Pramana – J. Phys. (2017) 88: 44
DOI 10.1007/s12043-016-1343-2

Refinement of the community detection performance by weighted relationship coupling

DONG MIN[1], KAI YU[1,*] and HUI-JIA LI[2]

[1]School of Computer Science and Engineering, Xinjiang University of Finance and Economics, Urumqi, Xinjiang 830001, China
[2]School of Management Science and Engineering, Central University of Finance and Economics, Beijing 100080, China
*Corresponding author. E-mail: yukai_dlut@163.com

MS received 17 July 2015; revised 29 March 2016; accepted 24 August 2016; published online 9 February 2017

Abstract. The complexity of many community detection algorithms is usually an exponential function with the scale which hard to uncover community structure with high speed. Inspired by the ideas of the famous modularity optimization, in this paper, we proposed a proper weighting scheme utilizing a novel k-strength relationship which naturally represents the coupling distance between two nodes. Community structure detection using a generalized weighted modularity measure is refined based on the weighted k-strength matrix. We apply our algorithm on both the famous benchmark network and the real networks. Theoretical analysis and experiments show that the weighted algorithm can uncover communities fast and accurately and can be easily extended to large-scale real networks.

Keywords. Network analysis; community structure; weighting scheme; k-strength relationship; modularity.

PACS Nos 89.75.Ck; 89.75.Fb; 89.75.Hc

1. Introduction

Recent years have witnessed an increasing interest in detecting communities in large-scale networks of various kinds. Communities are clusters of closely connected nodes within a network. Since massive online social networks have been deeply integrated into our daily life, detecting meaningful communities from them has become critical for research and applications of various purposes. Early work in graph partitions [1–3] could be adopted for community detection. However, these methods usually require the knowledge of the number of communities as part of the input, which is unrealistic in community detection problems. Modularity [4–6], proposed by Newman *et al*, is the first successful attempt to resolve this drawback. In real world, the correlation (edge) between two nodes owns different strength, for example, as shown in figure 1a, the bottleneck edges between two communities usually have larger influence or betweenness. Those partitions with high modularity score contain communities with an internal edge density larger than that expected in a given graph model. Several strategies have been proposed for its optimization, including greedy algorithms [7,8], simulated annealing [9], heuristic algorithms [10], spectral clustering [11,12], and genetic algorithms [13]. To the best of our knowledge, the most competitive approach of this kind is Louvain [14], which can scale to graphs with hundreds of millions of objects. Recently, Good *et al* [15] presented a broad characterization of the performance of modularity maximization in practical contexts. The limiting behaviour of the maximum modularity is derived, which shows that it depends strongly on both the size of the network and on the number of modules it contains. Arenas *et al* [16] propose a method that allows for multiple resolution screening of the modular structure. It provides a method that allows the full screening of the topological structure at any resolution level using the original definition of modularity.

Unfortunately, partition of the network into communities by maximizing modularity Q is shown to be an NP-complete problem [17]. In weighted networks, if the bottleneck edges do not have the largest weight,

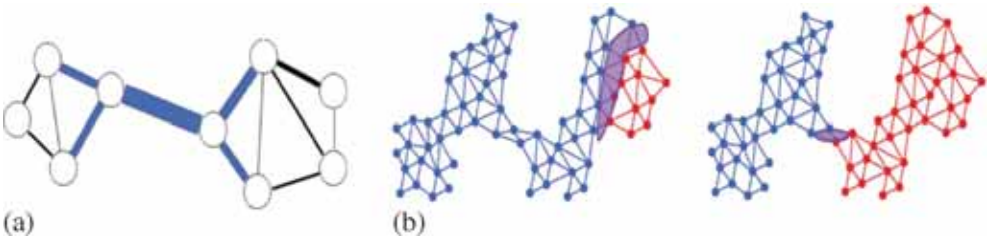


Figure 1. (**a**) The bottleneck edges between two communities usually own larger weight (influence or betweenness), and the values of weight are illustrated by thickness. (**b**) The edges with the largest weight are highlighted by enclosed circle, and if they are not bottleneck edges (left subgraph), the NP-complete problem appears. Here, different colours represent different communities.

the NP-complete problem appears, just as shown in figure 1b. Furthermore, the traditional optimization or heuristic methods are usually used based on the assumption that communities are subsets of nodes which have similar properties. They often compare the internal and external cohesions of a subgraph. However, to obtain an acceptable accuracy, these methods usually have a high-level computational complexity. In this paper, in order to design fast and accurate algorithm to detect communities, we proposed a new weighting scheme to enhance the community detection performance based on a novel correlation, i.e. k-strength relationship, which naturally represents the coupling distance between the two nodes. Community structure detection algorithm is presented using a generalized modularity measure based on the k-strength relationship weighted in various types of networks. Finally, we apply our algorithm on both benchmark network and real networks to evaluate its efficiency. Theoretical analysis and experiments show that the algorithm can uncover communities fast and accurately, which can be easily extended to large-scale real networks.

