x_1}^2 + ({\alpha_2}^2 + \lambda{\alpha_4}^2){x_2}^2 + 2(\alpha_1\alpha_2 + \lambda\alpha_3\alpha_4)x_1 x_2 \tag{6}$$

Assuming the rotation matrix is $\begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}$, f_2 can be written in the form of Eq. (7).

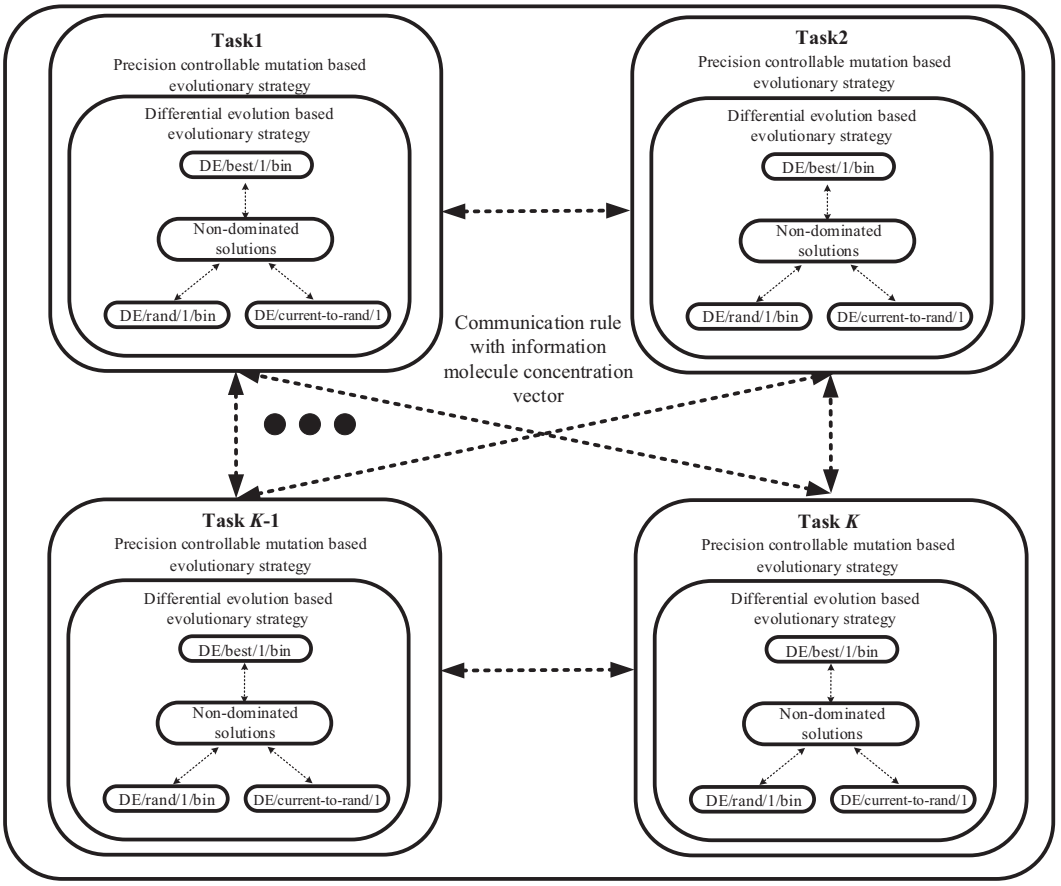

Fig. 5. The structure of the proposed EMT-MOMIEA.

$$f_2(x_1, x_2) = \frac{3}{2}{x_1}^2 + \frac{3}{2}{x_2}^2 + x_1 x_2 \tag{7}$$

Fig. 6 shows the contour plots of Eqs. (5) and (7). The contour indicates an area of fixed fitness value. From Fig. 6(a), it is evident that the function is aligned with the principle coordinate axis and the two decision variables are linearly separable. The two components, ${x_1}^2$and$\lambda{x_2}^2$ can be considered as two independent sub-problems to be solved. The algorithm can individually perturb x_1 or x_2 to search the global optimum. Take an extreme example, suppose point A is on the x_2 axis. Then, by interfering independently with x_1, point A can move to the global minimum at the origin point. After Eq. (5) is appended with the rotation matrix, parameter interaction is introduced through the term x_1x_2. It can be found from Fig. 6 (b) that, although independently changing x_1 or x_2 of A' along the direction of the principle coordinate axis can improve the quality of A'. However, it is impossible to climb to the contour line where the fitness value is 10, and the search will be in a dilemma. Improving all the decision variables simultaneously can the solution develop towards the global optimum effectively.

Due to problems with parameter interactions cannot be optimized by evolving a single variable alone [41]. Therefore, simultaneous perturbation of the non-separable variables is needed. However, it is not known whether the problem is linearly non-separable or which variables have interactions before optimization. Blindly perturbing all variables will reduce the search efficiency.

In this paper, a novel membrane system based two-stage evolution strategy algorithm is proposed as AIM. In the first stage, DMS-DES is utilized to simultaneously perturb all decision variables to optimize linearly non-separable variables roughly. In the second stage, PCM-ES is applied to mutate a single decision variable to optimize the linearly separable decision variables.

In the first stage, DMS-DES is designed based on the dynamic membrane structure combined with the differential evolution algorithm. The pseudocode of DMS-DES is shown in **Algorithm 3**. The allocation proportion of total evolution times of DMS-DES is a predetermined decimal value, namely p_{des}. First, create four elementary membranes by the division rule,

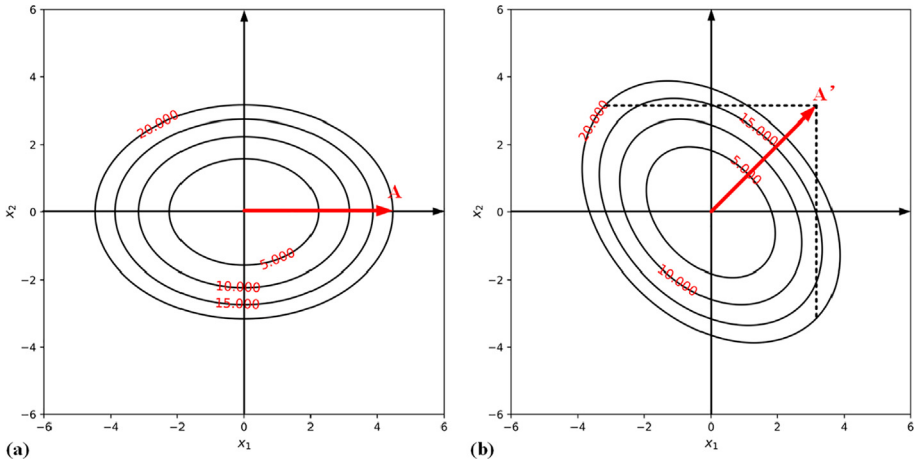

Fig. 6. The contour plots of the Eqs. (5) and (7).

namely *em1, em2, em3,* and *em4*. The DMS-DES has an elite guidance mechanism, which leads the remaining individuals to converge guided by the current non-dominated solutions. The current non-dominated solution set is obtained by **Algorithm 4**, where *DOMp* represents the total number of symbol objects that can dominate **p** in the multiset. If there is no symbol object in the multiset that can dominate **p**, storage **p** in the temporary storage set *NS*. Then equally distribute the symbol objects in the *P* into membranes *em1*, *em2*, and *em3*, and send the storage set *NS* to *em4*. To enable all membranes to obtain the elite information of the current population, call the communication rule as shown in Eq. (8) to send the non-dominated solutions in *em4* to *em1*, *em2*, and *em3*.

Communication rule:

$$\begin{aligned}
&[\mathbf{x}_{\boldsymbol{rand}1}]_{em4} \rightarrow [\mathbf{x}_{\boldsymbol{rand}1}]_{em1} \\
&[\mathbf{x}_{\boldsymbol{rand}2}]_{em4} \rightarrow [\mathbf{x}_{\boldsymbol{rand}2}]_{em2} \\
&[\mathbf{x}_{\boldsymbol{rand}3}]_{em4} \rightarrow [\mathbf{x}_{\boldsymbol{rand}3}]_{em3}
\end{aligned} \tag{8}$$

where $\mathbf{x}_{\boldsymbol{rand}1}$, $\mathbf{x}_{\boldsymbol{rand}2}$, and $\mathbf{x}_{\boldsymbol{rand}3}$ represent three elite non-dominated solutions randomly selected from *em*4 respectively. The *rand1*, *rand2*, *rand3* are the random integer values in [1, N_{em4}], where N_{em4} denotes the number of symbol objects in membrane *em4*. Then, the rewrite rule 1**,** rewrite rule 2, and rewrite rule 3 are utilized to evolve the symbol objects in elementary membranes *em1, em2,* and *em3*, respectively. The three rewrite rules are improved based on the differential evolution (DE) operator and have different search behaviors, as shown below.

Rewrite rule 1:

$$\mathbf{v} = \mathbf{x}_{1\boldsymbol{best}} + F_1 * (\mathbf{x}_{\boldsymbol{r}11} - \mathbf{x}_{\boldsymbol{r}12})$$

o = crossover $(\mathbf{v}, \mathbf{x})$

Rewrite rule 2:

$$\mathbf{v} = \mathbf{x}_{\boldsymbol{r}21} + F_1 * (\mathbf{x}_{\boldsymbol{r}22} - \mathbf{x}_{\boldsymbol{r}23})$$

o = crossover $(\mathbf{v}, \mathbf{x})$

Rewrite rule 3:

$$\mathbf{v} = \mathbf{x} + F_2 * (\mathbf{x}_{\boldsymbol{r}31} - \mathbf{x}) + F_1 * (\mathbf{x}_{\boldsymbol{r}32} - \mathbf{x}_{\boldsymbol{r}33})$$

$$\mathbf{o} = \mathbf{crossover}(\mathbf{v}, \mathbf{x} \tag{9}$$

where $\mathbf{v}$ denotes the trial vector, F_1 and F_2 are the two scaling factors. $\mathbf{x}_{1\boldsymbol{best}}$ is the non-dominated solutions obtained from membrane *em4* by communication rule. $\mathbf{x}_{\boldsymbol{r}11}$ and $\mathbf{x}_{\boldsymbol{r}12}$ are the two symbol objects randomly chosen in *em1* and $\mathbf{x}_{\boldsymbol{rand}1}$. $\mathbf{x}_{\boldsymbol{r}21}$, $\mathbf{x}_{\boldsymbol{r}22}$ and $\mathbf{x}_{\boldsymbol{r}23}$ indicate the three symbol objects randomly selected from *em2* and $\mathbf{x}_{\boldsymbol{rand}2}$. $\mathbf{x}_{\boldsymbol{r}31}$, $\mathbf{x}_{\boldsymbol{r}32}$ and $\mathbf{x}_{\boldsymbol{r}33}$ represent the three sym-

bol objects randomly picked from the *em3* and $\mathbf{x}_{rand3}$. Then, the crossover operator is applied to mate the trial vector **v** and the parent **x.** The SBX is used as the crossover operator here.

Rewrite rule 1 has a superduper global search ability, but the local search performance is relatively weak. Rewrite rule 2 possesses a slower global search speed, but the local search ability is more excellent. Rewrite rule 3 has rotation-invariant property. In the early iterations, the rewrite rule 1 can promote rapid convergence. In the middle and late iterations, rewrite rule 2 becomes preferable to generate more non-dominated solutions. The division and dissolution rules can dynamically adjust the membrane structure and apply the three rewrite rules evenly to the whole population. The communication rule can share the quality information from the elites to the other individuals.

