Application of Algorithm used in Community Detection of Complex Network

Guoshun Wang[1], Xuan Zhang[2], Guanbo Jia[3] and Xiaoping Ren[4]

[1, 2, 3] Central South University, Changsha, Hunan, 410083, China
[4] National institute of metrology, Beijing, 100013, China
[1] wanggs@th.btbu.edu.cn, [2] xuan20010@126.com,
[3] guanbo12@126.com, [4] 176317362@qq.com

Abstract

A new algorithm named Differential Evolution Algorithm for Community Detection (DEACD) was proposed in the paper. DEACD used DE as its search engine and used the network modularity as the fitness function to search for an optimal community partition of a network. In this algorithm, there is a modified binomial crossover mechanism to transmit some important information about the community structure in evolution effectively. In addition, a biased process and clean-up operation were employed in DEACD to improve the quality of the community partitions detected in evolution. Experimental results showed that DEACD has very competitive performance compared with other state-of-the-art community detection algorithms. In the process of evolution, the colony evolution was conducted under DE scheme, the network modularity was used to evaluate the fitness of individuals in the colony. The performance of DECD was analyzed by computer generated network and real-world network examples. The algorithm was implemented using matlab Genetic Algorithm Optimization Toolbox (GAOT), and the parametric analysis was performed in the experiment.

Keywords: *Complex Network; Differential Evolution Algorithm; Community Detection*

1. Introduction

In recent years, a plenty of studies about complex network have been conducted by researchers in fields like computer science, complexity science, *etc.*, and the study on complex network has become the hot spot among interdisciplinary studies [1]. Varies of complex systems in reality can be converted into complex networks, such as social network [2], internet [3], food chain [4], metabolism mechanism and e-mail, *etc*. Complex networks are of the basic statistical propertied, including small-world property and scale-free property [5], the community structure is one of the most important aspects of complex network [6]. The well accepted conception of community is defined as a set consists of nodes that are of certain similarity to each other while there are exclusive differences between them and other nodes in the network, and the edges among such nodes. The connectivity among the nodes inside a community is dense, while the connectivity among communities is sparse. Community structures are common in real complex networks. For example, the community groups based on occupation, the study community groups based on similar research topics inside the citation nets [7].

It is common in the complex networks that the community structure with dense links amongst nodes in the same individual community while sparse links amongst nodes in the different communities, aka. The personal relation community with same occupation in the personal relationship network, the community of the same or similar research field in the

bibliography, the web page community with the same theme in the internet, and the biometric organism module network based on the similar functionality within the metabolism mechanism.[8]. The study of detection of communities in complex network is of evident theoretical and practical value to the study of topology analysis, functional analysis, and the behavior prediction about complex network, and has been applied in various fields like management of organization, detection of terrorist organization, and cross effect network of proteins. DE algorithm's benefits includes simplicity in structure, effectiveness in optimization, and strong capability in search, it has received wide concern in the field of evolution computation.

2. State of the Art of the Community Detection in Complex Network

In the past decades, the study of community partition detection algorithm in complex network has aroused wide concern, and many community partition detection algorithms have been proposed.

The traditional algorithms can be divided into 3 kinds: graph dividing, hierarchical clustering, and divide-based clustering. The graph dividing is based on iterative bisection to divide the whole network into two optimal communities. Firstly, then doing this repeatedly until we obtain enough communities. The given size of the information network will be divided into specified community number and according to the optimization objective function principle constantly adjust each node of the cluster distribution until meet the iterative conditions in divide-based clustering.

Besides, the GN algorithm was proposed by Girvan and Newman [17], in which the 'between-nesses' of all edges are calculated first according to the algorithm. The community structure of the network is obtained by iteratively removing the edge with the max between-ness and recalculating the between-ness, until all the edges are removed. This algorithm has been a standard algorithm for network community analysis, and sees uses. Due to lack of knowledge of the proper number that how many partitions to be divided, the calculation is time consuming, and it lacks a quantitative description of the community partition of the network. The shortcoming is overcame later by a network modularity function to that evaluates the quality of community structure in the network, proposed by Girvan and Newman [18].GN algorithm is one of the best-known community detection algorithms, which start the community detection on a global network structural perspective, that avoids many defects of the traditional methods.

