Regular paper

A Decomposition-Based Multiobjective Chemical Reaction Optimization Algorithm for Community Detection in Complex Networks

Hongye Li[1,*], Wei Gan[2]

[1] *The Faculty of Computer Science and Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710121, China*
[2] *The Faculty of Electronic Engineering, Xi'an Shiyou University, Xi'an 710065, China*

ABSTRACT

Community detection structure is very important for understanding the organization of the complex networks. This problem is NP-hard, which is modeled as a seriously nonlinear optimization problem. Recently, different intelligence algorithm has shown promising results for this problem. The chemical reaction optimization (CRO) algorithm is a novel evolutionary algorithm which mimics the phenomenon of interactions among molecules in a container. The one characteristic of CRO is that the size of the population is changing. In this paper, we redefined the operator of CRO, and using the method of multiobjective decomposition decomposed the community detection problem into a scalar of sub-problems and using the proposed a discrete variant of CRO (MODCRO) to optimization. In the proposed method, neighbor-based turbulence of on-wall ineffective collision operator and decomposition operator are redefined which is responsible for searching local exploitation ability of algorithm, and the intermolecular ineffective collisions operator and synthesis operator is also redesigned which is responsible for searching global exploration ability of algorithm. Experimental results clearly demonstrate that the proposed algorithm outperforms a number of state-of-the-art multiobjective optimization evolutionary algorithms (MOEAs) on modularity.

1. INTRODUCTION

Complex networks have utilized in many fields such as computer science, biology, physics and mathematics, which represent different types of complex systems. Many examples of complex systems include collaboration networks, communication networks, biological networks, bibliography networks, technological networks, social networks and even political election networks. These complex networks have inhomogeneous and consisted some substructures. In the substructures, there have some vertices (or nodes) and connection (or links or edges) to form different network structure. The property of the complex network attracted many researchers from different field to study, is community structure. A community is defined as a subset of nodes in a network that connection between these nodes are denser than other nodes of the rest network, which called a cluster or module. Community detection is the process of discovering the hidden community structures in complex networks, and it can be used for discovering a complex network topology structures and understanding complex functions [1]. Therefore, the detection and analysis of community structures is of great significance for investigating the organization and function of complex systems [2,3].

Generally, a community refers to a subset of nodes within the network such that connections between the nodes are denser than connection with the rest of the network. Over the last decade, many different methods have been proposed for community detection [4]. Among these algorithms, evolutionary algorithms (EAs) have shown the competitive performance in finding the communities. Therefore, lots of studies have been done based on metaheuristic methods like: genetic algorithm (GA), chemical reaction optimization (CRO), particle swarm optimization, memetic algorithm, teaching learning optimization and so on. Pizzuti [5,6] has proposed a single objective GA-net for network clustering. Gong *et al.* [7] have suggested a memetic algorithm-based network clustering method. A discrete particle swarm optimization algorithm and a clonal selection algorithm for community detection are proposed by Cai *et al.* [8,9].

All of the abovementioned approaches optimize only one objective. However, community detection not only considers one objective, but also considers other objectives, at the same time. Recently, several multiobjective community detection methods were developed by different researchers. These methods include multiobjective genetic algorithm (MOGA-Net), in which the network communities were discovered by employing a GA and the number of clusters is automatically determined [10], multiobjective EA with decomposition (MOEA/D-Net) [11]. Zou *et al.* [12] proposed

*Corresponding author. Email: lihongye8@163.com

a discrete teaching–learning-based optimization for community detection. Shadi Rahimi *et al.* [13] proposed a novel multi-objective community detection method based on a modified version of particle swarm optimization. They innovation in PSO algorithm is changing the moving strategy of particles. Zhang *et al.* [14] proposed a local information based MOEA, termed LMOEA that an individual updating strategy is suggested to improve the quality of community detection. Jiao *et al.* [15] proposed quantum mechanism-based particle swarm optimization algorithm which attempt to apply the quantum mechanism-based discrete particle swarm optimization algorithm into network clustering.

In this work, a new multiobjective method of MODCRO algorithm is developed to solve the community detection problem. In the MODCRO, using the locally based adjacency representation strategy initializes a population, firstly. In the population, a molecular represents a value of the cluster. Then using the framework of decomposition decomposes the multiobjective community detection problem into a scalar sub-problem. In the last stage, those sub-problems were optimization redefined the CRO algorithm.

The remainder of this paper is organized as follows: Preliminaries are introduced in Section 2. Section 3 describes the proposed method for community detection. Section 4 presents the tests on real-world complex networks and the experiments are conducted along with statistical tests. Conclusions are given in Section 5.

2. PRELIMINARIES

2.1. Community Detection Problem

A network N can be modeled as an undirected graph [3] $G = (V, E)$, where $V = (v_1, v_2, \ldots, v_n)$ denote the sets of nodes and $E = (e_1, e_2, \ldots, e_m)$ is a set of edges (links) connecting two elements in V. In the graph G, if node i connect with j, then $A_{ij} = 1$, else $A_{ij} = 0$. Then the graph G can descript as a network adjacency matrix $A = (A_{ij})_{nxn}$, where n is the number of vertices or nodes in the complex network. Let $V_1 \subseteq V, V_2 \subseteq V$, and $V_1 \cap V_2 = \phi, L(V_1, V_2) = \sum_{i \in v_1, j \in v_2} A_{ij}, L(V_1, V_1) = \sum_{i \in v_1, j \in v_1} A_{ij}, L\left(V_1, \overline{V}_1\right) = \sum_{i \in v_1, j \in \overline{V}_1} A_{ij}$, among them, $\overline{V}_1 = V - V_1$. Let $x = (G_1(V_1, E_1), \ldots, G_k(V_k, E_k))$, denote a partition of network G, where, V_i and E_i represent vertex set and edge set of the sub-network $G_i (i = 1, 2, \ldots, k)$, respectively. Let n represents all possible partitions of network G, then network G, then the module density (abbreviated as D) of network G under x division is defined as follows:

$$D(x) = \sum_{i=1}^{k} d(G_i) = \sum_{i=1}^{k} \frac{L(V_i, V_i)}{|V_i|} - \sum_{i=1}^{k} \frac{L(V_i, \overline{V}_i)}{|V_i|}, x \in \Omega. \quad (1)$$

The two terms in Eq. (1) respectively reflect the tightness of the connection between the modules and between the modules, and reflect the topological characteristics of the community structure. These two items will be selected here as different optimization goals.

In [11], they formulated the complex network as a multiobjective optimization problem, in which two objectives termed as negative

ratio association (*NRA*) and ratio cut (*RC*) were to be minimized. The optimization can be defined as

$$\min \begin{cases} f_1 = -\sum_{i=1}^{k} \frac{L(V_i, V_i)}{|V_i|} = NRA \\ f_2 = \sum_{i=1}^{k} \frac{L\left(V_i, \overline{V}_i\right)}{|V_i|} = RC \end{cases} . \quad (2)$$

In this paper, we used the kernel k-means (KKM) to replace NRA [11], thus, the community detection problem of the unsigned complex network is defined as follows:

$$\min \begin{cases} f_1 = 2(n-k) - \sum_{i=1}^{k} \frac{L(V_i, V_i)}{|V_i|} = KKM \\ f_2 = \sum_{i=1}^{k} \frac{L\left(V_i, \overline{V}_i\right)}{|V_i|} = RC \end{cases} . \quad (3)$$

2.2. Multiobjective Optimization

A MOP can be defined as follows:

$$\begin{cases} \min F(x) = (f_1(x), f_2(x), \ldots, f_m(x)) \\ \qquad \text{subject to} : x \in \Omega. \end{cases} \quad (4)$$

where $\Omega \subset \mathbb{R}^n$ is the decision space and $x = (x_1, x_2, \ldots, x_n) \in \Omega$ is a decision variable which represents a solution to the target MOP. $F(x) : \Omega \rightarrow \mathbb{R}^m$ denotes the m-dimensional objective vector of the solution x. The attainable objective set is defined as the set $\{F(x)|x \in \Omega\}$.

