Evolutionary Multi-Objective Membrane Algorithm

CHUANG LIU, YINGKUI DU, AO LI, AND JIAHAO LEI
School of Information Engineering, Shenyang University, Shenyang 110044, China
Corresponding author: Chuang Liu (chuang.liu@mail.dlut.edu.cn)

This work was supported in part by the Startup Research Fund for Ph.D. of Liaoning Province, China, under Grant 20170520364, in part by the Guidance Plan for Key Research and Development Plans of Liaoning Province under Grant 2018104013, in part by the Shenyang Young and Middle-Aged Science and Technology Innovation Talents Support Plan under Grant RC180338, in part by the Innovative Talents Support Plan of Liaoning Education Department under Grant LR2016074, and in part by the Science and Technology Program of Shenyang City, China, under Grant F16-155-9-00 and Grant F17-180-9-00.

Abstract

Recent advances in evolutionary algorithms based on membrane computing have shown that the mechanism of membrane computing is an effective way to solve optimization problems. In this work, we propose a new evolutionary multi-objective algorithm that uses membrane systems to solve multi-objective optimization problems. Based on the mechanism of living cell structure and function, the algorithm introduces three factors, including membrane structure, multiset and reaction rules. The membrane structure of the proposed algorithm is inspired by the structure of the membrane system, which has multiple layers and nested structures in the skin membrane. Two special symbol-objects are designed to improve the search efficiency of the algorithm. In addition, some reaction rules are used to evolve the symbol-objects of multiset in the inner region of the membrane. In addition, the proposed method combines external archive to maintain the diversity of non-dominated solutions and enhance the search capabilities of the solutions. Our proposed method is compared to five state-of-the-art multi-objective heuristic algorithms. For comparison, six different criteria were used: the quality of the resulting approximate set, the diversity of candidate solutions, and the rate of convergence to the Pareto front. Experimental results show that the proposed method is competitive in performance in qualitative and quantitative measurements of selected test functions. Therefore, the algorithm is feasible and effective for solving multi-objective optimization problems.

Index Terms

Membrane computing, multiobjective optimization, evolutionary computation.

I. INTRODUCTION

A large number of real-world optimization problems, especially in scientific research and engineering applications, require the optimization of multiple targets that are often conflicting. These problems are called multi-objective problems (MOPs) and are usually solved by finding for their maximum or minimum values under certain constraints. Unlike a single-objective problem, MOP is usually not composed of a single solution, but consists of an entire set of solutions. In recent years, many scholars around the world have carried out a lot of research work in the field of multi-objective optimization [1]–[4]. Although many multiobjective algorithms may find some trade-off solutions, the solution still cannot spread along the entire Pareto front.

As a new branch of natural computing, membrane computing is proposed by Păun inspired by the structure and function of the biological cells [5], [6]. The device model of membrane computing is called membrane system or P system, which is a new distributed parallel computing framework. The basic model consists of a hierarchical structure, several multisets, and some reaction rules. Most membrane systems have been shown to have the universality and computational effectiveness of Turing [7]–[10]. Recently, research on membrane systems has begun to focus on multidisciplinary topics such as computer science, biology and artificial intelligence [11]. In particular, membrane systems have some applications to solve optimization problems. Nishida [12] first proposed a novel membrane algorithm to solve the traveling salesman problem. On the basis of this idea, Nishida incorporated a dynamic membrane structure and sub-algorithm, including Tabu search, Brownian motion and genetic algorithms to...
solve traveling salesman problem (TSP for short) [13], [14]. Leporati and Pagani proposed an improved membrane algorithm for the minimum storage problem based on Nishida’s previous work [15]. Huang et al. presented an optimization algorithm based on the membrane systems for large feasible space and parameters problems [16]. Zhang et al. introduced an evolutionary algorithm, which was on the basic of the quantum-inspired evolutionary approach and membrane systems, to solve a well-known combinatorial knapsack optimization problem [17]. Zaharie and Ciobanu proposed new strategies of applying the operators in evolutionary algorithms and new variants of distributed evolutionary algorithms for some continuous optimization problems [18]. Zhang et al. presented a hybrid approach based on the appropriate combination of Differential Evolution algorithms and Tissue P Systems to solve a class of constrained manufacturing parameter optimization problems [19]. Peng et al. proposed a multiobjective clustering framework to deal with fuzzy clustering problem by designing a tissue-like membrane system with a special cell structure [20]. Liu et al. proposed a new membrane algorithm to solve the community detection in complex networks [21].

