

Seeding the Kernels in graphs: toward multi-resolution community analysis

Jie Zhang^{1,4}, Kai Zhang², Xiao-ke Xu^{1,3}, Chi K Tse¹
and Michael Small¹

¹ Department of Electronic and Information Engineering,
Hong Kong Polytechnic University, Hong Kong,
People's Republic of China

² Life Science Division, Lawrence Berkeley National Laboratory,
Berkeley, CA 94720, USA

³ School of Communication and Electronics Engineering,
Qingdao Technological University, Qingdao 266520,
People's Republic of China

E-mail: enzhangjie@eie.polyu.edu.hk

New Journal of Physics **11** (2009) 113003 (12pp)

Received 24 August 2009

Published 2 November 2009

Online at <http://www.njp.org/>

doi:10.1088/1367-2630/11/11/113003

Abstract. Current endeavors in community detection suffer from the resolution limit problem and can be quite expensive for large networks, especially those based on optimization schemes. We propose a conceptually different approach for multi-resolution community detection, by introducing the *kernels* from statistical literature into the graph, which mimic the node interaction that decays locally with the geodesic distance. The modular structure naturally arises as the patterns inherent in the interaction landscape, which can be easily identified by the hill climbing process. The range of node interaction, and henceforth the resolution of community detection, is controlled via tuning the kernel bandwidth in a systematic way. Our approach is computationally efficient and its effectiveness is demonstrated using both synthetic and real networks with multiscale structures.

⁴ Author to whom any correspondence should be addressed.

Contents

| | |
|--|-----------|
| 1. Introduction | 2 |
| 2. Constructing interaction landscape by seeding kernels in graph | 3 |
| 3. Application to synthesized and real networks | 5 |
| 4. Conclusion and discussion | 11 |
| Acknowledgments | 11 |
| References | 11 |

1. Introduction

A community, or module, defined as a subgraph in a network, whose nodes are more densely inter-connected compared with the rest of the graph, is a universal feature in real-world complex networks lying between the microscopic and macroscopic levels of a system. With the explosion of available data in various disciplines including biology, sociology and engineering, unraveling community structures of the network [1]–[3] has become an increasingly important task, which holds the key to understanding the relationship between the structure and the function of real-world complex systems, ranging from a single cell to the human brain.

Girvan and Newman [4] first introduced a divisive algorithm, where the edges are removed successively according to their ‘betweenness’. A major step forward was made by the same authors who proposed the concept of modularity Q that evaluates the quality of a partition of a network into communities. The main stream of relevant algorithms either rely on or aim directly at optimizing Q [5]–[9]. Other approaches include spectral bisection [10] from computer scientists, hierarchical clustering developed by sociologists [11], and those based on physical intuition [12] and information-theoretic measures [13, 14].

Although it has attracted extensive attention, community detection still remains a challenging problem. Computationally, the exact optimization of Q is untractable for large networks due to the non-deterministic polynomial-time (NP)-hard nature of the problem [15]. On the other hand, many practical networks possess multiscale structures due to their hierarchical [16]–[18] or self-similar organizations [19]. In such circumstances, modularity optimization may fail to detect communities below a certain scale which is known as the resolution limit [20]. The resolution limit suggests the existence of a multiscale modular structure in the real-world networks, which calls for community detection schemes that are capable of probing the network at multiple, varying scales. To this end, Arenas proposed a modification on the modularity by a scale-dependent re-scaling factor, such that the modular structure of the network can be analyzed under different scales [21]. Serrano *et al* [22] use a filtering method to extract the relevant connection backbone from multiscale networks, which preserves the links that represent statistically significant deviations to the null model for local assignment of weights to links. This method does not ignore small-scale interactions and can operate at all scales specified by the weight distribution. In addition, a new method has been developed to extract the hierarchical organization of a network, which utilizes the local maxima of the so-called modularity landscape [23]. In fact, the concept of ‘multiscale’ was implicitly contained in the traditional hierarchical clustering [11], where a series of partitions takes place, ranging from a single cluster containing all nodes to N clusters, each containing a single object, represented by dendrograms. The community detection performed at different levels of the dendrograms is multiscale in nature.

