COMMUNITY DETECTION IN SOCIAL NETWORKS: AN OVERVIEW

Deepjyoti Choudhury[1], Arnab Paul[2]

[1]*Department of Information Technology, Triguna Sen School of Technology, Assam University, Silchar*
[2]*Department of Information Technology, Triguna Sen School of Technology, Assam University, Silchar*

Abstract

A social network can be defined as a set of people connected by a set of people. Social network analysis provides both a visual and a mathematical analysis of human relationship. The investigation of the community structure in the social network has been the important issue in many domains and disciplines. Community structure assumes more significance with the increasing popularity of online social network services like Facebook, MySpace, or Twitter. This paper reflects the emergence of communities that occur in the structure of social networks, represented as graphs. We have mainly discussed various community detection algorithms in real world networks in this paper. This paper represents as an overview of the community detection algorithms in social networks.

Keywords: *node, graph, community, algorithms, clustering*

----------***----------

1. INTRODUCTION

One of the most relevant features of graphs representing real systems is community structure, or clustering, i.e. the organization of vertices in clusters, with many edges joining vertices of the same cluster and comparatively few edges join vertices of different clusters. Detecting communities is of great importance in sociology, biology and computer science, disciplines where systems are often represented as graphs. Real networks are not random graphs, as they display big in homogeneities, revealing a high level of order and organization. The degree distribution is broad, with a tail that often follows a power law. Therefore, many vertices with low degree coexist with some vertices with large degree. Furthermore, the distribution of edges is not only globally, but also locally inhomogeneous, with high concentrations of edges within special groups of vertices, and low concentrations between these groups. This feature of real networks is called community structure, or clustering.

The term "Community" first appeared in the book "Gemeinschaft und Gesellchaft" published in 1887. There is no unique definition of community till to present which is widely accepted in social networks. A variety of definitions of community have been proposed according to different sides, which can be mainly classified to three categories: intuitive definition, functional definition and definition from the process of algorithm.

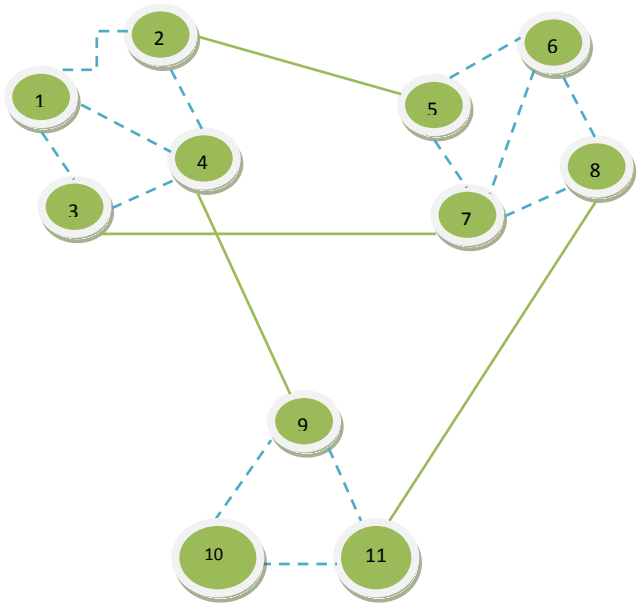

Fig 1: A simple graph with three communities.

In the above figure, we have seen there are three communities in which all the nodes contain in a community are dense intra-connected with each other and sparse inter-connected with the nodes contain in another community. In a community, nodes are connected with each other based on their human relationship like friendship, colleague etc.

In computer science, community can be regarded as sub-graphs of network. The whole complex network can be generated as a graph, which is consisted of many sub-graphs. Connection between nodes in a sub-graph is intra-dense, while connection between the nodes among sub-graphs is relatively sparse. Newman call this sub-graph community structure [1].This

definition emphasis on structural characteristic of community, with links inter community more dense than intra-community, which can be measured by degree of the module [2]. Most existing community detection algorithms are limited to deal with non-overlapping communities, which do not work well on overlapping community detection [3]. Overlapping community detection involves community definition, as well as the evaluation metric which especially focuses on analysis and comparison of the existing overlapping community detection algorithms including the basic ideas of the algorithms, and its performance. M. Girvan and M. E. J. Newman [4] had proposed community structure and detection algorithm in social and biological networks. The ability to detect community structure in a network could clearly have practical applications. Communities in a social network might represent real social groupings.