These fore-mentioned algorithms are of high computation complexity, that they can-not be applied on large scale complex networks. And most of them require prior information about the network community partitions, such as number of community partitions. In reality, such prior information is unavailable. Plenty of community detection algorithms used for large-scale network based on evolution computation have been proposed to overcome these limitations, which are of much lower computational complexity, and are suitable to be applied on large-scale networks. Tasgin and bingol proposed a community detection algorithm based on genre computing, which optimizes the network modularity to detect the community partition. Newman proposed a new aggregation clustering algorithm utilizing the network modality function. The algorithm iterately combines any pair of nodes, and merge the pair which maximized the network modality function. Clauset [4] improved the aggregation clustering algorithm by introducing some mechanism and complex data structures. Radicchi [21] defined the concepts of strong community and weak community, and introduced a dividing algorithm on the basis of the aggregation clustering algorithm, which detect the community structure by removing the edges with smallest edge clustering factor. Duch and

Arenas [12] employed a heuristic search algorithm based on extreme optimization to detect the community structure by optimizing the network modality function.

To effectively detect the community partition in complex network, a novel Differential Evolution Algorithm for Community Detection, DEACD is proposed in this paper, utilized the Differential Evolution Algorithm, DEA. DEACD employs DEA as a search algorithm to optimize the population, and network modularity function as fitness function to evaluate each individual in the evolution. Furthermore, Pizzuti [20] proposed a detection algorithm, which defined a community score for each community in the network to evaluate the quality of the divided communities, and optimize the score using genetic algorithm. Chen [9] proposed a Immune Clone Selection Algorithm to detect communities in the network by optimizing the modality density function.

3. The Detection Algorithm for Complex Network based on DEA

Differential evolution (DE) is an evolution algorithm based on colony, which optimizes its search space regarding the difference vectors between the agents in the colony, and form new agents in the colony by randomly cross-over operation on the known optimal agents, the colony is refreshed by a competitive selection mechanism.

The colony is optimized by DE using mutate, cross-over, and selection operation in the evolution process. The colony moves to the optimal part in the search space. In the process of mutation, the difference vector of two agents is multiplied by a factor, and added to a third agent vector, obtaining a mutated vector. During the cross-over operation, DE combines the mutated vector and the preset target vector(individual agent in the colony) under certain regulation, resulting a test vector. In the selection operation, DE compares the test vector and the target one according to the cost function of the optimized problem: if the cost of a test vector is better than the one of the target vector, the target vector was replaced by the test vector. Such operation would be conducted onto every individual agent in the colony, resulted a next generation evolutional colony.

3.1. Individual Expression

In the early phase of evolution, DE creates a initial colony to establish the starting search point for the evolution optimal search

$$P_G = \{\vec{x}_1, \ldots, \vec{x}_i, \ldots \vec{x}_{NP}\}, \; i = 1, \ldots, NP$$

Where G denotes for the generation of the evolution, starting from 0. NP denotes the scale of colony, which remains static in the evolution. The i th generation individual agent $\vec{x}_i$ is denotes as:

$$\vec{x}_i = (x_{i,1}, \ldots, x_{i,j}, \ldots, x_{i,n}), \; i = 1, \ldots, NP, \; j = 1, \ldots, n$$

The jth component $x_{i,j}$ of individual agent $\vec{x}_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,n})$ in colony P0 is randomly created in the search space S under the boundary constraints according to the optimal problem to be solved.

$$x_{i,j} = Lbound_j + rand \times (Ubound_j - Lbound_j), i = 1, \ldots, NP, j = 1, \ldots, n$$

