Detecting Community Structures in Social Networks with Particle Swarm Optimization

Yuzhong Chen[1,*] and Xiaohui Qiu[1]

[1] Fujian Provincial Key Laboratory of Networking Computing and Intelligent Information Processing, Fuzhou University, China
yzchen@fzu.edu.cn

Abstract. Community detection in social networks is usually considered as an objective optimization problem. Due to the limitation of the objective function, the global optimum cannot describe the real partition well, and it is time consuming. In this paper, a novel PSO (particle swarm optimization) algorithm based on modularity optimization for community detection in social networks is proposed. Firstly, the algorithm takes similarity-based clustering to find core areas in the network, and then a modified particle swarm optimization is performed to optimize modularity in a new constructed weighted network which is compressed from the original one, and it is equivalent to optimize modularity in the original network with some restriction. Experiments are conducted in the synthetic and four real-world networks. The experimental results show that the proposed algorithm can effectively extract the intrinsic community structures of social networks.

Keywords: Community Detection, Particle Swarm Optimization, Modularity.

1 Introduction

In recent years, community detection in social networks has attracted a lot of attention [1] [2]. Informally, communities are groups of nodes that are connected densely inside the group but connected sparely with the rest of the network. Community structure is the key feature for uncovering the global property in social networks, which is very important for studying social networks. The community can represent special role, group or a substructure of certain function. For example, communities in World Wide Web are considered as thematic clusters [3], communities in biological networks are widely believed to have a close connection to biological function [4], etc.

As an important attribute of the social networks, community detection has attracted lots of people's attention from different fields, like sociology, biology, computer science, etc. Many classic methods have been proposed to detect community structures in social networks. They can be roughly classified into two categories. The first category employ heuristic strategies, such as Girvan-Newman (GN) algorithm [5], Wu-Huberman (WH) algorithm [6] and Hyperlink Induced Topic Search (HITS)

[*] Corresponding author.

J. Su et al. (Eds.): ICoC 2013, CCIS 401, pp. 266–275, 2013.

algorithm [7] etc. The secondary category choose optimization methods or approximation methods, such as spectral method [8], Kernighan-Lin algorithm [9] and Guimera-Amaral algorithm (GA) [10] etc. In recent years, with the widely application of computational intelligence, some global optimization algorithms have been used in detecting community structure with good results. In [11], particle swarm optimization (PSO) is used to optimize modularity[12] for community detection. However, global optimization process always has high computation complexity, and resolution limit problem[13]. There are also some multi-objective optimization algorithms for community detection [14] [15], these algorithms are flexible, but it is hard to design an effective strategy that can automatically selects a proper solution from Pareto front.

In this paper, community detection is considered as an optimization problem. An algorithm named SCPSO (Similarity Based Clustering and Particle Swarm Optimization) is proposed. The rest of this paper is organized as follows. Section 2 designs a similarity clustering algorithm and then the construction of new weighted network will be introduced in section 3. Section 4 depicts an improved PSO algorithm for community detection. In section 5, experimental results in synthetic network and four real-world networks are presented and analyzed. Finally, Section 6 draws the conclusion.

2 Clustering Based on Similarity

Many algorithms can discover core clusters (dense-linked areas) of the network. For example, DBSCAN is able to discover arbitrary clusters in any database and detect noise at the same time in one scan [16]. SCAN [17] is also a structural clustering algorithm for networks based on DBSCAN. SCAN is effective and fast, but it depends on a sensitive parameter: minimum similarity threshold ε. In the proposed algorithm, we aimed at finding the core areas effectively, so a similarity based clustering method is introduced.

2.1 Basic Concepts

Here, for simplicity and without loss of generality, we only consider simple, undirected, and un-weighted networks. Let $N = (V,E)$ represents the network where V is the set of nodes and E is the set of edges. Some terms required for explaining the clustering algorithm is defined as follows [16][17].