Very often, since the objectives in (4) contradict each other, there is no point in Ω minimizes all the objectives simultaneously. The answer is set of solutions that define the best tradeoff between competing objectives. The best tradeoffs among the objectives can be defined in terms of Pareto optimality. The set of Pareto optimal solutions in the objective space is called the Pareto optima front (*PF*) and the set of all Pareto-optimal objective vectors is the Pareto set (*PS*) [16]. Generally, a decision maker requires an approximation to the *PF* for having a good insight to the problem and making her final choice.

2.3. Chemical Reaction Optimization

Lam and Li first developed a new computation intelligence method, called CRO for the optimization problems. It mimics the natural phenomenon to solve the optimization problem [17,18]. The two factors that have taken the researchers interests in a chemical reaction are

- Chemical reaction always gives a more stable product with minimum energy in it.

- The chemical reaction is a stepwise searching for an optimal solution.

CRO has emerged as one of the efficient techniques for different optimal problems [19–24]. The objective function of the optimization problem is represented by potential energy in CRO. Besides,

Table 1 | Attributes of a molecule [21].

Symbol	Chemical Meaning	Mathematical Meaning
w	Molecular structure	Solution
PE	Potential energy	Objective function value
KE	Kinetic energy	Measure of tolerance of having worse solutions
NumHit	Number of hits	Current total number of moves
MinPE	Minimum value	Current optimal function value
MinHit	Minimum hit number	Move number at current optimal solution

some other terminology of the chemical reaction is represented in CRO algorithm such as molecules, kinetic energy, the number of hits, etc. The meanings of these terminology used in CRO are given in Table 1.

The basic chemical reaction algorithm has four kinds of reaction operator including on-wall ineffective collision, decomposition, inter-molecular ineffective collision and synthesis.

1. **On-wall Ineffective Collision operator of CRO**
    The on-wall ineffective collision reaction occurs when a molecule hits the wall of the container and then bounces back. After the on-wall collision, the structure of molecules will change. This collision is used to implement local search in the search space. Basically, a very small change occurs in the molecular structure of the participant molecule and thus traverses neighborhood space. If the current molecular structure is w, it turns into another state w'. The reaction process is defined by Eq. (5).

    $$w(i) \rightarrow w(i)', \qquad (5)$$

    where, $w(i)' = Neighbor(w(i))$.
    As a rule of thumb, when a molecule hits a wall, a portion of its *KE* will be lost, the lost energy is stored in the central energy buffer, when the reaction completes. Its *KE* is updated by

    $$KE_{w'} = (PE_w - PE_{w'} + KE_{w'}) \times \alpha, \qquad (6)$$

    where α is a random number that lies in between [*KELossRate*,1], where *KELossRate* is a parameter of CRO.

    If there is high *KE* processed by the molecule, then there is a possibility that *PE* could be increased depicting the worst solution. This change is desirable to make the algorithm to escape from its local minimum. The *KE* drops as an effect of collision. Its level of tolerance in getting a worst solution is lowered.

2. **Decomposition operator of CRO**
    A decomposition operator means that a molecule w hits a wall of the container and then breaks into two or more molecules. Compared with the ineffective collision, the decomposition is more vigorous and the resultant molecule structures have greater differences from that of the original one. This operator can be considered as the situation when we finish the local search from w to w_1' and w_2'. Due to the conservation of energy, w may sometimes not have enough energy to sustain its transformation into w_1' and w_2'. A certain portion of energy in buffer accumulated from on-wall ineffective collisions can be utilized to support the change.

3. **Inter-molecular Ineffective Collision operator of CRO**
    An inter-molecular ineffective collision occurs when two or more molecules collide with each other and then separate. The number of molecules involved in this collision remains unchanged after the collision. If more molecules are involved in the reaction, more energy is needed, and the structure of the molecule is also more flexible. In the original implementation in the simulation, we assume only two molecules, e.g., with molecular structures w_1 and w_2, involved. Similar to the on-wall ineffective collision, this collision is also not vigorous and the new molecular structures w_1' and w_2' are produced from their own neighborhoods separately.

4. **Synthesis operator of CRO**
    Synthesis does the opposite of decomposition. A synthesis refers to the situation when two or more molecules collide and then combine to form one new single molecule. This process implies that the search regions are expanded, i.e., diversification of solutions.

2.3.1. Energy handling

Energy can be transformed from one type to another type but all energy manipulations must follow the first law of thermodynamics that states the energy can neither be created nor destroyed. A generalized form of elementary reaction can be written as follows:

$$w_1 + \dots + w_k \rightarrow w'_1 + \dots + w'_l, \qquad (7)$$
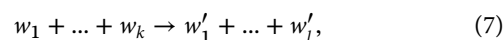
where k and l are the numbers of molecules involved before and after the reaction, respectively. For $k = 1$ and $l = 2$, the reaction can be said to be a decomposition reaction.

The corresponding energy equation can be written as follows:

$$\left(PE_{w_1} + \dots + PE_{w_k}\right) + \left(KE_{w_1} + \dots + KE_{w_k}\right) + buffer$$
$$= \left(PE_{w'_1} + \dots + PE_{w'_l}\right) + \left(KE_{w'_1} + \dots + KE_{w'_l}\right) + buffer'.$$

In the above equation, the change of the total energy of the molecules before and after the reaction is represented the left- and the right-hand side of the equality respectively.

The general acceptance rule for the new solution is given as follows:

$$\sum_{i=1}^{k} PE_{w_i} + \sum_{i=1}^{k} KE_{w_i} - \sum_{i=1}^{l} PE_{w_i} \geq 0. \qquad (8)$$

2.3.2. The sketch of CRO algorithm

As explained above, the step-wise procedure for the implementation of CRO can be summarized as follows:

In the iteration stage, one reaction from the designed reactions occurs to meet the termination condition. This is done by selecting a random variable from 0 to 1 and then checking the random variable with MoleColl. If the random variable is greater than MoleColl then an intramolecular reaction will be occurred otherwise inter-molecular reaction will be occurred. After each iteration, the *NumHit* of any particular molecule increases if the reaction is intramolecular or *NumHit* of two or more molecules increases if

the reaction is inter-molecular. Besides, the resultant molecule is checked with all other molecules to see whether the molecule is local best or not. In the finalization stage, the global best solution with the objective function value is returned as output. The pseudo code of CRO algorithm is given in Algorithm 1.

3. THE PROPOSED ALGORITHM MODCRO

In this section, the proposed MODCRO method is used to deal with community detection problem in a complex network. The proposed MODCRO algorithm is easy to implement, and the detailed description for the MODCRO algorithm is as follows:

3.1. Motivations

When a single-objective optimization algorithm solves community detection network, it only considers clustering, without considering other constraints such as degree. Therefore, single-objective algorithms may not be suitable for networks with multiple potential structures [10]. Community detection considered that the

connection within a community is distributed densely and the links between different communities are distributed sparsely. It can be formulated a two-objective function optimization problems, which one objective is minimized the inter-links. The other is maximum intra-links. However, as far as we know, there is no research of using the multiobjective CRO algorithm to detect the communities in complex networks till now. Based on these considerations, we put forward a discrete new multiobjective method of decomposition (MODCRO) for community detection in complex networks. The specific operation of the algorithm is shown in Section 3.2.3. The flow chart of the algorithm is shown in Figure 1. A detailed description of the MODCRO algorithm for community detection is shown in Algorithm 6.

3.2. MODCRO

In this section, we propose a decomposition-based multiobjective CRO algorithm for community detection in complex networks (MODCRO). The proposed MODCRO algorithm has two major contributions are proposed to solve the community detection in complex networks, and the details are listed in the following:

Algorithm 1: CRO( )

Begain:

    Step 1: Define the optimization problem and initialize the optimization parameters.

    Step 2: Initialize the population of molecules.

    Step 3: **while** (*FES< FESLimit*)

        If (rand > MoleColl) || (the number of population is equal to one)

          go to **Step4**;

        elese

          go to **Step5**;

        endif

      endwhile

    Step 4: Uni-molecular operator.

        If ($w_i.numHit$ - $w_i.minHit$ >*DecThres*)

          Decomposition operator of CRO ();

       eslse

          On-wall Ineffective Collision operator of CRO ();

       endif

      Update energy management rules ( );

    Step 5: Inter-molecular operator.