However, it is rare that the membrane systems are used to solve MOPs. Huang et al. proposed a new multi-objective optimization algorithm based on the membrane systems, named PMOA. The algorithm selected an important objective and optimized first, then optimized other objective. But, this algorithm could not effectively share the information among the objectives. Also, the selection of the important objective directly affects the final Pareto fronts [22]. Zhang et al. proposed a multi-objective membrane algorithm, which is based on the membrane systems and quantum-inspired evolutionary algorithms, to solve multi-objective knapsack problems [23]. Liu et al. proposed a novel algorithm based on membrane systems for solving multi-objective optimization problems [24]. Zhang et al. proposed an effective multi-objective membrane algorithm guided by the skin membrane where the information of solutions stored in the skin membrane is used to guide the evolution of internal membranes [25]. Liu et al. proposed a hybrid optimization algorithm to solve manyobjective optimization problems, which consists of evolutionary membrane algorithm and chemical reaction optimization algorithm [26].

Based on our previous work, we propose a new evolutionary multi-objective optimization algorithm based on membrane computing theory called multi-objective optimized membrane computing (MOMC). The main idea of the algorithm is described below. Membrane system mechanisms such as membrane structures, symbol-objects, multisets and reaction rules are introduced into the proposed algorithm. In the inner region of the membrane system, the symbol-object represents a candidate solution for MOPs. Next, the multiset consists of several symbol-objects. In addition, the reaction rule of the inner region of the membrane is used to obtain a non-dominated solution of MOPs. In addition, the proposed method includes several strategies to maintain the diversity of non-dominated solutions and speed up its own convergence speed. Finally, the non-dominated solutions from the elementary membrane region are atomically released into the inner region of the skin membrane in each cycle. These solutions form the final Pareto Front of MOPs.

The key contributions of this paper are summarized as follows:

- Solving multi-objective optimization problems based on membrane system
- Designing membrane structures, multisets and reaction rules based on the characteristics of the optimization problem
- Achieving a balance between exploration and exploitation

The remainder of this paper is organized as follows: Section II presents a brief review on background of multi-objective optimization. In Section III, the details of the proposed algorithm are elaborated. Comprehensive study and experimental results are discussed in Section IV, and finally, Section V provides concluding remarks of the study.

II. BACKGROUND OF MULTIOBJECTIVE OPTIMIZATION

Multi-objective optimization consists of multiple conflicting objects at the same time, which is a very interesting topic because many real-world problems have multi-objective properties. The minimized multi-objective optimization problem as an example can be described as follows:

\[
F(X) = \min (f_1(X), \ldots, f_m(X)),
\]

s.t. \( g_i(X) = 0, i = 1, \ldots, q, \)
\( h_j(X) \geq 0, j = 1, \ldots, p, \)
\( X_l \leq X \leq X_u, X \in R^n, F \in R^m \)  \hspace{1cm} (1)

where, \( X = (x_1, x_2, \ldots, x_n) \) is the vector of \( n \) decision variables. \( F(X) \) is the vector of \( m \) minimized objective functions. \( g_i(X) = 0 \) are \( q \) equality constraints. \( h_j(X) \geq 0 \) are \( p \) inequality constraints. \( X_l \) is the lower bound of the decision variable \( X \). \( X_u \) is the upper bound of the decision variable \( X \).

The following sections give some definitions to further understand multi-objective optimization.

Definition 1. Dominate. Let \( U = (u_1, u_2, \ldots, u_l), V = (v_1, v_2, \ldots, v_l) \), \( U \) is said to dominate \( V \) if and only if \( u_i \leq v_i \) for \( \forall i \in \{1, \ldots, l\} \) and \( u_i < v_i \) for at least one index.

Definition 2. Pareto optimal solution. A solution \( X^* \in \Omega \) is Pareto optimal to (1) if there is no other solution \( X \in \Omega \) such that \( F(X) \) dominates \( F(X^*) \). \( X^* \) is called a Pareto optimal solution.

Definition 3. Pareto Set (PS). The set of all the Pareto optimal solutions is called as the Pareto Set. The schematic diagram describe that all the hollow circles in decision space represent the PS in Figure 1.

Definition 4. Pareto Front (PF). The set of all the Pareto Set is called the Pareto Front, \( PF = \{F(X) \in R^m | X \in PS\} \). The schematic diagram describe that all the solid circles in objective space represent the PF in Figure 1.
III. EVOLUTIONARY MULTIOBJECTIVE MEMBRANE ALGORITHM

As described in Section I, the main purpose of this paper is to design an evolutionary multi-objective membrane algorithm inspired by the structure and function of cells in the membrane system. Therefore, we will propose an optimization algorithm based on the membrane systems to solve MOPs. In addition, the proposed method incorporates several strategies to maintain the diversity of non-dominated solutions and enhance its own global optimization capabilities. Finally, the implementation details of the proposed algorithm are described in the next section.