Where n is the dimension of the individual agent, Uboundj and Lboundj are the upper bound and the lower bound of xi, j respectively, rand is a random value uniformly distributed

in the interval of [0,1]

A complex network can be abstracted as a graph $G = (V, E)$, with n nodes, and the individuals can be regarded as chromosomes $\vec{x} = (x_1, x_2, ..., x_n)$ with n genes. The allelemorph gene j of each gene xi has its value {1,2,…,n}, $j \in \{1, 2, ..., n\}$ .. The gene and its allele indicate the node and the community identifier corresponding to the community, respectively. For example, $x_i = j$ means node I in the graph belongs to a community whose commID=j; and $x_i = x_d$ means node I and node d belong to the same community. During the initialization process, each node was assigned to any community randomly, to create the initial population, hence there may be n communities, for the maximum value of comm ID is n.

According to the definition of individual expression, it is very simple to handle the individuals by numbering them using community ID. Based on such expression, DECD could directly identify the community structures in the network amid the individuals of the colony, and the nodes belong to the same community. Besides, DECD could also identify the quantity of communities amid the individuals without any other decoding operation.

On the basis of such individual expression manner, the DEACD algorithm can identify all the community partitions and all the nodes belong to the same community partition from the individuals among the population.

3.2. Fittness Function

In the process of community detection in complex network, Newman and Girvan introduced a network modality function to quantitatively describe the community structure of the complex network, and to evaluate the quality of the detected community structure.

The modality is the difference subtracted from the proportion of the edges intra-linking the nodes within a community and the expectation of the proportion of the edges intra-linking within the community in a stochastic network. The creation of the stochastic network was conducted randomly creating linking edges between nodes according to the degree of nodes while keeping the community property of nodes invariant. If properly divided, the density of the edges within a community should be higher than the expectation of that in randomly created stochastic chases.

The network modality function Q is defined as: A matrix $e = (e_{ij})$ of m×m dimension is to be created with its element e_{ij} which means the proportion of the edges that linked different communities i and j among all the edges in the network. The sum of the matrix's diagonal is denoted as $Tre = \sum_i e_{ii}$, the elements e_{ii} stand for the proportion of the edges that linked nodes in the same community i internally among all the edges in the network. The sum of every column (or row) is denoted as $a_i = \sum_j e_{ij}$, it accounts for the proportion of the edges that linked with community I among all the edges in the network. The network modularity function Q can be defined as follow: [15]

$$Q = \sum_i (e_{ii} - a_i^2) = Tre - \|e^2\| \tag{1}$$

Where $\|e^2\|$ denotes the sum of elements of matrix e^2

A matrix of m×m dimension with its element

$$Q = \sum_{j=1}^{m} \left[\frac{l_j}{L} - \left(\frac{d_j}{2 \times L} \right)^2 \right]$$

(2)

If the proportion of intra-community edges is no greater than the expectation when randomly connected, the network modality will be negative. The upper bound of network modality is 1, a network modality value approximating 1 means that the divide of communities in the network is reasonable. In practice, the network modality value lies between 0.3-0.7.

Operation based on Differential Evolution Detection Algorithm

3.3.1. Initialization: According to the definition of community [5], the nodes in the same community should be inter-linked by edges so that visit from one to another could be achieved. In the idea case, the nodes should be neighbor to one-another, say, edges directly link each pair of nodes exists in the network. On such basis, the DEACD algorithm executes a biasing operation during the initialization to eliminate the un-proper community partition caused by the random individual generation method.

The detailed initialization goes as follow: after the individuals are generated randomly, some nodes are randomly chosen by the algorithm, whose commuID are copied to all their neighboring nodes to put them and their neighboring nodes in to the same community. By such a biasing operation, the size of the search set for the probable community partition created by the initialization is greatly reduced; the algorithm's speed is greatly improved. An initial population P0 with size NP is obtained by the DEACD algorithm.