Following the MFEA framework, for each generated offspring **o**, the skill factor is inherited from its parent and only be evaluated on the corresponding task. If the parent is dominated by the offspring, the offspring replaces the parent. If they are non-dominated, keep the lesser-dominated individuals. If the number of individuals dominating the two is the same, consider the crowding degrees. The *MED* [1] is the crowding degree indicator used in this paper, maintaining a uniform distance among individuals. The formal representation of *MED* is shown in Eq. (10).

$$MED(\mathbf{x}_i) = Totaldis(\mathbf{x}_i) \times Neardis(\mathbf{x}_i)$$

where

$$Totaldis(\mathbf{x}_i) = \sum_{j=1}^{N} \sum\nolimits_{m=1}^{M} \sqrt[2]{\left(f_m^i - f_m^j\right)^2}, i,j \in \{1,2,\cdots,N\}$$

$$Neardis(\mathbf{x}_i) = \min_{i,j \neq i} \sum\nolimits_{m=1}^{M} \sqrt[2]{\left(f_m^i - f_m^j\right)^2}, i,j \in \{1,2,\cdots,N\} \tag{10}$$

where $Totaldis(\mathbf{x}_i)$ indicates the sum of Manhattan Distance from symbol object $\mathbf{x}_i$ to all other symbol objects in the objective space. $Nieardis(\mathbf{x}_i)$ calculates the Minimum Manhattan Distance between the symbol object $\mathbf{x}_i$ and other symbol objects in the objective space. *N* represents the number of symbol objects, and *M* is the number of objective functions. The larger of *Totaldis* means the symbol object is farther away from other symbol objects. The larger *Neardis* signifies that the symbol object is farther from the nearest symbol object. Therefore, the larger *MED* indicates that the solution can extend to the boundary as much as possible and has better individual diversity. After all the symbol objects have evolved, call the dissolution rule to release all the elementary membranes and proceed to the next iteration.

Algorithm 3
The pseudocode of DMS-DES

Input: *P*: *k*th sub-multiset, *k*: the task index.
Output: *P'*: the multiset after ***DMS-DES***.
1. **While** differential evolution stage stop criterion is not met **do**
2. Call the ***Division rule*** to create four elementary membranes, namely *em1, em2, em3,* and *em4.*
3. *NS* = ***Obtain the non-dominated solution set*** (*P*) according to **Algorithm 4**.
4. Distribute *P* equally to the elementary membranes *em1*, *em2*, *em3* and send *NS* to *em4*.
5. Call the ***Communication rule*** to send the non-dominated solutions in *em4* to *em1*, *em2* and *em3*.
6. Each symbol object **x** in *em1* call the **o** = ***Rewrite rule 1*** (**x**).
7. Each symbol object **x** in *em2* call the **o** = ***Rewrite rule 2*** (**x**).
8. Each symbol object **x** in *em3* call the **o** = ***Rewrite rule 3*** (**x**).
9. **For each** offspring **o** in *em1, em2,* and *em3*
10. assign **o**' skill factor as *k,* and evaluate **o**.
11. **If o** $\prec$**x then x = o**.
12. **Else if o** and **x** are non-dominated **thena**
13. **If** *DOMo* < *DOMx* **then x = o**.
14. **Elseif** *DOMo* = *DOMx* and *MEDo* > *MEDx* **then x = o**.
15. **End if**
16. **End if**
17. **End for**
18. Call the ***Dissolution rule*** to release all the elementary membranes.
19. **End while**

Algorithm 4
The pseudocode of obtaining the non-dominated solution set

Input: P: the kth sub-multiset
Output: NS: the current non-dominated solution set of P.
1. $NS = \varnothing$.
2. **For each** symbol object **p** in P
3. $DOMp = 0$.
4. **End for**
5. **For each** symbol object **p** in P
6. **For each** symbol object **q** in P
7. **If** (**q** $\prec$ **p**) **then**
8. $DOMp = DOMp + 1$.
9. **End if**
10. **End for**
11. **If** ($DOMp == 0$) **then**
12. $NS = NS \cup$ **p**.
13. **End if**
14. **End for**

Algorithm 5
The pseudocode of PCM-ES

Input: **x**: the parent symbol object, d: the index of the mutation decision variable, k: the task index, p: the mutation precision.
Output: **x**: the symbol object after ***PCM-ES***.
1. $rand = random(4)$.
2. **If** $i == 1$ **then** call the **o** = ***Rewrite rule 4*** (**x**).
3. **Else if** $i == 2$ **then** call the **o** = ***Rewrite rule 5*** (**x**).
4. **Else if** $i == 3$ **then** call the **o** = ***Rewrite rule 6*** (**x**).
5. **Else if** $i == 4$ **then** call the **o** = ***Rewrite rule 7*** (**x**).
6. **End if**
7. **If o** is feasible **then** assign **o'** skill factor as k, and evaluate **o**.
8. **If o**$\prec$**x then x** = **o**.
9. **Else if o** and **x** are non-dominated **then**
10. **If** $DOMo < DOMx$ **then x** = **o**.
11. **Elseif** $DOMo = DOMx$ and $MEDo > MEDx$ **then x** = **o**.
12. **End if**
13. **End if**

In the second stage, PCM-ES [1] is used to perform a fine-grained search on each decision variable to facilitate exploration and exploitation after the coarse-grain search of DMS-DES. The pseudocode of PCM-ES is interpreted in **Algorithm 5.** Rewrite rules 4 and 5 are designed for exploitation, rewrite rules 6 and 7 are intended for exploration. The rewrite rules 4–7 detail is illustrated in Eq. (11).

Rewrite rule 4:

$$\mathbf{o} = \mathbf{x}$$

$$o_d = x_d + \Delta x$$

Rewrite rule 5:

$$\mathbf{o} = \mathbf{x}$$

$$o_d = x_d - \Delta x$$

Rewrite rule 6:

$$\mathbf{o} = \mathbf{x}$$

$$o_d = x_d \times \Delta x$$

Rewrite rule 7:

$$\mathbf{o} = \mathbf{x}$$

$$o_d = x_d \tilde{A} \cdot \Delta x$$

Where

$$\Delta x = \frac{1}{10^{Random(pcs)}} \times Random(9) \tag{11}$$

where x_d and o_d are the d dimensions of parent and offspring, respectively, the variable pcs is the manually set integer parameter to control the search precision, $Random$ (pcs) is utilized to generate random integers between [1, pcs],$Random(9)$ can generate random integers between [1,9]. Fig. 7 shows the frequency histogram of the mutated values obtained by independently repeating rewrite rules 4–7 20,000 times, in which the original value is set to 1. Fig. 7 (a)-(b), (e)-(f) show that rewrite rules 4–5 focus on exploitation. As shown in Fig. 7 (a)-(b), when pcs is set as 2, the value will be mutated in steps of 0.1 and 0.01 in the vicinity of 1. In Fig. 7 (e)-(f), when pcs takes to 5, the value does local searches near 1 in steps of 1E-1, 1E-2, 1E-3, 1E-4, and 1E-5. In contrast, Fig. 7(c)-(d), (g)-(h) point that rewrite rules 6–7 are adept at exploration. As shown in Fig. 7(c)-

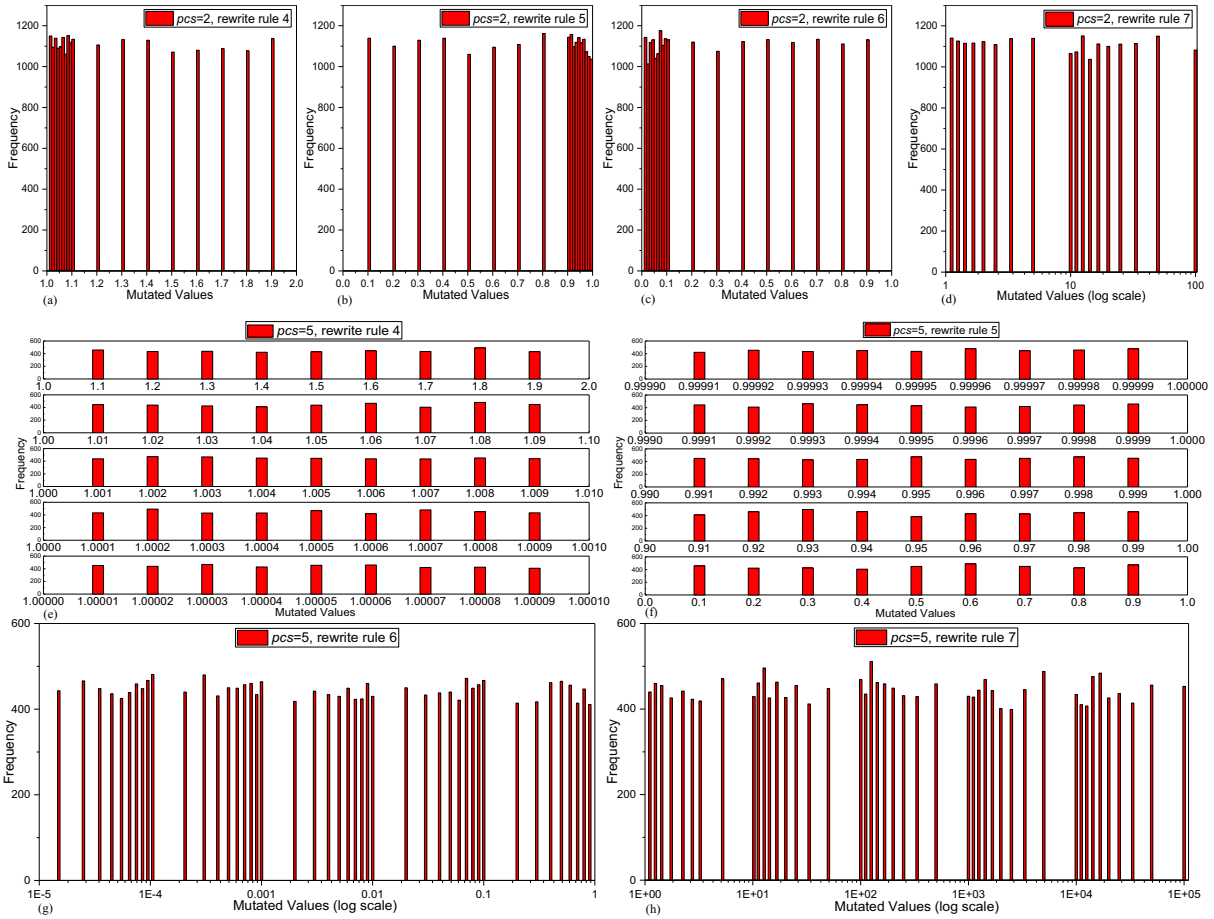

Fig. 7. The frequency histogram of the mutated values generated by rewrite rules 4–7.

(d), when *pcs* is set as 2, the value can magnify or shrink 1 by a factor of 10 or 100. In Fig. 7(g)-(h), when *pcs* takes to 5, the value can enlarge or shrink 1 by 10, 1E2, 1E3, 1E4, or even 1E5 times. Hence, the value can jump out of the local optimum.

To clearly show the advantages of the proposed two-stage algorithm in solving the non-separable problems, Fig. 8(a)-(d) demonstrate the performance of the proposed EMT-MOMIEA and famous NSGA-II on the sub-problems CIMS1 and PILS2 in the classical MOMTO test suite. CIMS1 and PILS2 are both non-separable problems. For a fair comparison, EMT-MOMIEA runs without using the information transfer mechanism. The population size is 100, and each individual is evolved in 5000 generations. In the proposed EMT-MOMIEA, the first 1250 generations use the DMS-DES strategy, and the PCM-ES strategy is utilized later. Fig. 8(a) shows that DMS-DES suffices to converge quickly and make the multiset rapidly reach a point on the PF in 1250 generations when solving the CIMS1 problem. Fig. 8(b) confirms that PCM-ES can perform a refined search after DMS-DES so that that multiset can extend to the entire PF when solving the CIMS1 problem. In Fig. 8(c), the DMS-DES can roughly trace to an approximate PF in solving the PILS2 problem. As shown in Fig. 8(d), the obtained PF can continuously advance to the true PF through PCM-ES's refined search for every variable.

3.3. The information transfer based on IMCV

Recently, many studies have made efforts to avoid the negative transfer [24–34]. However, these methods ignore the characteristics of decision variables [42]. The diversity-related variable emphasizes dispersion, but the convergence-variable is working on converging to the global optimum. When the communication decision variables have conflicting characteristics, the opposite evolution direction leads to negative transfer. Inspired by the phenomenon of the binding of information molecules and receptors during information transfer between cells, the IMCV concept is proposed to control the information transfer probability on a single variable to reduce negative transfer.