The complete membrane system consists of three factors, including multiset, reaction rules and membrane structures [5], [6]. The proposed algorithm involves three key strategies: how to establish a relationship between multiset of membrane systems and solutions of MOPs. In addition, how to design reaction rules to evolve the multiset to find the best approximate solution. Finally, how to define the membrane structure is a key issue because it describes the execution logic of the algorithm. This structure can improve the speed at which the proposed algorithm reaches Pareto Front. Additional designs are included to support the aforementioned three strategies. These designs include the following: Pareto diversity maintenance with crowded distances, and external archives to update globally optimal solutions. Algorithm 1 shows the general steps of MOMC. The details of some of the key steps in Algorithm 1 are detailed in the following sections.

A. SYMBOL-OBJECTS AND MULTISET

In membrane systems [6], symbol-objects represent abstract representations of atoms or molecules in liquid chemicals; and multisets consist of the multiple symbol-objects. In the proposed MOMC, the symbol-object represents the solution to the optimization problem; and the multiset presents a set of the multiple symbol-objects. In addition, unlike the symbol-object in membrane computing, the symbol-object in the proposed MOMC can be encoded in decimal code to represent a solution of optimization problems. Therefore, symbol-objects can be generated under the constraints of the optimization problem, and their specifications can be described in formula 2.

\[ S_{l,j} = S_{l,j} + (S_{a,j} - S_{l,j}) \times \text{rand}() \quad (2) \]

where, \( 1 \leq i \leq N, N \) denotes the number of symbol-objects in the multiset, \( 1 \leq j \leq D, D \) represents the dimension size of the optimization problems. \( S_{l,j} \) is the \( j \)-th dimension of the \( i \)-th symbol-object. \( S_{l,j} \) represents the \( j \)-th dimension of the lower boundary. \( S_{a,j} \) denotes the \( j \)-th dimension of the upper boundary. rand is a random function which can generate a random number between 0 and 1.

In order to improve the global optimization capabilities of the proposed algorithm, we construct two special symbol-objects in the proposed algorithm. These symbol-objects are advantageous for the algorithm to converge to Pareto Front with a high probability. The detail of their form is described as follows:

The first symbol-object \( S_a = S_{a,1}, \cdots, S_{a,j}, \cdots, S_{a,D} \) is named as a balanced symbol-object. The construction of the balanced symbol-object is inspired by multiple collisions of many symbol-objects in the membrane. Its form is described in formula 3.

\[ S_{a,j} = \alpha \times \frac{1}{M} \sum_{m=1}^{M} S_{m,j} \quad (3) \]

where, \( S_{a,j} \) is the \( j \)-th dimension of the balanced symbol-object. \( \alpha \) is a disturbance coefficient. \( M \) is the number of symbol-objects in the membrane.

The second symbol-object is called an intermediate symbol-object. The intermediate symbol-object can exchange information about symbol-objects between the skin membrane and the membrane. In other words, information

---

**Algorithm 1** The Pseudo-Code of the Proposed MOMC

**Input:** The parameters of the proposed algorithm, including the number of symbol-objects, each symbol-object within its boundaries.

**Output:** The best multiset is found.

1: Construct the skin membrane.
2: Initialize the symbol-objects and multiset.
3: Evaluate the symbol-objects while End Condition is not Met do
4: Evaluate the symbol-objects
5: Send the multiset into the different membranes
6: Send the multiset into the different membranes
7: Wait for the multiset from each membrane
8: Save the multiset from the different membrane into the archive.
9: end while

**Output:** Receive multiset from the outer membrane

1: Receive the multiset from the outer membrane. The intermediate symbol-object can exchange information about symbol-objects between the skin membrane and the membrane. In other words, information

**FIGURE 1.** Pareto set and Pareto front.
from symbol-objects of different membranes can be effectively shared by using the intermediate symbol-object. In our algorithm, the intermediate symbol-object is randomly chosen in the external archive. Its form is described in 4.

\[ S_k = W_a(i), \quad i = \text{rand()} \]  

(4)

where, \( S_k \) is the intermediate symbol-object. \( W_a \) is the multiset in the skin membrane which records the best symbol-objects from different membranes. \( W_a(i) \) represents a symbol-object selected from the archive. \( \text{rand()} \) is a random function.

The multiset is constructed by combining the initialized symbol-objects according to the formula 2. Note that there is only one multiset in the skin membrane, and the multiset in the membrane is constructed by separating the multiset in the skin membrane. In our proposed algorithm, the number of the divided multiset is determined according to the number of outermost membrane except the skin membrane. In the following section, we will describe how to create the multiset in each membrane. It is assumed that there are \( M \) outermost membranes in addition to the skin membrane, and there are \( N \) symbol-objects of the multiset in the skin membrane. Moreover, we need the same number of symbol-objects in each membrane. So, we can obtain \( M \) multisets according to the formula 5.

\[ W_k = S_{k(1)N/M}, \ldots, S_{kN/M} \]  

(5)

where, \( S_{k(1)N/M} \) is the \((k-1) \times N/M\) symbol-objects of multiset in the skin membrane. \( W_k \) represents the \( k \)-th multiset in the elementary membrane, \( k = 1, \ldots, M \).