        If ($w_i.KE< SynThres$) &&($w_j.KE < SynThres$

          Synthesis operator of CRO ();

       else

          Inter-molecular Ineffective Collision operator of CRO ();

       endif

    Step 6: Termination criterion. Stop if the maximum generation number is achieved;

        otherwise repeat from **Step 3**.

  end

- Firstly, the decomposition-based multiobjective CRO algorithm (MODCRO) is introduced as a new method to optimization the community detection in complex networks, which supplied the framework of TMOEA/D to solve the practical applications with unknown Pareto frontiers.

- Secondly, the multiobjective optimization of decomposition selection operation is adopted to overcome the modularity resolution limitation problem.

3.2.1. Representation

The MODCRO use a molecule (a real integer vector) as a partition of a complex network for community detection problem. In the population, each molecule (individual) of population can be expressed as a set of real integer values. In solution, if there have some integer values of different nodes are the same value, those nodes corresponding the node or vertices are the same community; otherwise, there have some integer values of different nodes are different, those nodes corresponding the node or vertices are different community. The dimension of each molecule (individual) is the same as the number of total nodes (vertices). Therefore, a molecule (individual) vector represents a partition of the network. As mentioned above, there are N vertices with C communities in a complex network, the molecule (individual) vector X_i is defined as follows:

$$X_i = (x_{i1}, x_{i2}, ..., x_{iN}). \tag{9}$$

Each dimension of position is a random integer between 1 and N, i.e., $x_{ij} \in [1, N]$, where N is equal to the total vertices number of the network. x_{ij} represents the community that the j-th vertex of the i-th individual belongs to. If $x_{ij} = x_{ik}$, it indicates that the j-th node (vertices) and k-th node (vertices) of i-th molecule (individual) belong to the same community. Figure 1 shows the representation of a discrete molecular, Figure (a) shows the graph topology of the graph G; Figure (b) shows the community and vertex of the molecular (individual) {1 3 2 3 1 1 2 2 3}; Figure (c) shows the community.

3.2.2. Initialization

The proposed MDCRO algorithm is easy to implement. First, initialize a population with N individuals (molecules) by using a problem-specific strategy, which initialize in the way of Figure 1 until all vertices are traversed, and then a valid molecule **x** is generated. After repeating the above operation for PopSize time, a set of valid individual was got as the initial population. In the proposed MODCRO, a label propagation-based initialization strategy [25] is used to initialization the population, which can quickly reach a consensus on a unique label by collection the densely connected groups of nodes. The label propagation-based initialization strategy can reduce the searching space and the same to save the time for the algorithm.
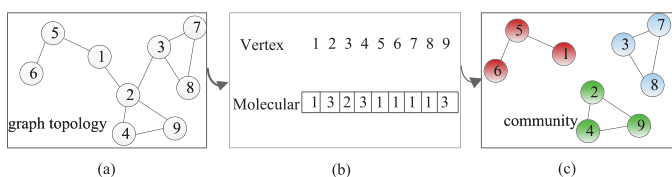
3.2.3. Reproduction procedure

MODCRO firstly decomposed the community detection problem into a set of single objective optimization problems, and then use the discrete CRO to optimize those problems at same time. The CRO has four operators: on-wall ineffective collision operator, decomposition operator, inter-molecular ineffective collisions operator and synthesis operator.

1. **On-wall Ineffective Collision operator of MODCRO**
   The on-wall ineffective collision operator and inter-molecular ineffective collisions operator are similar. The on-wall ineffective collision operator has one molecular involved but the inter-molecular ineffective collisions operator has two molecules involved. The on-wall ineffective collision operator is carried on the local search, which can be described in Algorithm 2 and figure 2. The on-wall ineffective collision operator is simple, but it does not take any guidance into account. From the Figure 1 and Section 3.1, we know that changing one vertex of individual may greatly change the division. Therefore, the procedure on-wall ineffective collision operator gives MODCRO the ability to explore other promising areas.

2. **Decomposition operator of MODCRO**
   Decomposition operator of MODCRO used the total half change method by A. Y. S. [17]. The pseudocode of the decomposition operator is shown in Algorithm 3. First, copies w_1' and w_2' come from one individual w. Then selecting $\lfloor n/2 \rfloor$ vertices of individual w_1' is randomly which can represent by $INDX_1$. For each of vertex, e.g., w_{1i}', selecting a vertex from the neighbors of i is randomly, and then assign it to w_{1i}'. Above shows the generation process of w_1', the generation process of w_2' is similar to w_1'. The Figure 3 shows the decomosition operator.

   Inter-molecular Ineffective Collision operator of MODCRO
   The Inter-molecular ineffective collision operator has two ways to search. The one way is greedy search the neighbor of molecular, and the other is searching by using on-wall ineffective collision operator.

3. **Synthesis operator of MODCRO**
   Combing two individual w_1 and w_2 into a new individual w is using the synthesis operator. The probabilistic selection strategy [17] is used to generate w. The pseudocode of the synthesis operator is shown in Algorithm 5. For each vertex w_i is coming from w, generating a random number $\eta \in [0, 1]$. If $\eta > 0.5$, w_i is replaced by w_{1i} otherwise, w_i is replaced by w_{2i}, which can show in Figure 5.

3.3. Framework of the Proposed Algorithm

In the proposed algorithm, the adopted decomposition method is widely used weight sum method, which is described for m-objective problems using a weight vector $\Lambda^i = (\lambda_{i1}, \lambda_{i2}, ..., \lambda_{im})$. The algorithm flow is shown in Figure 6.

$$\min g^{ws}\left(x^i | \Lambda^i\right) = \lambda_{i1} \cdot f_1\left(x^i\right) + \lambda_{i2} f_2\left(x^i\right) + \cdots + \lambda_{im} f_m\left(x^i\right), \tag{10}$$

(a)　　　　　　(b)　　　　　　(c)

Figure 1 | Illustration of the representation of a discrete molecular.

Algorithm 2: On-wall ineffective collision operator ( )

Input: A individual **x**

$\mathbf{x}_{temp} \leftarrow \mathbf{x}$

$\mathbf{x'} \leftarrow \mathbf{x}$

$Temp \leftarrow \varnothing$

while $| Temp | \mathrel{!=} |V|$

    Randomly select one vertex $v \in V$

    if $v \notin Temp$ **then**

      Find the individual $\mathbf{x}_{min}$ with the min KKM and RC in the neighbor of vertex v in $\mathbf{x}_{temp}$

      $Temp \leftarrow Temp \cup v$

      $\mathbf{x}_{temp} \leftarrow \mathbf{x}_{min}$

      if $PE(\mathbf{x}_{temp}) < PE(\mathbf{x'})$
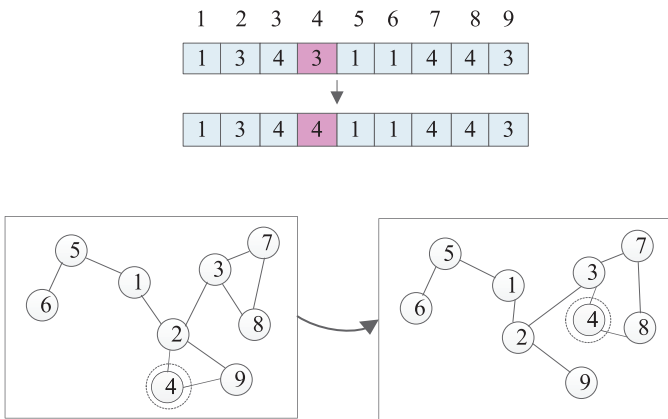
        $\mathbf{x'} \leftarrow \mathbf{x}_{temp}$
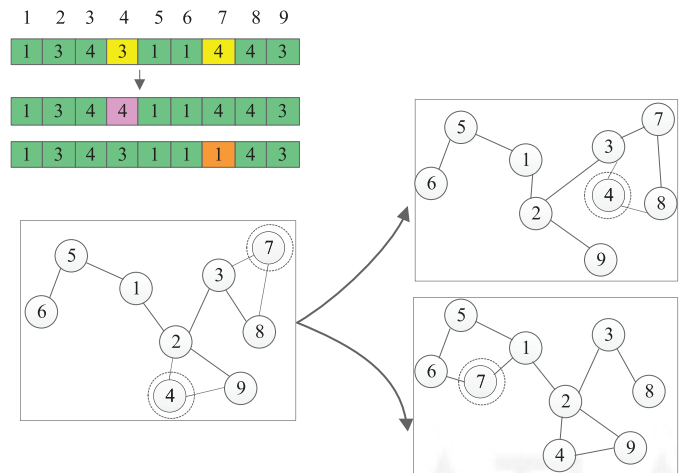
      endif

    endif

endwhile

Output: The best individual **x'**

Figure 2 | Diagram showing on-wall ineffective collision the operator.