The following explains the occurrence of negative transfer in MOMTO from the perspective of decision vectors. Assume that the advanced CIMS1 and CIMS2 functions [43] are optimized simultaneously, as shown in Eqs. (12), (13). Parameter n represents the number of dimensions of decision variables set as 6 and sv is the shift vector set as [1, 1, −1, −1, −1]. Then, the PS of CIMS1 and CIMS2 are $x_1 \in [0,1], x_i = 1, i = 2:6$ and $x_1 \in [0,1], x_2 = 1, x_3 = 1, x_i = -1, i = 3:6$, respectively. From Fig. 9(-a)-(b), it can be seen that the first dimension decision variable of CIMS1 and CIMS2 are both diversity-related variables, and each individual in the population has different values in this dimension, so that the population can be distributed on the PF uniformly. The 2–6 dimension decision variables are all convergence-related variables. Each individual in the population has

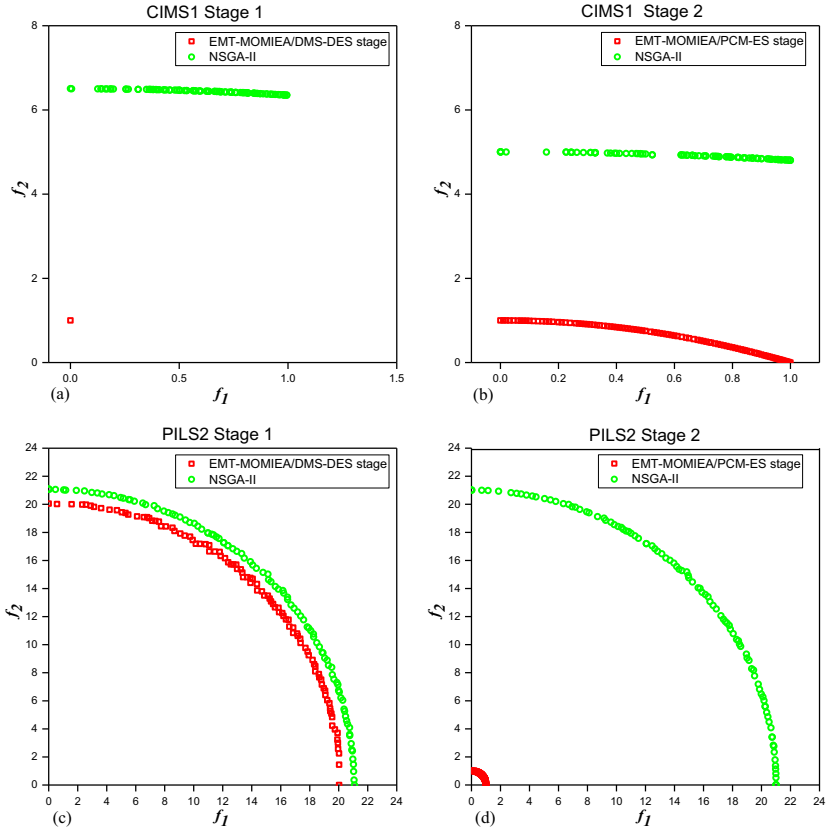

Fig. 8. The performance of two-stage evolution strategy on the CIMS1 and PILS2 problems.

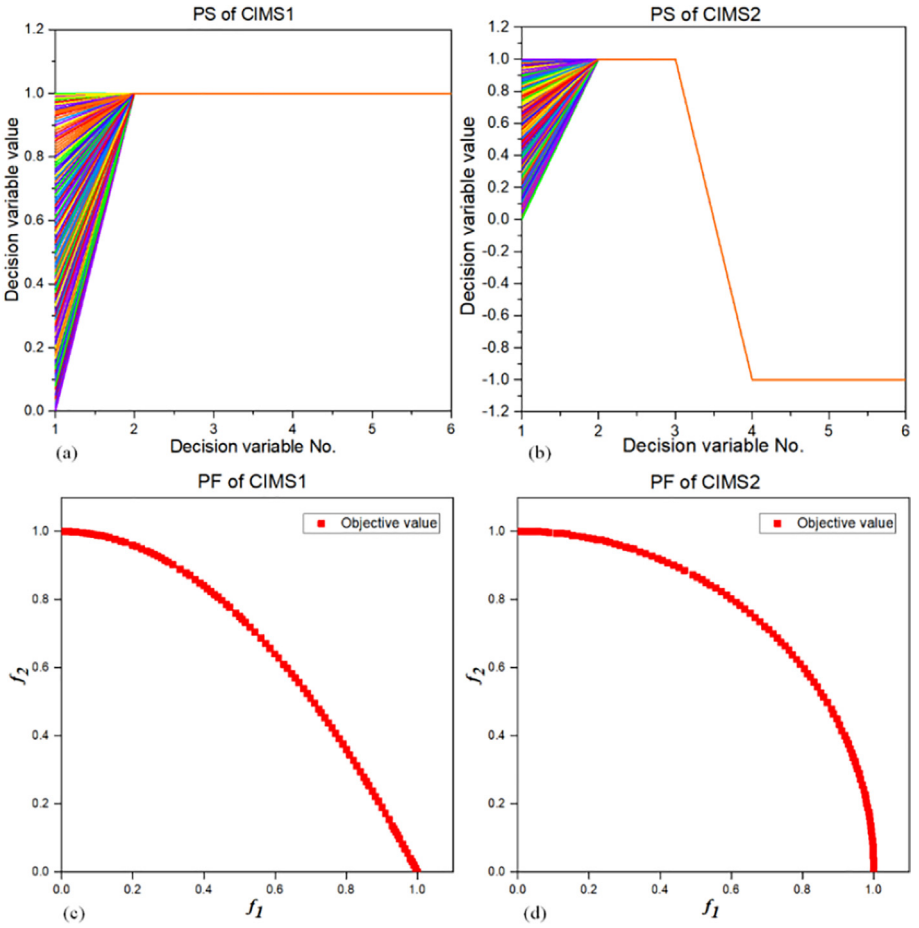

Fig. 9. The PS and PF of CIMS1 and CIMS2.

the same value in this dimension so that the population can be close to the true PF. Fig. 9(c)-(d) show the true PF of CIMS1-2. If the first dimension of the two individuals exchanges information, the diversity of the two can be improved. If the second and third dimensions of the two transfer information correspondingly, it will improve the convergence of the two tasks. However, if the diversity-related variable such as the first dimension of CIMS1 and the convergence-related variable such as the second dimension of CIMS2 exchange information, it will destroy CIMS1′s diversity and CIMS2′s convergence and result in a negative transfer. Moreover, suppose two task's convergence-related variables' global optimums are far apart. In that case, the traditional information transfer method may destroy the convergence of two tasks and result in a negative transfer.

$$\begin{aligned} &minf_1(\boldsymbol{x}) = x_1, \\ &minf_2(\boldsymbol{x}) = q(\boldsymbol{x})\left(1 - \left(\frac{x_1}{q(\boldsymbol{x})}\right)^2\right), \\ &q(\boldsymbol{x}) = 1 + \sum_{i=2}^{n-1}\left(100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2\right), \\ &x_1 \in [0,1], x_i \in [-5,5], i = 2,3,\ldots,n \end{aligned} \tag{12}$$

$$\begin{aligned} &minf_1(\boldsymbol{x}) = q(\boldsymbol{x})\cos\left(\tfrac{\pi x_1}{2}\right) \\ &minf_2(\boldsymbol{x}) = q(\boldsymbol{x})\sin\left(\tfrac{\pi x_1}{2}\right) \\ &q(\boldsymbol{x}) = 1 + \tfrac{9}{n-1}\sum_{i=2}^{n}|z_i| \\ &(z_2, z_3, \ldots z_n)^{\mathrm{T}} = \left((x_2, x_3, \ldots, x_n)^{\mathrm{T}} - sv\right) \\ &x_1 \in [0,1], x_i \in [-5,5], i = 2,3,\ldots,n \end{aligned} \tag{13}$$

In cytology, the information transmission between cells through body fluids requires the participation of information molecules. The concentration of information molecules is positively related to the sensitivity of information transmission between cells that is the higher the concentration, the more sensitive the information transmission between cells. Therefore, the sensitivity of information transmission between cells can be affected by adjusting the concentration of information molecules. Inspired by this phenomenon, the IMCV concept is proposed that only when the probability of the decision variable meets the threshold of IMCV in this dimension can information be transferred between tasks. The IMCV is designed as a threshold that is positively correlated to and adaptively adjusted by the success rate of information transfer. The proposed EMT-MOMIEA applies the crossover operator SBX as the information transfer method. The pseudocode of crossover with communication rule based on IMCV is given in **Algorithm 6**. First, a random decimal value *rand* is generated between (0, 1) in line 1. If *rand* is less than the value of IMCV on the d dimension $\overrightarrow{imcv}_d$, the transfer crossover strategy is invoked. The uppermost difference between the transfer crossover method and the conventional crossover method is the source of the other parent. Next, set the flag *Istransfer* as True, and add the statistical variable $\overrightarrow{ngt}_d$ by one. In the transfer crossover strategy, the communication rule $[\mathbf{t}]_{j\text{thsub}-\text{multiset}} \rightarrow [\mathbf{t}]_{k\text{thsub}-\text{multiset}}$ is called to transfer **t** from the source task j to the target task k in line 6. The crossover operator is performed at d the dimension between parents **x** and **t**, the result will assign to the offspring **o** in line 7. However, if *rand* is larger than $\overrightarrow{imcv}_d$, the transfer crossover strategy is abandoned. The flag *Istransfer* is set as False, and the $\overrightarrow{ngt}_d$ is added by one. The following process is the same as the transfer crossover method, but the difference lies in selecting the other parent **ns**. In the conventional transfer strategy, **ns** comes from the current non-dominated solutions of the target task.

Algorithm 6
The pseudocode of crossover with communication rule based on IMCV

Input: x: the parent symbol object, *NS*: the current non-dominated solutions set, d: the index of the decision variable, j: the index of the source task, $\overrightarrow{imcv}$: the information molecule concentration vector, $\overrightarrow{ngc}$: number of offspring generated by conventional crossover, $\overrightarrow{ngt}$: number of offspring generated by transfer crossover.
Output: o: offspring symbol object, *Istransfer*: the flag used to indicate the crossover method.
1. *rand* = *random* (0, 1)
2. **If** *rand* < $\overrightarrow{imcv}_d$ **then**
3. *Istransfer* = True.
4. $\overrightarrow{ngt}_d = \overrightarrow{ngt}_d + 1$.
5. **o** = **x**.
6. **t** = ***Communication rule*** (jth sub-multiset).
7. $\mathbf{o}^d$ = *crossover* ($\mathbf{x}^d$, $\mathbf{t}^d$).
8. **o**.skill factor = **x**.skill factor.
9. **Else**
10. *Istransfer* = False.
11. $\overrightarrow{ngc}_d = \overrightarrow{ngc}_d + 1$.
12. **o** = **x**.
13. **ns** = random (*NS*).
14. $\mathbf{o}^d$ = *crossover* ($\mathbf{x}^d$, $\mathbf{ns}^d$).
15. **o**.skill factor = **x**.skill factor.
16. **End if**

Algorithm 7
The pseudocode of update the influence factor of IMCV

Input: x: the parent symbol object, **o**: the offspring symbol object, d: the index of the decision variable, $\overrightarrow{nst}$: the number of offspring generated by transfer crossover surpass parents, $\overrightarrow{nsc}$: the number of offspring generated by conventional crossover surpass parents.
Output: the updated $\overrightarrow{nst}$ and $\overrightarrow{nsc}$.
1. **If o** $\prec$ **x then**
2. **If** (*Istransfer* == True) **then** $\overrightarrow{nst}_d = \overrightarrow{nst}_d + 1$.
3. **Else** $\overrightarrow{nsc}_d = \overrightarrow{nsc}_d + 1$.
4. **End if**
5. **Else if o** and **x** are non-dominated **then**

6. **If** *DOMo* < *DOMx* **then**
7. **If** (*Istransfer* == True) **then** $\overrightarrow{nst}_d = \overrightarrow{nst}_d+1$.
8. **Else** $\overrightarrow{nsc}_d = \overrightarrow{nsc}_d+1$.
9. **End if**
10. **Elseif** *DOMo* = *DOMx* and *MEDo* > *MEDx* **then**
11. **If** (*Istransfer* == True) **then** $\overrightarrow{nst}_d = \overrightarrow{nst}_d+1$.
12. **Else** $\overrightarrow{nsc}_d = \overrightarrow{nsc}_d+1$.
13. **End if**
14. **End if**
15. **End if**

As mentioned in **Algorithm 1**, after traversing all symbol objects in the *k*th task, IMCV needs to be updated. The IMCV is calculated by the statistical variables $\overrightarrow{nst}$ and $\overrightarrow{nsc}$ which indicate the number of offspring generated by transfer crossover and conventional crossover in a generation, respectively. The calculation method of $\overrightarrow{nst}$ and $\overrightarrow{nsc}$ are expressed as **Algorithm 7**. The rules that offsprings can replace their parents are the same as those in DMS-DES and PCM-ES, mentioned above. The identifier *Istransfer* indicates offspring's generation strategy, *d* represents the mutation position of the decision variable. If the superior offspring is generated by the transfer strategy, $\overrightarrow{nst}_d$ is added by 1. On the contrary, if it is generated by the conventional crossover, $\overrightarrow{nsc}_d$ is added by 1.