B. REACTION RULES

In membrane systems [6], each rule corresponds to a chemical reaction that can process the information of symbol-objects and the membrane structure. Rules for evolving symbol-objects are provided that work on a specified compartment or on a specified membrane. The most common types of rules are multiset rewriting rules (similar to chemical reactions) and transportation rules inspired by biological processes. Symbol-objects can not only evolve, but they also pass through the membrane through communication rules.

Based on the membrane system, there are two main types of rules in our algorithm: rewriting rules, and communication rules. The rewriting rules correspond to possible chemical reactions in the cell compartment. Communication rules can send symbol-objects from one membrane to another membrane. The details of the two types of reaction rules are described as follows:

Two rewriting rules, simulating the process of molecular motion, are designed to evolve symbol-objects. The rewriting rules including Formula 6 and Formula 7 incorporate Gaussian distribution functions, because the randomness of molecular motion can be described by Gaussian processes. The rewriting rules can be illustrated as follows:

\[ S_{i,j} = r \times (2 \times S_{a,j} - S_{i,j}) + (1 - r) \times S_{g,j} \]  

(6)

\[ S_{i,j} = (1 + 2 \times r) \times S_{g,j} - 2 \times r \times S_{i,j} - S_{a,j} \]  

(7)

where, \( S_{i,j} \) is the \( j \)-th dimension of the \( i \)-th symbol-object. \( r \) is a Gaussian distribution function. \( S_{a,j} \) is the \( j \)-th dimension of the balanced symbol-object. \( S_{g,j} \) is the \( j \)-th dimension of the intermediate symbol-object.

Communication rules enable the process of transferring and exchanging materials between membranes. By calling the rules in Formula 8, the multiset can be sent into the specific membranes which allow the algorithm to search for an approximate optimal solution in the specified search space. In addition, the multiset in the membrane can be sent into the skin membrane, which makes information from symbol-objects of the different elementary membranes to be shared with each other. The detail forms are described as follows:

\[ [W_1, W_2, \ldots, W_n]_0 \leftrightarrow [[W_1], [W_2], \ldots, [W_n]]_0 \]  

(8)

where, \( n \) is the amount of the outermost membrane except the skin membrane; \([_]_0 \) indicates the skin membrane; while \([_]\) is the \( i \)-th membrane; \( W_i \) is \( i \)-th sub-multiset based on the multiset divided equally in the skin membrane.

If a communication rule is invoked from the skin membrane to the membrane, then multiset need to be first divided into multiple sub-multisets. In other words, the multiset in the skin membrane is divided into several parts to send global information of the symbol-objects to different membranes. Therefore, a strategy for partitioning multiset is used in the proposed algorithm. The details of the strategy are described as follows: First, the symbol-objects of the multiset are sorted according to their fitness based on non-dominated sort [27]; second, the sorted symbol-objects are equally divided into several parts of the same size according to the non-dominated rank and crowding distance value, so that each membrane has its own part. Its form is described in Formula 9.

\[ W' = \text{nondomianted} - \text{sort}(W), \]  

\[ W_i = W'((i - 1) \times n + 1, i \times n)), \]  

\[ n = \text{sizeof}(W')/m \]  

(9)

where, \( n \) is the number of symbol-objects in the multiset; \( W' \) is a multiset sorted according to the fitness of symbol-objects based on non-dominated sort; \( W_i \) is \( i \)-th sub-multiset based on multiset divided equally in the skin membrane.

C. MEMBRANE STRUCTURE

Membrane systems have multiple layers and nested structure in which the skin membrane contains several membranes [6]. The main function of the membrane is to physically separate the two regions. In the proposed MOMC, the structure is simplified to a specific structure in which the skin membrane contains a number of special nested membranes. This particular structure of the proposed algorithm can be represented as Figure 2. In Figure 2, \( n \) is fixed when the algorithm is run. And \( e, h, \) and \( k \) are random integer values, respectively. In the simulation, \( n \) is set to 20. And \( e, h, \) and \( k \) are set to random integer values between 1 and 20, respectively.

In the proposed MOMC, information from symbol-objects of different membranes is shared in the region of the skin.
membrane, which can maintain the diversity of candidate solutions during the search process. The region of the membrane contains multiset and reaction rules. The symbol-objects in multiset are evolved by invoking reaction rules, which can generate new symbol-objects.

D. EXTERNAL ARCHIVE

We incorporate the external archive into our algorithm to solve MOPs. External archives are very necessary to save the best symbol-objects from different membranes. In other words, non-dominated candidate solutions can be found during the search process, which can help the algorithm produce a well-distributed Pareto Front. Moreover, an intermediate symbol-object is selected from the external archive. In addition, the key issue in archive management is to decide whether a new symbol-object should be added to it or not. Figure 3 depicts a new symbol-object being added into the external archive.