The outline of the paper is as follows: In §2 the fundamental definitions, such as k-strength relationship and its generalized modularity measure are introduced. In §3, the details of our weighting framework, including the procedures of algorithm and the analysis of computational complexity are presented. Then, some representative experiments on both benchmark and real networks are given to validate the effectiveness and efficiency of the algorithm in §4. Finally, §5 concludes this paper.

2. Definitions

First, some important definitions are provided in this paper. Let an undirected network G has n nodes and m links and without loop. The set of nodes and links are denoted by V and E, respectively. $A = (a_{ij})_{n\times n}$ is the adjacent matrix of network G, where $a_{ij} = 1$ if an edge exists between node i and j, and $a_{ij} = 0$ otherwise. If the graph is extended to a weighted form, the weight of each edge is represented by w_{ij} and the adjacent matrix is extended to the weight form $W = (w_{ij})_{n\times n}$.

If k is a specific positive number, then we denote the k-path as a path between nodes i and j if it is a walk chain pass over $k + 1$ nodes and without cycle. $A^k = (a^k_{ij})_{n\times n}$, $a^k_{ij} = \sum_{l=1}^{n} a^{k-1}_{il} \times a_{lj}$ is the number of k-paths from node i to j ($i \neq j$), if $i = j$, set $a^k_{ij} = 0$. For a specific positive integer k, $S^k = (s^k_{ij})_{N\times N}$ is denoted as a matrix of G. We define S^k as a kth-strength matrix of G. The detailed definition is shown as follows:

If $k = 0$,

$$S^0 = A. \tag{1}$$

If $k = 1$,

$$S^1 = (w_{ij})_{n\times n}. \tag{2}$$

For $k \geq 2$,

$$S^k = (s^k_{i,j})_{n\times n}, \quad s^k_{i,j} = \sum_{s=1}^{a^k_{i,j}} \frac{1}{k} \sum_{l=1}^{k} w_{i^s_{l-1} i^s_l}, \tag{3}$$

where $i = i^s_0, i^s_1, \ldots, i^s_{k-1}, i^s_k = j$ are k-path for $s = 1, 2, \ldots, a^k_{i,j}$. In order to calculate $s^k_{i,j}$, we fix $S^0 = A$ if the network is given. For each pair of nodes, all the k-paths between them can be obtained by A^k, where $S^{i,j}$ is an additive polynomial. Therefore, the value of $S^k_{i,j}$ can be calculated precisely.

Next, for a given k-strength matrix, we can induce the k-strength relationship: $R_k = \{(i, j, s_{i,j}) | s_{i,j} = \sum_{l=1}^{k} s^l_{i,j}\}$. That is, $s_{i,j}$ in R_k are the elements of $S = S^1 + S^2 + \cdots + S^k = (\sum_{l=1}^{k} s^l_{i,j} = s_{i,j})_{n\times n}$. We denote S as a k-strength matrix of G and the networks induced by S is a k-strength relationship network.

In the proposed framework, another useful definition is the minimal q-cut of a network, which is defined

as the cut edges with the smallest sum of weight. Here, q is a positive integer, $\{C_1, C_2, \ldots, C_q\}$ with $|C_i| = k_i$ and $\cup_{i=1}^{q} C_i \subseteq V(G)$ is a vertex subset where deleting all C_i will result in a disconnected graph with minimal sum of the edge weight. One can easily find that a minimum cut will result in a partition of G when $\cup_{i=1}^{q} C_i \subseteq V(G)$. Guttmann-Beck and Hassin [18] proposed an algorithm that detects the communities using the strength relationships by finding minimum cut in complete graphs. Inspired by this idea, we use the minimum cut based on maximizing modularity which is proposed by Newman. We extend the modularity Q to a generalized form based on strength relationship matrix of G. Suppose q ($q \leq n/2$) communities exist in network G, $\mathcal{C} = \{C_1, C_2, \ldots, C_q\}$. The weighted modularity Q of the strength relationship matrix is

$$Q = \max_{q} \sum_{i=1}^{q} (c_{i,i} - c_i^2), \tag{4}$$

where

$$c_{i,i} = \sum_{i,j} \frac{s_{i,j}}{\Delta} \delta_{i,j}, \quad c_i = \sum_{j} s_{i,j}$$

and

$$\Delta = \sum_{i,j} s_{i,j}.$$

$\delta_{i,j} = 1$, if the nodes i and j belong to the same community, $\delta_{i,j} = 0$, otherwise. $c_{i,i}$ is the fraction of strength within the same community partition C_i, c_i is the proportion of strength with only one end of edges in C_i. If the network is unweighted (binary network), Q is just the Newman's modularity. Based on this, the new measure can capture the properties of the real social systems. One can find that both direct and indirect information between two nodes can be used within our framework.