Algorithm 8
The pseudocode of update the IMCV

Input: $\overrightarrow{nst}$: the number of offspring generated by transfer crossover surpass parents, $\overrightarrow{nsc}$: the number of offspring generated by conventional crossover surpass parents. $\overrightarrow{ngc}$: the total number of offspring generated by conventional crossover, $\overrightarrow{ngt}$: the total number of offspring generated by transfer crossover, $\overrightarrow{imcv}$: the information molecule concentration vector.

Output: the updated $\overrightarrow{imcv}$.

1. **For** *d* = 1 to *DimMin*
2. *src* = $\overrightarrow{nsc}_d/\overrightarrow{ngc}_d$.
3. *srt* = $\overrightarrow{nst}_d/\overrightarrow{ngt}_d$.
4. **If** $\overrightarrow{ngt}_d$ == 0 **then**
5. $\overrightarrow{imcv}_d = \overrightarrow{imcv}_d + src$, $\overrightarrow{imcv}_d = \min(\overrightarrow{imcv}_d, 0.5)$.
6. **Else if** *srt* > *src* **then**
7. $\overrightarrow{imcv}_d = \overrightarrow{imcv}_d + srt$, $\overrightarrow{imcv}_d = \min(\overrightarrow{imcv}_d, 0.5)$.
8. **Else**
9. $\overrightarrow{imcv}_d = \overrightarrow{imcv}_d - srt$, $\overrightarrow{imcv}_d = \max(\overrightarrow{imcv}_d, 0)$.
10. **End if**
11. **End if**
12. $\overrightarrow{imcv}_d = \min(\max(\overrightarrow{imcv}_d, 0), 1)$.
13. **End for**

The update method of the IMCV vector is shown in **Algorithm 8**. The purpose is to dynamically adjust the intensity of information transfer on each decision variable to avoid the negative transfer. The adjustment of IMCV is based on the reward and punishment mechanism of *src* and *srt* which represent the proportion of offspring that can surpass the parent generated by the conventional crossover method and the information transfer method, respectively. Considering a single decision variable, if *srt* is greater than *src*, the information transfer strategy is superior at this generation, so $\overrightarrow{imcv}_d$ is better to increase in line 7. Conversely, if *srt* is less than *src*, the conventional evolution way is more suitable for the current population, so $\overrightarrow{imcv}_d$ is better to reduce in line 9. However, even if the success rate of the transfer crossover is meager, there should reserve some opportunities for the information transfer. If there are no individuals generated by the information transfer in a generation, increase $\overrightarrow{imcv}_d$ to the *src* in line 5. If $\overrightarrow{imcv}_d$ is too large that information transfer occupies all evolutionary resources, $\overrightarrow{imcv}_d$ is restricted to 1 in line 12.

Table 1
The parameters for the experiments.

Parameter	NSGA-II	MOMFEA	MOMFEA-II	MFEA-GHS	MFEA-SADE	EMT-MOMIEA
Random mating probability *rmp*	–	0.3	–	0.3	0.3	–
Initial value of IMCV of each dimension						0.5
Population size for all the tasks in the classical and complex MOMTO test suites	200	200	200	200	200	200
Population size for all the tasks in the MOMaTO test suite	–	500	–	–	–	500
Maximum evaluation times for all the tasks in the classical and complex MOMTO test suite	200,000	200,000	200,000	200,000	200,000	200,000
Maximum evaluation times for all the tasks in the MOMaTO test suite	–	500,000	–	–	–	500,000
Crossover probability p_c	0.9	0.9	0.9	0.9	–	0.9
Distribution index of crossover η_c	20	20	20	20	–	20
Mutation probability p_m	$1/D$	$1/D$	$1/D$	$1/D$	–	–
Distribution index of mutation η_m	20	20	20	20	–	–
Scaling factor F_1 in DE	–	–	–	–	0.6	0.6
Scaling factor F_2 in DE	–	–	–	–	0.5	0.5
Experience period e in MFEA-SADE	–	–	–	–	60	–
Allocation proportion of evolution times in DMS-DES in classical MOMTO test suite p_{des}						1/4
Allocation proportion of evolution times in DMS-DES in complex MOMTO and MOMaTO test suites p_{des}						1/8
Mutation precision *pcs* in PCM-ES						3

4. Experiment

In our experimental study, the performance of the proposed EMT-MOMIEA is compared with four advanced multi-objective EMT algorithms MOMFEA-II [25], MFEA-SADE [29], MOMFEA [32], MFEA-GHS [33], and the famous single-task MOEA NSGA-II [2]. The performance of EMT-MOMIEA is estimated on the classical and complex MOMTO test suites [43,44], and the MOMaTO benchmark test suite [44].

4.1. Test suites introduction

In EMT theory, the decisive factors affecting information transfer efficiency are the intersection degree of the optimal solutions and the similarity of fitness landscapes [43]. If the optimal solutions of the source and the transfer tasks are approximate, the evolution of the population can bring dividends to all these tasks. If the fitness landscapes of the tasks are similar, the information transfer can promote the convergence. Hence, the classical MOMTO test suite [43] is designed into the complete intersection (CI), partial intersection (PI), and no intersection (NI), three categories based on the intersection degree of the global optimums. Based on the similarity degree of the fitness landscape, the classical MOMTO test suite can be classified as high similarity (HS), medium similarity (MS), and low similarity (LS). Combining the above two classification standards, the classical MOMTO test suite includes nine continuous multi-objective sub-problems from CIHS to NILS. To study the performance of the EMT algorithms on the problems with complex PS [45], the proposed algorithm is also compared with other algorithms on the complex MOMTO test suite named CPLX [44]. Moreover, the MOMaTO benchmark test suite [44], which optimizes 50 different tasks simultaneously, is proposed to estimate algorithms' performance in processing many tasks.

4.2. Compared algorithms

The proposed EMT-MOMIEA is compared with four advanced multi-objective EMT algorithms MOMFEA-II [25], MFEA-SADE [29], MOMFEA [32], MFEA-GHS [33], and the famous single-task MOEA NSGA-II [2]. MOMFEA [32] is the first multi-objective EMT algorithm with a landmark significance. It introduces the well-known MOEA NSGA-II as the evolutionary operator within the classic MFEA framework. It can reflect the pros and cons of EMT by comparing the performance of NSGA-II and MOMFEA. MOMFEA-II [25] introduces an online learning method into MOMFEA to adaptively adjust the information transfer intensity. The MFEA-SADE [29] introduces the subspace alignment and adaptive differential evolutionary to MOMFEA. The MFEA-GHS [33] presents the genetic transform and hyper-rectangle search into MOMFEA. All the algorithms are implemented utilizing the Jmetal 4.5.2, a famous object-oriented Java-based framework [47] to avoid the differences brought by the implementation platform of algorithms [48]. The platform on which the algorithm runs is a PC with Intel Core i5-9400F CPU 2.90 GHz, and 16.00 GB of RAM.

4.3. Performance indicator

The inverted generational distance (IGD) [46], which can simultaneously evaluate the convergence and diversity of the obtained population, is used as the performance indicator in this paper. IGD is the average value of the minimum distance from the sampling points on the true PF to the obtained PF. The smaller the IGD value is, the better the convergence and diversity of the population are. The mathematical form of the IGD is shown in Eq. (14). N_{PF} represents the number of the sampling points in the true PF, and $Dist_i$ indicates the Euclidean Distance from the ith point on the true PF to the closest point in the obtained population.

$$IGD = \frac{\left(\sum_{i=1}^{N_{PF}} Dist_i^2\right)^{1/2}}{N_{PF}} \tag{14}$$

4.4. Parameter settings

For a fair comparison, the population size for a single-task MOEA NSGA-II is set as 100 in the classical and complex MOMTO test suites. The population size of multi-objective EMT algorithms MOMFEA [32], MOMFEA-II [25], MFEA-GHS [33], MFEA-SADE [29], and EMT-MOMIEA are all set as 200. As for the MOMaTO benchmark test suite, which optimizes 50 tasks simultaneously, the population size is set to 500. The parameter settings of the comparison algorithms are consistent with the original paper. For the proposed EMT-MOMIEA, the values of scaling factors F_1 and F_2 are the same as MFEA-SADE. The allocation proportion of evolution times in DMS-DES in classical MOMTO, complex MOMTO, and MOMaTO test suites are set as 1/4, 1/8, and 1/8, respectively, the mutation precision *pcs* in PCM-ES is set as 3. The details of all the parameter settings are summarized in Table 1.

It is worth noting that all comparison algorithms run independently 20 times on all test suites. In all the experimental results tables such as Tables 2–5, the mean of IGD values is exhibited. The best result among the algorithms on each sub-problem is highlighted in gray. The Wilcoxon rank-sum test at the 95% confidence level is utilized for the experimental results, the remarkable better, remarkable worse, and not equivalent employing "+", "−" and "=" to express respectively.

4.5. Performance on classical MOMTO test suite

The comparative experiments on the classical MOMTO test suite are revealed in Table 2. On most test problems, the multi-objective EMT algorithms perform better than classic MOEA NSGA-II. This proves that the information transfer mechanism in EMT algorithms is indeed effective. In comparison with the advanced multi-objective EMT algorithms MOMFEA, MOMFEA-II, MFEA-GHS, and MFEA-SADE, the proposed EMT-MOMIEA obtains 15, 14, 16, and 17 superior results out of 18 sub-problems, respectively.

The proposed EMT-MOMIEA can get the best results on the high similarity problems as CIHS, PIHS, and NIHS. This is because the proposed IMCV-based information transfer strategy can adjust the information transfer intensity adaptively. When solving the high similarity problems, the success rate of information transfer will increase, which will lead to an increase in IMCV and promote the information transfer between the two tasks. It can accelerate the convergence of the two tasks. For problems with no intersection, such as NIHS, NIMS1, and NILS2, among these problems, the global optimum of simultaneously optimized tasks do not intersect and are more difficult to solve simultaneously. The proposed EMT-MOMIEA also accomplishes the best results on these sub-problems. This is because the IMCV theory changes the information transfer intensity from the decision variable level so that EMT-MOMIEA can adapt to the simultaneous optimization of two utterly different tasks. When the two tasks are not similar, or the global optimum do not overlap, the concentration of information molecules will decrease, which will reduce the sensitivity of information transmission between the membranes. Incidentally, the IMCV curves on the classical MOMTO test problems are shown in Fig. 10.

4.6. Performance on complex MOMTO test suite

The comparative experiments on the complex MOMTO test problems are revealed in Table 3. The advanced EMT algorithms can achieve better results than NSGA-II on most sub-problems, mainly because the EMT algorithm can bring more bias to the target task's decision space through information transfer. In comparison with the state-of-the-art multi-objective EMT algorithms MOMFEA, MOMFEA-II, MFEA-GHS, and MFEA-SADE, the proposed EMT-MOMIEA obtains 16, 15, 18, and 15 superior results out of 20 sub-problems, respectively.

Solving the MOMTO problems with complex PS requires an algorithm having more robust search capability. The advantages of the proposed EMT-MOMIEA mainly come from the proposed two-stage evolution strategy algorithm based on the membrane system. In the first stage, the DMS-DES is applied as AIM. Firstly, the DMS-DES can endow the population with more evolutionary biases to improve the search performance and balance exploration and exploitation by embedding three different DE rewrite operators. Secondly, the DMS-DES gives full play to the advantages of the dynamic membrane structure, which employs unique division rule and dissolution rule to adjust membrane structure in the evolutionary process dynamically. The division rule can divide the membrane into four sub-membranes. The dissolution rule can dissolve the membrane

Table 2
Mean values of the IGD obtained by six algorithms on the classical MOMTO test problems.