In the region of skin membrane, the archive is used to hold any good symbol-objects found from different membranes during the search process. The new symbol-object is compared with respect to all symbol-objects in the archive at each iteration. If the new symbol-objects are not dominated by any symbol-objects in the archive, they are inserted directly into the archive. Otherwise, the dominated symbol-objects are removed from the archive. When the size of the archive exceeds the threshold, a crowded distance mechanism will be employed to remove the symbol-objects with the least crowding distance. Finally, the proposed algorithm updates the archive and forms the next generation of symbol-objects. The archive allows the algorithm to improve the convergence speed to reach the known Pareto Front and increase the number of excellent non-dominated solutions.

IV. EXPERIMENT

This section presents some benchmark problems and six performance metrics to compare the performance of the proposed algorithm with other state-of-the-art algorithms, such as AbYSS [28], OMOPSO [29], MOCell [30], SPEA2 [31], and NSGAII [27]. Finally, the simulation results obtained by each algorithm are discussed.

A. TEST PROBLEMS

To evaluate the performance of the proposed algorithm, we used multi-objective function benchmarks covering various problems with different features (concave, convex, disconnected, deceptive, etc.). These benchmarks include the test suite Zitzler-Deb-Thiele (ZDT) [32] and the Deb-Thiele-Laumanns-Zitzler (DTLZ) problem series [33]. These problems include characteristics that are suitable for examining the effectiveness of multiobjective approaches in maintaining population diversity and converging to the Pareto Front.

The ZDT family of functions [32] has been selected because it is a comprehensive and popular set of test functions for benchmarking the performance of multi-objective Pareto optimization methods. Each of these test functions contains a specific feature that represents a real world optimization problem that can make it difficult to converge to Pareto Front. The ZDT1 function has the best front of the convex Pareto. The ZDT2 function has a non-convex Pareto optimal front. The ZDT3 function adds discrete functionality to the front. Its Pareto optimal front is composed of several noncontiguous convex parts. The introduction of a sine function into the objective function results in discontinuity in Pareto optimal front, but does not cause discontinuities in the parameter space. The ZDT4 function has 21 local Pareto-optimal fronts and is therefore highly multimodal. The ZDT6 function has a non-uniform search space: the Pareto optimal solutions are non-uniformly distributed over the global Pareto Front, and the density of the solutions is lowest near the Pareto optimal front and highest away from the front. Of all the ZDT functions, two design variables were selected. The design variable ranges are from 0 to 1, except for the ZDT4 problem, where the second design variable is in the interval [-5, 5]. The first one is also included in the interval [0, 1].

The DTLZ suite of benchmark problems, created by Deb et al. [33], is unlike most multi-objective test problems,
because the problems can be extended to any number of objectives, DTLZ2, DTLZ6 and DTLZ7 test problems are included in this family, three of which are addressed in this work. Test problem DTLZ2 is a three-objective minimization problem with standard size of 12 variables. Test problem DTLZ6 is a three-objective minimization problem with 12 decision variables and four disconnected sets of the Pareto Front regions. This problem tests the ability of the experimental algorithms to maintain subpopulations in different Pareto-optimal regions. DTLZ7 has a three-objective minimization problem with 22 decision variables.

**B. PERFORMANCE METRICS**

Many performance metrics proposed by many scholars will evaluate the performance of the experimental algorithms to solve MOPs. The goal of MOPs is to approximate the optimal boundary of the true Pareto. More specifically, it is necessary to calculate the distance between the approximate optimal solutions and the true Pareto Front; and the diversity of the approximate optimal solution is as large as possible so that the solution can extend the boundary of Pareto Front. Therefore, we use six metrics to evaluate the performance of these experimental algorithms, including: General Distance (GD) [34], Error Rate (ER) [35], Inverse General Distance (IGD) [36], the Hypervolume (HV) [37], Spread (SP) [38], and Maximum spread (MS) [32]. Please note that all metrics require the known true Pareto Front of the test problems.

![Performance Metrics](image)

**FIGURE 4.** The quality indicators of the MOPs.

Some of them are designed to measure convergence or diversity, while others take into account two criteria. Fig 4 depicts indicators they measure in our work. The GD and ER indicators are designed to measure the convergence of the experimental algorithms. The IGD and HV indicators are used to measure the trade-off performance between diversity and convergence. The Spread and MS metrics describe the diversity of the solutions obtained by the experimental algorithms.