- DEFINITION1 (NODE STRUCTURE)

The common neighborhoods of two connected nodes are important for measuring similarity. So in this paper, we define the structure of node ν as the node set including node ν and its neighborhoods , denoted by $\Gamma(\nu)$ as follows.

$$\Gamma(\nu) = \{\mu \in V \mid \{\nu, \mu\} \in E\} \cup \{\nu\} \tag{1}$$

- DEFINITION2 (NODE SIMILARITY)

Nodes in the same community share similar structure, the value of structural similarity metric will be large. The larger similarity value a pair of nodes have, the more likely they are in the same community. Here a normalized similarity function extended from Jaccard index is defined as follows.

$$sim(\mu, v) = \frac{|\Gamma(\mu) \cap \Gamma(v)|}{|\Gamma(\mu) \cup \Gamma(v)|} \tag{2}$$

- DEFINITION 3 (ε-NEIGHBORHOOD)

A minimum similarity threshold is used to be a cut to the similarity value. In other words, a node's ε-neighborhood is selected from its neighbors through threshold ε.

$$N_{\varepsilon}(v) = \{\omega \in \Gamma(v) \mid sim(v, \omega) \geq \varepsilon\} \tag{3}$$

- DEFINITION 4 (CORE NODE)

Core node represents a special node which have enough members in ε-neighborhood. Cluster(core area) are grown up from the core node. Here μ represents the minimum threshold of ε-neighborhood of the core node.

$$CORE_{\varepsilon,\mu}(v) \Leftrightarrow |N_{\varepsilon}(v)| \geq \mu \tag{4}$$

- DEFINITION 5 (DIRECT STRUCTURE REACHABILITY)

Core node is expanded to cluster(core area) according to the direct reach-ability rule formulized in the following definition. Node v is direct-connected to node ω if and only if v is a core node and ω is in the ε-neighborhood of v.

$$DirREACH_{\varepsilon,\mu}(v, \omega) \Leftrightarrow CORE_{\varepsilon,\mu}(v) \wedge \omega \in N_{\varepsilon}(v) \tag{5}$$

2.2 Clustering Algorithm

In this sub-section, a basic structural clustering algorithm that searches for core areas and isolated nodes in a network is discussed.

Firstly, all nodes are labeled as unclassified. For each node v that is unclassified, if node v is a core node, a new cluster_ID will be generated and all nodes which satisfy direct reach-ability rule will be inserted into a seed queue, otherwise node v will be labeled as NOISE which means isolated node. Moreover, the cluster_ID will be assigned to all the nodes appeared in the seed queue.

Secondly, node y is pick up from the top of the queue. If node y is a core node, its unclassified neighborhood which satisfy the direct reach-ability rule will be added to the queue. Then remove node y from the queue. The operation will be repeated until the queue is empty.

Finally, the network is partitioned into some clusters(core areas) and isolated nodes. The pseudocode of the algorithm is depicted as follows.

Clustering Algorithm

```
assign all nodes as unclassified;
for each unclassified node $v$
  if ($CORE_{\varepsilon,\mu}(v)$) then
       generate new cluster_ID;
       insert $DirREACH_{\varepsilon,\mu}(v)$  into queue Q;
       while(Q != 0) do
           y = first node in Q;
           assign cluster_ID to y;
           if ($CORE_{\varepsilon,\mu}$(y)) then
               for each $x \in DirREACH_{\varepsilon,\mu}(y)$ do
                   if x is unclassified  then
                      insert  x into queue Q;
                  if x is NOISE then
                     assign cluster_ID to x;
        remove y from Q;
  else
     label v as NOISE;
```

3 Constructing New Weighted Network

After clustering process, now the original network consists of some core areas and isolated nodes. Meanwhile, in order to reduce the network scale, a new compressed weighted network will be constructed by abstracting each core area and isolated node in the original network as a super node in the new weighted network. For edges that are in the same core area of the original network, a self-join edge will be added to the corresponding super node in the new network. And for edges that are between core areas or isolated nodes in the original network, an edge between the corresponding super nodes will be added. Finally, a new weighted network is constructed. Fig.1 shows the conversion. Then the optimization based on modularity will be performed on the new constructed network which has a smaller scale than the original one.