3. The framework

3.1 *The weighted k-strength relationship matrix*

Here, we analyse the property of the weighting scheme in detail. As described above, the k-strength relationship matrix is fundamental to the whole framework. Here, we focus on determining the k-strength relationship matrix. The following theorem not only shows the detailed process of computing, but also reveal the important time complexity information.

Theorem 1. *One can compute the k-strength relationship matrix W^k in polynomial time.*

Proof. For a given network G, the adjacent matrix is A, the number of all paths with k-length from i to j is $a_{i,j}^k$, and $A^k = A^{k-1} \times A = (a_{i,j}^k)$ [19]. A path is called a k-path if its length is k. Denote the k-path by $\{i_0, i_1, \ldots, i_{k-1}, i_k\}$ with $k+1$ nodes and $i_s \neq i_j$ for all s and j, which means that no cycle exists in the path.

To obtain the k-strength relationship matrix, an operation $\oplus$ is defined on the weight matrix of network G. $\oplus : W^k = W^{k-1} \oplus W = (w_{i,j}^k)_{n \times n}$, where $w_{i,j}^k$ is defined as: If $\sum_{l=1}^{n} (w_{i,l}^{k-1} \times w_{l,j} \neq 0)$, there are k edges between nodes i and j. Similarly, more than one term exist in $\sum_{l=1}^{n} (w_{i,l}^{k-1} \times w_{l,j} \neq 0)$. Generally, it is supposed that h terms in W are not equal to 0, $w_{i,l^1}^{k-1} \times w_{l^1,j} \neq 0$, $w_{i,l^s}^{k-1} \times w_{l^s,j} \neq 0$, $w_{i,l^h}^{k-1} \times w_{l^h,j} \neq 0$. Then $w_{i,j}^k = \sum_{s=1}^{h} (w_{i,l^s}^{k-1} + w_{l^s,j})$; otherwise, $w_{i,j}^k = 0$ (that means no edge exists between nodes i and j). $w_{i,j}^k$ is defined as the sum of all weights in each k-path between nodes i and j. It can be easily found that $s_{i,j}^k = w_{i,j}^k / k$. That means

$$s_{i,j}^k = \sum_{s=1}^{a_{i,j}^k} \frac{1}{k} \sum_{l=1}^{k} w_{i_{l-1}^s, i_l^s} = w_{i,j}^k / k.$$

The k-path can be determined when $\sum_{l=1}^{N} (w_{i,l}^{k-1} \times w_{l,j})$ is given. If $w_{i,l^1}^{k-1} \times w_{l^1,j} \neq 0$, $w_{i,l^s}^{k-1} \times w_{l^s,j} \neq 0$, $w_{i,l^h}^{k-1} \times w_{l^h,j} \neq 0$ for a specific positive integer $k \geq 2$. Suppose a k-path exists between nodes i and j by $P_{i,j}^k$. It can be easily found that $a_{i,j}^k$ k-paths exist between nodes i and j and hence,

$$P_{i,j}^k = \left\{ P_{i,l^1}^{k-1} \vee (l^1, j), P_{i,l^2}^{k-1} \vee (l^2, j), \ldots, P_{i,l^h}^{k-1} \vee (l^h, j) \right\},$$

where $P_{i,l}^{k-1} \vee (l, j)$ indicates that all the k-paths formed by the $(k-1)$-paths in set $P_{i,l}^{k-1}$ join the edge (i, j).

Finally, all the k-paths can be determined iteratively. That means computing $\oplus$ costs polynomial time. In general, one can determine the strength matrix W and the computational complexity is $O(n^2 m)$. Output $S^k = (s_{i,j}^k) N \times N$ for a fixed positive integer k. Summarize all the paths existing between nodes i and j with length k, $i = i_0^s, i_1^s, \ldots, i_{k-1}^s, i_k^s = j$ for $s = 1, 2, \ldots, a_{i,j}^k$.