3.3.2. Mutation: During the evolution, the mutation applied to each individual agent $\vec{x}_i$ （i=1,…,NP） (also called the target vector) was conducted under mutation strategy by DE, which combines the differences of several individual agents under certain rule with the agents to be mutated, producing a mutated vector $\vec{v}_i$, that improves the capability of global search while keeps the diversity of the colony.

Three different individuals $\vec{x}_{r_1}$, $\vec{x}_{r_2}$ and $\vec{x}_{r_3}$ were randomly chosen from population PG, $r_1, r_2, r_3 \in \{1,...,NP\}$. NP stands for the size of the population; G stands for the generations of evolution. Mutation is applied on the three individuals using 'rand/1' strategy, resulted a mutated vector $\vec{v}$ which is one member of the mutated population VG. Such operation is executed iteratively until the size of the mutated vector population VG reached NP.

To ensure that the colony is searching within the search space of the optimal problem during the evolution, DE check every mutated vector $\vec{v}_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,n})$ （i=1,···,NP） to verify whether its components satisfy the corresponding boundary constraints after the mutation. If the jth component of mutated vector $v_{i,j}$ （j=1,…,n） violates the boundary constraints, that $Lbound_j \leq v_{i,j} \leq Ubound_j$, DE will modify the component to ensure it a component satisfying the constraints.

$$v_{i,j} = \begin{cases} \max\{Lbound_j, 2 \times Lbound_j - v_{i,j}\}, & if\ v_{i,j} < Lbound_j \\ \min\{Ubound_j, 2 \times Ubound_j - v_{i,j}\}, & if\ v_{i,j} > Ubound_j \\ v_{i,j}, & otherwise \end{cases} \quad (3)$$

Next step is a out-of-boundary check on the individual mutated vector in the mutated vector population, to judge whether the components of vector have violated the constrain condition. If violation accurse, the components should be processed to fulfill the condition. A mutated vector population VG that fulfills the contain condition is obtained finally.

3.3.3. The crossover mechanism: A test vector $\vec{u}_i = \vec{x}_i$ ($i \in \{1,2,...,NP\}$) is utilized.

Consider the jth component of the mutated vector $\vec{v}_i$ and the test vector $\vec{u}_i$, if $rand \leq CR$ or $j = j_{rand}$, all the nodes whose commID equals vi, j are searched for according to the mutated vector $\vec{v}_i$. The commIDs of all the nodes corresponding to the fore-mentioned ones according to $\vec{u}_i$ are set as $v_{i,j}$. The manipulation moves the nodes indicated by test vector $\vec{u}_i$ to the community whose $commID = v_{i,j}$. If $rand > CR$ and $j \neq j_{rand}$, no manipulation will be performed.

3.3.4. The clean-up operation: The algorithm might obtain results that are obviously erroneous in the community partition detection processes, in which some nodes are wrongly assigned into other community. Once such wrong assignments occurred, the performance of the algorithm will be lowered, and the algorithm might easily get trapped into a local optimum, and the final resulting outcome would be a wrong community partition.