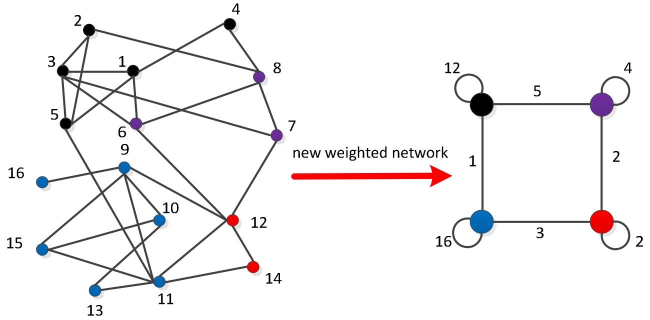

Fig. 1. Conversion from the original network to a new weighted network

Calculating modularity [12] in the new weighted network plays a significant role in community detection. Modularity has high computation complexity and is highly dependent on the scale of network. The optimization efficiency will be improved if we can prove that searching modularity optimum in the new constructed network and the original one is equivalent.

The equivalence here means optimization modularity in the new constructed network equals optimization in the original one with fixed combination of core areas. In another word, core area will not split when applying random optimization method. The proof of the equivalence of optimization modularity is presented briefly in the following paragraph.

Let G denotes the original network and A=$(A_{ij})_{n\times n}$ denotes the adjacency matrix of G where A_{ij} is the weight of the edge from node i to j. $k_i = \sum_{ij} A_{ij}$ is the degree of node i, and $m = \frac{1}{2}\sum_{ij} A_{ij}$ is the total of the edge weight of G, c_i is the identifier of the community that node i belongs to in certain iteration. If node i and node j are in the same community, $\delta(c_i, c_j) = 1$, otherwise 0. Q denotes the modularity of G.

$$Q = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)\delta(c_i, c_j), 1 \le i, j \le n \quad \text{s} \tag{6}$$

Since all the edges in G are kept in the new constructed network. According to the definition of modularity, it is easy to find out that the modularity of the new constructed network equals Q. Therefore, searching modularity optimum in the new constructed network is equivalent to searching in the original one.

4 Modularity Optimization

Particle Swarm Optimization (PSO) is a computational intelligence algorithm proposed by Kennedy and Eberhart in 1995 [18]. It is a swarm intelligence algorithm that simulates the movements of a flock of birds which seek food. Its relative simplicity and fast convergence have made it a popular optimization method in many research fields including community detection [11].

Fitness Function

Each particle represents a potential community structure of the network. Modularity which is a popular evaluation index for community detection is chosen as the fitness function. It is based on the intuitive idea that random networks do not have community structure, a good division into communities should have a high value of modularity[12]. PSO will select the particle with the maximum modularity as the best solution.

Particle Encoding

A particle encoding scheme based on local neighbor list is adopted in the proposed algorithm. Such a particle encoding scheme does not require apriori knowledge of the number of communities.

Fig.2 shows an example of the particle encoding scheme based on the local neighbor list. Fig.2(a) is the topology of a constructed weighted network obtained by the framework, Fig.2(b) shows one possible particle encoding based on the local neighbor list. For a particle $P_i = (P_{i,1}, P_{i,2},.., P_{i,n})$, if $P_{i,k} = m$, it means particle i represents that V_k and V_m are in the same community while m is chosen from the neighborhoods of particular k. Fig.2 (c) reveals how to convert a particle encoding into the community structure. Fig.2 (d) shows the community structure relevant to the particle encoding in Fig.2 (b).