Problem	Task	NSGA-II	MOMFEA	MOMFEA-II	MFEA-GHS	MFEA-SADE	EMT-MOMIEA
CIHS	T1	2.32E-03(+)	4.76E-04(+)	4.54E-04(+)	2.37E-03(+)	3.64E-03(+)	2.46E-04
	T2	4.56E-03(+)	2.38E-03(+)	2.37E-03(+)	4.60E-03(+)	4.92E-04(+)	2.87E-04
CIMS	T1	1.06E-01(+)	3.93E-02(+)	1.84E-01(+)	1.82E-01(+)	3.52E-04(+)	1.98E-04
	T2	3.31E-02(+)	5.72E-03(+)	3.71E-04(+)	6.20E-02(+)	8.63E-04(+)	3.22E-04
CILS	T1	2.78E-01(+)	4.21E-04(+)	3.95E-04(+)	3.03E-03(+)	3.87E-03(+)	2.38E-04
	T2	2.03E-04(+)	3.88E-04(+)	4.13E-04(+)	3.16E-03(+)	4.98E-04(+)	1.61E-04
PIHS	T1	1.18E-03(+)	5.37E-04(+)	4.72E-04(+)	2.88E-03(+)	4.68E-04(+)	2.34E-04
	T2	4.94E-02(+)	1.85E-02(+)	1.53E-02(+)	3.64E-01(+)	2.32E-01(+)	8.18E-03
PIMS	T1	5.44E-03(-)	3.64E-03(-)	2.24E-03(-)	9.30E-02(+)	2.28E-02(+)	1.69E-02
	T2	1.55E+01(+)	1.03E+01(+)	1.21E+01(+)	2.80E-03(-)	1.62E+01(+)	5.96E+00
PILS	T1	3.36E-04(+)	5.23E-04(+)	4.36E-04(+)	2.69E-03(+)	1.20E-02(+)	3.02E-04
	T2	6.35E-01(+)	7.78E-03(-)	5.42E-03(-)	6.35E-01(+)	1.73E+00(+)	1.66E-01
NIHS	T1	6.97E+00(+)	1.53E+00(+)	1.53E+00(+)	1.21E+01(+)	4.52E+01(+)	1.44E+00
	T2	7.86E-04(+)	5.08E-04(+)	4.77E-04(+)	2.77E-03(+)	4.59E-04(+)	2.36E-04
NIMS	T1	5.27E-01(+)	3.28E-01(+)	3.69E-01(+)	2.51E+00(+)	1.52E+01(+)	1.18E-01
	T2	1.54E-01(+)	3.44E-02(+)	1.52E-02(-)	3.68E-03(-)	1.61E-03(-)	3.22E-02
NILS	T1	8.58E-04(-)	1.05E-03(-)	1.12E-03(-)	4.50E-02(+)	1.68E-02(+)	1.24E-03
	T2	6.42E-01(+)	6.42E-01(+)	6.42E-01(+)	1.72E-02(+)	1.94E+01(+)	6.66E-04
		+16/-2/=0	+15/-3/=0	+14/-4/=0	+16/-2/=0	+17/-1/=0	

Table 3
Mean values of the IGD obtained by six algorithms on the complex MOMTO test problems.

Problem	Task	NSGA-II	MOMFEA	MOMFEA-II	MFEA-GHS	MFEA-SADE	EMT-MOMIEA
CPLX1	T1	4.13E-04(-)	4.49E-04(-)	4.19E-04(-)	4.87E-03(+)	2.73E-04(-)	5.51E-04
	T2	5.57E-03(+)	5.01E-03(+)	4.42E-03(+)	5.47E-03(+)	1.18E-03(+)	4.87E-04
CPLX2	T1	4.08E-04(-)	3.72E-04(-)	3.96E-04(-)	4.61E-04(+)	2.87E-04(-)	4.40E-04
	T2	1.06E-02(+)	4.37E-03(-)	4.02E-03(-)	3.98E-03(-)	7.92E-04(-)	9.04E-03
CPLX3	T1	5.47E-03(+)	5.08E-03(+)	4.15E-03(+)	3.80E-03(+)	1.67E-03(+)	5.56E-04
	T2	2.59E-03(+)	3.49E-03(+)	2.58E-03(+)	2.74E-03(+)	1.38E-03(+)	9.15E-04
CPLX4	T1	5.43E-03(+)	4.36E-03(+)	4.03E-03(+)	2.80E-02(+)	1.78E-03(+)	5.33E-04
	T2	8.25E-03(+)	4.39E-03(+)	4.26E-03(+)	4.41E-03(+)	2.09E-03(+)	7.33E-04
CPLX5	T1	3.11E-03(+)	2.96E-03(+)	2.19E-03(+)	6.80E-03(+)	2.17E-03(+)	5.65E-04
	T2	6.09E-03(+)	5.90E-03(+)	6.54E-03(+)	5.65E-03(+)	3.54E-03(+)	2.35E-03
CPLX6	T1	2.65E-03(+)	2.92E-03(+)	2.33E-03(+)	2.60E-03(+)	2.18E-03(+)	5.42E-04
	T2	7.71E-03(+)	5.27E-03(+)	5.19E-03(+)	4.87E-03(+)	3.27E-03(+)	7.76E-04
CPLX7	T1	3.11E-03(+)	2.44E-03(+)	2.69E-03(+)	2.70E-03(+)	1.21E-03(+)	9.43E-04
	T2	2.02E-03(+)	2.38E-03(+)	2.54E-03(+)	2.46E-03(+)	1.38E-03(+)	5.22E-04
CPLX8	T1	1.85E-03(+)	3.31E-03(+)	2.28E-03(+)	1.60E-03(+)	9.16E-04(+)	4.61E-04
	T2	1.33E-02(-)	1.46E-02(+)	1.17E-02(-)	8.38E-03(-)	2.19E-03(-)	1.42E-02
CPLX9	T1	6.43E-03(+)	6.19E-03(+)	6.43E-03(+)	1.74E-01(+)	4.23E-03(+)	2.38E-03
	T2	7.45E-03(+)	5.03E-03(+)	5.91E-03(+)	5.36E-03(+)	3.64E-03(+)	6.12E-04
CPLX10	T1	1.19E-02(+)	9.95E-03(-)	9.92E-03(-)	2.70E-01(+)	7.23E-03(-)	1.04E-02
	T2	9.06E-03(+)	8.19E-03(+)	7.69E-03(+)	9.03E-03(+)	9.45E-03(+)	7.28E-03
		+17/-3/=0	+16/-4/=0	+15/-5/=0	+18/-2/=0	+15/-5/=0	

Table 4
Mean values of the IGD obtained by MOMFEA and EMT-MOMIEA on the MOMaTO test problems.

	MOMFEA						EMT-MOMIEA					
Task	MATP1	MATP2	MATP3	MATP4	MATP5	MATP6	MATP1	MATP2	MATP3	MATP4	MATP5	MATP6
T1	3.64E+01	5.52E-01	8.26E-02	1.02E+00	6.45E-03	3.51E-02	2.19E-01	8.73E-02	1.96E-02	2.67E-02	2.33E-03	9.92E-03
T2	2.91E+01	5.54E-01	1.68E-01	6.70E-01	6.24E-03	3.73E-02	1.59E-01	7.02E-02	2.56E-02	2.09E-02	2.19E-03	1.13E-02
T3	1.89E+01	4.87E-01	1.70E-01	7.87E-01	6.29E-03	3.72E-02	3.77E-01	7.11E-02	4.81E-02	2.38E-02	2.17E-03	1.11E-02
T4	2.10E+01	4.65E-01	1.76E-01	7.79E-01	6.00E-03	3.61E-02	8.50E-02	7.63E-02	1.76E-02	4.64E-02	2.05E-03	1.14E-02
T5	1.73E+01	5.23E-01	2.04E-01	7.90E-01	6.54E-03	3.77E-02	1.91E-01	8.34E-02	1.98E-02	2.41E-02	1.98E-03	1.18E-02
T6	2.29E+01	6.20E-01	2.28E-01	9.42E-01	6.37E-03	3.62E-02	3.30E-01	1.01E-01	4.33E-03	2.23E-02	2.00E-03	9.44E-03
T7	2.11E+01	5.35E-01	1.97E-01	8.09E-01	8.19E-03	3.66E-02	2.79E-01	7.80E-02	3.91E-02	3.58E-02	1.97E-03	1.21E-02
T8	2.23E+01	5.09E-01	1.75E-01	8.03E-01	5.56E-03	3.69E-02	2.34E-01	8.50E-02	8.45E-02	4.39E-02	2.22E-03	1.19E-02
T9	2.62E+01	5.90E-01	2.40E-01	7.04E-01	7.06E-03	3.62E-02	1.63E-01	7.01E-02	1.33E-01	1.43E-02	2.22E-03	1.01E-02
T10	1.63E+01	6.87E-01	1.77E-01	6.47E-01	6.56E-03	3.56E-02	2.23E-01	9.01E-02	6.48E-02	3.63E-02	2.16E-03	1.17E-02
T11	1.63E+01	6.06E-01	1.90E-01	9.97E-01	7.61E-03	3.45E-02	2.38E-01	8.86E-02	4.36E-03	6.32E-02	2.01E-03	1.30E-02
T12	1.84E+01	6.69E-01	2.46E-01	8.14E-01	6.94E-03	3.63E-02	3.07E-01	6.71E-02	1.62E-01	4.70E-02	1.95E-03	1.31E-02
T13	1.66E+01	6.72E-01	2.31E-01	7.94E-01	8.44E-03	3.66E-02	3.70E-01	8.88E-02	1.96E-01	2.24E-02	2.06E-03	1.13E-02
T14	1.62E+01	6.47E-01	9.76E-02	9.50E-01	7.26E-03	3.55E-02	9.39E-02	8.07E-02	1.44E-01	3.91E-02	1.99E-03	1.19E-02
T15	2.06E+01	4.95E-01	1.82E-01	7.75E-01	7.28E-03	3.76E-02	4.45E-01	6.01E-02	1.26E-02	3.65E-02	2.02E-03	1.17E-02
T16	1.94E+01	5.15E-01	9.43E-02	8.18E-01	6.47E-03	3.50E-02	3.10E-01	1.03E-01	5.32E-03	4.68E-02	2.15E-03	1.20E-02
T17	3.07E+01	5.97E-01	1.91E-01	7.52E-01	5.01E-03	3.75E-02	1.17E-01	7.46E-02	4.34E-02	3.48E-02	1.95E-03	1.14E-02
T18	1.68E+01	5.65E-01	2.17E-01	7.02E-01	6.66E-03	3.61E-02	4.93E-01	7.03E-02	1.65E-02	4.29E-02	1.96E-03	1.39E-02
T19	1.64E+01	6.64E-01	2.02E-01	7.87E-01	5.24E-03	3.59E-02	3.79E-01	8.97E-02	3.34E-02	2.46E-02	2.40E-03	1.17E-02
T20	1.97E+01	6.52E-01	1.97E-01	7.57E-01	7.46E-03	3.62E-02	2.40E-01	7.49E-02	1.93E-01	4.03E-02	2.04E-03	1.41E-02
T21	1.97E+01	5.83E-01	1.45E-01	6.15E-01	6.62E-03	3.58E-02	1.52E-01	7.06E-02	1.01E-01	1.40E-02	2.04E-03	1.24E-02
T22	1.57E+01	4.39E-01	1.75E-01	6.37E-01	6.69E-03	3.53E-02	4.07E-01	5.75E-02	2.27E-03	2.67E-02	1.94E-03	1.00E-02
T23	2.02E+01	6.56E-01	2.18E-01	8.43E-01	7.18E-03	3.67E-02	1.72E-01	1.03E-01	8.31E-03	4.63E-02	1.95E-03	8.56E-03
T24	1.78E+01	5.36E-01	1.44E-01	5.79E-01	5.58E-03	3.66E-02	3.08E-01	8.16E-02	9.29E-03	4.43E-02	2.07E-03	9.99E-03
T25	3.14E+01	4.19E-01	2.35E-01	7.73E-01	5.86E-03	3.50E-02	2.89E-01	7.82E-02	7.22E-02	4.80E-02	1.96E-03	1.03E-02
T26	2.49E+01	7.55E-01	1.35E-01	6.98E-01	5.37E-03	3.69E-02	1.59E-01	7.93E-02	3.94E-02	1.99E-02	2.07E-03	1.27E-02
T27	1.66E+01	6.49E-01	2.20E-01	7.45E-01	6.40E-03	3.55E-02	3.03E-01	8.07E-02	1.30E-01	2.43E-02	2.05E-03	1.16E-02
T28	1.41E+01	4.32E-01	2.26E-01	8.80E-01	5.34E-03	3.76E-02	1.92E-01	7.63E-02	1.76E-01	1.84E-02	1.88E-03	1.44E-02
T29	2.07E+01	5.27E-01	2.17E-01	9.01E-01	8.04E-03	3.61E-02	2.55E-01	6.30E-02	3.22E-02	5.91E-02	2.15E-03	9.59E-03
T30	2.69E+01	4.92E-01	1.39E-01	6.97E-01	5.97E-03	3.63E-02	2.15E-01	7.33E-02	7.08E-03	3.45E-02	2.20E-03	1.06E-02
T31	1.42E+01	6.12E-01	2.38E-01	6.23E-01	5.93E-03	3.64E-02	1.92E-01	8.54E-02	7.83E-02	3.29E-02	1.94E-03	1.11E-02
T32	1.98E+01	7.08E-01	2.19E-01	6.58E-01	7.00E-03	3.57E-02	3.76E-01	7.68E-02	1.48E-02	2.79E-02	2.03E-03	1.32E-02
T33	2.02E+01	5.64E-01	2.40E-01	8.48E-01	6.23E-03	3.63E-02	3.53E-01	7.92E-02	8.53E-03	3.15E-02	2.13E-03	1.28E-02
T34	2.02E+01	4.42E-01	2.15E-01	6.76E-01	6.68E-03	3.48E-02	4.64E-01	6.06E-02	3.64E-02	3.26E-02	2.18E-03	1.05E-02
T35	2.62E+01	5.02E-01	1.67E-01	7.78E-01	6.80E-03	3.52E-02	4.46E-01	7.69E-02	4.20E-03	2.77E-02	1.81E-03	1.31E-02
T36	1.54E+01	6.01E-01	9.87E-02	6.24E-01	5.66E-03	3.60E-02	2.82E-01	9.52E-02	2.86E-02	4.49E-02	2.03E-03	1.30E-02
T37	1.62E+01	4.86E-01	1.36E-01	5.99E-01	7.47E-03	3.66E-02	1.21E-01	9.22E-02	8.58E-03	1.27E-02	2.22E-03	1.20E-02
T38	1.98E+01	7.48E-01	1.20E-01	7.54E-01	7.02E-03	3.65E-02	2.62E-01	7.97E-02	1.22E-02	2.96E-02	1.95E-03	1.15E-02
T39	1.56E+01	5.45E-01	1.87E+00	6.88E-01	6.28E-03	3.59E-02	4.40E-01	6.28E-02	2.01E-03	3.03E-02	2.06E-03	1.09E-02
T40	2.76E+01	5.23E-01	2.21E-01	6.70E-01	6.45E-03	3.54E-02	4.43E-01	7.01E-02	1.62E-01	3.27E-02	1.90E-03	1.11E-02
T41	2.54E+01	5.85E-01	2.20E-01	8.67E-01	6.35E-03	3.59E-02	2.99E-01	6.50E-02	4.14E-03	3.49E-02	2.04E-03	1.18E-02
T42	1.98E+01	6.88E-01	9.59E-02	9.44E-01	7.48E-03	3.60E-02	1.79E-01	9.80E-02	5.67E-03	3.37E-02	2.25E-03	1.16E-02
T43	1.90E+01	5.49E-01	1.95E-01	7.77E-01	6.28E-03	3.51E-02	4.75E-01	8.95E-02	2.37E-02	6.42E-02	2.03E-03	1.01E-02
T44	1.46E+01	5.44E-01	2.03E-01	7.64E-01	6.87E-03	3.69E-02	3.72E-01	6.80E-02	1.71E-02	2.34E-02	1.94E-03	1.05E-02
T45	2.29E+01	4.25E-01	1.76E-01	7.88E-01	5.46E-03	3.59E-02	7.16E-01	6.79E-02	1.34E-01	2.06E-02	1.93E-03	9.38E-03
T46	2.40E+01	6.79E-01	1.30E-01	7.76E-01	6.28E-03	3.68E-02	2.57E-01	5.16E-02	3.49E-02	6.96E-02	1.98E-03	1.47E-02
T47	2.19E+01	6.66E-01	2.12E-01	6.73E-01	7.44E-03	3.56E-02	2.55E-01	8.31E-02	1.28E-01	2.25E-02	2.13E-03	1.29E-02
T48	2.85E+01	6.19E-01	2.39E-01	7.49E-01	6.54E-03	3.70E-02	1.78E-01	1.08E-01	4.54E-02	2.68E-02	2.08E-03	1.02E-02
T49	2.25E+01	5.11E-01	2.35E-01	7.74E-01	5.61E-03	3.56E-02	3.93E-01	7.37E-02	4.43E-03	1.86E-02	2.01E-03	1.11E-02
T50	1.99E+01	5.18E-01	2.29E-01	7.12E-01	5.35E-03	3.49E-02	8.46E-01	7.94E-02	2.07E-02	2.78E-02	2.00E-03	1.03E-02