In this paper, we propose a conceptually different, heuristic approach that can resolve the multiscale community structure of a network without a resolution limit. Our key observation is that communities can be identified due to the existence of ‘borderline’ nodes, which separate the groups of densely inter-connected vertices. The borderline nodes typically have few edges associated with them, while in contrast nodes inside communities usually have denser connections (for example, the ‘core’ nodes are usually involved in a large number of connections). Therefore, the amount of connections associated with each node (possibly include the non-immediate, higher order links of a node) is expected to provide valuable information on identifying frontiers of community, hence their global structure. To measure it, we introduce a well-known smoothing technique called *kernel density estimation* in the statistical literature [24–26]. Basically, we seed a kernel function at each node to describe its interaction with its neighbors, with the strength of interaction decaying monotonously with their shortest path distance. By doing this, we transform the topology of connection into a *multiple interaction system* where the accumulated interaction levels of the nodes reflect intrinsic structures of the communities. Note that this scheme takes into account the high level topological information by considering node interaction at various levels. In fact, the scale of node interactions is effectively controlled by the bandwidth of the kernel, leading naturally to a convenient, multi-resolution approach. A small bandwidth causes the nodes to be involved only into local interactions thus the small-scale structure of the network is revealed. A large bandwidth, on the contrary, enforces global interaction among the nodes and provides information about the large-scale organization of the network.

2. Constructing interaction landscape by seeding kernels in graph

Suppose we have a network of n nodes with adjacency matrix A , where $A_{ij} = 1$ if an edge connects nodes i and j and $A_{ij} = 0$, otherwise. Let G_{ij} be the shortest path (geodesic) distance between nodes i and j . For each node i , we will then seed a smoothly decaying function called ‘kernel’ to model its *interaction* with its neighbors in the form of $K(G_{ij}/h)$. Here, $K(\cdot)$ is a non-negative, symmetric kernel function usually satisfying [24]

$$\int_{\mathbb{R}} K(x)dx = 1, \quad \int_{\mathbb{R}} x K(x)dx = 0, \quad \lim_{x \rightarrow \infty} x K(x) = 0,$$

and h is the *bandwidth* of the kernel that controls its spread. A popular example is the Gaussian $K(x/h) = \exp(-x^2/2h^2)$ (other choices include linear, quadratic or cubic kernels). As can be seen in figure 1, the interaction between a node pair will decrease smoothly with their geodesic distance, and the decaying rate is controlled by h . In other words, the node i will only impact its neighbors $j \in \mathcal{N}_i$, where the geodesic distance G_{ij} is not too large compared with the bandwidth h . Therefore, by introducing the kernel K with a tunable bandwidth, we build a node interaction system where the range of interactions are flexibly controlled by varying h .

Having defined the interactions between all pairs of nodes in the network, we can measure the level of connection for each node i by summing up all the interactions that node i has been actively involved with. We call this an *accumulated impact score* (AI score) which is defined as

$$AI(i) = \sum_{j=1}^n K\left(\frac{G_{ij}}{h}\right). \quad (1)$$

Sometimes the node connections are highly heterogeneous, and we may want to enforce a normalization by dividing the impact of node i on node j , $K(G_{ij}/h)$, by $\sum_j K(G_{ij}/h)$. This