Figure 3 | Diagram showing the decomposition operator.

Algorithm 3: Decomposition operator( )

Input: A individual w in the population P

$w_1' \leftarrow w, w_2' \leftarrow w$

Randomly get a set $INDX_1$ of size $\lfloor n/2 \rfloor$ from $\{1, \cdots, n\}$

Randomly get a set $INDX_2$ of size $\lfloor n/2 \rfloor$ from $\{1, \cdots, n\}$

for $i = 1 : \lfloor n/2 \rfloor$ do

    $j \leftarrow INDX_1[i], k \leftarrow INDX_2[i]$

    Selecting a vertex g from the neighbors of j randomly

    $w_{1j}' \leftarrow g$

    Selecting a vertex h from the neighbors of k randomly

    $w_{2j}' \leftarrow h$

endfor

Output: Two new individual w_1' and w_2'

---

where m equals five in this paper $\Lambda^i = (\lambda_{i1}, \lambda_{i2}, \dots, \lambda_{im})$ corresponds to a sub-problem i ($i = 1, 2, \dots, Q$), $\lambda_1 + \lambda_2 + \cdots + \lambda_m = 1$, and x^i is a solution to be optimized.

Many practical optimization problems have a Pareto front with an irregular shape. To approximate the Pareto optimal solutions of a multiobjective optimization problem, Q Zhang *et al.* [26] recently developed a novel multiobjective EA based on decomposition (MOEA/D). It can work well if the curve shape of the Pareto-optimal front is friendly. However, it shows poor performance about the irregular shape of the Pareto-optimal front. For this problem, HL Liu *et al.* proposed an improved MOEA/D algorithm (denoted as TMOEA/D [27]) that utilizes a monotonic increasing function to transform each individual objective function into one so that the curve shape of the nondominant solutions of the transformed multiobjective problem is close to the hyper-plane whose intercept of coordinate axes is equal to one in the original objective

---

Algorithm 4: Inter-molecular Ineffective Collision operator( )

Input: Two individual w_1 and w_2

if rand<0.5

    Selecting one vertex $i \in V$

    best← w_1

    for all neighbors j of i other than w_i **do**

       temp← w_1

       $temp_j$←j

       if $PE(temp) < PE(best)$ **then**

         best← *temp*

       endif

    endfor

else

    w'_1← *best*

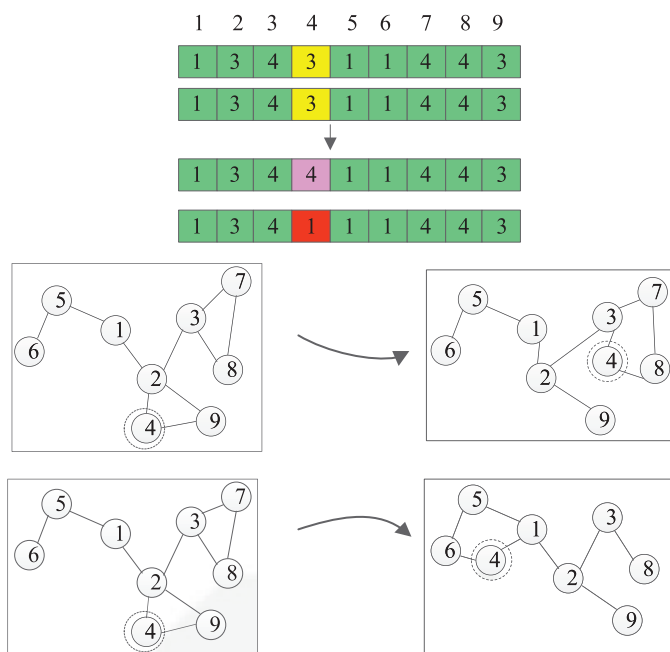   Carry on **On-wall ineffective collision operator** (w_2)

endif

Output: Two individual w'_1 and w'_2

Figure 4 | Diagram showing the inter-molecular ineffective collision operator.

function space. We consider the community detection problem as a discrete optimization problem, so TMOEA/D [27] was applied in this work.

3.4. Computational Complexity

MODCRO is a decomposition-based algorithm, where the computational complexity is O(MNT), where N is the population size, M is the number of objectives and T is the number of the weight vectors in the neighborhood of each weight vector.

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

For all experiments, 20 independent runs are carried out on the same machine with a Celeron 3.40 GHz CPU, 4GB memory and windows 7 operating systems, and conducted with the maximum number of function evaluations (*MAX_FES*) as the termination criterion. The proposed MODCRO and comparison algorithms were implemented in Microsoft Visual Studio 2015 (C++).

In this section, six real world datasets were used to evaluate the effectiveness and efficiency of the proposed MODCRO. The results of GA-Net [5], Meme-Net [7], MOGA-Net [10], MOEA/D-Net [11], MODPSO [4], QDM-PSO [15], MODTLBO/D [12] and LMOEA [14] algorithms are compared with our proposed algorithm.

4.1. Parameter Settings

The D-MOCRO algorithm was implemented in visual studio 2012. In experiments have been performed on a computer having Intel Core i5 CPU 2.67 GHz and 8GB of memory. The *MAX_FES* = 10,000, the population size $N = 100$, the neighborhood list size $T = 5$. Parameters of D-MOCRO are set as follows: The initKE equals to 10,000, the initBuffer equals to 100, the decThres equals to 800, the synThres equals to 15, the lossRate equals to 0.1, the collRate equals to 0.2. The initKE controls the molecular kinetic energy. The init-Buffer controls container total energy which plays a keeping energy conservation role. The decThres is a threshold value of carrying on decomposition operator. The synThres is a threshold value of carrying on synthetic operator. The collRate is a threshold value of carrying on whether the molecules collide with the container wall or collide with each other. All the compared algorithms stop when the number of function evaluation reaches the maximum number. The parameters of compared algorithms are listed in Table 2.

4.2. Test Data Sets

In this section, six real-world datasets are illustrated, i.e., the Zachary's karate club network [29], the dolphin social network [30], the American college football network [31], the Santa Fe Institute (SFI) network [32], the netscience network [33] and the power grid network [34]. The characteristics and parameters of the networks are given in the Table 3. In this work, each social network can be considered as undirected and unweighted.

4.3. Evaluation Metrics

In this work, the modularity (Q) and the normalized mutual information (*NMI*) are adopted to evaluate the quality of solution [36]. The modularity method is the most widely used in the community detection of complex networks. It is measured the difference between the actual fraction of edges within communities and its expected value. The modularity (Q) can be expressed as follows:

$$Q = \sum_{i=1}^{N_c} \left(\frac{l_i}{2m} - \left(\frac{d_i}{2m} \right)^2 \right), \tag{11}$$

where m is the edges, N_C is the communities, l_i is the total number of links in the community i and d_i the total degree of vertices in the community i.
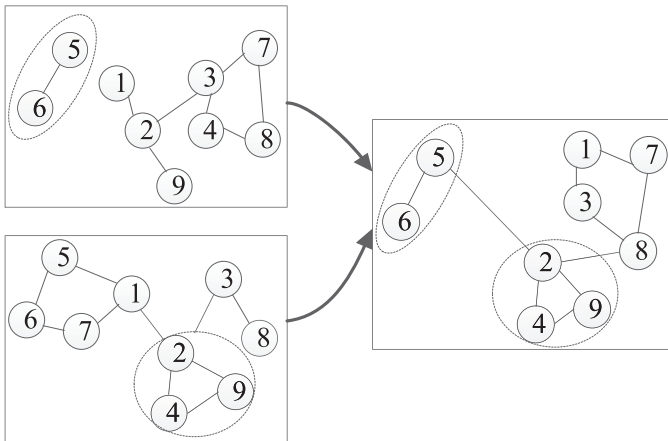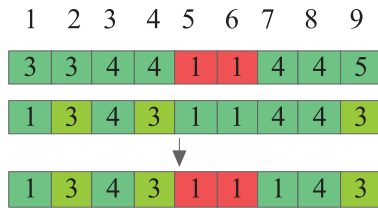
Algorithm 5: Synthesis operator( )

Input: Network $G=(V, E)$ and two individuals in the population representation w_1 and w_2

for i=1: n do

   Generation a random number $\alpha \in [0, 1]$

   if $\alpha > 0.5$ then

      $w_i \leftarrow w_{1i}$

   else

      $w_i \leftarrow w_{2i}$

   endif

endfor

Output: A new individual generated w

Figure 5 | Diagram showing the synthesis operator.