structure to disperse all the objects in the membrane to the outer membrane to redistribute the object resources. Thirdly, the DMS-DES makes full use of the communication rule in membrane calculation. By continuously sending the non-dominated solutions in the current population to other sub-membranes, it can effectively promote the convergence of the population. In the second stage, the rewrite rules in the PCM-ES have asynchronous lengths and search modes, which can mutate a single variable with different precisions. The IMCV curves on the complex MOMTO test problems are shown in Fig. 11.

4.7. Performance on MOMaTO test suite

The comparative experiments on the MOMaTO benchmark test suite are revealed in Table 4. There are six sub-problems in the MOMaTO benchmark test suite, namely MATP1-MATP6. In each sub-problem, there are 50 two-objective tasks optimized simultaneously. As shown in Table 4, the EMT-MOMIEA shows apparent advantages in all the tasks in the MOMaTO benchmark test suite. EMT-MOMIEA does not use a specific source task selection method but randomly selects a source task from 50 tasks. The advantages of the proposed EMT-MOMIEA can be summarized into three main points. Firstly, the proposed IMCV-based information transfer strategy can effectively reduce the negative transfer. Secondly, the proposed two-stage evolution strategy based on membrane structure can apply a variety of evolutionary operators to introduce multitudinous search modes, expand the search range and improve the search efficiency. Finally, all the tasks are optimized using the same total population in the MOMaTO benchmark test suite, so the search resources allocated to each task become sparse. The PCM-ES has been proven to have strong search ability even with tiny population size, approach the true PF, and extend to the whole PF [1]. The proposed EMT-MOMIEA applies the PCM-ES as AIM to a detailed search on a single decision variable. Therefore, EMT-MOMIEA can achieve impressive performance even with tiny population size.

4.8. Discussion on the proposed strategies

This section discusses the contribution of each proposed strategy to the performance of the proposed EMT-MOMIEA. Table 5 shows the results of the basic method with different proposed operators on the classical MOMTO test suite. The EMT-DMS only uses the DMS-DE as AIM, EMT-PCM only applies PCM-ES as AIM, EMT-TS employs the integrated membrane system-based two-stage evolution strategy as AIM. The above three algorithms do not use the IMCV-based information transfer strategy, and the *rmp* of the above three algorithms are all set as a fixed value of 0.3. The EMT-MOMIEA is the complete proposed algorithm. Comparing the performance of EMT-DMS and EMT-PCM, EMT-DMS only dominates on CIMS2. This is because the decision variables in CIMS2 problem have parameter interactions, while the DE operator in EMT-DMS has parameter invariance and can solve non-separable problems. However, applying the EMT-DMS as AIM solely also has the following disadvantages. First, changing multiple decision variables simultaneously can lead the population to converge quickly, but it will also lose some diversity. Second, EMT-DMS changes all variables concurrently. It is difficult to determine

Table 5
Mean values of the IGD obtained by four algorithms on the classical MOMTO test problems.

Problem	Task	EMT-DMS	EMT-PCM	EMT-TS	EMT-MOMIEA
CIHS	T1	1.13E+02(+)	9.50E-04(+)	2.51E-04(=)	2.46E-04
	T2	2.60E+00(+)	1.62E-03(+)	3.14E-04(+)	2.87E-04
CIMS	T1	6.91E-02(+)	6.68E-02(+)	2.27E-04(+)	1.98E-04
	T2	2.07E-04(-)	2.16E-02(+)	3.44E-04(+)	3.22E-04
CILS	T1	1.56E+00(+)	1.99E-03(+)	4.88E-04(+)	2.38E-04
	T2	1.37E-02(+)	2.13E-04(+)	1.63E-04(=)	1.61E-04
PIHS	T1	2.37E+02(+)	1.06E-03(+)	2.42E-04(+)	2.34E-04
	T2	2.83E+02(+)	8.26E-02(+)	9.04E-03(+)	8.18E-03
PIMS	T1	3.04E-01(+)	1.94E-02(+)	2.20E-02(+)	1.69E-02
	T2	2.48E+01(+)	6.81E+00(+)	1.13E+01(+)	5.96E+00
PILS	T1	3.94E-02(+)	1.16E-03(+)	3.84E-04(+)	3.02E-04
	T2	6.13E-01(+)	4.67E-01(+)	1.70E-01(+)	1.66E-01
NIHS	T1	8.74E+05(+)	1.51E+00(+)	1.45E+00(=)	1.44E+00
	T2	1.42E+02(+)	1.01E-03(+)	2.41E-04(+)	2.36E-04
NIMS	T1	3.28E+00(+)	1.28E-01(+)	1.24E-01(+)	1.18E-01
	T2	1.49E+00(+)	1.36E-03(-)	5.34E-02(+)	3.22E-02
NILS	T1	1.22E-02(+)	7.19E-03(+)	1.25E-03(=)	1.24E-03
	T2	6.43E-01(+)	4.16E-03(+)	7.24E-04(+)	6.66E-04
		+17/-1/=0	+17/-1/=0	+16/-0/=4	

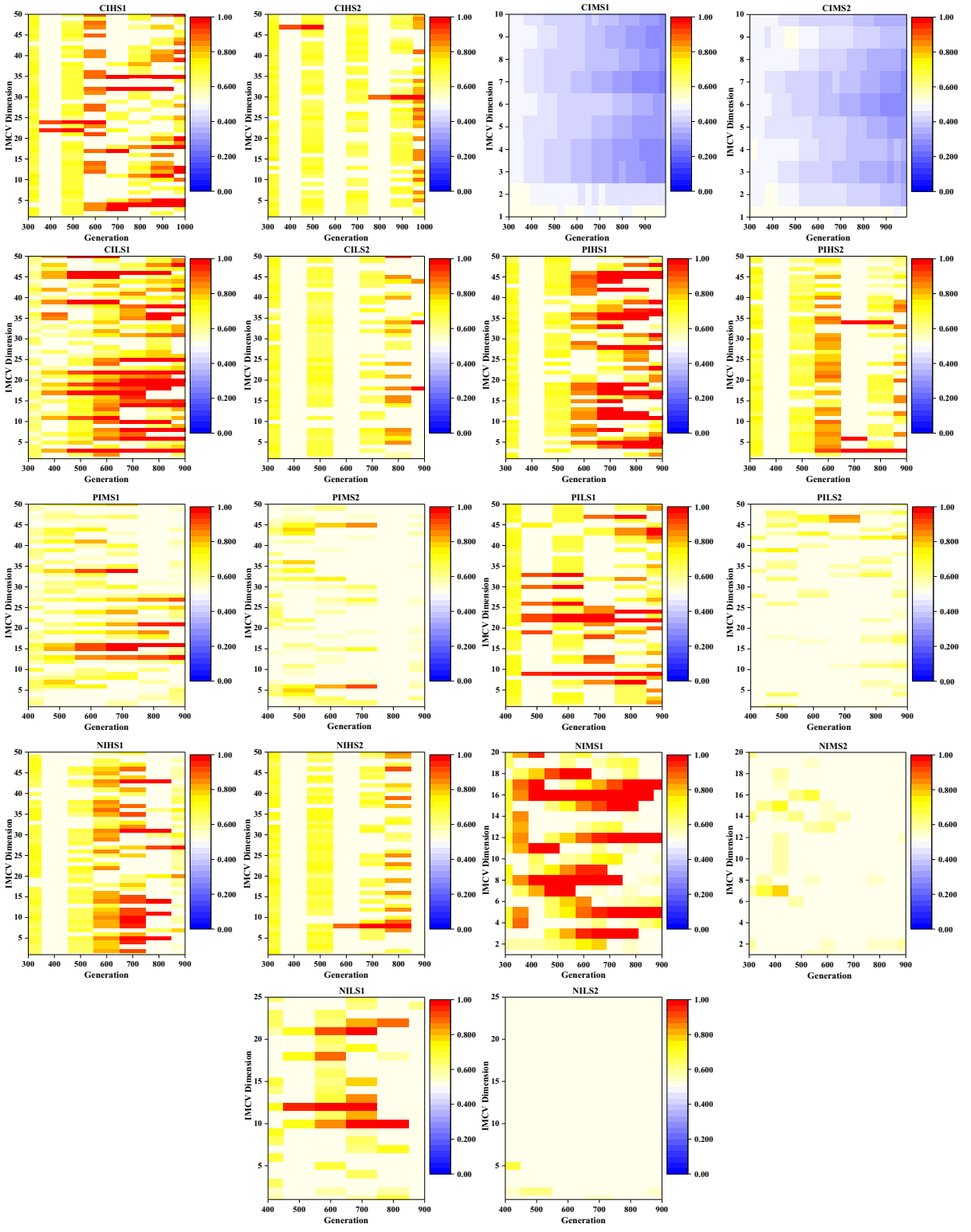

Fig. 10. The values of IMCV in the classical MOMTO test problems.