The proof ends here. □

3.2 *An improved community detection algorithm*

A minimal q-cut $\tilde{E}$ of G is a set of edges with minimal sum of weight that the remaining graph of deleting

the edges set, $G - \tilde{E}$, is an isolated graph. We need to choose the elements that $\sum_{i<j} w(C_i, C_j)$ is the minimal one or $\sum_{i=1}^{c} w(C_i, C_i)$ is the maximal one by maximum flow and minimum cut theorem [20]. However, the minimal q-cut problem is NP-complete and it is difficult to find a polynomial algorithm. The detailed procedures of the proposed method are shown in Algorithm 1.

3.3 *Computational complexity*

For a specific positive integer q, the transport problem [21,22] can be solved in $O(n)$ time, because it belongs to 0-1 transportation problem. There are $C_n^q\, O(qn)$ subsets of V. Totally, the computational complexity is $O((q+1)n)$. Here, we propose two important claims:

Claim 1. If $\{C_1, C_2, \ldots, C_q\}$ is a partition of $\tilde{G}$, if and only if $\{C_1, C_2, \ldots, C_q\}$ is also a partition of G.

Proof. This claim can be easily verified, because $\tilde{G}$ and G own the same set of nodes. □

Claim 2. Suppose $\{C_1, C_2, \ldots, C_q\}$ are q partition of $\tilde{G}$ such that $\sum_{i<j, C_i, C_j \subset \tilde{G}}^{q} w(C_i, C_j)$ is a minimum. Then there exists a minimum partition, called $\{\bar{C}_1, \ldots, \bar{C}_q\}$, such that $\sum_{i<j, \bar{C}_i, \bar{C}_j \subset G}^{q} w(\bar{C}_i, \bar{C}_j)$ is minimum.

Proof. By the definition of $\tilde{G}$,

$$\Delta = \sum_{i<j, C_i, C_j \subset \tilde{G}}^{q} w(C_i, C_j).$$

As

$$w(C_i, C_j) = \sum_{i \in C_i, j \in C_j} s_{ij}^1 + \sum_{i \in C_i, j \in C_j} \sum_{k=2} s_{ij}^k,$$

the value of Δ is fixed because $\{C_1, C_2, \ldots, C_q\}$ is the minimum partition of $\tilde{G}$. We know that

$$\sum_{i \in C_i \subset \tilde{G}, j \in C_j \subset \tilde{G}} s_{ij}^1 = \sum_{i \in C_i \subset G, j \in C_j \subset G} s_{ij}^1$$

using the k-strength relationship. Therefore, a minimum partition $\{\bar{C}_1, \ldots, \bar{C}_q\}$ is constructed in $\tilde{G}$ using Algorithm 1 such that $\sum_{i \in \bar{C}_i \subset \tilde{G}, j \in \bar{C}_j \subset \tilde{G}} s_{i,j}^1$ is a minimum. Then, $\{\bar{C}_1, \ldots, \bar{C}_q\}$ is the minimum q partition of G. The proof ends here. □

Table 1. The computational complexity of classical community detection algorithms for sparse networks.

Algorithms	Order
CNM [24]	$O(n \log^2 n)$
DA [25]	$O(n^2 \log n)$
Louvain [14]	$O(n \log n)$
OCR-HK [26]	$O(n^2)$
RN Potts model [27]	$O(n^{1.3})$
The proposed model	$O((q+1)n)$

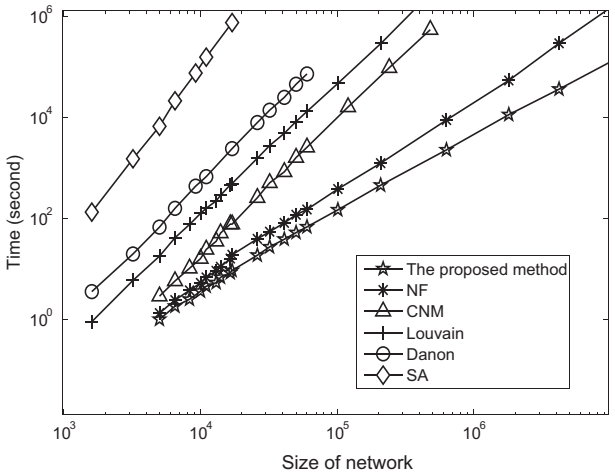


Figure 2. The computational time of six algorithms with the change of network scales.

Generally, the convergence speed of Algorithm 1 is very fast.The computational complexity of the proposed method is pretty low compared to many classical methods (see table 1). Based on the analysis above, the computational time is correlated to the number of communities q, which depends on the network structure. For many real networks, the size of the network is much larger than the number of communities, such as famous Zachary karate club network [23]. In these networks, it can be considered that the complexity is $O(n)$ for sparse cases.