Community detection in dynamic networks [5] is a challenging task since such networks are multi-graphs and a pair of nodes can have links appearing or disappearing at different time points. Instability of link configurations leads to constant changes in graph partitions between slices of a multi-graph which make community detection difficult in dynamic networks. Mobility is used to be a network transport mechanism for distributing data in many networks. However, many mobility models are set up based on individual movement case which ignores the fact that peer nodes often carried by people and thus move in community pattern according to some kind of social relation. GuoDong Kang et.al have proposed two new mobility models [6], as called Social Community Partner Mobility Model (SCP) and Social Community Leader Mobility Model (SCL) in 2011.

Minimum-cut method is one of the oldest algorithms for dividing networks into parts. This method uses in load balancing for parallel computing in order to minimize communication between processor nodes. However, this method always finds communities regardless of whether they are implicit in the structure, and it can only find a fixed number of them. So it is less than ideal for finding community structure in general networks [1]. In simulation environment, SCP model [6] will regard the office, restaurant and cinema to be small squares in the given simulation area. When the community moves from the office to the restaurant, the restaurant is called the community destination. In simulation, the community destination is the square which is chosen to correspond to the restaurant. When the community moves from restaurant to the cinema, one new square in the simulation will be chosen as a new community destination which corresponds to the cinema. In Partner Movement Case, the members in one community will also have their own destinations in the restaurant or cinema.

Jie Jin et. Al [7] have proposed a new center-based method, which is especially designed for weighted networks. And the method is also suitable for large-scale network because of its low computational complexity. They demonstrated the method on a synthetic network and two real-world networks. There are always some important nodes in the real networks, which are often the cores of the community and organize the whole community structure.

Most known techniques for community detection use only the information about the linkage behavior [8] for the purposes of community prediction and clustering. Some recent work has shown that the use of node content can be helpful in improving the quality of the communities. Moreover, we can see that edge content [9] provides a number of unique distinguishing characteristics of the communities which cannot be modeled by node content. Some examples of networks with edge-content are as follows:

In email networks, a communication between two participants can be considered an edge, which has content corresponding to the text which is communicated between two participants. Clearly, participants containing the similar content of communication are much likely to belong to the same community.

Complex networks in nature and society range from the immune system and the brain to social, communication and transport networks [10]. The key issue to develop these algorithms is able to automatically detect communities in complex networks. Camelia Chira et. al [10] propose a new fitness function for the assessment of community structures quality which is based on the number of nodes and their links inside a community versus the community size further reported to the size of the network.
In this paper, we have first discussed the fundamental concepts of community in social networks like walk, trail, path and chain. We have elaborated various community detection algorithms in social networks in the next section. We have separated the community detection algorithms under the sub-section as traditional methods, divisive algorithm, modularity based methods, spectral algorithms and dynamic algorithms. Then we have concluded this paper.

2. FUNDAMENTAL CONCEPTS OF COMMUNITY IN SOCIAL NETWORKS

To study and analyze the community structure in social networks the following concepts are necessary which are given in below:

2.1 Walk

A walk in a network is described as a sequence of nodes which hold the relations among themselves [11]. A walk starts with the source node and end with the target node. While a walk starts with a source node and ends with the same node then that walk is considered as the closed walk. In Fig. 2 we can see A-B-C-F-G is a walk and D-C-F-E-D is a closed walk in that network. C. Gkantsidis, M. Mihail, and A. Saberi [12] have described the effectiveness of the random walks for searching and

construction of unstructured peer-to-peer network. With a natural way, Supervised Random Walk [13] combines the information from the network structure with the characteristics of the nodes and the edge level attributes.

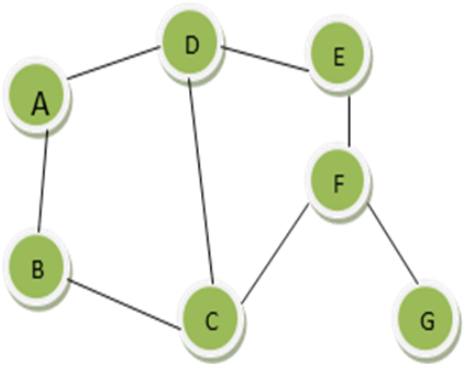

Fig 2: An undirected graph is used to show walk, trail and path

2.2 Trail

A trail in a network is considered as a walk between the nodes where a relation occurs between the nodes is never repeated for twice. The same nodes in a trail can be part of another trail for several times but the same relation between the nodes occurs not more than once. In Fig. 2 we can take A-D-E is a trail but F-E-D is not a trail since the relation between D and E is already established once in the trail A-D-E and the graph is undirected. The length of a trail is calculated based on the total number of relations in it. Trail supports distance-sensitive tracking of mobile objects in a network and trail does not partition the network into a hierarchy of clusters [14].