Other popular metric is normalized mutual information (NMI) [35,36], which is used to estimate the similarity between the true partition and the detected partition for a network with the known partition. It defined as follows: given two partitions A and B of a network, let C be the confusion matrix whose element C_{ij} is the number of nodes shared in common by community i in partion A and by community j in partition B. the $NMI(A, B)$ is then defined as follows:

$$NMI(A, B) = \frac{-2 \sum_{i=1}^{C_A} \sum_{j=1}^{C_B} C_{ij} \log \left(C_{ij} N / C_{i.} C_{.j} \right)}{\sum_{i=1}^{C_A} C_{i.} \log (C_{i.}/N) + \sum_{j=1}^{C_B} C_{.j} \log \left(C_{.j}/N \right)}, \quad (12)$$

where C_A and C_B are the number of clusters in the community partitions A and B. C_{ij} is the number of vertices of community i of the partition A that are also in the community j of the partition B. $C_{i.}(C_{.j})$ is the sum of elements of C in row i (column j), and N is the number of nodes of the network. The higher NMI (A, B) value represents a greater similarity between community A and community B. If $A = B$, $NMI(A, B)$ reaches the maximum value which equals to 1. If $A \neq B$, $NMI(A, B)$ reaches the minimum value which equals to 0.

4.4. Comparison of Results on the Extension of GN Benchmark Networks

In order to verify the performance of proposed MODCRO algorithm, the *GN* benchmark networks [37] are used. The *GN* benchmark networks include 128 vertices, and each vertex has an average degree 16. Those vertices of *GN* benchmark network can be divided into four clusters with the same number of nodes in every cluster. The different mixing parameter effects on the structure of *GN* benchmark networks. When the mixing parameter γ is bigger than 0.5, the structure of networks is rather sparse that means those networks has weak community structure. When the mixing parameter γ is smaller than 0.5, vertices of the networks have a tight network structure. Therefore, the range of the mixing parameter γ is changing from 0.0 to 0.5. In the experiment, using eleven synthetic networks verify the proposed algorithm, and using the *NMI* measure the similarity between the true partitions and the detected ones. The larger value of *NMI*, the better clustering performance of the algorithm archives. Figure 7 shows the average NMI values over 30

Table 2 | The parameter settings of the different compared algorithms.

Algorithm	pop	maxFES	pc	pm	ns	Reference
GA-Net	100	10000	0.9	0.1	–	[5]
Meme-Net	100	10000	0.9	0.1	–	[7]
MOGA-Net	100	10000	0.9	0.1	–	[10]
MOEA/D-Net	100	10000	0.9	0.1	40	[11]
MODPSO	100	10000	0.9	0.1	40	[4]
QDM-PSO	100	10000	0.9	0.1	–	[15]
MODTLBO/D	100	10000	0.9	0.1	40	[12]
LMOEA	100	10000	0.9	0.1	40	[14]

Table 3 | Network parameters.

Network	Node Number	Edge Number	Real Clusters	Reference
Karate	34	78	2	[29]
Dolphin	62	159	2	[30]
Football	115	613	12	[31]
SFI	118	200	unknown	[32]
Net-science	1589	2742	unknown	[33]
Power grid	4941	6594	unknown	[34]

Algorithm 6: MODCRO ( )

Input: An instance of MODCRO, the number of subpopulation N, a set of uniformly distributed direction vectors $\lambda_1, \lambda_2..., \lambda_N$, the maximal number of generations FES_{max}.

Begain

> **Step 1 Initialization**
>
> **Step 2**: Initialize the population of molecules.
>
>> Create the initial population P_0 with N random molecules, set the external elitism archive $X^* = \varnothing$;
>>
>> Using the Direction Vectors Generation Mechanism to generate N uniformly distributed direction vectors $\lambda^1, \lambda^2..., \lambda^N$,. Computing the Euclidean distances between any two weight vectors to each weight vector. For each $i = 1,..., N$, set $B(i)=\{i_1,...,i_T\}$ where $\lambda^{i_1}, \lambda^{i_2}..., \lambda^{i_T}$ are the T closest weight vectors to λ^i.
>>
>> Decomposition the original multiobjective community detection of complex networks into N single-objective community detection optimization problem $g_1,..., g_N$ with $\lambda^1, \lambda^2,..., \lambda^N$.
>
> **Step 3**: **while** (FES $\leq MaxFunEvals$) **do**
>
>> **Step 4**: **Reproduction:**
>>
>> Decompose the original multiobjective community detection of complex networks into N scalar subproblems with weight set $\lambda^1, \lambda^2,..., \lambda^N$;
>>
>> Selection of Reaction/Upadate Range: Uniformly randomly generates a number *rand* from [0, 1]. Then set
>>
>> $$P = \begin{cases} B(i) & \text{if } rand < \delta, \\ Archive(i) & \text{otherwise.} \end{cases}$$
>>
>> Selection of the number of molecular: Uniformly randomly generate a number of *rand* from [0,1].
>>
>> **If** ((*rand* > collRate) || (pop.size ()==1))
>>
>>> **If** ((subproblems (*i*). curHitIndex – subproblems(*i*).minHitIndex) > threshold)
>>>
>>>> **Decomposition** (subproblems (*i*));
>>>> $FES = FES + 2$;
>>>
>>> **else**
>>>
>>>> **On-wall** (subproblems (*i*));
>>>> $FES = FES + 1$;
>>>
>>> **endif**
>>
>> **else**
>>
>>> Selection subproblems (*j*) from the neighborhood of the subproblems (*i*).
>>>
>>> **If** (subproblems (*i*).KE < SynThres) && (subproblems (*j*).KE < SynThres)
>>>
>>>> **Synthesis**(subproblems (*i*), subproblems (*j*));
>>>> $FES = FES + 2$;
>>>
>>> **else**
>>>
>>>> **InterMolecular**(subproblems (*i*), subproblems (*j*));
>>>> $FES = FES + 2$;
>>>
>>> **endif**
>>
>> **endif**
>>
>> **Repair:** the neighborhoods of the subproblems(*i*) and the subproblems(*j*).
>>
>> **Update** of *z*: For each *j*, *i* = 1, ...,*m*, if $z_j > f_j(y)$, then set $z_j = f_j(y)$,
>>
>> **If** $z_i > f_i(y)$, **then** set $z_i = f_i(y)$.
>>
>> **Update of Solutions**: Set *count* = 0 and then do the following:
>>
>> **If** *count* = PopNum or the number of subproblems is empty
>>
>>> go to **Step 3**.
>>
>> **else** randomly pick an index *j* from the set of subproblems.
>>
>>> $Z = X \cup Y$, Sort the molecules in Z by Efficient non-dominated sort on sequential search strategy[28]
>>
>> **endif**
>
> **endwhile**

end

Output: A set of nondominated solutions X^*.
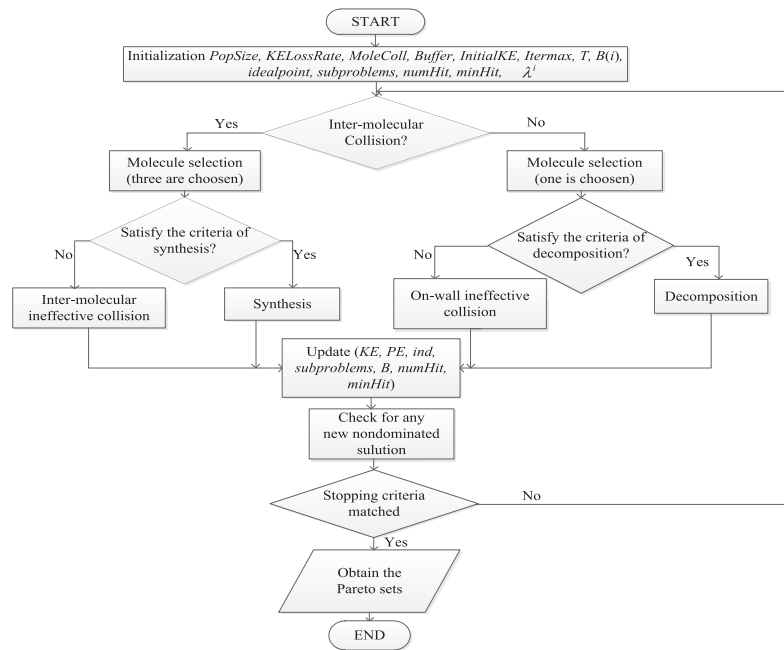

Figure 6 | Flow chart showing the working of MODCRO algorithm.

independent runs, obtained by compared algorithms for extended GN benchmark. Our proposed algorithm MODCRO has the best performance than comparing algorithms from the Figure 4. In Figure 4, MODCRO has highly robust, due to the γ is equals to 0.45, MOCRO archives high *NMI* value, 0.94.