We also compare the performance of Algorithm 1 with several famous methods, including Newman fast (NF) algorithm [1], CNM algorithm [4], Louvain algorithm [14], Danon algorithm [28] and SA algorithm [9], which is shown in figure 2. Here, the device we used is a desktop computer with a 2 GHz CPU and a 4 GB memory. The operating system is Windows 7, and the programming software is Matlab 2010b. To apply these algorithms, we generate a sparse network using LFR benchmark with n nodes and $m = O(n)$ edges. In figure 2, first, we observe that our algorithm is much faster than all the other methods with all scales

Algorithm 1. An improved community detection algorithm based on weighting scheme

Input: a network $G = (V, E)$;

Output: Maximize the modularity Q using a minimum q-partition.;

1: **Step 1:** Delete all leaf nodes with one degree. This procedure does not affect the result of community partition. The shrink network is still represented by G.

2: **Step 2:** Set the number of communities q.

3: **Step 3:** For a specific positive integer q, the minimum q-cut problem can be solved in a polynomial time, i.e. $O(|V|^{q^2})$ [18].

4: Therefore, for a specific partition $\mathcal{C} = \{C_1, C_2, \ldots, C_q\}$ in $\tilde{G} = (V, \tilde{E})$, and $|C_i| = k_i$, $\sum_{i=1}^{q} k_i = n$, $C_i \cap C_j = \emptyset$. We determine the minimum q-cut in $\tilde{G}$, and prove it is the minimum q-cut in network G. Let $v_i \in C_i$ for $i = 1, 2, \ldots, q$. $x_{i,j} = \begin{cases} 1, & u_j \in C_i \\ 0, & \text{otherwise} \end{cases}$

Begin

For $\{v_1, v_2, \ldots, v_q\} \subset V$, $v_i \in C_i$.

For $u_j \in V - \{v_1, v_2, \ldots, v_q\}$, the following transport problem is optimal.

$$\min : \sum_{i=1}^{q} \sum_{j=1}^{n-q} w(C_i, u_j)(1 - x_{i,j})$$

$$\text{subject to} \begin{cases} \sum_{j=1}^{N-q} x_{i,j} = k_i - 1, \ i = 1, 2, \ldots, q \\ \sum_{i=1}^{q} x_{i,j} = 1, \ j = 1, 2, \ldots, N - q \\ x_{i,j} \in \{0, 1\}, \ i = 1, 2, \ldots, q \text{ and } j = 1, 2, \ldots, N - q \end{cases}$$

End

$$C_i = C_i \cup \{u_j | x^*_{i,j} = 1, 1 \le j \le N - q\} \quad \text{for } 1 \le i \le q.$$

End

Back to begin

5: **Step 4:** Output: $\{C_1, C_2, \ldots, C_q\}$ with $v_i \in C_i$ is a minimum q-cut on $\tilde{G}$.

of network; and second, the computational time of our method is scalable to the scales of networks, and it can be efficiently manipulated in the extremely large network with millions or even billions of nodes.

4. Experiments and results

In order to verify the effectiveness of our weighting method, we apply it on two famous benchmarks. First, an artificial random network generated by Girvan and Newman, GN network, is used [29,30]. This benchmark was used by many researchers for comparing the efficiency of partition result. The established mechanism is as follows: a 128-nodes network is a partition of four communities, and each community owns 32 nodes. Every innercommunity edge is linked independently with probability p_{in} and every intercommunity edges is linked with probability p_{out}. For each node, the expected innercommunity degree is $z_{\text{in}} = 31p_{\text{in}}$ and the expected intercommunity degree is $z_{\text{out}} = 31p_{\text{out}}$.

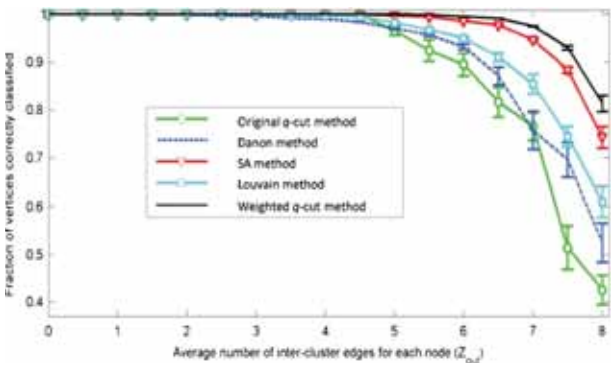


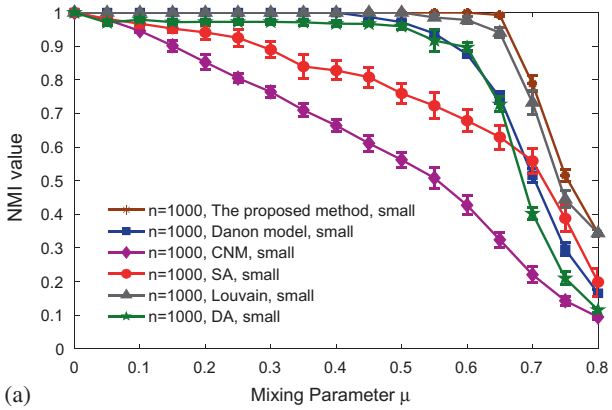
Figure 3. Computational results by five algorithms including our method as a function of z_{out} in GN network. Each point shows the average and variance over 50 times.