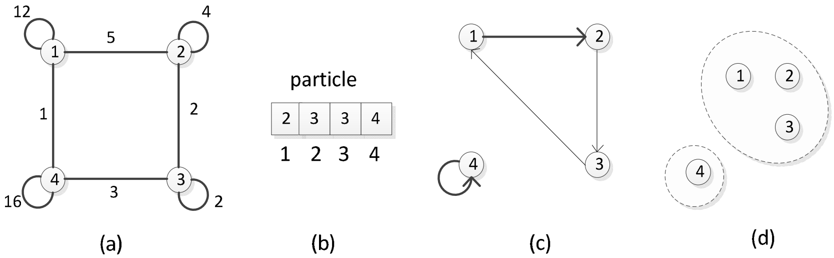

Fig. 2. Illustration of particle encoding based on neighbor list

Update Strategy

A potential solution is represented as a particle, each particle adjusts its trajectory based on the activities of its neighbor or the whole population. In each iteration, the particle adjusts velocity by the following formula (7) where $x_{ij}^{(p)}$ is the personal best position, $x_{ij}^{(g)}$ is the global best position of swarms, ω is the inertia weight, c_1 and c_2 are learning factors, r_1 and r_2 are random values in the region [0,1].

The update strategy in this paper refers to the update strategy of PSO in continuous space optimization [18]. However, an additional modular operation is added to ensure $x_{ij}(t)$ lies within $[0, \deg(v_j)]$, $x_{ij}(t)$ rounded upwards to the nearest integer. The resulting change in position is defined by formula (8).

$$v_{ij}(t+1) = w v_{ij}(t) + c_1 r_1 (x_{ij}^{(p)} - x_{ij}(t)) + c_2 r_2 (x_{ij}^{(g)}(t) - x_{ij}(t)) \tag{7}$$

$$x_{ij}(t+1) = (x_{ij}(t) + v_{ij}(t+1)) \operatorname{mod}\deg(v_j) \tag{8}$$

Algorithm Description

The detail procedure of the optimization algorithm is shown as follows:

Step1. Build the local-neighbor-list based on the new constructed weighted network.

Step2.Set the parameters, and initiate each particle with a random value selected based on the local-neighbor-list.

Step3. For each particle, calculates its current fitness, copies its current position (fitness) to its local best position (fitness).

Step4. Perform the update strategy to each particle, if the fitness is better than its local best; update its local best,

Step5. Select the global best from the local best of each particle. If the fitness is better than the current global value, update current global value $gbest$.

Step6. If the stop condition is met, the community structure relevant to the global best particle is selected as the best partition of the network, otherwise go to step 4.

5 Experimental Results

In this section, extensive experiments are carried out to validate the proposed algorithm SCPSO(Similarity Based Clustering and Particle Swarm Optimization). The experimental results of SCPSO are compared with several classical methods (GN [5] and MOGA-Net [14]) on both synthetic and real world networks. In the clustering process, the minimum similarity threshold ε is set to 0.5 and the minimum ε-neighborhoods of core node is set to 2.In the optimization process, the mutation rate is set to 0.1.

- Evaluation metric

In order to compare the performance of different solutions, two evaluation metrics are introduced.

One metric adopted in the experiments is Normalized Mutual Information (NMI) [19]. NMI is always used to calculate the similarity between two partitions. NMI is defined as follows. A higher NMI value represents a greater similarity between partition A and partition B. When partition A and partition B are exactly the same, NMI will reach the max value of 1.

$$NMI\left(A,B\right) = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} C_{ij} \log\left(C_{ij}N / C_i C_j\right)}{\sum_{i=1}^{C_A} C_i \log\left(C_i / N\right) + \sum_{j=1}^{C_B} C_j \log\left(C_j / N\right)} \tag{9}$$

The other metric adopted in the experiments is modularity Q. Modularity Q is always used in estimating the quality of community structure discovered by different solutions if the community structure of a network is unknown. The community partition with a larger modularity usually indicates a better solution. Modularity and NMI are two commonly used metrics for evaluating the quality of community structure.