2.3 Path

A path is a walk where each and every node and each other relation is used at most one time in a network. The single exception is carried out to path is closed path where the path begins and ends with the same node. In Fig. 2 we can see C-D-E is a path but D-A-B is not a path again since the node D has already been used. The length of a path can be calculated based on the total number of its link. There may two paths between the nodes A and E. Let A is the source node and E is the target node then the paths will be A-D-E and A-B-C-F-E. If all the nodes appeared in those two paths are connected based on the social relationship then we can say both the paths are social trust paths [15]. There may several social trust paths between the source node and the target node. Trustworthiness of the target node can be generated and evaluated by the source node based on the trust information of all the intermediate nodes along the path. This process is called as the trust propagation [16][17]. IBM has launched one online social network for its employee which is named as Small Blue. There are 16 social paths at maximum with not more than 6 employees between the source and the target employees in this system. Peng Huatao [18] has described the four hypotheses about path selection in a social network.

2.4 Chain

A chain is a walk in a social network which is considered only in directed graph. Let us consider Fig. 2 is the directed graph. If there is single flow from the node A to the node C then A-B-C is a chain but C-B-A is not a chain.

3. COMMUNITY DETECTION ALGORITHMS IN SOCIAL NETWORKS

There are several community detection algorithms in social networks. Several methods have been proposed to identify and analyze the communities in social networks. We have briefly discussed here some of those algorithms in below:

3.1 Traditional Methods

There are three types of traditional methods to detect communities in a social network.

3.1.1 Graph Partitioning

Graph partitioning method represents to divide the nodes in g groups of predefined size, such that the number of edges lying between the groups is minimal. The number of edges running between clusters is called cut size. Fig. 3 presents the solution of the problem for a graph with six nodes, for g = 2 and clusters of equal size.

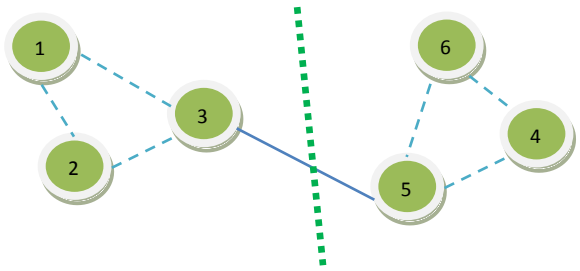

Fig 3: Graph partitioning method. The green dashed line shows the solution of the minimum bisection problem for the graph illustrated.

Most variants of the graph partitioning problem are NP-hard. If the solutions are not necessarily optimal, then also there are several algorithms that can do a good job [19]. Many algorithms perform a bisection of the graph. Generally, partitions into more than two groups are achieved by iterative bi-sectioning.

The Kernighan-Lin algorithm [20] is one of the earliest proposed methods and is still frequently used. The problem of partitioning electronic circuits onto boards motivated the authors as the nodes contained in different boards need to be linked to each other with the least number of connections. The Kernighan-Lin algorithm was extended to get partitions in any number of parts [21]; however the run-time and storage costs increase rapidly with the number of clusters.

There are several efficient routines to compute maximum flows in graphs, like the algorithm of Goldberg and Tarjan [22]. In the graph of the World Wide Web, Flake et al. [23][24] have used maximum flows to identify communities. The web graph is directed but Flake et al. treated the edges as undirected for the purposes of the calculation. The internal degree of each node must not be smaller than its external degree [25] in a community. So, Web communities are defined to be strong. An artificial sink t is added to the graph and one calculates the maximum flows from a source node s to the sink t: the corresponding minimum cut identities the community of node s, provided s shares a sufficiently large number of edges with the other vertices of its community.

It is necessary to provide as input the number of groups and their sizes in some cases. So, Algorithms for graph partitioning are not good for community detection. Besides, it is not a reliable procedure using iterative bi-sectioning to split the graph in more pieces.

3.1.2 Hierarchical Clustering

Hierarchical clustering is a widely used data analysis tool. The idea behind this clustering is to build a binary tree of data that merges similar groups of points. If the graph is split then it is not easy to know the total number of clusters. If the graph is in hierarchical structure with small groups included within larger groups, in that case hierarchical clustering algorithm [26] may be used.

