Threshold Random Walkers for Community Structure Detection in Complex Networks

Xianghua Fu
College of Computer Science and Software Engineering, Shenzhen University, Shenzhen Guangdong, 518060, China
Email: fuxh@szu.edu.cn, 971914846@qq.com, {wangzq, Mingz}@szu.edu.cn

Chao Wang, Zhiqiang Wang, Zhong Ming
College of Computer Science and Software Engineering, Shenzhen University, Shenzhen Guangdong, 518060, China
Email: fuxh@szu.edu.cn, 971914846@qq.com, {wangzq, Mingz}@szu.edu.cn

***Abstract*—There exist large amounts of complex networks in different areas nowadays, which have aroused great interest in detecting community structures. Although diverse community detection algorithms have been proposed, most of them perform poorly in large scale complex networks. According some social principles, we proposed a scalable *C*ommunity *D*etectio*n* method based on *T*hreshold *Rand*om *walk*ers, which is called CD-*TRandwalk*. CD-TRandwalk selects active nodes with high degree as seed nodes, and detects the core communities through random walkers according to predefined thresholds at first. Because the threshold random walkers start from the active seed nodes and only randomly walk to those nodes which association degrees are larger than a given threshold, the processes of detecting core communities work quickly. After that, the remaining non-core nodes are allocated into the core communities according their common degrees between these nodes and the core communities with a voting strategy. Compared with some other community detection algorithms such as Affinity Propagation (AP), Walktrap, Newman Fast, and ComTector in several social networks, the experimental results show that CD-TRandwalk is faster than the other methods without worse quality of community detection quality. Furthermore, CD-Trandwalk is adaptable to large scale networks and unbalance networks. CD-TRandwalk also has some other advantages, such as it is unsupervised and not need to set the community number beforehand, and it only needs local information of the networks to support local community detection.**

***Index Terms*—community detection; threshold random walk; social network analysis; complex networks;**

I. INTRODUCTION

Nowadays, there exist large amounts of complex networks in different areas, such as social networks, Internet networks, biological networks, mobile phone communication networks, and micro-blogs networks. Despite the diverse physical meaning behind those networks, they usually exhibit common topological properties. For example vertexes of the networks are often organized into communities or groups with dense connections within groups and sparse connections between groups. Efficiently detecting communities can help us to understand the nature of those networks better and facilitate the analysis on large scale complex networks.

During the last decades, community detection methods have received an enormous amount of attention in many disciplines. Examples range from social network analysis[1] to Natural language processing[2], and from analyzing protein interaction networks[3] to the distributed Very-large-scale integration (VLSI) simulation [4]. Because community detection is very important, many methods have been proposed in the literature. Fortunator divided the existing community detection methods roughly into traditional methods, divisive methods, modularity based methods, spectral methods and dynamic methods et al[5]. The idea of divisive algorithms is to detect the edges that connect vertexes of different communities and remove them, and disconnect the clusters from each other. The most popular algorithm is that proposed by Girvan and Newman[6, 7]. A notable work of them defines betweenness and introduces modularity as a posterior measure of network structure, which is very important significance to community detection and gains success in many applications. Modularity is a best known quality function to measure the community structure. The modularity based methods use different clustering technique such as greedy techniques[8], simulated annealing, external optimization, spectral optimization to maximize the modularity. Some researchers also proposed improved modularity measurement such as extension to directed graph, accounts for positive and negative edges. Spectral methods detect communities according the eigenvectors and eigenvalues of the graphs' feature matrixes. And dynamic methods employ processes running on the graph, such as spin-spin interactions, random walks and synchronization. The detailed survey of community detection can be found in Fortunator's paper[5].

This work is supported by Science and Technology Foundation of Shenzhen City (JC201005280463A, No.JC201105160498A). Corresponding author's email addresses: fuxh@szu.edu.cn (Xianghua Fu).

doi:10.4304/jsw.8.2.286-295

Although we have got many research results, there are some unresolved problems of community detection. For example, the concept of community is still not unified and has no rigorous mathematical definition. Most existing methods perform poorly in large scale networks because of the cost in memory space or time.

We have proposed a scalable community detection method based on threshold random walk which is called *CD-Trandwalk* [9]. CD-TRandwalk implements two intuitions in our social life: (1) there always exists some people which are more active than others, and the active people generally have much more connections to others; (2) if two persons have many common friends, the two persons belong to the same social community with large probability. CD-TRandwalk is a two-stage community detection method. At first, the core nodes of the communities are detected by the threshold random walk; and then the remaining non-core nodes are allocated by a voting policy. Because the active nodes are selected as the seed node of the threshold random walkers beforehand, and the community detection process is executed at the same time of the random walk, the time cost of CD-TRandwalk is not high. So it is scalable to large scale social networks. Moreover, CD-TRandwalk has some other advantages, such as it is unsupervised and not need to set the community number beforehand. CD-TRandwalk only needs local information of the networks to support local community detection. In this paper, we will give more detail of CD-Trandwalk algorithm and experiment results.

This paper is organized as follows. In section 2 we mainly review some related work. Section 3 introduces Preliminaries of random walk on community detection. In section 4 we describe our CD-TRandwalk method. The experiment results and analysis are presented in section 5, and the conclusions are given in section 6.

II. Related Works

The most related works are the random walk based community detection methods. A widely popular approach in graph mining and machine learning literature is to compute proximity between nodes by using random walk on graphs: diffusion of information from one node to another. Given a graph and a start node, random walk algorithm finds next node by transition probability. The sequence of random walk is a Markov chain. Random walk provides a simple framework for unifying the information from ensembles of paths between two nodes.