- Experimental results and analysis on synthetic network

The synthetic network is a benchmark proposed by Girvan and Newman [5]. The network consists of 128 nodes and is divided into four equally-sized communities, each with 32 nodes. Each node has z_{out} links to nodes of other communities, z_{in} links to nodes in the same community, and has an average degree of 16, namely

$Z_{out} + Z_{in} = 16$. When $Z_{out} < Z_{in}$, the network generated have relatively clear community structure while the community structure becomes obscure when $Z_{out} \geq 8$. The benchmark networks are generated with Z_{out} varying from 0 to 8, each Z_{out} with 50 networks.

Fig.3 shows the distribution of NMI results of the three algorithms averaged over 100 runs for Z_{out} ranging from 0 to 8. The difference in the three algorithms increases when Z_{out} grows. Random optimization method like MOGA-NET become instable with the increase of Z_{out}. The performance of GN drop rapidly when $Z_{out} > 5$. On the contrary, SCPSO can always detect the community structure effectively when Z_{out}<=6, the NMI value is close to 1.0. When Z_{out}> 6 and even Z_{out}=8, the NMI value achieved by SCPSO is still close to 0.6. It indicates that SCPSO can still find out some high quality community structure even when the network structure becomes obscure.

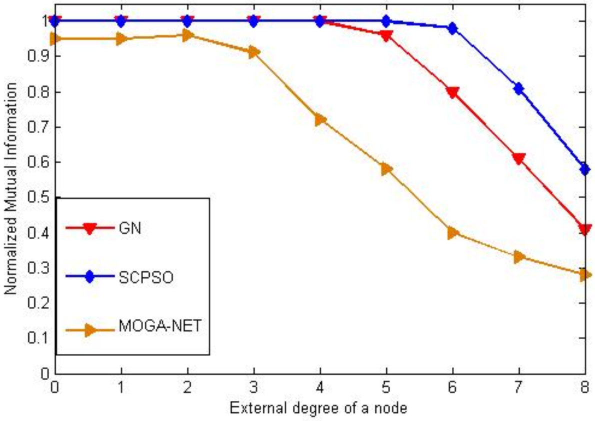

Fig. 3. The comparison of NMI in synthetic network

Experimental Results and Analysis on Real Networks

Four well studied real-world networks whose community structures are known in prior, including Zachary's Karate Club [20], Bottlenose Dolphins [21], the American College Football [5], and the Krebs' books on American politics [22], are selected as benchmark networks to verify the performance of SCPSO.

In the experiments on real networks, we run MOGA-NET 100 times over each real network and calculate the average value of modularity Q and NMI, since MOGA-NET are random optimization algorithms and the result of each run may be different. In addition, MOGA-NET is also a multi-objective optimization algorithm and returns a set of solutions called Pareto front. For the convenience of comparison, the solution with max modularity Q is selected as the single recommendation solution from the solution set of MOGA-NET and the corresponding Q and NMI are chosen as the final result.

Table1 illustrates the experimental results of three candidate algorithms. It is clear to find out that SCPSO outperforms the other three algorithms in most cases. In Karate network, SCPSO gets a little worse modularity Q than MOGA-NET. The reason is that clustering process in the framework has combine some nodes, SCPSO may not retrieval the global best modularity compared to random optimize in the original network. If splitting some core area increases the modularity index, SCPSO gets lower modularity index compared to optimize in origin network. In Dolphins, Krebs and Football, SCPSO outperforms its competitors in both modularity and NMI.