**C. EXPERIMENTAL SETTING**

The experiment will run on Intel Pentium dual-core 2.93 GHz and 4G memory hardware environment as well as the Windows 7 operating systems. In the simulation experiment, the proposed algorithm was compared with some of the state-of-the-art algorithms, including AbYSS, OMOPSO, MOCell, SPEA2, and NSGAII. These algorithms were chosen because NSGA-II and SPEA2 are two state-of-the-art algorithms, and OMOPSO is a very salient MOPSO according to a comparative study. AbYSS and MOCell are two competing multi-objective evolutionary algorithms. Therefore, these algorithms are representative and help to make the comparisons more comprehensive and persuasive. The parameters of the aforementioned algorithms are configured according to the proposals in their corresponding references [27]–[31]. The parameters to be determined in the proposed algorithm can be described as follows:

\[ \Pi = \{O, \mu, w_1, \cdots, w_n, R_1, R_2\} \] (10)

where, \( \Pi \) denotes a membrane system. \( O \) denotes the symbol-object in the membrane algorithm. \( \mu \) is a membrane structure, consisting of \( n \) membranes. \( w_1, \cdots, w_n \) denotes the multiset present in regions 1, 2, \cdots, \( n \) of membrane structure. \( R_1 \) and \( R_2 \) denote two evolutionary rules.

**D. EXPERIMENTAL RESULTS**

In this section, simulations are performed to examine the performance of the proposed algorithm from several aspects. More specifically, the performance of the proposed algorithm is compared with other state-of-the-art algorithms for eight multi-objective problems. In order to provide a fair comparison, all algorithms run independently 30 times for each test problem, and the function evaluation for each run is 25,000 times. The statistical results are calculated, including Mean and Standard deviation. Based on the mentioned above simulation results, the statistical performance is analyzed, including consistency and robustness of the algorithms. All simulation results are given in the following section, where the best results for each test problem have a gray colored background. The distribution of 30 independently running simulation data is represented in a box plot format. Each box plot represents the distribution of the sample set, where the thick horizontal line within the box encodes the median, while the upper and lower ends of the box are the upper and lower quartiles. The dashed appendages illustrate the spread and shape of distribution, and dots represent outside values.

1) **COMPARSED RESULTS IN GD**

This indicator was introduced by Van Veldhuizen and Lamont [34] to measure the distance between the elements in the computed approximation and the elements in the optimal Pareto Front. The smaller the GD value is, the closer the distance is between the approximate solutions obtained by the algorithms and the optimal Pareto Front. A value of GD = 0 means that all generated elements are in Pareto Front.

Figure 5 shows a box plot of the GD indicator found by all selected multi-objective algorithm for all test problems. The figure shows that MOMC reach a high GD value except DTLZ2. In Figure 5, the GD values of AbYSS attained significantly poor results on ZDT3, DTLZ6 and DTLZ7. The OMOPSO algorithm achieves better results than AbYSS, MOCell, SPEA2 and NSGAII except ZDT4. MOCell is clearly better than the results of other comparison algorithms on ZDT2, ZDT4. Two classical multi-objective algorithms, including SPEA2 and NSGAII, can attain similar results for all test problems.
All results in Figure 5 are discussed by observing the box plot of the different multi-objective algorithms in the experiment. Table 1 clearly illustrates the discussion of quantifiable results. In Table 1, we can observe that MOMC achieves the best results on ZDT1, ZDT2, ZDT3, and ZDT4. The proposed MOMC gets the second best result on ZDT6, DTLZ6 and DTLZ7. The best results are attained by AbYSS on ZDT6, and SPEA2 on DTLZ2 achieves the best results, and OMOPSO on DTLZ6 get the best results, and NSGAII on DTLZ7 get the best results. Most importantly, the box plots and results in Table 1 clearly show that the performance of the proposed MOMC is significantly better than the compared algorithms. Lastly, the low standard deviation of all test problems in Table 1 indicates that the proposed MOMC can produce reliable solutions.
2) COMPARED RESULTS IN ER
This metric was proposed by Van Veldhuizen [35] to represent the percentage of solutions (from the non-dominated set found so far) that are not part of the Pareto Front set. The results of the ER indicator for all test functions are shown in Figure 6. By observing the figure, we found that MOMC achieve similar ER values for all test problems in comparison with the selected multi-objective algorithms.

In order to distinguish the similarity and difference of the box plot, we give a statistical analysis of the results obtained by the experimental algorithms in Table 2. The proposed MOMC achieves the best results on ZDT1, ZDT4, DTLZ2, and DTLZ7. In addition, MOCell attains the best result on ZDT2, and SPEA2 gets the best result on ZDT6 and DTLZ6, and NSGAIi finds the best result on ZDT3.
3) COMPARED RESULTS IN IGD

The IGD indicator is a variant of Generational Distance [36]. It measures the distances between each solution that makes up the optimal Pareto Front and the approximation solutions obtained by the experimental algorithms. The smaller IGD is, the better the performance of the algorithm is.