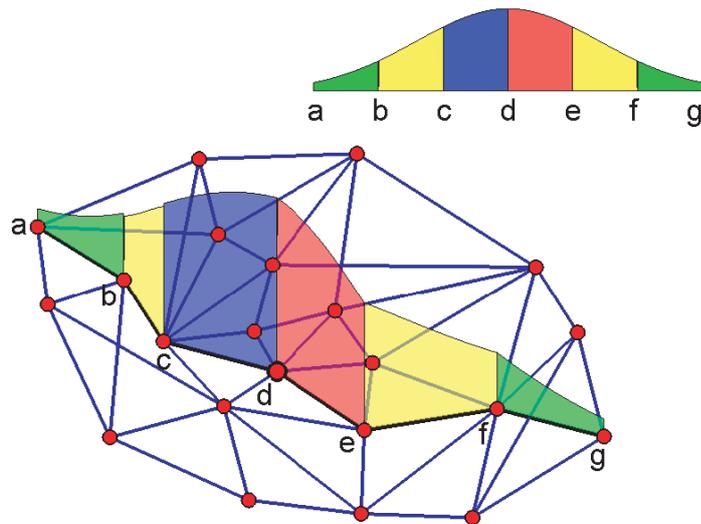


Figure 1. Illustration of building the node interaction system. Consider the impact of the central node d on its neighbors (a , b , c , e , f and g). The kernel function is Gaussian with $h = \sqrt{2}$. The decay of interaction strength is shown as the Gaussian curve in the inset. Ultimately, each node will be seeded with such a kernel and their AI scores are computed. Here we use a kernel with medium bandwidth, i.e. $h = \sqrt{2}$. It is immediately evident that a small bandwidth will lead to a very narrow kernel, while a large h renders the kernel wide.

can be imagined as a voting system, where each node has only a fixed amount of tickets to vote for other nodes. In figure 1, we schematically illustrate the kernel-based node interaction on a simple graph.

The accumulated impact of the nodes provides vital information on the modular structures of the network. On the one hand, as we have discussed, the contrast between the amount of connections for each node, now quantified by the AI score, clearly distinguishes the ‘core’ nodes (large AI) and the ‘boundary’ nodes (low AI) of a community. In particular, imagine that the nodes of the network are laid in a space according to the topology (the pairwise geodesic distances), with each node endowed with an impact level (AI). Then, the AI score can be considered as a discrete function defined node-wisely in this space: ‘peaks’ of the AI function correspond naturally to the core of communities, while ‘valleys’ act as boundaries that separate the communities.

On the other hand, remember that the pairwise node interactions that determine the computation of AI score are controlled by the bandwidth parameter. A small h corresponds to a thin, rapidly decaying kernel which only allows highly localized node interactions within close neighbors. This makes the AI scores fluctuate and more refined structures may arise, providing a delicate reflection on the network structure. Conversely, a large h leads to a wide kernel that decays slowly and favors long-range node interactions. Therefore, the resultant AI distribution becomes smoother with fewer converged clusters, furnishing us with a more global perspective. In other words, the kernel bandwidth h serves naturally as a ‘scale adjuster’, controlling the level on which we prefer to examine the network, with the resultant AI distribution reflecting the community structure in the impact score domain.

Next, we describe an efficient algorithm to identify meaningful communities based on the AI scores. In the landscape of AI scores, nodes can be deemed as lying on the surface of the

AI function composed of peaks and valleys. Thus we can iteratively shift each node along the ridges to the peak, and nodes gathering in the same peak will be naturally partitioned into the same community. Algorithmically, we start from a node and always move it toward its neighbor with the largest increase on the AI score, until no improvement can be achieved. By doing this, each node travels iteratively along edges on the graph, experiencing a path with a monotonously rising AI score until they touch the core node of the nearest community. The number of communities are automatically determined by calculating how many clusters the whole network ‘shrinks’ to.

The ‘hill climbing’ procedure designed here can actually be thought of as finding a partition of the network with a large modularity Q in a greedy manner ($Q = \sum_i (e_{ii} - a_i^2)$, where e_{ii} and a_i are the fraction of edges from the community i to itself and the whole graph, respectively [3]). Note that each node is always shifted to its neighbor that has the largest AI score. Therefore, the procedure will push together the nodes such that the sum of their AI scores is large. Also note that the AI score is a general measure of the amount of connections of each node and in particular, by choosing the kernel as a rectangular function

$$K\left(\frac{x}{h}\right) = \begin{cases} 1 & |x| \leq h \\ 0 & |x| > h \end{cases}$$

and setting $h = 1$, $AI(i)$, will be exactly the degree of node i . Therefore, in this special case, our algorithm will try to increase e_{ii} , leading to a large Q value. Moreover, the hill climbing also implicitly enforces a small a_i by using the boundary nodes to separate different communities, making the links among communities sparse.

Occasionally, we find that the hill climbing process may cause a node (say node k) to be misclassified due to its *direct link* to some hub nodes that always have large AI scores. To solve this problem, we can modify node k ’s membership by considering which community its neighbors mostly belong to. Suppose node k ’s neighbors are already assigned to various communities. We aggregate the neighbors that are in the same community and their AI scores also sum up. We then reassign node k to the community with the largest added-up AI score. In practice, a hub node may also be ‘absorbed’ to another hub that has a larger AI score, given the direct link between them. This can happen for some large-scale networks and may influence the accuracy of community detection. In this case, we just need to disconnect these hub nodes (usually by a very small amount) and then apply our algorithm.