Random walk has been used for community detection by many methods, where the measure of vertexes similarity is based on the properties of random walk on graphs. Such as Zhou and Lipowsky[10] proposed to calculate the similarities matrix of vertexes by random walk at first, and then cluster communities with the similarities matrix. Pons and Latapy proposed the *Walktrap* method[11] alike with Zhou and Lipowsky, but the transition probability of the random walk is different. Saerens and his coworkers[12] used commute-time as similarity measure: the larger the time, the farther the vertexes. Another similarity measure is the escape probability[13], which defined as the probability that the walker reaches the target vertex before coming back to the source vertex. Delvenne [14] introduce the quality measurement function of network partition, which defined in terms of the clustered autocovariance of a Markov process taking place on the graph. And Backstrom and Leskovec [15] developed an algorithm based on Supervised Random Walks that naturally combines the information from the network structure with node and edge level attributes. The goal of the supervised learning task is to learn a function that assigns strengths to edges such that a random walker tends to visit those nodes to which new links will be created in the future.

Unlike above random walk based methods, Alamgir et al[16] proposed a multi-agent random walk to improve the problem of local graph clustering. All agents move independently like a standard random walk on the graph, but they are constrained to have distance at most l from each other. In multi-agent random walk, the node sequence of random walk is directly selected as the community members. This idea is very like our CD-TRandwalk method. CD-TRandwalk also identifies the community members at the same time of the random walk. The difference is that our threshold random walkers start from the active seed nodes and only random walk to the nodes which association degrees are larger than a given threshold. Of course our CD-TRandwalk also can be modified as a version of multiple random walkers.

The other related works are the large scale community detection methods. In order to processing large scale networks, some novel methods are proposed recently, such as probability model based method[17], Heuristic based method[18], and local community detection method[16]. Henderson et al[17] introduced the Latent Dirichlet Allocation model (LDA) for graphs clustering, where the edges are viewed as words and vertexes are viewed as documents. Waikita et al [18]showed that the inefficiency of CNM method is caused from merging communities in unbalanced manner, and they found a simple heuristics that attempts to merge community structures in a balanced manner can dramatically improve community structure analysis. The proposed techniques are tested using datasets obtained from existing social networking service that hosts 5.5 million users. Satuluri et al [1] presented a multi-level algorithm for graph clustering using flows that delivers significant improvements in both quality and speed.

Furthermore, the research works of community detection are diversification. Such as Katzir et al [19] focused their attention on assessing the size of the online community. They did this work with the API of the online community. They selected the samples randomly and assess the size of online community by the number of Non-unique nodes. Kim and Leskovec[20] wanted to solve the problem that data collection is incomplete. They put the graph data into the kronecker graphs model and evaluate the expectation until the expectation is maximization. Simultaneously, they predicted the link with this idea. Link prediction is also a hot spot in

community detection. Many scholars are studying this aspect. Jure Leskovec et al [21] divided the link into passive and active links and found that link prediction can have high accuracy.

III. Preliminaries of Random Walkers on Community Detection

Let $G=(V,\mathrm{E})$ be our input network, where V denotes the node set and E denotes the edge set. We only consider undirected graph in this paper. Let A be the adjacency matrix of the input network with $|V|\times|V|$ elements, and any element A_{ij} denotes the weight of the edge between the vertex v_i and the vertex v_j. In unweighted network, if the vertex v_i connects with the vertex v_j, then $A_{ij}=1$; otherwise $A_{ij}=0$. Let $N(i)$ denote the neighbors set of the node v_i, and $d(i)$ denote the degree of the vertex v_i, where $d(i)=\sum_j A_{ij}$, and $d(i)$ is the size of $N(i)$.

D is a $|V|\times|V|$ diagonal matrix, where $D_{ii}=d(i)$. Let P denote the transition probability matrix, if $(v_i,v_j)\in E$, then $P_{ij}=A_{ij}/D_{ii}$; otherwise $P_{ij}=0$. A random walk on this graph is a Markov chain with transition probabilities specified by this matrix. The random walk process is driven by the powers of the matrix P: the probability of going from v_i to v_j through a random walk of length t is $(P^t)_{ij}$. In the subsequent part, we will denote this probability with P_{ij}^t. It satisfies two well-known properties of the random walk process [11]:

Property 1: When the length t of a random walker starting at the vertex v_i to the vertex v_j tends towards infinity, the probability of being on a vertex v_j only dependents on the degree of the vertex v_j:

$$\forall v_i,\lim_{t\to+\infty}P_{ij}^t=\frac{d(j)}{\sum_k d(k)} \tag{1}$$

Property 2: the probabilities of going from v_i to v_j and from v_j to v_i through a random walker of a fixed length t have a ration that only depends on the degree $d(i)$ and $d(j)$:

$$\forall v_i,\forall v_j,\ d(i)P_{ij}^t=d(j)P_{ji}^t \tag{2}$$

The standard random walker based community detection methods calculate the measure of vertex similarity based on the properties of random walks on graphs, such as Zhou and Lipowsky[10], Pons and Latapy [11]. For example, in Walktrap algorithm of Pons and Latapy, a similarity matrix between vertexes is calculated based on random walks at first, and then an agglomerative algorithm is used to compute the community structure of the network. The distance of two vertexes in Walktrap is defined as:

$$r_{ij}=\sqrt{\sum_{k=1}^{n}\frac{(P_{ik}^t-P_{jk}^t)^2}{d(k)}}=\|D^{-\frac{1}{2}}P_{i\bullet}^t-D^{-\frac{1}{2}}P_{j\bullet}^t\| \tag{3}$$

And let C_1 and C_2 denote two different communities, the distance between two communities can be defined as:

$$r_{C_1C_2}=\|D^{-\frac{1}{2}}P_{C_1\bullet}^t-D^{-\frac{1}{2}}P_{C_2\bullet}^t\|=\sqrt{\sum_{k=1}^{n}\frac{(P_{C_1k}^t-P_{C_2k}^t)^2}{d(k)}} \tag{4}$$

Where $P_{C_j}^t$ denotes the probability of going from community C to vertex v_j in t steps, and $P_{C_j}^t=\frac{1}{|C|}\sum_{v_i\in C}P_{ij}^t$.

