Journal of Mathematics and Informatics
Vol. 1, 2013-14, 43-51
ISSN: 2349-0632 (P), 2349-0640 (online)
Published on 1 October 2013
www.researchmathsci.org

Journal of
Mathematics and Informatics

Detecting communities of directed networks via a local algorithm

Shigang Liu

School of Information Technology, Deakin University
Melbourne, Australia. Email: shigang@deakin.edu.au

Received 30 June 2013; accepted 16 July 2013

Abstract. Targeted to the division of community structure in directed networks, this paper gives a definition of the node strength. Based on the principle of the node strength, the division of the local community structure algorithm is proposed in directed and overlapping networks. The basic idea is to find an initial community from a node with maximal node strength and to expand the community by adding nodes. The algorithm only requires local information of nodes used, so the time complexity is very low, reaching a linear complexity. Finally, the algorithm is applied to the classic Zachary network to verify the validity of the algorithm.

Keywords: Directed network, Overlapping community, Node strength

1. Introduction

Various complex systems are described as graphs and networks [1-3] in nature. With the development of network in the field of physics and mathematics, the important property of the social network has been recognized—community structure [4]. That is to say, the whole network is made up of several groups and every vertex of the related group is linked to others tightly, while the link between group and group is very sparse. Therefore, research about complex network plays an important role in realizing that structure of the social network and analysis its property. Also community structure analysis is widely used in management, sociology, biology, physics and computer science and related area[5].

Research about how to divide communities among complex network has been taken for long times. Meanwhile researchers have proposed many different algorithms toward various group structures, in order to divide the communities into certain kind of groups: (1) Eigenvalues spectral bisection algorithm of Laplacian matrix based on graphs[6]; (2)K-L algorithms based on greedy algorithm[7]. In recent years, mang algorithm about dividing communities have been invited based on modularity[8]. In 2004, Newman proposed a fast algorithm that for dividing communities[9], which is based on modularity and have a good result in dividing networks, especially for Sparse Network. Currently, how to find out the local community and its vertices is becoming a hot topic [10]. However, there doesn't exist any kind of communities that separated to each other. It is different to classify the network into several separated part, because they have connections. Therefore, Palla et al. presented a clique percolation algorithm[11] which is good at dividing overlap networks.

Aimed at the community structure partitioning problem of the directed network, this paper first described the definition of node strength, neighbors. We will use the local community divide method into the directed network based on node strength. The main thought of this algorithm is begin with the largest node strength and then find an initial community, after that expanded the community by adding nodes. Finally, we use this approach into the classical Zachary network; the result shows this method good at processing overlapped community problems.

This paper is structured as follows. In section 2, we introduce several basic function and modularity function related to community structure partition. In section 3, we presented a local community partition algorithm based on node strength. In section 4 is devoted to an example that uses the classical Zachary network, in order to show the effectiveness of the proposed algorithm. Section 5 concludes.

2. Definitions and Modularity Function

2.1. Definitions

First of all, we will introduce some basic knowledge of network, suppose $G = G(V, E)$ is an undirected graph where $V = \{v_1, v_2, \cdots, v_n\}$ is vertex set and $E = \{(v_i, v_j) | v_i \in V, v_j \in V\}$ the set of directed edges. $d(v)$ is the degree of v in graph G, which is also regard as the neighbors of v. Suppose $D = (V(D), E(D))$ is a directed graph and $V(D) = \{v_1, v_2 \cdots, v_n\}$ is a vertex set. $v_i, v_j \in V(D)$ and $(v_i, v_j) \in E$. A directed edge from a vertex v_i to a vertex v_j is denoted by (v_i, v_j). Therefore, (v_i, v_j) and (v_j, v_i) are to different edges. We say $d_{in}(v)$ and $d_{out}(v)$ denote the in-degree and out-degree of v in D. In the following part some definition about dividing the community structure of directed graph will be presented.

Definition 1. Suppose $D = (V(D), E(D))$ is a directed graph, vertices $v_i, v_j \in V(D)$, where v_i and v_j are neighbors if there is an edge from v_i to v_j.

For example, from figure 1 we can see there is an edge from v_5 to v_3, so v_5 is the neighbor of v_3.

Definition 2. Suppose $D = (V(D), E(D))$ is a directed graph, $v \in V(D)$, the note strength of vertex v is the number of all the neighbors of v, which is the in-degree of v note as $d_{in}(v)$.