As z_{out} increases, the community structures becomes more and more ambiguous, and correspondingly, the fraction of correctly classified nodes decreases.

To illustrate the efficiency of our refinement algorithm, we compare the percentage of nodes that are classified correctly for different methods including,

original minimum q-cut algorithm, refined minimum q-cut algorithm, two famous modularity optimization methods, Louvain method [14], Danon method [28], and SA heuristic method [9]. As can be observed in figure 3, the refinement process enhances the performance of community detection a lot. The proposed algorithm has the best performance even when z_{out} increases to 8. Furthermore, we analyse the performance of variance of community partition. As refinement algorithm is sensitive to the initial condition, the original minimum q-cut algorithm method shows the largest variation, while the Louvain and Danon methods have less variation compared to GA heuristic method, and as it was expected, the variance of refined minimum q-cut algorithm is the best among the above-mentioned approaches.

We have also tested our algorithm for LFR benchmark networks [31] and compared its performance



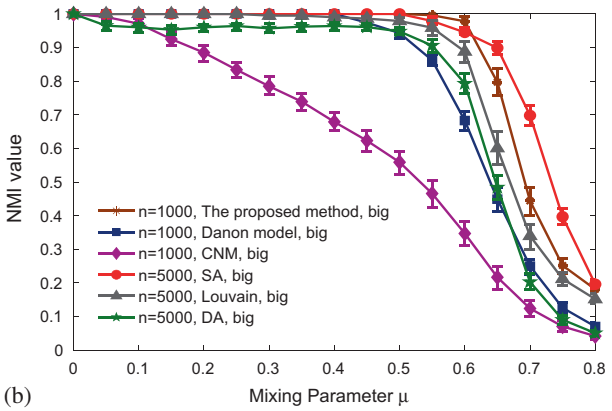


Figure 4. The comparison of different algorithms with the proposed one for (**a**) $n = 5000$ small LFR, (**b**) $n = 5000$ big LFR benchmark networks. Each point is averaged over 50 realizations.

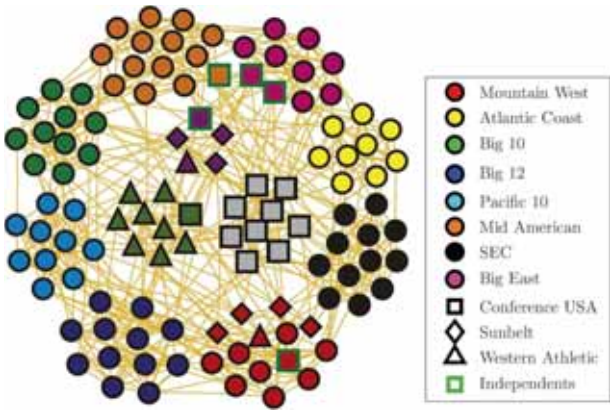


Figure 6. The partition result of the football network. Almost all the football teams coincide with the known structure, indicating that our algorithm performs remarkably well.

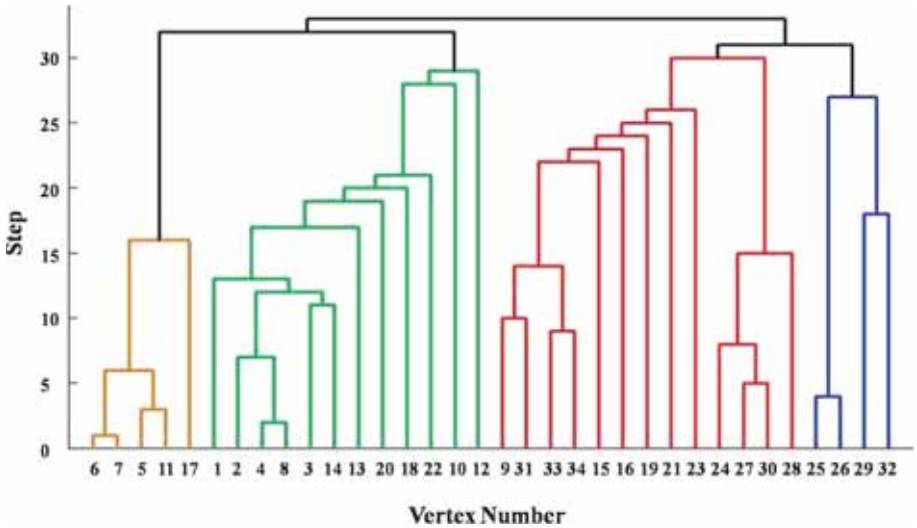


Figure 5. Using the weighting scheme, we apply our framework on Zachary karate network identified by Newman fast algorithm. Dendrogram of communities are shown and different colours correspond to different community structures.