The time complexity of calculating the vertex similarity is $O(t|E||V|)$, the time complexity of the agglomerative algorithm is $O(|V|)$. So in the worst case, the time complexity of Walktrap is $O(t|E||V|^2)$.

IV. Community detection Based on Threshold Random Walkers

A. Problem Formulation

The goal of community detection in networks is to find groups of vertexes connected densely. In our real social life, we associate with other people and build our social circles through our social interaction activities. The process of community detection can be viewed as a process to build social circles. Usually we try to associate with new friends based on our existing friends randomly. Furthermore, we would rather to associate with those active people who have big power or large influence.

According above the property 1 of the random walk process, the degree of a node represents the node's power or its influence. The larger degree means the larger attraction to other nodes to associate with it. The larger degree also means more easily to become the core member of a community. And the property 2 of the random walk process shows that the random walk is not symmetrical. It is also consistent with our social interaction activities. The average persons have greater desire to associate with the power persons. Based on those social principles, we propose a scalable community detection method based on the threshold random walks.

Given a network $G=(V,\mathrm{E})$, we first define some concepts as following.

Definition 1. **Active degree:** $\forall v_i\in V$, the active degree of v_i denotes as $a(i)$, which is proportional to the degree of v_i. That is $a(i)=a\cdot d(i)$, where a is a constant number.

For simplicity, we can set $a(i)=d(i)$. The average active degree of the network G denotes as $\bar{a}(G)$, $\bar{a}(G)=\sum_k d(k)/|V|=2|E|/|V|$.

Definition 2. **Active node:** $\forall v_i\in V$, if $a(i)>\bar{a}(G)$, we called it is an active node.

Definition 3. **Common degree:** $\forall v_i,v_j\in V$, the set of their common neighbor nodes is $N(i,j)=N(i)\cap N(j)$.

So the common degree $C(i,j)$ of v_i and v_j can be defined as $C(i,j)=\sum_k (A_{ik}=A_{jk})$. The common degree between a node and a community also can be defined analogously.

Definition 4. **Association degree:** $\forall v_i, v_j \in V$, the association degree from the node v_i to the node v_j is defined as: $R(i,j)=C(i,j)+\beta \cdot a(j)$.

The association degree is asymmetric. It is different with other random walk based similarity measure. Besides the common degree of two nodes, the node's active degree also is considered in our association degree. In social community, it is reasonable that we much more want to associate with the active person.

Definition 5. **Core community:** Given a community C, the core community $core(C)$ is a subset of the community, where the nodes' association degrees are larger than others. i.e. $core(C)\subset C$, $\forall v_i, v_j \in core(C)$, $R(i,j)\geq TR$, where TR is a threshold. The nodes in the core communities are called core nodes, and the others are called non-core nodes.

In our threshold random walk, some active seed nodes are selected according their active degrees. The threshold of the seed nodes is defined as: $TR_{seed}=\alpha \cdot \bar{a}(G)$, where α is an adjustable parameter. When the value of α increments, the number of the seed nodes decrements.

The transition probability of a threshold random walk changes from one node to next according the association degree, which is defined as: $TR_{transition}=\xi \cdot R(i,j)$, where ξ is another adjustable parameter. When the value of ξ increments, the number of the non-core nodes increments.

B. Algorithm Description

Because the nodes in a core community connect closely, it is relative easy to identify the core communities. CD-TRandwalk detects communities with two stages: detecting the core communities according the threshold random walk at first, and then allocating the non-core nodes into the core communities.

The seed nodes of the threshold random walk are selected according the nodes' active degree. We believe that the core members of a community should be active nodes. To detect the core communities, the random walkers start from the seed nodes and only accesses those nodes with large associate degree. This strategy will reduce time cost.

At the start time, a seed node is selected randomly. Then the random walker looks for the next node from the neighbors of the current node. The next node must satisfy following conditions: it has not been accessed, and the associated degree from the current node to the next node is larger than the threshold. We can set a tag to represent whether a node has been accessed. The nodes accessed by the random walker are added into the same core community. If there are not nodes meet these conditions, this random walker finishes. And if there are remaining seed node, then a new seed node is selected to start a new random walker. When all the seed nodes are accessed, the core community detection process finishes. To those non-core nodes, by a voting strategy, we allocate them to a core community which has the maximum common degree with it.

The community detection algorithm based on the threshold random walk can be described as following.

```
// select seed nodes
FOR ∀v_i ∈ V ,
  IF a(i) ≥ TR_seed , THEN push v_i into the seed node stack S_seed .
  END IF
END FOR

// identify the core community
WHILE S_seed ≠ ∅ , DO
  Pop a seed node v_i ∈ S_seed .
  IF v_i has been accessed, THEN reselect another node from S_seed .
  ELSEIF
    Create a new stack S_core , R_max = ∞
    push v_i into the core node stack S_core .
  END IF
  WHILE S_core ≠ ∅ , Pop a node v_i ∈ S_core , add v_i into a core
        node set C_core .
   FOR ∀ v_j , v_j ∈ N (i) , Calculates R(i, j) .
    IF R(i, j) ≥ TR_transit , THEN
      Label v_j has been accessed.
      Let v_i = v_j and random walk with probability R(i, j)/||V|| .
      push v_j into S_core with probability (1 − R(i, j)/|V|) .
    END IF
  End FOR
  END WHILE
  END WHILE

// allocate non-core nodes
FOR ∀v_i ∈ V , v_i ∉ ⋃C_core^j
  Calculates the common degree C(i, C_core^j) .
  Add v_i into the core community C_core^j which has the maximum
    common degree.
END FOR
```

In our CD-TRandwalk, the time complexity of selecting seed nodes is $O(2|E|)$, the time complexity of identifying core community is $O(\bar{a}(G)\cdot|V|)=O(2|E|)$, the time complexity of allocating non-core node is lower than $O(|V|^2/4)$. So the total time complexity of CD-TRandwalk is $O(|V|^2/4+4|E|)$.