A clean-up operation is introduced to avoid this kind of error. The operation is based on the community related differences between nodes in the network. The difference of node i is defined by the ratio between the total number of communities that node I and its neighboring nodes belong to and the degree of node i, as illustrated in (4)(5). The deg(i) is the degree of node I, E accounts for the set of all the edges in the network, the commID(i) accounts for the community ID of which the node i belongs to.
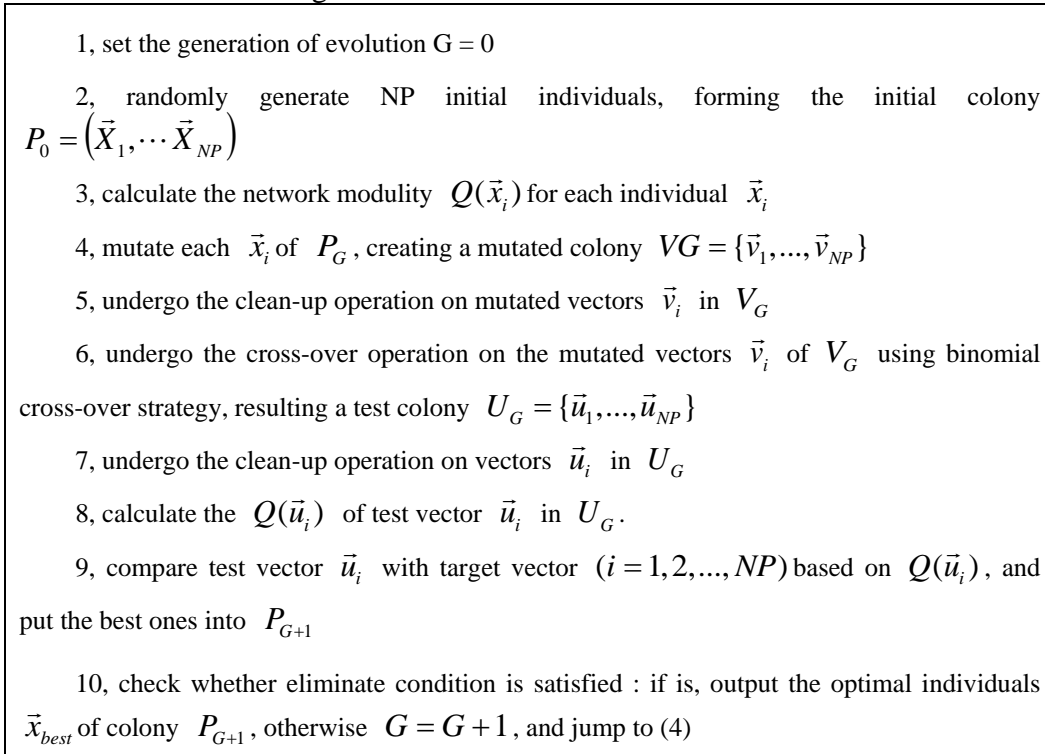
$$CV(i) = \frac{\sum_{(i,j) \in E} f(i,j)}{\deg(i)} \quad (4)$$

$$f(i,j) = \begin{cases} 1, & commID(i) \neq commID(j) \\ 0, & commID(i) = commID(j) \end{cases} \quad (5)$$

The operation carried on the individuals of the population follows the procedural: Select nodes in the network randomly, calculate the CV of each node, compare the CVs with a threshold value η: if the CV is greater than the threshold, it can be inferred that the node might be wrongly assigned to other community, in this case a survey over all the commits of the node's neighbors is conducted, the IDs of the node and its neighboring nodes will be replaced by the most frequently appeared ID; if the CV is smaller than the threshold, no further operation is needed. Using italic to emphasize a word or phrase. Do not use boldface typing or capital letters except for section headings.

According to the definition of community structure, the total amount of edges intra-linking the nodes inside a community should be greater than that of edges inter-linking nodes across communities. Hence, it is highly probable that a node with its linked neighbors belong to the same community, and a well divided community should be with nodes that have the minimal community difference.

The work flow of the algorithm is shown below:

1, set the generation of evolution G = 0

2, randomly generate NP initial individuals, forming the initial colony $P_0 = (\vec{X}_1, \cdots \vec{X}_{NP})$

3, calculate the network modulity $Q(\vec{x}_i)$ for each individual $\vec{x}_i$

4, mutate each $\vec{x}_i$ of P_G, creating a mutated colony $VG = \{\vec{v}_1, ..., \vec{v}_{NP}\}$

5, undergo the clean-up operation on mutated vectors $\vec{v}_i$ in V_G

6, undergo the cross-over operation on the mutated vectors $\vec{v}_i$ of V_G using binomial cross-over strategy, resulting a test colony $U_G = \{\vec{u}_1, ..., \vec{u}_{NP}\}$

7, undergo the clean-up operation on vectors $\vec{u}_i$ in U_G

8, calculate the $Q(\vec{u}_i)$ of test vector $\vec{u}_i$ in U_G.