Table 2. The summary of results using our algorithm in real-world networks. For the sake of comparison, the best published modularity Q results are provided.

Network	Ref.	Best published Q	Source	Q of the proposed model
Zachary karate club	[23]	0.420	[32]	0.416
US football	[4]	0.606	[32]	0.602
Les miserables	[33]	0.561	[32]	0.554
Jazz	[34]	0.446	[32]	0.439
PGP-key signing	[35]	0.878	[32]	0.883
Dolphin social network	[36]	0.531	[32]	0.527
Email	[9]	0.579	[32]	0.568

with other methods. The LFR benchmark graph is introduced by Lancichinetti, Fortunato and Radicchi, which provides more realistic scale-free graphs by tuning the relevant parameters. In this graph, the number of nodes in communities and the degree of the nodes follow power-law distributions, showing scale-free property with predefined exponents. We need to set some parameters, such as the maximal degree of nodes, the average degree of nodes, the total number of nodes, the maximal and minimal sizes of communities, and the most important mixing parameter (μ). The value of mixing parameter (μ) varies within the interval $\lfloor 0, 1 \rfloor$, which determines the degree of fuzziness of the communities in the LFR graph and the larger the μ, the more fuzzy the communities. One can note that the GN benchmark is a special case of the LFR benchmark. The outcome is shown in figure 4. In this simulation, the initial guess for the number of communities for small (big) LFR benchmark networks with $n = 5000$. As can be seen in figure 4, the experimental results of the proposed algorithm are fabulous in almost all cases.

Next, we test whether the communities identified are completely correct. Unlike with the GN network, if such a group is not entirely contained in the same community, then all vertices of the group are assumed to be incorrectly identified. We test our framework in a famous real-world network, i.e. Karate club network [31]. It is a standard network which used to compare the precision between different community detection algorithms. Here, we use Newman fast (NF) method on weighted network, in which the number of communities need not be specified. In figure 5, the dendrogram obtained by using our weighting scheme is represented. We report the modularity measure obtained from original NF method and weighted NF as 0.397 and 0.432. From these results, one can conclude that the weighting scheme improves NF considerably.

Finally, we investigate a college football network, which represents the game schedule of the US college football league in 2000. The nodes in the network represent 115 teams and the edges represent regular season games between the two teams. The teams are divided into 12 conferences containing around 8–12 teams each. Such a known community structure makes this network interesting to investigate.

We set the number of communities as 12 and the community detection result is shown in figure 6. According to our result, we identify the community structure with a high degree of accuracy. Almost all of the football teams are correctly clustered with the others in their conference. Only one member in Conference USA (shaded black-edge box), Texas Christian, is grouped with most of the teams in the Western Athletic conference (shaded triangle). All the other communities (shaded coloured ellipses) coincide with the known structure, which indicates that our algorithm performs remarkably well.

Finally, we evaluate our model for more widely used real-world data, which are shown in table 2. Seven widely used networks are employed and the corresponding references are represented. For the sake of comparison, the best published modularity Q results are provided, which are obtained by the computation of a lot of partition methods. As can be seen in table 2, the results are very close to the best published values after applying our algorithm, but at a small computational cost.

5. Discussion

In this paper, we have designed an efficient algorithm to detect community in social network using a new definition, i.e. k-strength relationship, which naturally represents the coupling degree between two nodes. Theoretical analysis shows that this algorithm is polynomial time which is much better than most of the

existing ones. Finally, we apply our algorithm on both benchmark network and real networks to evaluate its efficiency. Theoretical analysis and experiments show that the algorithm can uncover the communities fast and accurately. This algorithm can be easily extended to large-scale real networks.

Acknowledgements

The authors are grateful for the detailed reviews and constructive comments of the reviewers, which have greatly improved the quality of this paper. The research was supported in part by MOE (Ministry of Education in China), Liberal Arts and Social Sciences Foundation Grant No. 12YJA870013, NSFC grants 71561025, 71401194, 91324203 and Ph.D. research foundation of Xinjiang University of Finance and Economics Grant No. 2015BS004.