the position of non-separable variables and mutate a single decision variable to perform a fine-grained search. Therefore, EMT-DMS can quickly converge to the approximate PF, but it is hard to continue to be better and likely to mature prematurely. Comparing EMT-PCM and EMT-TS, EMT-PCM is only better on PIMS and NIMS2. Although PCM-ES can carry out a

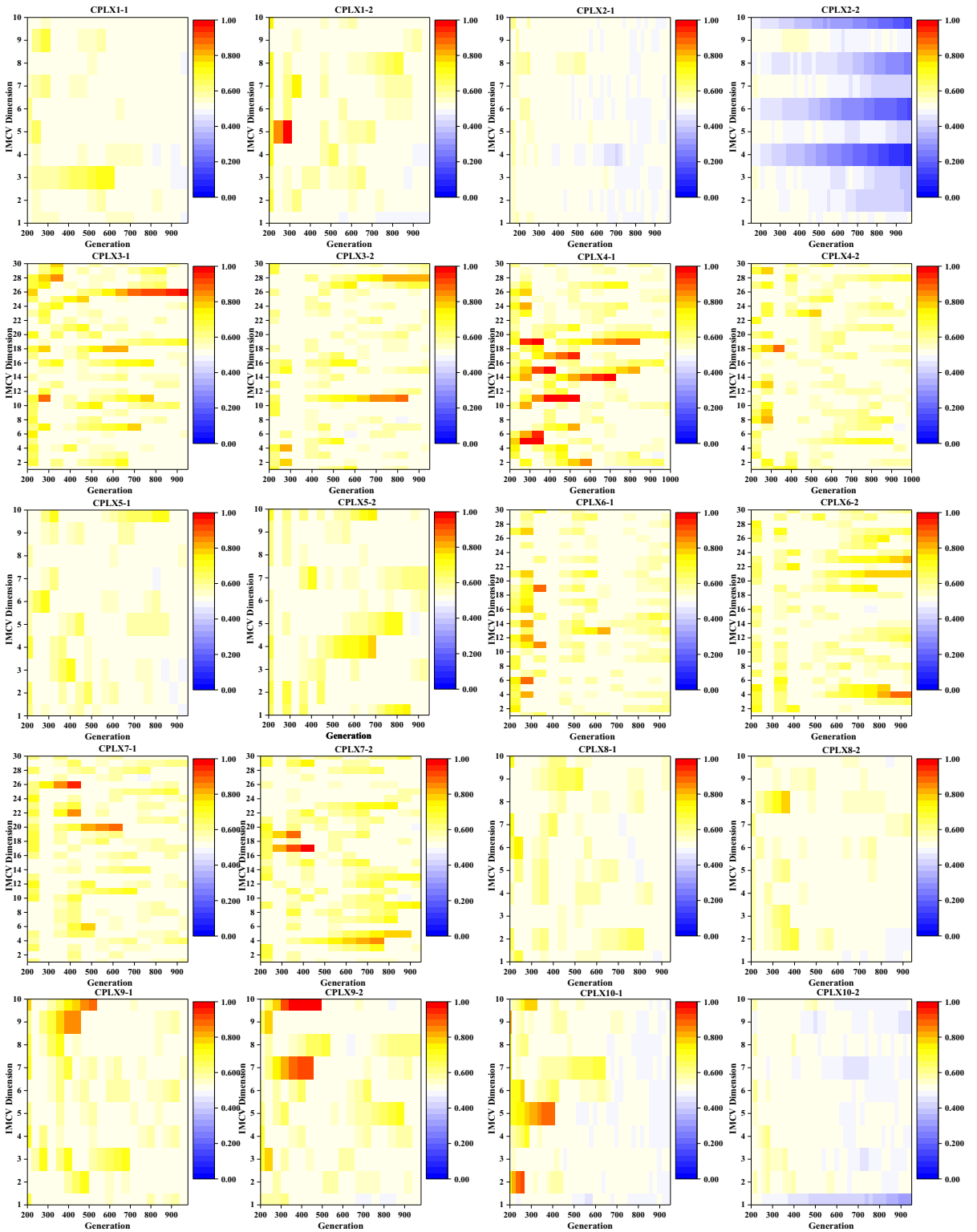

Fig. 11. The values of IMCV in the complex MOMTO test problems.

subtle variation of unsynchronized length on a single variable, PCM-ES is poor at solving the non-separability problem. EMT-TS takes advantage of two evolutionary operators. It can quickly guide the population to an approximate PF or a point on the PF in the early iterations and then use PCM-ES to fine-tune a single decision variable, which can fully play MIEA's strength

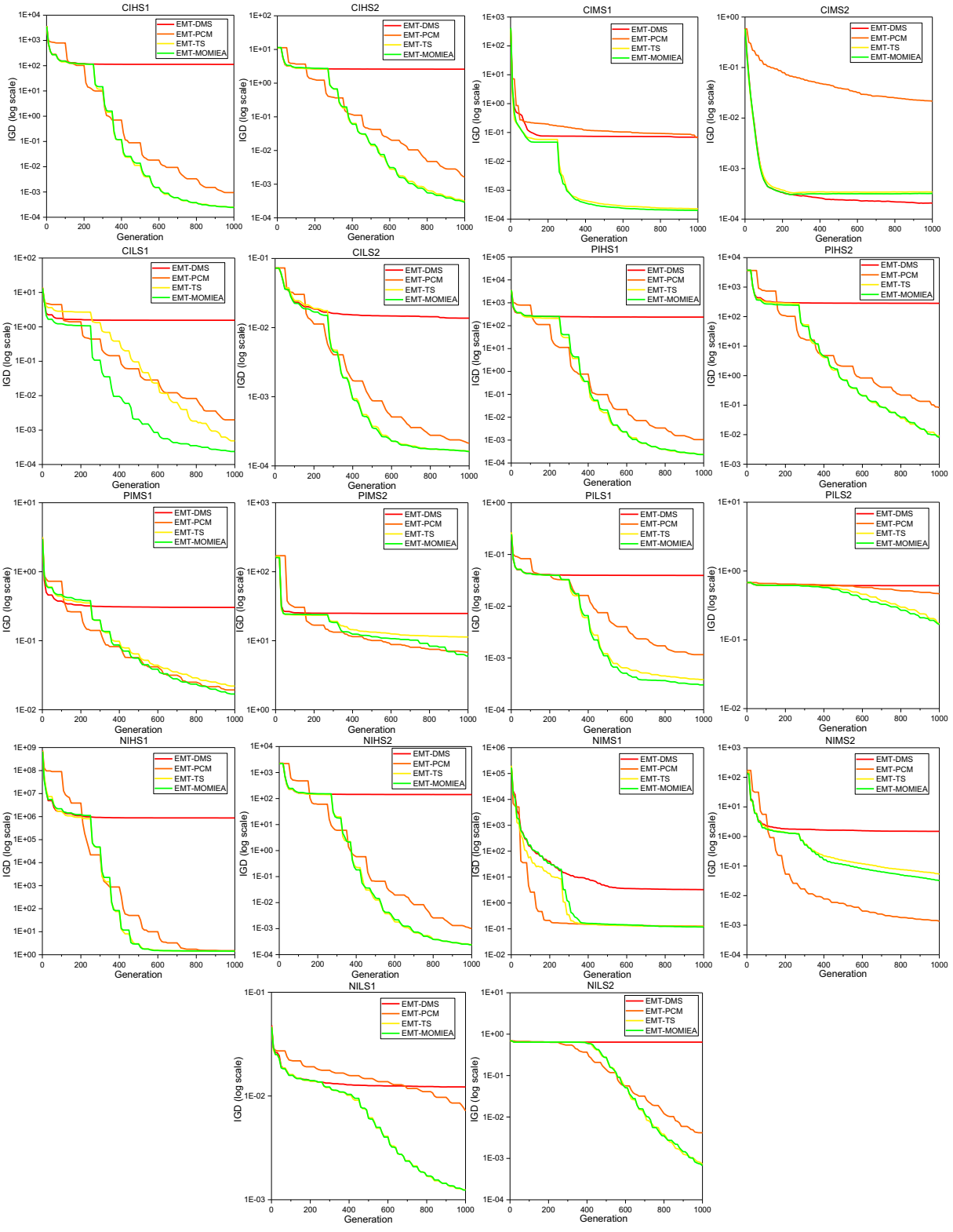

Fig. 12. The IGD curves by EMT-DMS, EMT-PCM, EMT-TS, EMT-MOMIEA on the classical MOMTO test problems.

and potential. Comparing EMT-TS and EMT-MOMIEA, EMT-TS performs worse than EMT-MOMIEA in all sub-problems. This is because EMT-MOMIEA utilizes the proposed IMCV based adaptive information transfer strategy, which can dynamically change the information transfer probability, thus reducing the occurrence of negative transfer.

Fig. 12 shows the mean IGD curves by EMT-DMS, EMT-PCM, EMT-TS, EMT-MOMIEA after 20 independent runs on the classical MOMTO test problems. Comparing EMT-DMS and EMT-PCM, it can be found that in most sub-problems, EMT-DMS can evolve faster in about the first 100 generations, thanks to the fast convergence of the DE operator based on the dynamic membrane structure. However, after about 100 generations, the IGD of EMT-PCM tends to continue to decline, while EMT-DMS tends to converge. This is because EMT-DMS modifies all decision variables each time, and it isn't easy to generate a better individual than the parent in this way. Comparing EMT-DMS, EMT-TS, and EMT-MOMIEA, it can be seen that these three algorithms have the same iterative trend in about the first 250 generations because they all use the EMT-DMS evolutionary operator. After about 250 generations, EMT-TS and EMT-MOMIEA can give full play to the advantages of the PCM-ES operator, which can independently mutate a single decision variable to make the population continue to improve. Comparing EMT-TS and EMT-MOMIEA, although the performance of the two algorithms is not much different at the beginning of the iteration, as the iteration progresses, the advantages of EMT-MOMIEA gradually become prominent. The adaptive transfer strength strategy based on IMCV can effectively reduce negative transfer.