V. EXPERIMENT RESULTS AND ANALYSIS

In this section, we present several applications with our CD-TRandwalk. The algorithm is test on the Zachary Karate Club [22], Dophins [23] and Polbooks [24] to uncover the performance influence of the parameters α, β and ξ. Then we compare our CD-TRandwalk with some other algorithms such as Newman Fast[25], ComTector[26], Affinity Propagation (AP) [27, 28] and Zhou[10] on the Zachary Karate Club, American College

Football[6], Netscience[29], Celegansneural[30], ErdÄos[29], CA-GrQc[31], Ca-HepPh[31] and Email-EuAll[31] et al.

A. Experiment Setup

Several datasets with different scales are selected to evaluate our CD-TRandwalk and other algorithms. The datasets used in our experiments are list in the following Table 1.

TABLE I.
THE DATA USED IN EXPERIMENT

Data Name	Node	Edges
Zachary Karate Club	34	78
Dolphins	62	159
Polbooks	105	441
American College Football	115	613
Celegansneural	297	2345
Netscience	1641	2742
Ca-HepPh	5242	28980
Erodas97	5482	8972
Erodas98	5816	9505
Erodas99	6094	9939
CA-GrQc	9877	51971
Scientific Collaboration Networks	39577	175693
Email-EuAll	265214	420045

The data of Zachary Karate Club[22] contains the network of friendships between the 34 members of a karate club at a US university, as described by Wayne Zachary in 1977. Wayne Zachary thinks that there are two communities in this club by actual observation.

The data of Football [6] contains the network of American football games between Division IA colleges during regular season Fall 2000, which compiled by M. Girvan and M. Newman.

The data of Netscience [29] contains a coauthorship network of scientists working on network theory and experiment, which compiled by M. Newman in May 2006. The network was compiled from the bibliographies of two review articles on networks, M. E. J. Newman, SIAM Review 45, 167-256 (2003) and S. Boccaletti et al., Physics Reports 424, 175-308 (2006), with a few additional references added by hand. The version given here contains all components of the network, for a total of 1589 scientists, and not just the largest component of 379 scientists previously published.

The data of Celegansneural [30] describes a weighted, directed network representing the neural network of C. Elegans. The data were taken from the web site of Prof. Duncan Watts at Columbia University. The nodes in the original data were not consecutively numbered, so they have been renumbered to be consecutive. The original node numbers from Watts' data file are retained as the labels of the nodes.

The data of Dolphins [23] contains an undirected social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand, as compiled by Lusseau et al. (2003).

The data of Polbooks [24] contains a network about US politics published around the time of the 2004 presidential election and sold by the online bookseller Amazon.com. Edges between books represent frequent copurchasing of books by the same buyers. The network was compiled by V. Krebs and is unpublished, but can found on Krebs' web site.

The data of Scientific Collaboration Networks [29] contains an updated version of cond-mat, the collaboration network of scientists posting preprints on the condensed matter archive at www.arxiv.org. This version is based on preprints posted to the archive between January 1, 1995 and March 31, 2005.

Paul ErdÄos [32] was one of the most prolific mathematicians in the history, with more than 1500 papers to his name. He is also known as a promoter of collaboration and as a mathematician with the largest number of different co-authors, which was a motivation for the introduction of the ErdÄos number collaboration network [29].

The data of CA-GrQc [31] is come from e-print arXiv and is about the general relativity and quantum universe. This version is based on preprints posted to the archive between January 1, 1993 and April 31, 2003.

The data of Ca-HepPh [31] is come from e-print arXiv and is about the theory of high energy physics. This version is based on preprints posted to the archive between January 1, 1993 and April 31, 2003.

The data of Email-EuAll [31] is come from[5] . This data is the graph of the email of members in Europe a large research institution between October 1, 2003 and May 31, 2005.

In all the above datasets, the real community structures of some dataset are known, such as Zachary Karate Club, Football; and the exact community structures of some dataset are unknown, such as Netscience and Scientific Collaboration Networks. To the dataset which community structures we have known, it is easy to evaluate the performance of the community detection algorithm. To those dataset which community structures we have not known, we need a quality function to evaluate the algorithms. The most popular quality function is the modularity of Newman and Girvan[7]. We also use the modularity to measure the community quality. Modularity can be written as following equation:

$$Q = \sum_{c=1}^{n_c} \left[\frac{l_c}{m} - \left(\frac{d_c}{2m} \right) \right] \tag{5}$$

Here, m is the total number of edges of the graph, n_c is the number of clusters, l_c the total number of edges joining vertexes of the module C and d_c the sum of the degrees of the vertexes of C . The first term of each summand is the fraction of edges of the graph inside the module, whereas the second term represents the expected fraction of edges that would be there if the graph were a random graph with the same expected degree for each vertex.

The computer environment of the experiments is : AMD Dual-Core 2.2GHz CPU, 2.00GB memory and windows 7 OS. We analyze the influence of parameters α and ξ at first, then confirm the accuracy of our CD-TRandwalk in the data of Zachary Karate Club and American College Football, at last compare our CD-TRandwalk with other algorithms such as Newman

Fast, ComTector, Walktrap, AP and Zhou.

B. Parametric Analysis

CD-TRandwalk has three parameters α, β and ξ. To analyze their influence to the community detection, We run CD-TRandwalk in the datasets of Zachary Karate Club [22], Dophins [23], Polbooks [24] and Dophines[23]. In our experiments, α changes between 0 to 3, ξ changes between 0 and 5. Each parameter starts from 0, and increase with a step 0.5. Every combination with different values of the parameters α, β and ξ is tested in the four datasets. Each value combination is executed with ten times, and the average modularity is calculated. Because the influence of β is relatively small, in order to present visually, we only show the average modularity with different value combination of α and ξ as following figure 1-4.