For example, let's look at figure 1, the note strength of vertex v_4 is 3, the note strength of vertex v_6 is 0 etc.. To be note that if a network is a undirected graph, we say that the note strength of every vertex is equal to there degree.

Definition 3. Suppose $D=\left(V\left(D\right),E\left(D\right)\right)$ is a directed graph, $v\in V\left(D\right)$, where the compactness of vertex v and community c is described as follows:

$$B\left(v,c\right)=\frac{d_{out}^{c}\left(v\right)+1}{d_{out}\left(v\right)+1} \tag{1}$$

An example can be given from Figure 1, if a community c includes vertices $\{v_1,v_2,v_3,v_4\}$, the compactness of vertex v_5 and community c is $B(v_5,c)=\frac{2+1}{3+1}=0.75$; similarly, $B(v_6,c)=\frac{2+1}{4+1}=0.6$, $B(v_8,c)=\frac{0+1}{3+1}=0.25$.

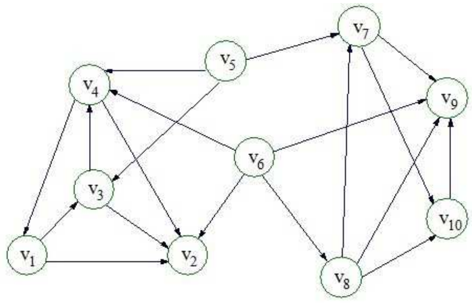

Figure 1: A directed network

2.2. Modularity Function

Girban and Newman have given the definition of modularity function in their papers, which is as follows:

$$Q=\sum_{i=1}^{k}(e_{ii}-a_i^2) \tag{2}$$

Note that e_{ii} is the percentage of the edges that connect to community c_i divide all the edges, a_i is the percentage of the edges that connect communities divide all the edges of the graph. We can generate a network by the related definition, and each community is not changed, the edges of vertex and vertex can join to each other freely, where the function Q is used to describe the level of divided community.

When the communities are known in the graph, c_v is noted as the community of vertex v. The proportion of the edges that belongs to the community among all the edges in the network is:

$$\frac{\sum_{uv}A_{uv}\left(c_u,c_v\right)}{\sum_{uv}A_{uv}}=\frac{1}{2m}\sum_{uv}A_{uv}\delta_{c_u,c_v} \tag{3}$$

where A_{uv} is a matrix that contains elements from network. For example, if there is an edge between u and v, then $\delta_{c_u,c_v}=1$, otherwise $\delta_{c_u,c_v}=0$; m is the number of edges in the network. In random network, the probability between vertices u and v can be expressed as $\frac{d(u)d(v)}{2m}$, where $d(v)$ is the degree of vertex v. Therefore the expression of the modularity function is:

$$Q=\frac{1}{2m}\sum_{uv}\left[A_{uv}-\frac{d(u)d(v)}{2m}\right]\delta_{c_u,c_v} \tag{4}$$

Now we will talk about the modularity function in directed network [12]. Note that the community structure in directed network problems is also connected with the proportion of the related edges in all the edges and the proportion between communities. However, the difference is that the direction should be considered and the position of an edge is related to its direction. For example, there are vertices A,B, where vertex A has a high out-degree and low in-degree, meanwhile vertex B is opposite. That is to say, the direction between A and B is more likely to be A to B, rather than B to A. Therefore, if there is an edge from B to A, we should pay more attention. In other words, if we know all the in-degree and out-degree of the whole network and their connection structure, then the related structure model can be obtained. The probability from u to v is $\frac{d_{in}(v)d_{out}(u)}{m}$, where $d_{in}(v)$ and $d_{out}(u)$ denote the in-degree and out-degree of vertex v and u, A_{uv} is a matrix that contains elements from network, if u and v are neighbors then $A_{uv}=1$, otherwise $A_{uv}=0$. And the related modularity function is defined as following:

$$Q^{'}=\frac{1}{m}\sum_{uv}\left[A_{uv}-\frac{d_{in}(v)d_{out}(u)}{m}\right]\delta_{c_u.c_v} \tag{5}$$

3. Description of community structure partition algorithm based on note strength

Two steps are included in this algorithm: (1) find the initial community; (2) expand the community

3.1. Find the initial community

Given a directed network D, first we calculate its vertex note strength and then begin to find the initial community from the largest note strength vertex; the process is detailed as following:

(1) With every $v\in V(D)$, calculate $d_{in}(v)$;

(2) Choose $\max\{d_{in}(v),v\in V(D)\}$ as vertex v and its neighbor $N(v)$, these vertices form a community c;

(3) Given vertex $u\in c$, if $B(u,c)\le B^C$ (we choose $B^C=0.5$), delete vertex u from community c;

(4) Repeat step (3) until for every $u\in c$, $B(u,c)>B^C$ is true. And we obtain a new initial community, which we write it as c.