References

[1] M E J Newman, *Phys. Rev. E* **69**, 066133 (2004)
[2] M E J Newman and M Girvan, *Phys. Rev. E* **69**, 026113 (2004)
[3] H J Li, Y Wang, L Y Wu, Z P Liu, L Chen and X S Zhang, *Eur. Phys. Lett.* **86(1)**, 012801 (2012)
[4] M Girvan and M E J Newman, *Proc. Natl Acad. Sci.* **99**, 7821 (2002)
[5] X S Zhang, R S Wang, Y Wang, J Wang, Y Qiu, L Wang and L Chen, *Eur. Phys. Lett.* **87**, 38002 (2009)
[6] X S Zhang, Z Li, R S Wang and Y Wang, *J. Comb. Optim.* **23(4)**, 425 (2012)
[7] L C Huang, T J Yen and S C T Chou, *International Conference on Advances in Social Networks Analysis and Mining*, IEEE Computer Society, pp. 110–117 (2011)
[8] Peter J Mucha *et al*, *Science* **328**, 876 (2010)
[9] R Guimera and L A N Amaral, *Nature* **433**, 895 (2005)
[10] B W Kernighan and S Lin, *Bell System Tech. J.* **49**, 291 (1970)
[11] H J Li and X S Zhang, *Eur. Phys. Lett.* **103**, 58002 (2013)
[12] H J Li, Y Wang, L Y Wu, J Zhang and X S Zhang, *Phys. Rev. E* **86**, 016109 (2012)
[13] F Radicchi, C Castellano and F Cecconi, *Proc. Natl Acad. Sci.* **101**, 2658 (2004)
[14] V D Blondel, J L Guillaume, R Lambiotte and E Lefebvre, *J. Stat. Mech.* **10**, 10008 (2005)
[15] B H Good, Y-A de Montjoye and A Clauset, *Phys. Rev. E* **81**, 046106 (2010)
[16] A Arenas, A Fernandez and S Gomez, *New J. Phys.* **10(5)**, 053039 (2008)
[17] M Latapy and P Pons, *Proceedings of the 20th International Symposium on Computer and Information Sciences*, *Lect. Notes Comput. Sci.*, **3733**, 284 (2005)
[18] N Guttmann-Beck and Hassin, *Algorithmica* **27**, 198 (2000)
[19] H W Su, *Int. Rev. Comput. Software* **7(7)**, 3782 (2012)
[20] M R Garey and D S Jonson, *Computers and intractability*: *A guide to the theory of NP-completeness* (Freeman, San Francisco, CA, 1979)
[21] H J Li, H Wang and L Chen, *Eur. Phys. Lett.* **108(6)**, 68009 (2015)
[22] M Rosvall and C T Bergstrom, *Proc. Natl Acad. Sci.* **105(4)**, 1118 (2008)
[23] W Zachary, *J. Anthropol. Res.* **33**, 452 (1977)
[24] A Clauset, M E J Newman and C Moore, *Phys. Rev. E* **70(6)**, 066111 (2004)
[25] J Duch and A Arenas, *Phys. Rev. E* **72(2)**, 027104 (2005)
[26] S Boccaletti, M Ivanchenko, V Latora and A Pluchino, *Phys. Rev. E* **75(4)**, 045102 (2007)
[27] P Ronhovde and Z Nussinov, *Phys. Rev. E* **81(4)**, 046114 (2010)
[28] L Danon, J Duch, D Guilera and A Arenas, *J. Stat. Mech.* **29**, 09008 (2005)
[29] H J Li and J Daniels, *Phys. Rev. E* **91(1)**, 012801 (2015)
[30] Z P Li, S H Zhang, R S Wang, X S Zhang and L Chen, *Phys. Rev. E* **77**, 036109 (2008)
[31] A Lancichinetti and S Fortunato, *Phys. Rev. E* **80**, 056117 (2009)
[32] G Agarwal and D Kempe, *Eur. Phys. J. B* **66(3)**, 409 (2008)
[33] D E Knuth, *The Stanford GraphBase*: *A platform for combinatorial computing* (Addison Wesley Professional, Reading, CA, 1993) Vol. 37, p. 592
[34] P Gleiser and L Danon, *Adv. Complex Syst.* **6**, 565 (2003)
[35] M Boguna, R Pastor-Satorras, A Diaz-Guilera and A Arenas, *Phys. Rev. E* **70(5)**, 056122 (2004)
[36] D Lusseau, K Schneider, O J Boisseau, P Haase, E Slooten and S M Dawson, *Behav. Ecol. Sociobiol.* **54(4)**, 396 (2003)