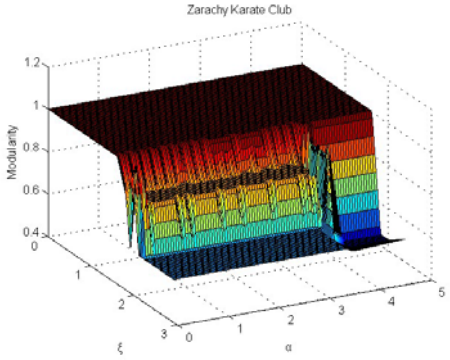


Figure 1. The modularity of different α and ξ in Zarachy Karate Club

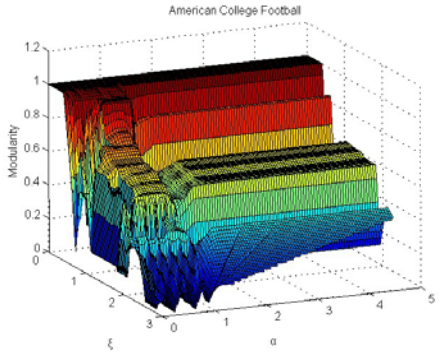


Figure 2. The modularity of different α and ξ in American College Football

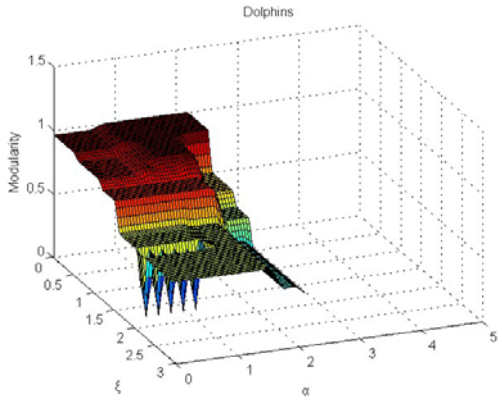


Figure 3. The modularity of different α and ξ in Dolphins

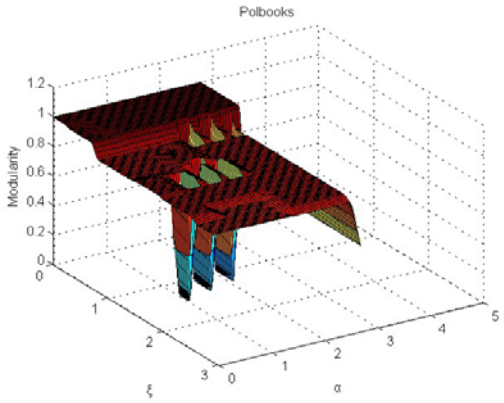


Figure 4. The modularity of different α and ξ in Polbooks

From figure1-4, we can see that α has a big influence to the modularity measures. And it also influence the run time greatly. The reason is that α determines the number of seed nodes. If we set α a small value, there will be a lot of seed nodes, and it will increase the run time. But if we set α a too large value, we cannot find all the core communities. The value of ξ is useful to control the number of the communities. Increasing the value of ξ will increase the number of the communities. In our experiments, we find that when the value of α is between 1 and 2, and the value of ξ is between 1.5 and 3, the CD-TRandwalk can get the best modularity. So in our following experiments, the value of α is 1.5 and the value of ξ is 3. Besides the value of β is 0.8.

C. Experimental Comparison with Other Algorithms

In this section, we compare our CD-TRandwalk with some other algorithms in the datasets of Zarachy Karate Club, American College Football, Netscience, CA-GrQc, ca-HepPh and Email-EuAll. Because the community structures of these datasets are known, we can see the results of community detection directly. Figure 5 is the community detection result of Zarachy Karate Club with our CD-TRandwalk, where the nodes with green color belong the same community, and the nodes with yellow color belong another community. In all of 34 nodes, comparing with the real club relationship, we find that only node 9 and node 10 are allocated wrongly. Figure 7 is the community identification result of our CD-TRandwalk in American College Football. CD-TRandwalk detects all the 115 teams into 7 communities exactly. Moreover, we can see the visual result has small world features and obvious community structures.

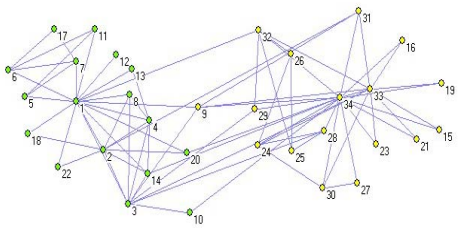


Figure 5. The community structures detected by CD-TRandwalk in Zarachy Karate Club

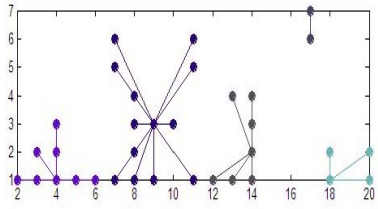


Figure 6. The community detection results by AP in Zarachy Karate Club

We select AP to compare with our CD-TRandwalk, the code of AP download from Frey Lab of PSI Group in University of Toronto[1]. AP is an efficient algorithm for identifying communities in social and biological networks[27, 33]. We find that AP is fast and effective for detecting spherical community; however it is not suitable for small networks. Figure 6 is the community identification result of AP in Zarachy Karate Club. We can see that all the nodes in Zarachy Karate Club are allocated into 5 different communities. Obviously it is not consistent with the real community structures. Figure 8 is the result of AP in American College Football. All the 115 teams are divided into 9 communities. Although it is not consistent with the real communities, it is better than the community identification result in Zarachy Karate Club.