For example, figure 2 shows that after calculating the node strength of every vertex, v_2 and its neighbors were chosen as a community $\{v_1, v_2, v_3, v_4, v_6, v_8\}$. For every vertex, after calculating $B(\cdot, c)$, v_8 is deleted from the community, finally we get the initial community $\{v_1, v_2, v_3, v_4, v_6\}$.

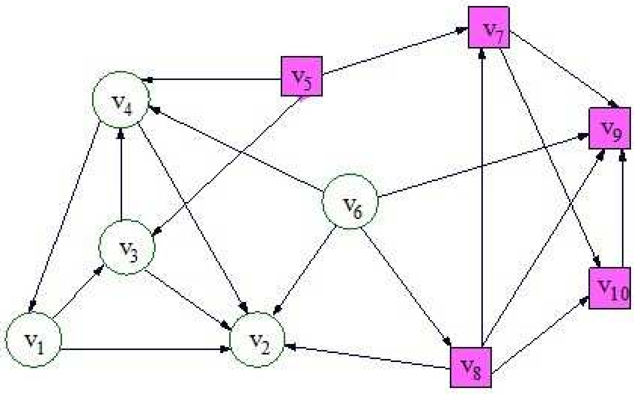

Figure 2: Find the initial community

3.2. Expand the community

First find out all the neighbors $N(c)$ of community c. If there exist $v \in N(c)$, such that $B(v,c) > B^C$, we add vertex v into community c; if for every vertex $v \in N(c)$, $B(v,c) < B^L$, ($B^L = 0.4$), then we stop expanding the community; now we consider a kind of vertex v which meet the need of $B^L \le B(v,c) \le B^C$, and we add this kind of vertex v into the community c, after that the modularity Q' become larger, finally the vertex v was added to the community c. The detailed expanded progress is as following:

(1) Find out all the neighbors $N(c)$ of initial community c, for every node $v \in N(c)$, calculate $B(v,c)$;

(2) Find out vertices which meet the need of $B(v,c) > B^C$ and $B^L \le B(v,c) \le B^C$, write as $N_v = \{v \mid B(v,c) > B^C\}$ and $N_{lv} = \{v \mid B^L \le B(v,c) \le B^C\}$;

(3) If $|N_v| > 0$, we add all the vertices of N_v into the community c which is larger than before, and we still write as c, back to step (1);

(4) If $|N_{lv}| > 0$, we add all the vertices of N_{lv} into the community c which is larger than before, and we still write as c, back to step (1);

(5) If $|N_v| = 0$ and $|N_{lv}| = 0$, stop expanding the community, finally we obtain a community.

As shown in figure 2, $\{v_5, v_8\}$ are the neighbors of initial community $c=\{v_1,v_2,v_3,v_4,v_6\}$, then we get $B(v_5,c)=\frac{2+1}{3+1}=0.75$, $B(v_8,c)=\frac{1+1}{4+1}=0.4$. As we said before, v_5 can be added to community c which is larger than before, still denoted as c. After v_5 is added to the community, c has only one neighbor vertex v_8, because $B(v_8,c)=0.4$ and the modularity before adding v_8 is $Q'=3.25$, after adding v_8 is $Q'=2.25$, that is to say modularity become small. Therefore vertex v_8 cannot be added to the community; finally we get the ultimate community $c=\{v_1,v_2,v_3,v_4,v_5,v_6\}$.

Extract the community c, the make note of every vertex of c with sign 'T', repeat process 3.1 and 3.2 until find out all the communities. As shown in figure 3, extract the first community $c=\{v_1,v_2,v_3,v_4,v_5,v_6\}$, follow the algorithm we proposed above, we can find out anther community $\{v_6,v_7,v_8,v_9,v_{10}\}$, as can be seen in figure 4, vertex v_6 is the common vertex of these two communities.