The IGD indicator for each function is summarized as a box plot in Figure 7. Except for DTLZ2, the MOMC algorithm can achieve the best results for all test problems. Furthermore, AbYSS attains the worst results on ZDT3 and DTLZ7, and OMOPSO gives the worst result on ZDT4, and MOCell found the worst results on DTLZ6 and DTLZ7. Similar results are obtained for SPEA2 and NSGAII on ZDT2, ZDT3, ZDT4 and DTLZ6.

From Table 3, we can observe that the MOMC algorithm achieves the best results on ZDT1, ZDT2, ZDT3, and ZDT4, and it gets the second results on ZDT6, DTLZ6, and DTLZ7. AbYSS can attain the best result on ZDT6. On DTLZ2, SPEA2 achieves the best result. On DTLZ6, OMOPSO find the best result. On DTLZ7, NSGAII achieve the best result. Therefore, the proposed algorithm can obtain the better results on IGD than the comparison algorithms.

4) COMPARED RESULTS IN HV

The HV indicator [37] calculates the volume in the objective space covered by non-dominated solution members. Algorithms with larger HV values are desirable. The hypervolume indicator measures the degree of convergence of the algorithm and produces non-dominated solutions that are evenly distributed and well-expanded on the Pareto Front.

Figure 8 shows a box plot of the results obtained by the multi-objective algorithms on the HV indicator. For all test problems, the proposed MOMC achieves the best results except DTLZ2. On ZDT1, NSGAII obtains the worst the results in comparison with the other algorithms. On ZDT2 and ZDT3, all the algorithms in simulation achieve the similar results. On ZDT4, OMOPSO attains the worst result. On DTLZ6, only MOMC and OMOPSO can get good results.
As shown in Table 3, we can observe that the proposed algorithm achieves the best results on ZDT1, ZDT2, and ZDT3. It attains the second best results on ZDT4, ZDT6, DTLZ6, and DTLZ7. MOMC finds the best result on ZDT4. OMOPSO attains the best results on ZDT6 and DTLZ6. SPEA2 finds the best result on DTLZ7.

5) COMPARED RESULTS IN SPREAD

The indicator [38] measures the range of spread by computing solutions. This indicator uses a zero value for the ideal distribution, pointing out the perfect spread of the solutions in Pareto Front.

As shown in Figure 9, the box plot of all algorithms is different for all test problems. On ZDT1, ZDT2, and ZDT3, NSGAII attains worse results than other algorithms. OMOPSO attains bad result on ZDT4. On ZDT6, MOMC and OMOPSO attain larger changes than other algorithms.

Observing the statistical analysis results in Table 5, we can know that the proposed MOMC has attained the best results on ZDT1, ZDT2, and ZDT3. It attains the second results on ZDT4, DTLZ6, DTLZ7. However, MOCell attains the best result on ZDT6. SPEA2 gets the best results on DTLZ2 and DTLZ7. OMOPSO attains the best result on DTLZ6.

6) COMPARED RESULTS IN MS

The MS metric is presented in [32], which shows the extent to which the Pareto Front obtained by the algorithms covers the true Pareto Front. An algorithm with a larger MS value is needed, and $MS = 1$ means that the true Pareto Front is completely covered by the obtained Pareto Front.

Figure shows a box plot of the MS indicator found by all selected multi-objective algorithms for all test problems. The figure shows that MOMC achieves high MS values for all test problems. On ZDT3 and DTLZ7, the MS value of MOMC is invalid. MOCell is clearly better than the results of other comparison algorithms on ZDT2,
ZDT4. Two classical multi-objective algorithms, including SPEA2 and NSGAII, can attain the similar results for all test problems.

As seen in Table 6, the proposed algorithm attains the best MS value on ZDT6.

In summary, the experimental approach we followed consists of calculating a predetermined number of function evaluations and then comparing the results obtained by considering six different quality indicators. The simulation results are analyzed for each test problem for each performance metric. By observing the simulation results, it is proved that the proposed algorithm has better performance on the ZDT series than other state-of-the-art algorithms, because the proposed algorithm obtains better ER, GD, IGD, Spread, HV and MS metric values in comparison with them. Our results reveal that many multiobjective algorithms have difficulties when facing some multi-frontal problems. This means that the structure of the membrane system can help drive the approximate solutions obtained closer to the optimal Pareto than other state-of-the-art algorithms. In addition, the results obtained by the proposed algorithm are more stable, which means that the membrane structure helps to improve the robustness of the algorithm. Therefore, it can be concluded that the proposed algorithm has promising behavior on those problems that other experimental algorithms fail.

V. CONCLUSION

This paper has proposed a new multiobjective technology based on membrane system to solve MOPs. The proposed algorithm uses membrane structures, reaction rules, and multisets in the membrane system. Multisets in different membranes work together and communicate through an external shared archive. When solving MOPs, the proposed algorithm benefits from the following three aspects. Since the membrane system is distributed, it accelerates the optimization of speed and maintains the diversity of the Pareto Front.