Table 1. Comparison of Modularity in Real Networks

	MODULARITY COMPARISON			**NMI COMPARISION**		
	SCPSO	**MOGA-NET**	**GN**	**SCPSO**	**MOGA-NET**	**GN**
Karate	0.400	0.415	0.380	0.803	0.602	0.692
Dolphins	0.528	0.505	0.495	0.581	0.506	0.573
Krebs	0.521	0.518	0.502	0.549	0.536	0.530
Football	0.617	0.515	0.577	0.801	0.775	0.762

6 Conclusion

Our goal is to reduce the scale of network and accelerate the convergence in optimization process. In addition, optimization in the new constructed network has shown its advantage, and the rationality has been briefly proof. In the optimization process, a mutation strategy had been proposed to accelerate the convergence. In comparison with GN, MOGA-Net in synthetic and four real networks, SCPSO exhibits its advantage. Therefore, the proposed algorithm SCPSO is an effective optimization algorithm in community detection. Expanding the algorithm to dynamic networks is our next job.

Acknowledgment. The authors would like to thank the support of the Technology Innovation Platform Project of Fujian Province under Grant No. 2009J10027, the Key Project of Fujian Education Committee under Grant No. JK2012003, the Program of National Natural Science Foundation of China under Grant No. 60171 009.

References

1. Fortunato, S.: Community detection in graphs. Physics Reports 486, 75–174 (2010)
2. Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D.: Defining and identifying communities in networks. Proceedings of the National Academy of Sciences of the United States of America 101, 2658–2663 (2004)
3. Flake, G.W., Lawrence, S., Giles, C.L., Coetzee, F.M.: Self-organization and identification of web communities. Computer 35, 66–70 (2002)
4. Ravasz, E., Somera, A.L., Mongru, D.A., Oltvai, Z.N., Barabási, A.-L.: Hierarchical organization of modularity in metabolic networks. Science 297, 1551–1555 (2002)

5. Girvan, M., Newman, M.E.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences 99, 7821–7826 (2002)
6. Wu, F., Huberman, B.A.: Finding communities in linear time: a physics approach. The European Physical Journal B-Condensed Matter and Complex Systems 38, 331–338 (2004)
7. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM) 46, 604–632 (1999)
8. Smyth, S., White, S.: A spectral clustering approach to finding communities in graphs. In: Proceedings of the 5th SIAM International Conference on Data Mining, pp. 76–84 (2005)
9. Newman, M.E.: Detecting community structure in networks. The European Physical Journal B-Condensed Matter and Complex Systems 38, 321–330 (2004)
10. Guimera, R., Amaral, L.A.N.: Functional cartography of complex metabolic network. Nature 433, 895–900 (2005)
11. Xiaodong, D., Cunrui, W., Xiangdong, L., Yanping, L.: Web community detection model using particle swarm optimization. In: IEEE Congress on Evolutionary Computation, CEC 2008, IEEE World Congress on Computational Intelligence, pp. 1074–1079 (2008)
12. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E 69, 026113 (2004)
13. Fortunato, S., Barthelemy, M.: Resolution limit in community detection. Proceedings of the National Academy of Sciences 104, 36–41 (2007)
14. Pizzuti, C.: A multi-objective genetic algorithm for community detection in networks. In: 21st International Conference on Tools with Artificial Intelligence, ICTAI 2009, pp. 379–386 (2009)
15. Shi, C., Yan, Z., Cai, Y., Wu, B.: Multi-objective community detection in complex networks. Applied Soft Computing 12, 850–859 (2012)
16. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD, pp. 226–231 (1996)
17. Xu, X., Yuruk, N., Feng, Z., Schweiger, T.A.: SCAN: a structural clustering algorithm for networks. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 824–833 (2007)
18. Kennedy, J.: Particle swarm optimization. In: Encyclopedia of Machine Learning, pp. 760–766. Springer (2010)
19. Danon, L., Diaz-Guilera, A., Duch, J., Arenas, A.: Comparing community structure identification. Journal of Statistical Mechanics: Theory and Experiment, 09008 (2005)
20. Zachary, W.W.: An information flow model for conflict and fission in small groups. Journal of Anthropological Research, 452–473 (1977)
21. Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M.: The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology 54, 396–405 (2003)
22. Krebs, V.: Unpublished, http://www.orgnet.com/