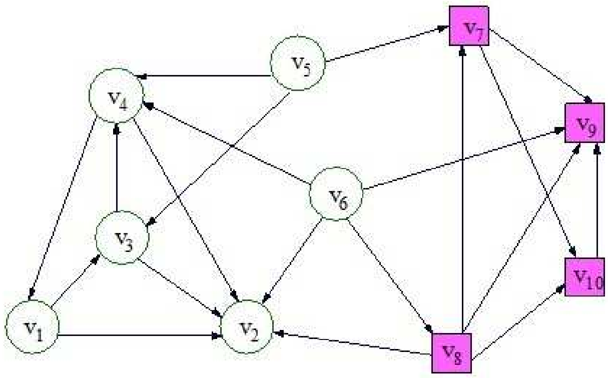

Figure 3: Expand community

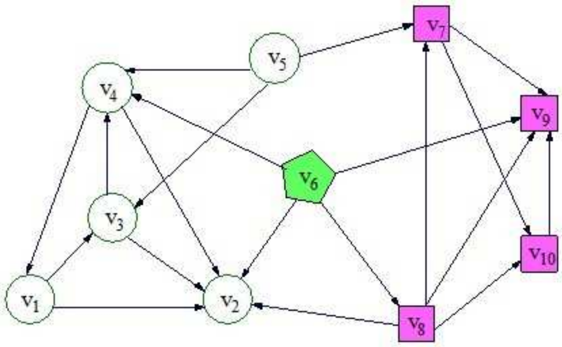

Figure 4: partition of network

4. Application of the algorithm

We choose the Zachary network as an example in order to show the effectiveness of the proposed algorithm which is a famous classic problem about social club network. In the early 1970s, it took two years for Zachary to observe the social relations among the Karate club members in one of American university and finally Zachary constructed the social network[13] among them. See figure 5. Interestingly, during his survey he found that the boss of the club and the supervisor has different opinion about whether charge more fees to the members of the club, thus the club become two communities under the leadership of boss and the supervisor. Zachary club social network has become a classical problem in complex network community structure and a standard in measuring whether a network structure partition is good or not.

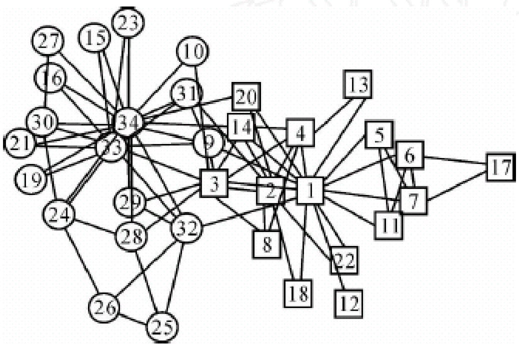

Figure 5: Zachary social karate club network model and the relations

We will analysis the Zachary social network with the algorithm that proposed above. Zachary social network made up of 34 vertices and 78 edges. First of all, we will treat the network as two-way network, that is to say, the in-degree and out-degree of every vertex is equal to its vertex degree. Then we find that vertex v_{34} and its 17 neighbors constitute a community, after computing $B(\cdot,c)$, v_{14} and v_{20} are deleted and form a new community c which is made up of vertex v_{34} and others vertices without v_{14}、v_{20}. Because the in-degree and out-degree of vertex v_{10} is equal, v_{10} is regard as the common vertex of both communities. Furthermore, through add vertices and expand the community. We then find out the neighbors $\{v_1,v_2,v_3,v_{14},v_{20},v_{25},v_{26}\}$ of community c, compute $B(\cdot,c)$, we find that the vertices compactness $B(\cdot,c)$ of vertices v_1, v_2, v_{14} and community c are all smaller than 0.4, thus these vertices cannot be added to the community. While $B(\cdot,c)$ of vertices v_{25}, v_{26} and community c is 1, and vertices v_{25}, v_{26} are added to the community c. Meanwhile vertices v_3 and v_{20} are all included in N_{lv}, we can compare the Q' before and after adding the vertices, it is can be calculated that Q' is become smaller, so v_3 and v_{20} are not allowed to be added. Stop expand and the ultimate community is obtained which is made up of vertex v_{34}, v_{25}, v_{26} and other neighbors vertices except v_{14}, v_{20}. This can be seen from the square in figure 5. Then we can find the largest note strength, follow the above steps, we can obtain the roundness in figure 5. And vertex v_{10} is the common vertex between the two communities.