3.1.3 Spectral Clustering

Donath and Hoffmann [28] contributed first on spectral clustering in 1973. They used eigen vectors of the adjacency matrix to partition the graph. Spectral clustering makes use of eigen values of the similarity matrix of the data. The similarity matrix is provided as an input and consists of a quantitative assessment of the relative similarity of each pair of points in the dataset. Andrew Y. Ng et. al [27] have analysed the algorithm of spectral clustering as the ideal case and the general case.

3.2 Divisive Algorithm

3.2.1 Newman-Girvan Algorithm [4]

This algorithm follows the steps stated in below:
1. Calculate the betweenness for all edges in the network.
2. Remove the edge with the highest betweenness.
3. Recalculate betweennesses for all edges affected by the removal.
4. Repeat from step 2 until no edges remain.

If a graph contains groups that are inter-connected each other and loosely connected by few edges, then all shortest paths between different groups must go along one of these few edges. Thus, the edges connecting groups will have high edge betweenness. Betweennesses can be calculated by using the fast algorithm of Newman [29], which calculates betweenness for all m edges in a graph of n vertices in time O(mn). Because this calculation has to be repeated once for the removal of each edge, the entire algorithm runs in worst-case time $O(m^2n)$. Rattigan et al. [30] proposed a fast version of Newman-Girvan algorithm in 2007.

3.3 Modularity-Based Methods

High values of modularity represent good partitions of a graph. There are four techniques we have discussed in below:

3.3.1 Greedy Techniques

Newman [2] proposed first greedy method to maximise modularity. It is a hierarchical clustering method where edges do not contain in the graph initially; edges are added one by one during the procedure.

3.3.2 Simulated Annealing

To get global optimization, simulated annealing [31] is probabilistic procedure used in different fields and problems. This procedure consists of the space of possible states looking for the maximum global optimum of a function F. Guimera et al. [32] first applied simulated annealing for modularity optimization. The standard implementation of them [33] combines two types of moves: local moves, where a single node is shifted from one cluster to another randomly; and global moves, which consist of mergers and splits of communities.

3.3.3 Extremal Optimization

Boettcher and Percus [34] proposed Extremal optimization and it is a heuristic search procedure.

This technique is based on the optimization of local variables. Duch and Arenas [35] used this technique for modularity optimization. Modularity can be measured as a sum over the nodes in the graph. We can get a fitness measure for each node by dividing the local modularity of the node by its degree. Degree of the node does not define the measure.

3.3.4 Spectral Optimization

By using the eigen values and eigen vectors of a spectral matrix, modularity can be optimized. Wang et al. [36] used community vectors to achieve high-modularity partitions into a number of communities smaller than a given maximum. If the eigenvectors is taken corresponding to the two largest eigenvalues, then we can obtain a split of the graph in three clusters. In 2009, Richardson et al. [37] presented a fast technique to achieve graph tri-partitions with large modularity along these lines.

3.4 Spectral Algorithms

In the previous sub-section, we have learnt the spectral properties of graph matrices that are frequently used in finding the partitions in a graph. In 2005, Slanina and Zhang [38] have shown that if the graph has a clear community structure, then eigen vectors of the adjacency matrix may be localized. In 2009, Mitrovic and Tadic [39] presented a comprehensive analysis of spectral properties of modular graphs.

In 2007, Alves [40] used eigen values and eigen vectors of the Laplacian matrix to compute the effective conductances for pairs of nodes in a graph. We compute the transition probabilities by enabling the conductances for a random walker moving on the graph, and from the transition probabilities, we can build a similarity matrix between the node pairs. Hierarchical clustering is applied to join nodes in communities. If we need to compute the whole spectrum of the Laplacian matrix, the time taken by this algorithm is $O(n^3)$, i.e. the algorithm proposed by Alves [40] is slow.

3.5 Dynamic Algorithms

There are three algorithms we have discussed in this section: Spin models, Random walk, and Synchronization.

3.5.1 Spin Models

In statistical mechanics, the Potts model [41] is the most popular models. This model elaborates a system of spins that can be in q different states. It favours spin alignment such that all spins are in the same state at zero temperature. That means the interaction is ferromagnetic in this model. The ground state of the system may not be the one where all spins are aligned if antiferromagnetic interactions are also present. But, different spin values coexist in homogeneous clusters in a state. If Potts [41] spin variables are assigned to the nodes of a graph with community structure, then the structural groups could be recovered from like-valued spin clusters of the system while the interactions are between neighbouring spins, as there are many interactions inside communities than outside. Based on Potts [41] model, in 2004, Reichardt and Bornholdt [42] proposed a method to detect communities that maps the graph onto a zero-temperature q-Potts model with nearest-neighbour interactions.