2) Two special symbol-objects are designed to guide the search direction of the proposed algorithm towards the global solution of MOPs. 3) As an external shared archive for storing non-dominated solutions discovered by different membranes, and sharing archive information for directing particle updates, the algorithm can use the entire search information to quickly approximate the whole Pareto Front.

The experimental results show that the proposed method is competitive in terms of performance in qualitative and quantitative measures of selected test functions with various objective functions, Pareto Front, and Pareto Set. The proposed method has the ability to find the Pareto Front, which approximates to the known Pareto Front in each test function. We can clearly see that the proposed algorithm is effective by comparing its mean and variance with the state-of-the-art algorithms. In addition, the proposed method has better value than the analytical algorithm to solve ZDT and DTLZ in the metrics of ER, GD, IGD, Spread, HV and MS.

In the future, there are several interesting areas to pursue. The parameters of MOMC are the key to the performance of the algorithm, and it is very difficult to accurately determine, for example the number of membranes, and the number of symbol-objects. In addition, evolutionary rules need to be improved to strike a balance between explorative and exploitative. The proposed algorithm will be utilized to solve the multiobjective problems in engineering applications.

ACKNOWLEDGMENT

The authors would like to appreciate editors and reviewers for providing useful comments and suggestions to improve the quality of this article.

REFERENCES

[1] M. Wu, K. Li, S. Kwong, Q. Zhang, and J. Zhang, “Learning to decompose: A paradigm for decomposition-based multiobjective optimization,” IEEE Trans. Evol. Comput., vol. 23, no. 3, pp. 376–390, Aug. 2018.
[2] K. Deb, R. Hussein, P. C. Roy, and G. Toscano-Pulido, “A taxonomy for metamodeling frameworks for evolutionary multiobjective optimization,” IEEE Trans. Evol. Comput., vol. 23, no. 1, pp. 104–116, Feb. 2019.
[3] Z. He, G. G. Yen, and Z. Yi, “Robust multiobjective optimization via evolutionary algorithms,” IEEE Trans. Evol. Comput., vol. 23, no. 2, pp. 316–330, Apr. 2019.
[4] L.-X. Wei, X. Li, R. Fan, H. Sun, and Z.-Y. Hu, “A hybrid multiobjective particle swarm optimization algorithm based on R2 indicator,” IEEE Access, vol. 6, pp. 14710–14721, 2018.
[5] G. Paun, “Computing with membranes,” J. Comput. Syst. Sci., vol. 61, no. 1, pp. 108–143, Aug. 2000.
[6] G. Paun, G. Rozenberg, and A. Salomaa, The Oxford Handbook of Membrane Computing, Oxford, U.K.: Oxford Univ. Press, 2010.
[7] G. Zhang, H. Rong, F. Neri, and M. J. Pérez-Jiménez, “An optimization spiking neural P system for approximately solving combinatorial optimization problems,” Int. J. Neural Syst., vol. 24, no. 05, 2014, Art. no. 1440006.
[8] L. Pan and G. Paun, “Spiking neural P systems: An improved normal form,” Theor. Comput. Sci., vol. 411, no. 6, pp. 906–918, 2010.

TABLE 5. SPREAD. mean and standard deviation.

| Test Function | ER | GD | IGD | Spread | HV | MS |
|---------------|----|----|-----|--------|----|----|
| ZDT1          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|
| ZDT2          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|
| ZDT3          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|
| ZDT4          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|

TABLE 6. MS. mean and standard deviation.

| Test Function | ER | GD | IGD | Spread | HV | MS |
|---------------|----|----|-----|--------|----|----|
| ZDT1          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|
| ZDT2          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|
| ZDT3          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|
| ZDT4          | 0.00| 0.00| 0.00| 0.00   | 0.00| 0.00|
[36] C. Liu et al.: Evolutionary Multi-Objective Membrane Algorithm

CHUANG LIU received the Ph.D. degree in control theory and control engineering from the Dalian University of Technology, Dalian, China, in 2014. He is currently an Associate Professor with the School of Information Engineering, Shenyang University. His current research interests include multiobjective optimization, evolutionary computation, and control of complex industrial systems.

YINCKUI DU received the Ph.D. degree from the Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, China, in 2010. He is currently an Associate Professor with the Key Laboratory of Equipment Manufacturing Comprehensive Automation, School of Information Engineering, Shenyang University. His current research interests include visual perception technology and the Internet-of-Things technology.

AO LI is currently pursuing the bachelor’s degree in the Internet of Things with Shenyang University, China. His current research interests include evolutionary computation and system optimization theory and applications.

JIAHAO LEI is currently pursuing the bachelor’s degree in the Internet of Things with Shenyang University, China. His current research interests include evolutionary computation and multiobjective optimization.