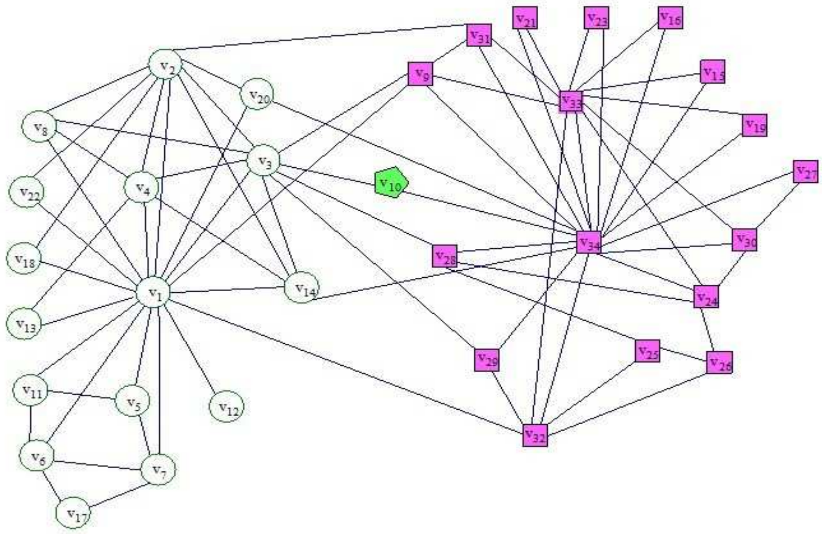

Figure 5: The result after using the proposed algorithm into Zachary

Zachary as a social network, when we study its structure, we didn't consider the other effect factors from the outside. Moreover, in different social conditions, the partition of Zachary network has different results. Therefore, as long as the error is acceptable, we say the algorithm is accurate. Figure 6 also shows the results is good enough to meet the real world problem.

5. Conclusion

This paper introduces an algorithm about how to divide the local community structure of directed network, which is based on the note strength. Comparatively, the advantage of this algorithm is that we don't have to know the whole network structure and the number of communities before we use the algorithm. And local information is enough for this approach. Therefore, this method is easy to use with low complexity. Moreover, this algorithm is also can be used to find the community structure of the whole network. Finally, after we use this approach into Zachary social network, a good result has been achieved.

REFERENCES

1. M.E.J.Newman, The structure and function of complex networks, SIAM Rev, 45 (2003) 167-256.
2. R.Albert and A.L.Barabási, Statistical mechanics of complex networks, Rev Modern Phys, 74(1) (2002) 47-97.
3. S.H.Strogatz, Exploring complex networks, Nature, 410 (2001) 268-276.
4. X.F.Wang, X.Li and G.R.Chen, The theory and applications of complex networks, Beijing, Qinghua University Press, 2006.

5. L.Wang and G.Z.Dai, Community finding in complex networks—theory and applications, Sci Tech Rev, 23(8) (2005) 62-66.
6. M.Fiedler, Algebraic connectivity of graphs,. Czech Math. J, 23(98) (1973) 298-305.
7. B.W.Kernighan and S.Lin, An efficient heuristic procedure for partition graphs, Bell Sys Tech J., 49 (1970) 291-307.
8. M.E.J.Newman and M.Girvan, Finding and evaluating community structure in networks, Phys. Rev E, 69 (2004), p.026113.
9. M.E.J.Newman, Fast algorithm for detecting community structure in networks, Phys Rev E, 69(6) (2004) 1-5.
10. A.Clauset, Finding local community structure in networks, Phys Rev E, 72(2) (2005), p.026132.
11. G.Palla, I.Derenyi, I.Farkas I, et al., Uncovering the overlapping community structure of complex networks in nature and society, Nature, 435(7043) (2005) 814-818.
12. D.R.Lai, H.T.Lu and C.Nardini, Finding communities in directed networks by PageRank random walk induced network embedding, Phys A, 389 (2010) 2443-2445.
13. W.Zachary, An information flow conflict and fission in small groups, J Ant Res, 33 (1977) 452-473.