3.5.2 Random Walk

In 1995, Hughes [43] showed that random walk can be useful to detect the clusters in a graph. If a graph contains several clusters, a random walker spends a long time inside a cluster due to the high intra-connections among all the nodes. All of clustering algorithms based on the random walk can be trivially extended to the case of weighted graphs.

In 2004, Zhou and Lipowsky [44] used biased random walkers, where the bias happens to the fact that walkers usually move towards the nodes sharing a large number of neighbours with the starting node in a graph. A proximity index is defined to show that how much a pair of nodes is closer to all other nodes in the graph. The procedure is called NetWalk to detect the communities in a graph, where NetWalk is a hierarchical clustering method, where the proximity defines the similarity between nodes. The time complexity of this method is O(n^3). In 2008, Weinan et al. [45] described that the best partition of a graph in k communities, where the chain describing a random walk on the meta-graph provides the best approximation of the full random walk dynamics on the whole graph.

3.5.3 Synchronization

Synchronization [46] is an excellent process occurs in the systems and interacts among the units in nature and technology. All the units of the system are in the similar state at every moment while the system is in synchronized state. To detect the communities in a real world network, synchronization can also be applied. In 2007, Boccaletti et al. [47] have designed a method for community detection applying the concept of synchronization.

CONCLUSIONS

In this paper, we have discussed fundamental concepts of community in social network. Then in the next section, we have elaborated the existing algorithms to detect the communities in social networks. In this paper, we have briefly illustrated the traditional methods, divisive algorithm, modularity based methods, spectral algorithms and dynamic algorithms to detect the communities in real world networks. We hope the concepts demonstrated in this paper to detect the communities in real world networks will help us to study the community structure in social networks deeply in future.

REFERENCES

[1] M E J Newman, "Detecting Community Structure in Networks", *Eur. Phys. J. B 38*, pp 321-330 2004.

[2] M.E.J. Newman and M Girman, "Finding and evaluation community structure in networks", *Physical Review E, 69(2)*, 2004.

[3] L. Zhubing, W. Jian, and Li yuzhou, "An Overview on Overlapping Community Detection", *The 7th International Conference on Computer Science & Education (ICCSE 2012)* , Melbourne, Australia , July 14-17, 2012.

[4] M. Girvan and M. Newman, "Community Structure in Social and Biological Networks", *Proceedings of the National Academy of Scinces*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002.

[5] L.C. Huang, T.J Yen, and S.C.T. Chou, "Community Detection in Dynamic Social Networks".

[6] G. Kangl, M. Diaz, T. Perennou, P. Senac, and L. Xul, "Mobility Model Based on Social Community Detection Scheme", *2011 Cross Strait Quad-Regional Radio Science and Wireless Technology Conference.*

[7] J. Jin, L. Pan, C. Wang, and J. Xie, "A Center-based Community Detection Method In Weighted Networks", *2011 23rd IEEE International Conference on Tools with Artificial Intelligence.*

[8] Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks", *In Phys. Rev. E 70, 066111*, 2004.

[9] G.J. Qi, C. C. Aggarwal, and T. Huang, "Community Detection with Edge Content in Social Media Networks", *2012 IEEE 28th International Conference on Data Engineering.*

[10] C.Chira, A. Gog, and D. Icl˘anzan, "Evolutionary Detection of Community Structures in Complex Networks: a New Fitness Function", *WCCI 2012 IEEE World Congress on Computational Intelligence*, Brisbane, Australia, June, 10-15, 2012.

[11] J. Rupnik, "Finding Community Structure in Social Network Analysis–overview", *Journal of Mathematical Sociology*, 2006.

[12] Gkantsidis, M. Mihail, and A. Saberi, "Random Walks in Peer-to-Peer Networks", *IEEE INFOCOM*, pp. 1–12, 2004.

[13] L. Backstrom and J. Leskovec, "Supervised Random Walks: Predicting and Recommending Links in Social Networks", *WSDM '11*,Feb. 9-12, 2011.

[14] Kulathumani, A. Arora, M. Sridharan,and M. Demirbas, "Trail: A Distance-Sensitive Sensor Network Service for Distributed Object Tracking", *ACM Transactions on Sensor Networks*, vol. 5, no. 2, article 15, pp. 1–40, Mar. 2009.