Figure 9 shows the community structures detected by our CD-TRandwalk in the data of Netscience. There are 1461 nodes and 2742 edges. All the nodes are allocated to 95 communities. The modularity of this data is 0.73. Because there are too many edges and nodes in Netscience, it is not easy to see the community structure clearly in Figure 9. We give an enlarged local view of Netscience in Figure 10. From the zoom local view, we find that CD-TRandwalk also can detect many small communities, and reduces the number of isolated nodes. Moreover, the unbalanced community structure is a basic feature of social networks. CD-TRandwalk also can identify unbalanced community structures. Such as in the communities in Netscience, some communities detected by CD-TRandwalk only have four nodes, and some communities detected by CD-TRandwalk have more than 100 nodes. So CD-TRandwalk is scalable and has good performance for unbalanced community structure detection.

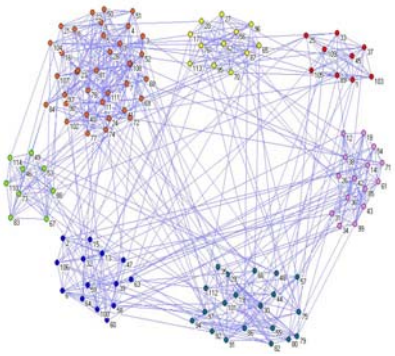


Figure 7. The community structures detected by CD-TRandwalk in American College Football

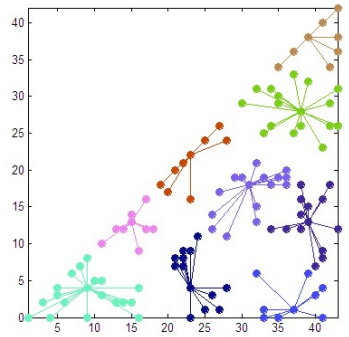


Figure 8. The community detection results by AP in American College Football

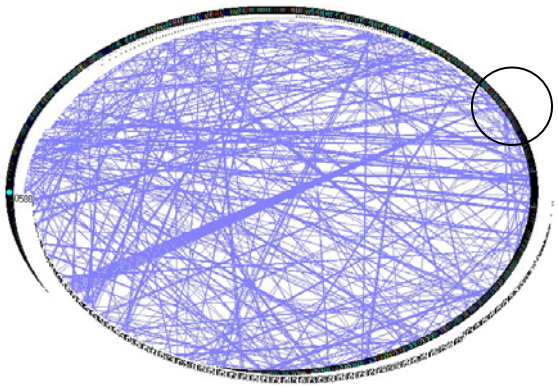


Figure 9. The overall community structures detected by CD-TRandwalk in Netscience

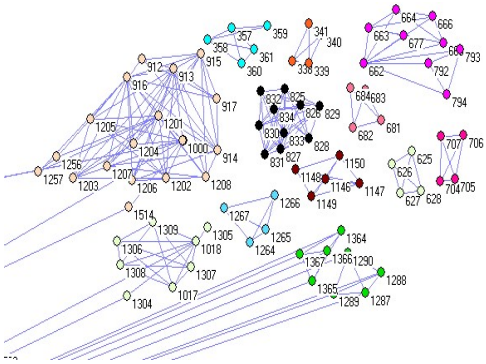


Figure 10. A local community structures detected by CD-TRandwalk in Netscience

Zhou and Walktrap are standard random walk based algorithms. They calculate the similarities between nodes with random walk, and then hierarchically clustering communities according the similarity matrix. In fact to compute the similarity will consume a lot of time. We compare our CD-TRandwalk with Zhou in Zarachy Karate Club, American College Football and Celegansneural. The experiment results are listed in Table 2. We can see that in the three datasets, CD-TRandwalk is faster than Zhou's algorithm. In Celegansneural, Zhou's algorithm runs 39 minutes, but CD-TRandwalk only runs 4.5 seconds. It shows that Zhou's algorithm and Walktrap are not adaptive to large scale network.

[1] http://*www.psi.toronto.edu/affinitypropagation/*

TABLE II.
TIME COST COMPARED WITH ZHOU'S ALGORITHM

Algorithm	Data	Time
Zhou	Zarachy Karate Club	8.9s
CD-TRandwalk	Zarachy Karate Club	0.375s
Zhou	American College Football	54s
CD-TRandwalk	American College Football	1.6s
Zhou	Celegansneural	39min
CD-TRandwalk	Celegansneural	4.5s

Although the similarity calculated by Zhou is very accurate, we find that there exists incorrect community merge phenomenon in Zhou's algorithm during the process of hierarchical clustering. For example, figure 11 gives the hierarchy community structures detected by Zhou's algorithm in Zarachy Karate Club. It is obviously that the hierarchical clusters are not corresponding with the actual relationships.

We also compare our algorithm with Newman Fast and ComTector in some large networks. In the literatures, both the author of the Newman Fast and ComTector Next present that their algorithms can handle large scale networks. We run CD-TRandwalk, Newman Fast and ComTector in the data of Scientific Collaboration Network, ErdÄos 99, ErdÄos 98 and ErdÄos 97. The modularity and the run time are tested in our experiments. The results are listed in Table 3 and Table 4. In Scientific Collaboration Network, the Newman Fast runs 3.7 hours, and ComTector runs 2.2 hours. Our CD-TRandwalk is much faster than Newman Fast and ComTector, it only run 3 minutes. Moreover, the modularity of our algorithm is 0.73, which better than Newman Fast's 0.31 and ComTector's 0.65.