The Table 4 shows the modularity Q of the proposed algorithm MODCRO and compared algorithms of GA-Net, Meme-Net, MOGA-Net, MOEA/D-Net, MODPSO and QDM-PSO on extensive classical *GN* benchmark networks. Those values of modularity Q are obtained over 30 independent runs. Each algorithm represents the mean value of the first line of the data, and the data in the second row brackets denote the standard deviation. In the Table 4, when the mixing parameter γ increased, the modularity values of Q are decreased. The Table 4 shows the proposed algorithm MODCRO obtained good modularity values of Q, which has good performance on mixing parameter value γ on 0, 0.10, 0.25, 0.30, 0.35, 0.40, 0.45 and 0.50. In the early stages of evolution, inter-molecular ineffective collisions operator and synthesis operator play the leading role, which can help the algorithm to explore the unknown space, in contrast, in the later evolution stage of the algorithm, the algorithm develops and utilizes the accumulated useful knowledge information, which is conducive to the maintenance of diversity and convergence of solution sets. Moreover, the on-wall effect collisions operator and decomposition operator can enhance the diversity of MODCRO algorithm to accelerate algorithm converge optimal solutions during the search. In addition, the "-" represents the corresponding algorithm cannot obtain data.

4.5. Comparison of Results on Real-World Benchmark Networks

Table 5 shows the modularity of compared algorithms and the proposed algorithm on different real-world networks. The Zachary's karate club network is a classical dataset from the literature in the field of social network analysis which is a social network of friendship between 34 members of a karate club. Every node of the network represents a member of the club. From the Table 3, we can see the Zachary's karate club network is divided into two groups. The Figure 8 (a) shows the proposed algorithm MODCRO successfully detect the clustering which is divided into two groups. The Figure 8 (b) shows the maximum modularity of karate club network by the partition in 4 communities, which obtained the highest modularity Q is equals to 0.4198. From the Table 5, we can see the MODCRO achieve the highest average value of *NMI* is equals to 1, and corresponding to the highest average of Q is 0.9498.

The Bottlenose Dolphin social network is a network with 62 bottlenose dolphins. This network is separate two large groups. The one is the female group and the other is the male one. From the Figure 9 (a) shows the true partitioning of dolphin network by MODCRO (*NMI* = 1) which obtained from Table 5 (a), and (b) shows the structure with the highest modularity value (Q = 0.5264) which can be seen from Table 5.

The American college Football network represents American football game, which was constructed by M. Girvan and M. E.J. Newman. The network concludes 115 nodes, 616 edges and 12 clusters. Every node of the network represents football team and the each edge denotes the regular season game between two teams they connect. Compared with the Zachary's karate club network and the American college Football network, the structure of American college Football network is relatively complex, the proposed algorithm MODCRO can search the best value of modularity (Q = 0.6045), the Figure 10 (a) and (b) shown the division of the network.

From the Table 3, we can see the SFI network have 118 nodes, 200 edges and the real cluster is unknown. From the Table 5, the highest modularity Q is equals to 0.7662, which is the best value of Q compared with MOGA-net, MOEA/D-net, QDM-PSO and MODPSO.
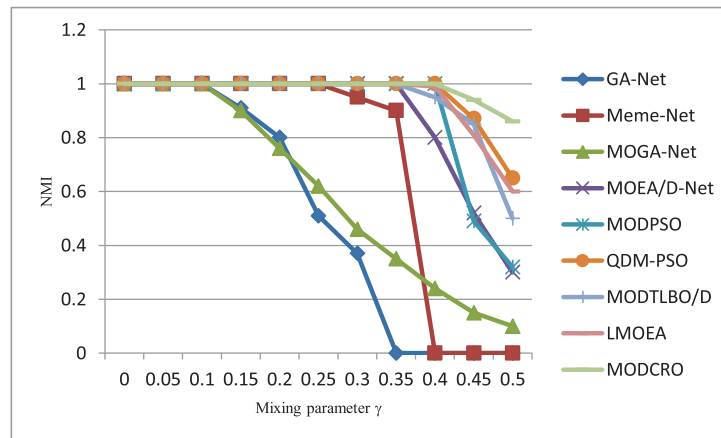
Figure 7 | Average NMI values obtained over 30 runs for the proposed MODCRO and compared algorithms on the extended GN benchmark networks.

Table 4 | Q (Modularity) obtained by seven algorithms on different eleven mixing parameters.

Mixing Parameter	0	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
GA-Net	0.6879 (0.2361)	0.6695 (0.0854)	0.6533 (0.1512)	0.5992 (0.1362)	**0.5366** (0.0110)	0.4454 (0.0156)	0.3464 (0.0208)	0.2623 (0.0172)	0.1869 (0.0102)	0.1428 (0.0076)	0.1274 (0.0050)
Meme-Net	0.7405 (0.1568)	0.6533 (0.0914)	0.5996 (0.1467)	0.5418 (0.1109)	0.4627 (0.1532)	0.3207 (0.0091)	–	–	–	–	–
MOGA-Net	0.6942 (0.0208)	0.7011 (0.0605)	0.5016 (0.1384)	0.5464 (0.0302)	0.3900 (0.0327)	0.2026 (0.0500)	0.0503 (0.0638)	0.0210 (0.0531)	–	–	–
MOEA/D-Net	0.6840 (0.0863)	0.6440 (0.0808)	0.4897 (0.2313)	0.4512 (0.2099)	0.4658 (0.1202)	0.4753 (0.0071)	0.2199 (0.0079)	–	–	–	–
MODPSO	0.7117 (0.0102)	0.65332 (0.0115)	0.5996 (0.0012)	0.5508 (0.0213)	0.5 (0.0109)	0.4502 (0.0157)	0.4023 (0.0162)	0.3496 (0.0147)	0.3105 (0.0056)	0.2564 (0.0186)	0.1426 (0.0067)
QDM-PSO	0.7415 (0.0077)	**0.6879** (0.0541)	0.6498 (0.0071)	**0.6498** (0.0071)	0.5358 (0.0418)	0.4975 (0.0036)	0.4981 (0.0041)	0.4479 (0.0030)	0.3724 (0.0329)	0.2578 (0.0412)	0.1472 (0.0234)
MODCRO	**0.7934** (0.0065)	0.6935 (0.0044)	**0.6541** (0.0067)	0.6029 (0.0046)	0.5303 (0.0079)	**0.5208** (0.0031)	**0.5016** (0.0032)	**0.4991** (0.0042)	**0.4432** (0.0069)	**0.3136** (0.0056)	**0.1993** (0.0325)

The bold values are represented the best performance value.