9, compare test vector $\vec{u}_i$ with target vector $(i = 1, 2, ..., NP)$ based on $Q(\vec{u}_i)$, and put the best ones into P_{G+1}

10, check whether eliminate condition is satisfied : if is, output the optimal individuals $\vec{x}_{best}$ of colony P_{G+1}, otherwise $G = G + 1$, and jump to (4)

Figure 1. Community Detection Algorithm in Complex Network based on Differential Evolution

4.The Experimental Evaluation

4.1. Computer Generated Network

Network community structure detected by DEACD algorithm in the experimental process. This network was divided into four communities and each community has 32 nodes. The number of edges connected within the community between each nodes is z_{in} while the number of edges connected with the node outside the community is z_{out} which meet the condition of $z_{in} + z_{out} = 16$. The parameter setting of algorithm are as follows: population scale $NP = 200$, zoom factor $F = 0.9$, cross control parameter $CR = 0.3$, threshold $\eta = 0.35$, maximum evolution algebra is 200.

Alongside adopting network modality Q function as a criterion, another quantitative criterion was also used, the accuracy of the result of DECD community divide, to evaluate the quality of the algorithm.

$$Accuracy = \frac{\sum_{k-1}^{n} equal(l_{tk}, l_{pk})}{n} \qquad (6)$$

$$equal(x, y) = \begin{cases} 1, & if\ commID(x) = commID(y) \\ 0, & otherwise \end{cases} \qquad (7)$$

Where l_{tk} was the actual community ID in the community network, while l_{tk} was the community ID algorithm produced for node K.

We used algorithm of DEACD, GACD and GN to detect the community structure generated by computer separately. Paragraph 2 shows the proportion of those nodes which were divided correctly into the four communities accounting for the total number of nodes related to z_{out} through the three algorithms in community dividing of the network generated from computer.
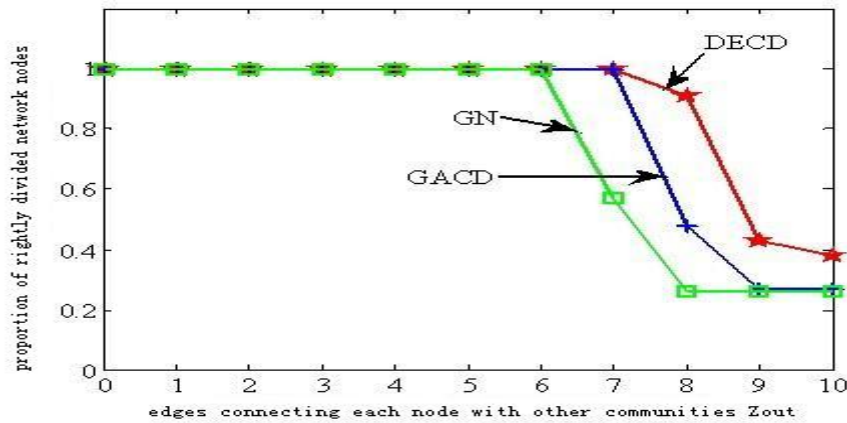
Figure 2. The Relation between the Proportion of Rightly Divided Network Node using DECD, GACD and GN Algorithms and the Edges Connecting each Node with other Communities Zout

4.2. Networks in real-world

Networks in real-world are of different ne work topology property comparing to computer generated networks, hence a further evaluation of DECD algorithm was conducted using two currently wide used real world network database (aka, the 'Karate network' and the 'Football network'). The performance of DECG algorithm was compared with those of GACD algorithm and GN algorithm. The introduction of the 'Karate network' and the 'Football network' are illustrated below.