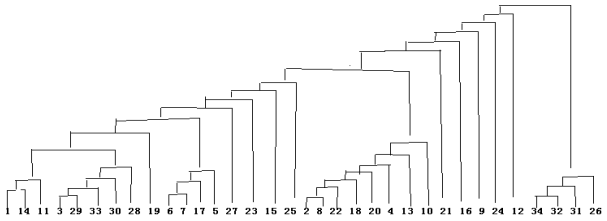


Figure 11. The hierarchy community structures detected by Zhou's algorithm in Zarachy Karate Club

TABLE III.
EXPERIMENT RESULTS COMPARED WITH NEWMAN FAST AND COMTECTOR IN SCIENTIFIC COLLABORATION NETWORK

Algorithm	Data	Q value	Time
Newman Fast	Science Collaboration Network	0.31	3.7h
ComTector	Science Collaboration Network	0.65	2.2h
CD-TRandwalk	Science Collaboration Network	0.73	3min

TABLE IV.
EXPERIMENT RESULTS COMPARED WITH NEWMAN FAST AND COMTECTOR IN ERDÄOS NETWORKS

Algorithm	Data	Q value	Time
Newman Fast	ErdÄos 99	0.35	29s
ComTector	ErdÄos 99	0.69	23s
CD-TRandwalk	ErdÄos 99	0.76	15s
Newman Fast	ErdÄos 98	0.34	35s
ComTector	ErdÄos 98	0.69	26s
CD-TRandwalk	ErdÄos 98	0.76	15s
Newman Fast	ErdÄos 97	0.43	40s
ComTector	ErdÄos 97	0.69	27s
CD-TRandwalk	ErdÄos 97	0.73	12s

TABLE V.
EXPERIMENT RESULTS RUNNING ON SOME OTHER DATASETS

Algorithm	Data	Communities	Q value	Time
CD-TRandwalk	email-EuAll	128	0.78	0.5h
CD-TRandwalk	CA-GrQc	75	0.85	13s
CD-TRandwalk	ca-HepPh	147	0.7	2m

In Table 4, all the experiment results in ErdÄos 99, ErdÄos 98 and ErdÄos 97 show that CD-TRandwalk is better than Newman Fast and ComTector. So we can conclude that CD-TRandwalk is also adaptive to large scale networks. To verify this conclusion, we run CD-TRandwalk in some other large network, such as email-EuAll, CA-GrQc and ca-HepPh, the experiment results show in Table 5. The modularity is from 0.7 to 0.85, and the run time is not long. Although email-EuAll have 265214 nodes and 420045 edges, the run time is only 0.5h. The experiment results in table 5 indicate that our CD-TRandwalk can adapt to different types of social networks.

VI. CONCLUSIONS AND FUTURE WORKS

The CD-Trandwalk is different with the traditional random walk based methods, where the random walk is used to calculate the node similarity. CD-TRandwalk selects active nodes as seed nodes, and detects the core communities through threshold random walkers at first. Because the threshold random walkers start from the active seed nodes and only random walk to those nodes which association degrees are larger than a given threshold, the core community can be detected fast. After that, the remaining non-core nodes are allocated into the core communities according the common degrees between the nodes and the core communities. This has been implemented by simple voting strategy in our CD-TRandwalk.

We have compared our CD-TRandwalk with several community detection algorithms such as Affinity Propagation (AP), Newman Fast, and ComTector in many social networks. The experiment results show that CD-TRandwalk is better than the other methods both in the quality of community detection and run time. Our algorithm also has good adaptation to large scale networks. It also can adapt to unbalance networks, from little communities with only several nodes to large communities with thousands of nodes. Moreover, our CD-TRandwalk is an unsupervised method; it is not need to give the number of communities beforehand the community detection.

There also exist some disadvantages of our algorithm. The first problem is how to select reasonable parameters. There is not automatic technique to implement it now. We can optimize the parameters through maximize the modularity or other community quality measurement; however this will lead to large cost of run time. The second problem is that there possibly generate many isolated nodes with our CD-TRandwalk. In our future work, we will continue our research to resolve these problems. We will also to analyze our CD-TRandwalk algorithm from the theoretical viewports.

ACKNOWLEDGEMENT

This work is supported Science and Technology Foundation of Shenzhen City (JC201005280463A, No.JC201105160498A).

REFERENCES

[1] Satuluri, V. and S. Parthasarathy, *Scalable graph clustering using stochastic flows: applications to community discovery*, in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*2009, ACM: Paris, France. p. 737-746.

[2] Amancio, D.R., et al., *Distinguishing between Positive and Negative Opinions with Complex Network Features*, in *Proceedings of the 2010 Workshop on Graph-based Methods for Natural Language Processing, ACL2010*2010: Uppsala, Sweden. p. 83-87.

[3] Cannataro, M., P.H. Guzzi, and P. Veltri, *Protein-to-protein interactions: Technologies, databases, and algorithms.* ACM Comput. Surv., 2010. 43(1): p. 1-36.

[4] Li, L. and C. Tropper, *A Multiway Design-driven Partitioning Algorithm for Distributed Verilog Simulation.* Simulation, 2009. 85(4): p. 257-270.

[5] Santo, F., *Community detection in graphs.* Physics Reports, 2010. 486(3-5): p. 75-174.

[6] Girvan, M. and M.E.J. Newman, *Community structure in social and biological networks.* Natl. Acad. Sci. USA 99, 2002: p. 7821-7826.

[7] Newman, M.E.J. and M. Girvan, *Finding and evaluating community structure in networks.* Physics Review E, 2004. 69(2): p. 1-15.

[8] M E J Newman, M.G., *Fast algorithm for detecting community structure in networks.* Physics Review E, 2004. 69(6): p. 066133.

[9] Fu, X., et al., *Scalable Community Discovery Based on Threshold Random Walk.* Journal Of Computational Information Systems, 2012. 8(21): p. 8953-8960.

[10] Zhou, H. and R. Lipowsky. *Network Brownian Motion: A New Method to Measure Vertex-Vertex Proximity and to Identify Communities and Subcommunities*. in *International Conference on Computational Science (ICCS2004)*. 2004. Springer-Verlag Berlin Heidelberg.