5. Conclusion

In this paper, a novel membrane-inspired evolutionary framework with a hybrid dynamic membrane structure is proposed to solve the MOMTO problems. The overall algorithm is contained in a skin membrane, and each task is involved in a separate sub-membrane. The rewrite rules in the membrane system are introduced to evolve symbol objects to converge. The communication rules are adopted to communicate between the membranes that solve different tasks to exchange and reuse the information between tasks. A novel two-stage evolution strategy algorithm based on the membrane system is proposed as AIM to improve the convergence and diversity. In the first stage, a novel differential evolution strategy based on dynamic membrane structure is proposed to promote multiset to converge quickly. In the second stage, the precision controllable mutation-based evolution strategy is adopted to promote the further convergence of the multiset while improving the diversity of the population. The information molecule concentration vector inspired by the binding process of information molecules and receptors during information exchange between cells is designed for adjusting transfer intensity to avert the negative information transfer. The performance is compared against four state-of-the-art multi-objective EMT algorithms and NSGA-II on classical MOMTO, complex MOMTO, and MOMaTO test suite. The experiment results clearly show that the proposed EMT-MOMIEA provides a competitive edge over the state-of-the-art EMT algorithms.

In the future, we will conduct further research in the following aspects. First, the proposed EMT-MOMIEA is designed based on the cell-like membrane structure. In the future, MIEAs based on tissue-like membrane structures can be considered to study the performance difference of different membrane structures in solving the MOMTO problems. Then, the proposed EMT-MOMIEA can also be used to solve large-scale MTO, dynamic MTO problems, and other complex optimization problems in the real world.

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.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant Nos. 62176191, 61702383). The authors would like to thank Yuan Yuan, Yew-Soon Ong, Liang Feng, A.K. Qin, Abhishek Gupta, Bingshui Da, Qingfu Zhang, Kay Chen Tan, Yaochu Jin, and Hisao Ishibuchi for friendly supplying the code of [43]. We would thank Zhengping Liang, Jian Zhang, Liang Feng, Zexuan Zhu for giving the code of [33]. We also appreciate Zhengping Liang, Hao Dong, Cheng Liu, Weiqi Liang, and Zexuan Zhu for the code of [29].

References

[1] K. Zhang, Z. Xu, S. Xie, G.G. Yen, Evolution strategy-based many-objective evolutionary algorithm through vector equilibrium, IEEE Trans. Cybern. 51 (2021) 5455–5467, https://doi.org/10.1109/TCYB.2019.2960039.

[2] 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, https://doi.org/10.1109/4235.996017.

[3] J. Xu, X. Chen, M. Wu, W. Cao, Highest wellbore stability obstacle avoidance drilling trajectory optimization in complex multiple strata geological environment, in: IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society, IEEE, Toronto, ON, Canada, 2021: pp. 1–6. https://doi.org/10.1109/IECON48115.2021.9589734.

[4] J. Bader, E. Zitzler, HypE: an algorithm for fast hypervolume-based many-objective optimization, Evol. Comput. 19 (2011) 45–76, https://doi.org/10.1162/EVCO_a_00009.

[5] S. Rostami, F. Neri, A fast hypervolume driven selection mechanism for many-objective optimisation problems, Swarm Evol. Comput. 34 (2017) 50–67, https://doi.org/10.1016/j.swevo.2016.12.002.

[6] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (2007) 712–731, https://doi.org/10.1109/TEVC.2007.892759.
[7] H.-L. Liu, F. Gu, Q. Zhang, Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems, IEEE Trans. Evol. Comput. 18 (2014) 450–455, https://doi.org/10.1109/TEVC.2013.2281533.
[8] 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, https://doi.org/10.1016/j.neucom.2017.02.065.
[9] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (2010) 1345–1359, https://doi.org/10.1109/TKDE.2009.191.
[10] A. Gupta, Y. Ong, L. Feng, Multifactorial evolution: toward evolutionary multitasking, IEEE Trans. Evol. Comput. 20 (2016) 343–357, https://doi.org/10.1109/TEVC.2015.2458037.
[11] 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, https://doi.org/10.1016/j.engappai.2017.05.008.
[12] Y.-S. Ong, A. Gupta, Evolutionary multitasking: A computer science view of cognitive multitasking, Cogn. Comput. 8 (2016) 125–142, https://doi.org/10.1007/s12559-016-9395-7.
[13] Q. Xu, N. Wang, L. Wang, W. Li, Q. Sun, Multi-task optimization and multi-task evolutionary computation in the past five years: A brief review, Mathematics 9 (2021) 864, https://doi.org/10.3390/math9080864.
[14] K.C. Tan, L. Feng, M. Jiang, Evolutionary transfer optimization - A new frontier in evolutionary computation research, IEEE Comput. Intell. Mag. 16 (2021) 22–33, https://doi.org/10.1109/MCI.2020.3039066.
[15] G. Zhang, M. Gheorghe, L. Pan, M.J. Pérez-Jiménez, Evolutionary membrane computing: A comprehensive survey and new results, Inf. Sci. 279 (2014) 528–551, https://doi.org/10.1016/j.ins.2014.04.007.
[16] T. Wu, S. Jiang, Spiking neural P systems with a flat maximally parallel use of rules, Journal of Membrane Computing 3 (2021) 221–231, https://doi.org/10.1007/s41965-020-00069-5.
[17] S. Verlan, R. Freund, A. Alhazov, S. Ivanov, L. Pan, A formal framework for spiking neural P systems, Journal of Membrane Computing 2 (2020) 355–368, https://doi.org/10.1007/s41965-020-00050-2.
[18] Y. Liu, R. Nicolescu, J. Sun, An efficient labelled nested multiset unification algorithm,Journal of Membrane Computing 3 (2021) 194–204, https://doi.org/10.1007/s41965-021-00076-0.
[19] J. Xiao, X. Zhang, J. Xu, A membrane evolutionary algorithm for DNA sequence design in DNA computing, Chin. Sci. Bull. 57 (2012) 698–706, https://doi.org/10.1007/s11434-011-4928-7.
[20] M. Zhu, Q. Yang, J. Dong, G. Zhang, X. Gou, H. Rong, P. Paul, F. Neri, An adaptive optimization spiking neural P system for binary problems, Int. J. Neur. Syst. 31 (2021) 2050054, https://doi.org/10.1142/S0129065720500549.
[21] G. Zhang, H. Rong, F. Neri, M.J. Pérez-Jiménez, An optimization spiking neural P system for approximately solving combinatorial optimization problems, Int. J. Neur. Syst. 24 (2014) 1440006, https://doi.org/10.1142/S0129065714400061.
[22] J. He, J. Xiao, X. Liu, T. Wu, T. Song, A novel membrane-inspired algorithm for optimizing solid waste transportation, Optik 126 (2015) 3883–3888, https://doi.org/10.1016/j.ijleo.2015.07.152.
[23] J. He, J. Xiao, Z. Shao, An adaptive membrane algorithm for solving combinatorial optimization problems, Acta Math. Sci. 34 (2014) 1377–1394, https://doi.org/10.1016/S0252-9602(14)60090-4.
[24] Z. Xu, X. Liu, K. Zhang, J. He, Cultural transmission based multi-objective evolution strategy for evolutionary multitasking, Inf. Sci. 582 (2022) 215–242, https://doi.org/10.1016/j.ins.2021.09.007.
[25] K.K. Bali, A. Gupta, Y.-S. Ong, P.S. Tan, Cognizant multitasking in multiobjective multifactorial evolution: MO-MFEA-II, IEEE Trans. Cybern. 51 (2021) 1784–1796, https://doi.org/10.1109/TCYB.2020.2981733.
[26] S. Yao, Z. Dong, X. Wang, L. Ren, A Multiobjective multifactorial optimization algorithm based on decomposition and dynamic resource allocation strategy, Inf. Sci. 511 (2020) 18–35, https://doi.org/10.1016/j.ins.2019.09.058.
[27] M. Gong, Z. Tang, H. Li, J. Zhang, Evolutionary multitasking with dynamic resource allocating strategy, IEEE Trans. Evol. Computat. 23 (2019) 858–869, https://doi.org/10.1109/TEVC.2019.2893614.
[28] 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, https://doi.org/10.1109/TCYB.2018.2845361.
[29] 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. (2020) 1–14, https://doi.org/10.1109/TCYB.2020.2980888.
[30] J. Zhang, W. Zhou, X. Chen, W. Yao, L. Cao, Multisource selective transfer framework in multiobjective optimization problems, IEEE Trans. Evol. Comput. 24 (2020) 424–438, https://doi.org/10.1109/TEVC.2019.2926107.
[31] Y. Chen, J. Zhong, L. Feng, J. Zhang, An adaptive archive-based evolutionary framework for many-task optimization, IEEE Trans. Emerg. Top. Comput. Intell. 4 (2020) 369–384, https://doi.org/10.1109/TETCI.2019.2916051.
[32] A. Gupta, Y.S. Ong, L. Feng, K.C. Tan, Multiobjective multifactorial optimization in evolutionary multitasking, IEEE Trans. Cybern. 47 (2017) 1652–1665, https://doi.org/10.1109/TCYB.2016.2554622.
[33] 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), https://doi.org/10.1016/j.eswa.2019.07.015 112798.
[34] Z. Xu, K. Zhang, Multiobjective multifactorial immune algorithm for multiobjective multitask optimization problems, Appl. Soft Comput. 107 (2021) 107399. https://doi.org/10.1016/j.asoc.2021.107399.
[35] J.A. Andreu-Guzmán, L. Valencia-Cabrera, A novel solution for GCP based on an OLMS membrane algorithm with dynamic operators, Journal of Membrane Computingt 2 (2020) 1–13, https://doi.org/10.1007/s41965-019-00026-x.
[36] J. Cheng, G. Zhang, X. Zeng, A novel membrane algorithm based on differential evolution for numerical optimization, Int. J. Unconven. Comput. 7 (2011).
[37] H. Peng, J. Shao, B. Li, J. Wang, M.J. Pérez-jiménez, Y. Jiang, Y. Yang, Image thresholding with cell-like P systems, in: Proceedings of the Tenth Brainstorming Week on Membrane Computing. Volume II, n.d.: pp. 75–88.
[38] S. Mi, L. Zhang, H. Peng, J. Wang, Medical image fusion based on DTNP systems and Laplacian pyramid, Journal of Membrane Computing 3 (2021) 284–295, https://doi.org/10.1007/s41965-021-00087-x.
[39] B. Song, X. Zeng, Solving a PSPACE-complete problem by symport/antiport P systems with promoters and membrane division, Journal of Membrane Computing 3 (2021) 296–302, https://doi.org/10.1007/s41965-021-00084-0.
[40] G. Zhang, J. Cheng, M. Gheorghe, F. Ipate, X. Wang, QEAM: an approximate algorithm using P systems with active membranes, Int. J. Comput. Commun. Control. 10 (2015) 263, https://doi.org/10.15837/ijccc.2015.2.1757.
[41] A.W. Iorio, X. Li, Rotated test problems for assessing the performance of multi-objective optimization algorithms, in: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation - GECCO '06, ACM Press, Seattle, Washington, USA, 2006: p. 683. https://doi.org/10.1145/1143997.1144118.
[42] X. Ma, F. Liu, Y. Qi, X. Wang, L. Li, L. Jiao, M. Yin, M. Gong, A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables, IEEE Trans. Evol. Comput. 20 (2016) 275–298, https://doi.org/10.1109/TEVC.2015.2455812.
[43] 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, ArXiv:1706.02766 [Cs]. (2017). http://arxiv.org/abs/1706.02766 (accessed September 29, 2019).

[44] L. Feng, K. Qin, A. Gupta, Y. Yuan, Y.-S. Ong, and X. Chi, IEEE CEC 2020 Competition on Evolutionary Multi-task Optimization, http://www.bdsc.site/websites/MTO_competition_2020/MTO_Competition_WCCI_2020.html, 2020.
[45] H. Li, Q. Zhang, Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II, IEEE Trans. Evol. Comput. 13 (2009) 284–302, https://doi.org/10.1109/TEVC.2008.925798.
[46] P. Czyżżak, A. Jaszkiewicz, Pareto simulated annealing—a metaheuristic technique for multiple-objective combinatorial optimization, J. Multi-Crit. Decis. Anal. 7 (1998) 34–47, https://doi.org/10.1002/(SICI)1099-1360(199801)7:1.
[47] J.J. Durillo, A.J. Nebro, jMetal: A Java framework for multi-objective optimization, Adv. Eng. Softw. 42 (2011) 760–771, https://doi.org/10.1016/j.advengsoft.2011.05.014.
[48] S. Rostami, F. Neri, K. Gyaurski, On algorithmic descriptions and software implementations for multi-objective optimisation: A comparative study, SN Comput. Sci. 1 (2020) 247, https://doi.org/10.1007/s42979-020-00265-1.