Table 5 | Q (Modularity) obtained by seven algorithms on six different real-world networks.

| Network	Karate	Dolphin	Football	SFI	Net-science	Power Grid
GA-Net	0.4197 (0.3601)	0.5277 (0.5028)	0.5738 (0.5275)	0.7505 (0.7486)	–	–
Meme-Net	0.4020 (0.4013)	0.5072 (0.4901)	0.5597 (0.5330)	0.6881 (0.6817)	–	–
MOGA-Net	0.3714 (0.3654)	0.3735 (0.3715)	0.5661 (0.5487)	0.7486 (0.7452)	0.8906 (0.0092)	0.6927 (0.0105)
MOEA/D-Net	0.4197 (0.4130)	0.5264 (0.5148)	0.6033 (0.5918)	0.7486 (0.7428)	0.9064 (0.0078)	0.7582 (0.0094)
MODPSO	0.4197 (0.4185)	0.5263 (0.5148)	0.6032 (0.5946)	0.7395 (0.7269)	0.9127 (0.0088)	0.8257 (0.0126)
QDM-PSO	**0.4198** (0.4146)	**0.5265** (0.5161)	0.6042 (0.5951)	0.7403 (0.7317)	–	–
MODTLBO/D	**0.4198** (0.00000)	0.52091 (0.00105)	0.60375 (0.00121)	–	–	–
LMOEA	**0.4198** (0.0000)	0.5206 (0.0010)	0.6044 (0.0000)	–	0.9370 (0.0036)	–
MODCRO	**0.4198** (0.0000)	0.5264 (0.0534)	**0.6045** (0.00034)	**0.7662** (0.0035)	**0.9443** (0.00351)	**0.8553** (0.0011)

The bold values are represented the best performance value.

The netscience network is a network of co-authorship of scientists working on network theory and experiment. In our experiment, we handle the network as an unweighted one. From the result of the Table 5, the MODCRO has the highest modularity value (*Q* = 0.9443) compared with MOGA-net, MOEA/D-net, QDM-PSO and LMOEA, which indicates the proposed MODCRO has a strong structure.

The large-scale power grid network represents the Western States Power Grid of the United States. The Table 3 shows the Power

grid network has 4941 nodes, 6594 edges and the real cluster is unknown. Every node of the Power grid network is a power base station and the edge denotes the transforming line between two stations. For this network, our algorithm obtained the highest modularity value of 0.8553 which compared with MOGA-net, MOEA/D-net and MODPSO (Table 6).

The Figure 11 shows the Pareto front of karate club and dolphin network by MODCRO.
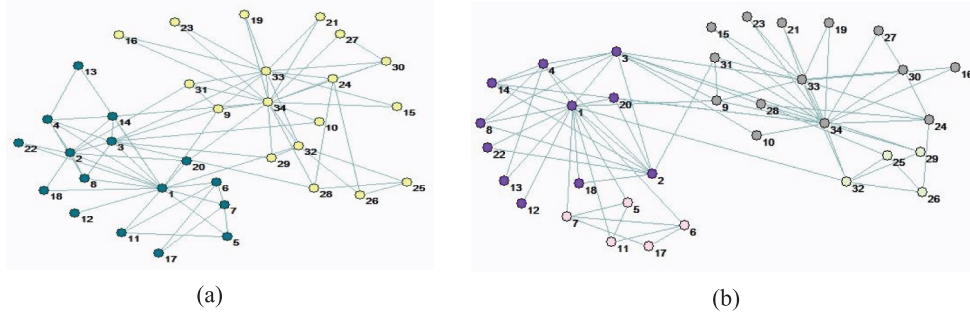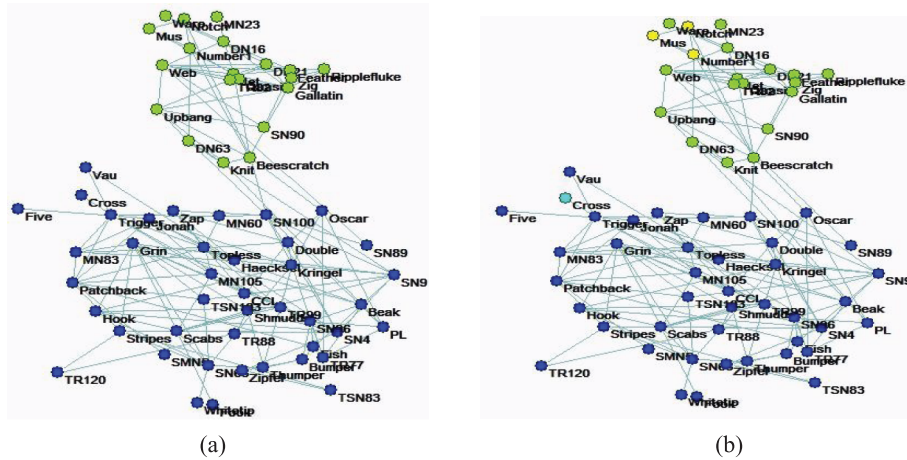
Figure 8 | Clustering results on karate club network by MODCRO. (a) Real structure detected by MODCRO. (b) Structure with highest Q value.

Table 6 | Normalized mutual information (NMI) obtained by seven algorithms on three different small-scale real-world networks.

Network	Karate	Dolphin	Football
GA-Net	0.6872 (0.6624)	0.5932 (0.5836)	0.8251 (0.7638)
Meme-Net	**1** (0.8635)	**1** (0.9449)	0.8525 (0.8500)
MOGA-Net	**1** (0.8593)	**1** (0.7853)	0.8019 (0.8011)
MOEA/D-Net	**1** (0.9455)	**1** (0.9154)	0.8930 (**0.8852**)
MODPSO	**1** (**0.9945**)	**1** (0.9152)	0.8616 (0.8566)
QDM-PSO	**1** (0.9498)	**1** (0.9160)	0.8929 (0.8834)
MODTLBO/D	**1** (0.9506)	**1** (0.9259)	0.8809 (0.8357)
LMOEA	**1** (0.9528)	**1** (0.9430)	0.8987 (0.8661)
MODCRO	**1** (0.9673)	**1** (**0.9459**)	**0.900** (0.8674)

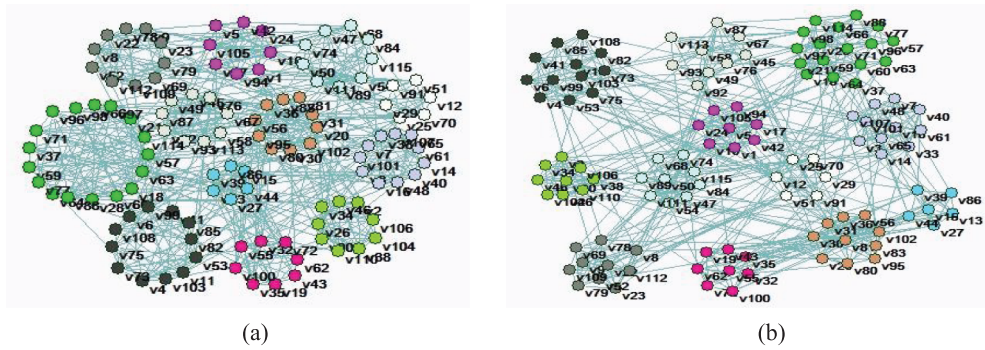The bold values are represented the best performance value.

Figure 9 | Clustering results on karate club network by MODCRO. (a) Real structure detected by MODCRO. (b) Structure with highest Q value.

5. CONCLUSION

CRO and MOCRO algorithms have been studied and used for continues and discrete optimization problems, but their applications in discrete multiobjective optimization are still at low pace. This paper first proposed a discrete multiobjective chemical reaction algorithm, which the CRO framework and the four operators of on-wall ineffective collision operator, decomposition operator, intermolecular ineffective collisions operator and synthesis operator are redefined. Moreover, based on the proposed MODCRO, a novel multiobjective discrete CRO algorithm is first presented to solve the multiobjective community detection problem in the complex network. MODCRO algorithm, the TMOEA/D framework can be used

to decomposition the multiobjective community detection problem into a scalar of sub-problems and using the redefined CRO algorithm to optimize. Neighbor-based turbulence of on-wall ineffective collision operator and decomposition operator can enhance the local exploitation ability of algorithm and the inter-molecular ineffective collisions operator and synthesis operator enhance the global exploration ability of algorithm. The multiobjective optimization about decomposition selection operation is adopted to overcome the modularity resolution limitation problem.