[11] Pons, P. and M. Latapy, *Computing Communities in Large Networks Using Random Walks Computer and Information Sciences - ISCIS 2005*, p. Yolum, et al., Editors. 2005, Springer Berlin / Heidelberg. p. 284-293.

[12] Yen, L., et al., *Graph nodes clustering with the sigmoid commute-time kernel: A comparative study.* Data Knowl. Eng., 2009. 68(3): p. 338-361.

[13] Tong, H., C. Faloutsos, and J.-Y. Pan, *Random walk with restart: fast solutions and applications.* Knowl. Inf. Syst., 2008. 14(3): p. 327-346.

[14] Delvenne, J.C., S.N. Yaliraki, and M. Barahona, *Stability of graph communities across time scales.* Proceedings of the National Academy of Sciences of the United States of America (PNAS), 2008. 107(29): p. 12755-12760.

[15] Backstrom, L. and J. Leskovec, *Supervised random walks: predicting and recommending links in social networks*, in *Proceedings of the fourth ACM international conference on Web search and data mining*2011, ACM: Hong Kong, China. p. 635-644.

[16] Alamgir, M. and U. von Luxburg. *Multi-agent Random Walks for Local Clustering on Graphs*. in *Data Mining (ICDM), 2010 IEEE 10th International Conference on*. 2010.

[17] Henderson, K. and T. Eliassi-Rad, *Applying latent dirichlet allocation to group discovery in large graphs*, in *Proceedings of the 2009 ACM symposium on Applied Computing*2009, ACM: Honolulu, Hawaii. p. 1456-1461.

[18] Wakita, K. and T. Tsurumi, *Finding community structure in mega-scale social networks: [extended abstract]*, in *Proceedings of the 16th international conference on World Wide Web*2007, ACM: Banff, Alberta, Canada. p. 1275-1276.

[19] Katzir, L., E. Liberty, and O. Somekh, *Estimating sizes of social networks via biased sampling*, in *Proceedings of the 20th international conference on World wide web*2011, ACM: Hyderabad, India. p. 597-606.

[20] Kim, M. and J. Leskovec, *The Network Completion Problem:Inferring Missing Nodes and Edges in Networks* SDM'11, 2011.

[21] Leskovec, J., D. Huttenlocher, and J. Kleinberg, *Predicting positive and negative links in online social networks*, in *Proceedings of the 19th international conference on World wide web*2010, ACM: Raleigh, North Carolina, USA. p. 641-650.

[22] Zachary, W.W., *Aninformation flow model for conflict and fission in small groups.* Journal ofAnthropological Research 33, 1977: p. 452-473

[23] Lusseau, D., et al., *The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations.* Behavioral Ecology and Sociobiology 54, 2003: p. 396-405.

[24] Adamic, L.A. and N. Glance, *The political blogosphere and the 2004 US Election.* WWW-2005 Workshop on the Weblogging Ecosystem, 2005.

[25] Clauset, A., M.E.J. Newman, and C. Moore, *Finding community structure in very large networks.* physics Review E, 2004. 70(6): p. 1-6.

[26] Du, N., et al., *Community detection in large-scale social networks*, in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*2007, ACM: San Jose, California. p. 16-25.

[27] Lai, et al., *Identification of community structure in complex networks using affinity propagation clustering method.* Modern Physics Letters B 22(16), 2008: p. 1547–1566.

[28] Frey, B.J. and D. Dueck, *Clustering by Passing Messages Between Data Points.* Science, 2007. 315(5814): p. 972-976.

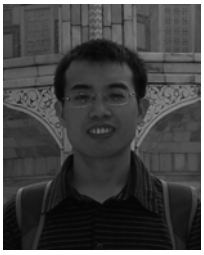
[29] Newman, M.E.J., *The structure of scientific collaboration networks.* Proc. Natl. Acad. Sci. USA 98, 2001: p. 404-409.

[30] Watts, D.J. and S.H. Strogatz, *Collective dynamics of `small-world' networks.* Nature 393, 1998: p. 440-442.

[31] Wan, X. and H. Su, *Recent Progress in Control of Complex Dynamical Networks.* Advances in Mechanics, 2008. 38(06): p. 751-765.

[32] Batagelj, V. and A. Mrvar, *Some Analyses of ErdosCollaboration Graph.* Social Networks, 2000. 22(2): p. 173-186.

[33] Jia, C., Y. Jiang, and J. Yu, *Affinity propagation on identifying communities in social and biological networks*, in *Proceedings of the 4th international conference on Knowledge science, engineering and management*2010, Springer-Verlag: Belfast, Northern Ireland, UK. p. 597-602.



Xianghua Fu received the M.Sc. degree from the Northwest A&F University, Yangling, China, in 2002 and Ph.D. degree in computer science and technology from Xi'an Jiaotong University, Xi'an, China, in 2005.

He is an associated professor and postgraduate director at College of Computer and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include machine learning, data mining, information retrieval, and natural language processing.

Dr. Fu is the member of China Computer Federation, project reviewer of Natural Science Foundation of Guang-dong, project review of Nanshan science and technology of Shenzhen.

Chao Wang received his B.Sc. degree in computer science and technology from Information Engineering University of the People's Liberation Army, Zhenzhou, China, in 2008.

He is M.Sc. degree candidate in computer application technology of College of Computer and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include data mining, information retrieval.

Zhiqiang Wang received his M.Sc. degree in automation from Wuhan University of Technology, Wuhan, China, in 1992.

He i**s** a professor and post postgraduate director at College of Computer and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include multimedia information processing.

Zhong Ming received his Ph.D. degree in computer science and technology from Sun Yat-Sen University, Guangzhou, China, in 2003.

He i**s** a professor and post postgraduate director at College of Computer and Software Engineering, Shenzhen University, Shenzhen, China. His research interests include software engineering, web intelligence.