The Karate club network [13] is built by Zachary, a social network demonstrating a karate club in an American university. This network includes 34 nodes representing members in the club, and 78 links representing the friendship among the members. A controversy arose between club manager and couch for whether to rise the charge, which lead to a split of the network into two smaller networks centering the manager and the couch respectively. The Karate club network has become a classical problem in community detection in complex network.

American football network is a complex social network built by Girvan and Newman on the basis of American college football league. The network includes 115 nodes representing

teams, and 616 edges representing games between the two teams linked by the edge. The teams belong to 12 unions, a team averagely matches 4 intra-union games and 7 inter-union games. The games within a union are more than that between any pair of particular unions, hence the union could be regarded as community within the network.

Network modality Q function is not sufficient and adequate to evaluate the quality of network community detection, as pointed out in [8, 9]. In addition to the network modality Q function, a new evaluation criterion was employed to evaluate the accuracy of the community detection of DECD algorithm

$$Accuracy = \frac{\sum_{k=1}^{n} equal(l_{tk}, l_{pk})}{n} \tag{8}$$

$$equal(x, y) = \begin{cases} 1, & if\ commID(x) = commID(y) \\ 0, & otherwise \end{cases} \tag{9}$$

5. Conclusion

In this paper we proposed an algorithm of complex network community detection based on DEA—DEACD algorithm to solve the problem of complex network community detection. We tested the property of the algorithm of DEACD by using some of the artificial network. The experimental result shows that proposed algorithm has an excellent performance on complex network community detection. Though the DECD algorithm showed excellent performance, there are still spaces for improvement. For example, local search operator could be added to better of the performance in detecting community structure in complex network. Besides, constraint handling technics could be combined to produce a newer DECD algorithm which successfully detects community structure in complex network under certain constraints.

References

[1] R. Albert, H. Jeong and A. Barabasi, "Error and attack tolerance of complex networks", Nature, vol. 406, **(2000)**, pp. 378-382.
[2] M. Faloutsos, P. Faloutsos and C. Faloutsos, "On power-law relationships of the internet topology", ACM SIGCOMM Computer Communications Review, vol. 29, no. 4, **(1999)**, pp. 251-262.
[3] R. J. Williams and N. D. Martinez, "Simple rules yield complex food webs", Nature, vol. 404, no. 6774, **(2000)**, pp. 180-183.
[4] A. Clauset, M. E. J. Newman and C. Moore, "Finding community structure in very large networks", Physical Review E, vol. 70, no. 6, **(2011)**, pp. 66-A12.
[5] L. Xiao-jia, Z. Peng, D. Zeng-ru and F. Ying, "Community Structure in Complex Networks", Complex Systems and Complexity Science, vol. 5, no. 3, **(2008)**, pp. 19-24.
[6] H. Faliang and X. Nanfeng, "Research on community detection in information network", Journal of Nanjing University of Information Science & Technology(Natural Science Edition, vol. 2, no. 1, **(2010)**, pp. 19-25.
[7] H. Dong-Xiao, Z. Xu1 and W. Zuo, "Community Mining in Complex Networks-Clustering Combination Based Genetic Algorithm", Acta Automatica Sinica, vol. 36, no. 8, **(2010)**, pp. 1160-1170.
[8] S. Fortunato and M. Barthelemy, "Resolution limit in community detection", Proceedings of the National Academy of Sciences, USA, vol. A13, **(2007)**, pp. 36-41.
[9] G. Chen, Y. Wang and Y. Yang, "Community detection in complex networks using immune clone selection algorithm", International Journal of Digital Content Technology and its Applications, vol. 5, no. 6, **(2011)**, pp. 182-189.
[10] Z. Li, S. Zhang, R. Wang, X. Zhang and L. Chen, "Quantitative function for community detection", Physical Review E, vol. 77, no. 3, **(2008)**, pp. 036109.
[11] M. E. J. Newman, "Fast algorithm for detecting community structure in networks", Physical Review E, vol. 69, **(2004)**, pp. 066133.

[12] J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization", Physical Review E, vol. 72, no. 2, **(2005)**, pp. 027104.