In order to verify the performance of MODCRO, the proposed algorithm compared with eight well-known algorithms, from the results of experiments on both synthetic and real life network, we can see

Figure 10 | Clustering results on karate club network by MODCRO. (a) Real structure detected by MODCRO. (b) Structure with highest Q value.
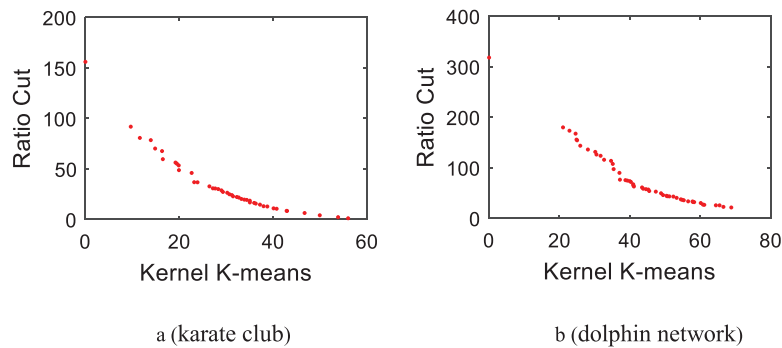
Figure 11 | Pareto front on karate club and dolphin network by MODCRO.

the MODCRO can achieve better modularity and can find community partitions close to the actual ones. It can be seen that the MODCRO is a good algorithm for solving the multiobjective community detection in the complex network problem.

AUTHORS' CONTRIBUTIONS

TMOEA/D is an algorithm framework, which can solve the unknown problems of pf frontier. For this, we use this framework to propose a multi-objective chemical reaction evolution algorithm for solving community detection problems. In addition, we have improve the multiobjective chemical reaction optimization for solving community detection in complex networks.

ACKNOWLEDGMENTS

REFERENCES

[1] M.E.J. Newman, The structure of scientific collaboration networks, Proc. Natl. Acad. Sci. USA. 98 (2001), 404–409.

[2] D. Watts, S. Strogatz, Collective dynamics of 'small-world' network, Nature. 393 (1998), 440–442.

[3] S. Fortunato, Community detection in graphs, Phys. Rep. 486 (2010), 75–174.

[4] M. Gong, Q. Cai, X. Chen, L. Ma, Complex network clustering by multiobjective discrete particle swarm optimization, IEEE Trans. Evol. Comput. 18 (2014), 82–97.

[5] C. Pizzuti, GA-Net: a genetic algorithm for community detection in social networks, in: G. Rudolph, T. Jansen, N. Beume, S. Lucas, C. Poloni (Eds.), Parallel Problem Solving from Nature, Lecture Notes in Computer Science, vol. 5199, Springer, Berlin, Heidelberg, 2008, pp. 1081–1090.

[6] H. Chang, Z. Feng, Z. Ren, Community detection using dual-Representation chemical reaction optimization, IEEE Trans. Cyber. 47 (2017), 4328–4341.

[7] M. Gong, B. Fu, L. Jiao, H. Du, Memetic algorithm for community detection in networks, Phys. Rev. E. 84 (2011), 056101.

[8] Q. Cai, M. Gong, L. Ma, S. Ruan, F. Yuan, L. Jiao, Greedy discrete particle swarm optimization for large-scale social network clustering, Inf. Sci. 316 (2015), 503–516.

[9] Q. Cai, M. Gong, B. Shen, L. Ma, L. Jiao, Discrete particle swarm optimization for identifying community structures in signed social networks, Neural Netw. 58 (2014), 4–13.

[10] C. Pizzuti, A multiobjective genetic algorithm to find communities in complex networks, IEEE Trans. Evol. Comput. 16 (2012), 418–430.

[11] M. Gong, L. Ma, Q. Zhang, L. Jiao, Community detection in networks by using multiobjective evolutionary algorithm with decomposition, Phys. A Stat. Mech. Appl. 391 (2012), 4050–4060.

[12] D. Chen, F. Zou, R. Lu, L. Yu, Z. Li, J. Wang, Multi-objective optimization of community detection using discrete teaching–learning-based optimization with decomposition, Inf. Sci. 369 (2016), 402–418.

[13] S. Rahimi, A. Abdollahpouri, P. Moradi, A multi-objective particle swarm optimization algorithm for community detection in complex networks, Swarm Evol. Comput. 39 (2018), 297–309.

[14] F. Cheng, T. Cui, Y. Su, Y. Niu, X. Zhang, A local information based multi-objective evolutionary algorithm for community detection in complex networks, Appl. Soft Comput. 69 (2018), 357–367.

[15] L. Li, L. Jiao, J. Zhao, R. Shang, M. Gong, Quantum behaved discrete multi-objective particle swarm optimization for complex network clustering, Pattern Recognit. 63 (2017), 1–14.

[16] K. Deb, Multi-objective optimization, in: E. Burke, G. Kendall (Eds.), Search Methodologies, Springer, Boston, MA, USA, 2014, pp. 403–449.

[17] A.Y. Lam, V.O. Li, Chemical reaction-inspired metaheuristic for optimization, IEEE Trans. Evol. Comput. 14 (2010), 381–399.

[18] A.Y.S. Lam, V.O.K. Li, J.J.Q. Yu, Real-coded chemical reaction optimization, IEEE Trans. Evol. Comput. 16 (2012), 339–353.

[19] J. Xu, A.Y.S. Lam, V.O.K. Li, Chemical reaction optimization for task scheduling in grid computing, IEEE Trans. Parallel Distrib. Syst. 22 (2011), 1624–1631.

[20] A.Y.S. Lam, J. Xu, V.O.K. Li, Chemical reaction optimization for population transition in peer-to-peer live streaming, in IEEE Congress on Evolutionary Computation, Barcelona, Spain, 2010, pp. 1–8.

[21] H. Li, L. Wang, X. Hei, Decomposition-based Chemical Reaction Optimization (CRO) and an extended CRO algorithms for multiobjective optimization, J. Comput. Sci. 17 (2016), 174–204.

[22] H. Li, L. Wang, X. Hei, A decomposition-based chemical reaction optimization for multi-objective vehicle routing problem for simultaneous delivery and pickup with time windows, Memetic Comput. 10 (2018), 103–120.

[23] Z. Li, Z. Li, T.T. Nguyen, S. Chen, Orthogonal chemical reaction optimization algorithm for global numerical optimization problems, Expert Syst. Appl. 42 (2015), 3242–3252.

[24] H. Duan, L. Gan, Orthogonal multiobjective chemical reaction optimization approach for the brushless DC motor design, IEEE Trans. Mag. 51 (2015), 1–7.

[25] J. Bagrow, E. Bollt, A local method for detecting communities, Phys. Rev. E. 72 (2005), 046108.

[26] Q. Zhang, H. Li, MOEA/D: a multi-objective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (2007), 712–731.

[27] H.L. Liu, F. Gu, Y. Cheung, T-MOEA/D:MOEA/D with objective transform in multi-objective problems, in 2010 International Conference of Information Science and Management Engineering (ISME), Xi'an, China, 2010, vol. 2, pp. 282–285.

[28] X. Zhang, Y. Tian, Y. Jin, A knee point-driven evolutionary algorithm formany-objective optimization, IEEE Trans. Evol. Comput. 19 (2015), 761–776.

[29] J. Duch, A. Arenas, Community detection in complex networks using extremal optimization, Phys. Rev. E. 72 (2015), 027104.

[30] D. Lusseau, K. Schneider, O.J. Boisseau, P. Haase, E. Slooten, S.M. Dawson, The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations, Behav. Ecol. Sociobiol. 54 (2003), 396–405.

[31] B.F. Auer, R.H. Bisseling, Graph coarsening and clustering on the GPU, in: Graph Partitioning and Graph Clustering, vol. 588 (2012), p. 223.

[32] J. Hardin, G. Sarkis, P.C. Urc, Network analysis with the Enron email corpus, J. Stat. Educ. 23 (2014), 1–27.

[33] V. Krebs, A network of books about recent us politics old by the online bookselleramazon.com, 2014. http://www.orgnet.com.

[34] M. Rosvall, C.T. Bergstrom, An information theoretic framework for resolving community structure in complex networks, Proc. Natl. Acad. Sci. 104 (2007), 7327–7331.

[35] R. Guider, L. Danon, A. Díaz-Guilera, F. Giralt, A. Arenas, Self-similar community structure in a network of human interactions, Phys. Rev. 68 (2004), 93–108.

[36] F. Wu, B.A. Huberman, Finding communities in linear time: a physics approach, Eur. Phys. J. B. 38 (2004), 331–338.

[37] M.E.J. Newman, Fast algorithm for detecting community structure in networks, Phys. Rev. E. 69 (2004), 066133.