[15] G. Liu, Y. Wang, and M. A. Orgun, "Finding K Optimal Social Trust Paths for the Selection of Trustworthy Service Providers in Complex Social Networks", *IEEE International Conference on Web Services*, pp. 41–48, 2011.

[16] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, "Propagation of Trust and Distrust", *Proceeding of the thirteenth International Conference on World Wide Web, ACM*, pp. 403–412, May. 17-22, 2004.

[17] Gray, J. Seigneur, Y. Chen, and C. Jensen, "Trust Propagation in Small Worlds", *Trust Management*, 2003.

[18] P. Huatao, "Path Selection for Social Network Evolution Map Formation of Start-up Enterprises", *International Conference on Computer and Communication Technologies in Agriculture Engineering, IEEE*, pp. 47–50, 2010.

[19] Pothen, "Graph Partitioning Algorithms with Applications to Scientific Computing", *Technical Report*, Norfolk,VA, USA, 1997.

[20] Kernighan, B. W., and S. Lin, *Bell System Tech. J. 49, 291*, 1970.

[21] Suaris, P. R., and G. Kedem, *IEEE Trans. Circuits Syst.35, 294*, 1988.

[22] Goldberg, A. V., and R. E. Tarjan, *Journal of the ACM 35, 921*, 1988.

[23] Flake, G. W., S. Lawrence, and C. L. Giles, *Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ACM Press, Boston, USA),*pp. 150-160, 2000.

[24] Flake, G. W., S. Lawrence, C. Lee Giles, and F. M. Coetzee, *IEEE Computer 35, 66,* 2002.

[25] Radicchi, F., C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, *Proc. Natl. Acad. Sci. USA 101, 2658,* 2004.

[26] Hastie, T., R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning (Springer, Berlin, Germany),ISBN 0387952845*, 2001.

[27] A.Y. Ng, M.I. Jordan, Y. Weiss, "On Spectral Clustering: Analysis and Algorithm", *Stanford AI Lab.*

[28] Donath, W., and A. Hoffman, *IBM Journal of Research and Development 17(5), 420,* 1973.

[29] Newman, M. E. J. *Phys. Rev. E 64, 016131*, 2001.

[30] Rattigan, M. J., M. Maier, and D. Jensen, *ICML '07: Proceedings of the 24th international conference on Machine learning (ACM, New York, NY, USA)*, pp. 783-790, 2007.

[31] Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi, *Science 220, 671*, 1983.

[32] Guimera, R., M. Sales-Pardo, and L. A. N. Amaral, *Phys. Rev. E 70(2), 025101 (R)* , 2004.

[33] Guimera, R., and L. A. N. Amaral, *Nature 433, 895*, 2005.

[34] Boettcher, S., and A. G. Percus, *Phys. Rev. Lett. 86, 5211*, 2001.

[35] Duch, J., and A. Arenas, *Phys. Rev. E 72(2), 027104*, 2005.

[36] Wang, G., Y. Shen, and M. Ouyang, *Comput. Math. Appl. 55(12), 2746*, 2008.

[37] Richardson, T., P. J. Mucha, and M. A. Porter, *Phys. Rev. E 80(3), 036111*, 2009.

[38] Slanina, F., and Y.C. Zhang, *Acta Phys. Pol. B 36, 2797*, 2005.

[39] Mitrovic, M., and B. Tadic, *Phys. Rev. E 80(2), 026123*, 2009.

[40] Alves, N. A., *Phys. Rev. E 76(3), 036101*, 2007.

[41] Wu, F. Y., *Rev. Mod. Phys. 54(1), 235*, 1982.

[42] Reichardt, J., and S. Bornholdt, *Phys. Rev. Lett. 93(21), 218701*, 2004.

[43] Hughes, B. D., "Random Walks and Random Environments: Random Walks Vol. 1", *Clarendon Press, Oxford, UK*, 1995.

[44] Zhou, H., and R. Lipowsky, *Lect. Notes Comp. Sci. 3038, 1062*, 2004.

[45] Weinan, E., T. Li, and E. Vanden-Eijnden, *Proc. Natl. Acad. Sci. USA 105, 7907*, 2008.

[46] Pikovsky, A., M. G. Rosenblum, and J. Kurths, "Synchronization : A Universal Concept in Nonlinear Sciences", *Cambridge University Press, Cambridge, UK*, 2001.

[47] Boccaletti, S., M. Ivanchenko, V. Latora, A. Pluchino, and A. Rapisarda, *Phys. Rev. E 75(4), 045102*, 2007.