[13] W. W. Zachary, "An information flow model for conflict and fission in small groups", Journal of Anthropological Research, vol. 33, **(1977)**, pp. 452-473.

[14] K. Steinhaeuser and N. V. Chawla, "Identifying and evaluating community structure in complex networks", Pattern Recognition Letters, vol. 31, **(2009)**, pp. 413-421.

[15] S. Fortunato and M. Barthelemy, "Resolution limit in community detection", Proceedings of the National Academy of Sciences, USA, vol. 104, **(2007)**, pp. 36-41.

[16] Y. Wang and Z. Cai, "Constrained evolutionary optimization by means of differential evolution and improved adaptive trade-off model", Evolutionary Computation, vol. 19, no. 2, **(2011)**, pp. 249-285.

[17] J. Scott, "Social network analysis: A handbook", London, Sage Publications, 2nd edition, **(2000)**.

[18] J. M. Han, M. Kamber and J. Pei, "Data mining: Concepts and techniques", San Francisco: Morgan Kaufmann Publishers, **(2005)**.

[19] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks", Physical Review E, vol. 69, no. 2, **(2004)**, pp. 026-113.

[20] K. Price, R. Stom and J. Lampinen, "Differential evolution - A practical approach to global optimization", Berlin, Germany: Springer-Verlag, **(2005)**.

[21] J. Scott, "Social network analysis: A handbook", London, Sage Publications, 2nd edition, **(2000)**.

[22] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks", Proceedings of the National Academy of Sciences, vol. A7, no. 12, **(2002)**, pp. 7821-7826.

[23] M. Tasgin and H. Bingol, "Community detection in complex networks using genetic algorithm", Proceedings of the European Conference on Complex Systems (ECCS'06), **(2006)**.

[24] C. Pizzuti, "GA-Net: A genetic algorithm for community detection in social networks", Proceedings of the 10th International Conference on Parallel Problem Solving from Nature, **(2008)**, pp. 1081-1090.

[25] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto and D. Parisi, "Defining and identifying communities in networks", Proceedings of the National Academy of Sciences (PNAS), vol. 101, no. 9, **(2004)**, pp. 2658-2663.

[26] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks", Proceedings of the National Academy of Sciences, vol. 99, no. 12, **(2002)**, pp. 7821-7826.

[27] Y. Yang, Y. Sun, S. Pandit, N. V. Chawla and J. Han, "Is objective function the silver bullet? A case study of community detection algorithms on social networks", Social Network Analysis and Mining, Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2011), **(2011)**, pp. 394-397.

[28] J. Liu and J. Lampinen, "Adaptive parameter control of differential evolution", Proceedings of the International Mendel Conference on Soft Computing, **(2009)**, pp. 11-18.

[29] J. Brest, V. Zumer and M. S. Maucec, "Self-adaptive differential evolution algorithm in constrained real-parameter optimization", Proceedings of the Congress on Evolutionary Computation (CEC2006),Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada:IEEE Press, **(2008)**, pp. 215-222.

Authors

Guoshun Wang received his PhD in Management Science and Engineering, and he was a visiting scholar as Warwick University's the Department of Economics. His current research interests include complex networks, business efficiency theory and evaluation. He is the executive director and deputy secretary general of the system engineering association.

Xuan Zhang is a Management Science and Engineering doctoral student at present, his current research interests in complex networks and related.
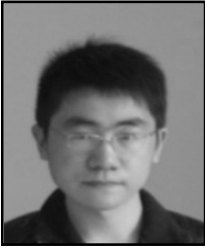
Guanbo Jia is currently now a Phd student and current research interests include statistical decision, data fusion evolutionary algorithms.

Xiaoping Ren is currently working toward the mass measurement and postdoctoral at national institute of metrology, China. He has published 18 academic papers since 2006. His current research interests include GPS/INS integrated navigation system, fault diagnosis and application of intelligent vehicle.