3. Application to synthesized and real networks

First, we illustrate our method with the network constructed from the x component of the chaotic Rössler system [27, 28], where the nodes represent the cycles in the time series and edges are assigned to cycles that are close in phase space. The cycles are attracted by the stable manifold of the unstable period orbits (UPOs) [29] and cluster around them. Since a chaotic attractor has infinitely many UPOs organized in a hierarchical manner, the corresponding network is multiscale, demonstrating smaller clusters (around higher-order UPOs) at a finer scale. It thus serves as a good example to test our algorithm in unraveling the multiscale modular structure. As shown in figures 2(a) and (b), small h leads to the detection of communities on small scales; as h increases, small communities begin to merge, forming coarser clusters with increasing scales. We find that communities identified at various resolutions essentially reflect the UPOs of different orders. For example, the central node of the two communities at the top of figure 2(a) correspond to the two branches from the UPO of order 2.

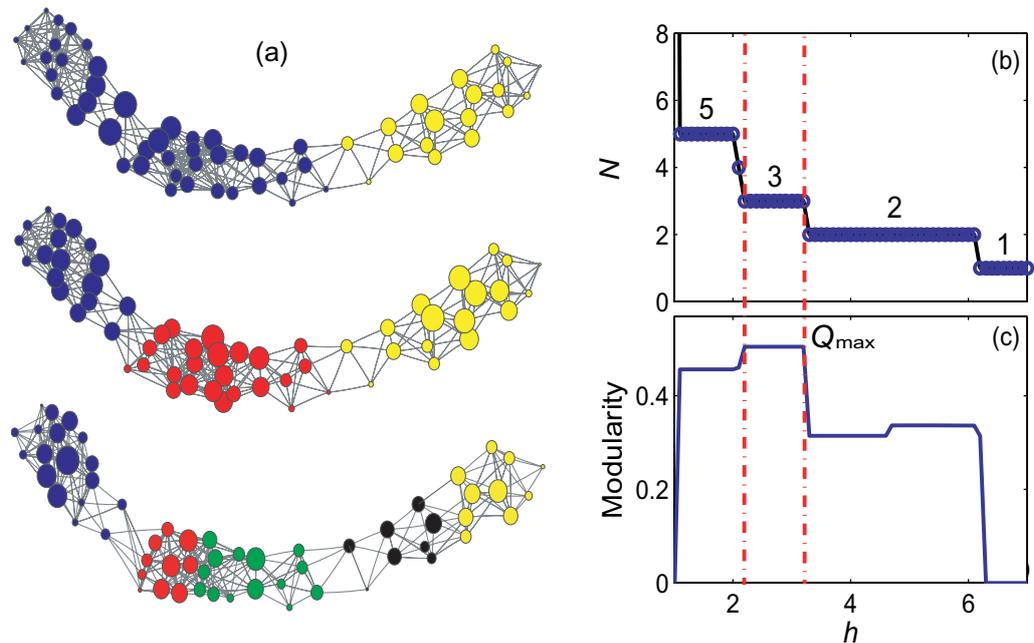


Figure 2. Multiresolution community structure for networks of the chaotic system. From top to bottom we use three bandwidths of cubic kernels that are decreasing, $h = 4, 3, 2$, etc, with the node sizes reflecting their AI scores. (b) and (c) show how community number and modularity change with h . A global maximum of Q appears at a mesoscale of h indicated by the dash-dot line.

In the case of multiscale networks, the smaller communities found at fine scales will merge into larger ones as h increases. Therefore, the number of detected communities N versus the scale parameter h will provide valuable information on the inner structures of the network. For example, we find that for the networks from chaotic systems, N decreases in a power-law manner with h , indicating that a strong multiscale property that is associated with the hierarchical and self-similar structure induced by the UPOs underlies the chaotic attractor; see figure 3. For random networks, however, N will encounter a radical decrease to 1 at a critical value of h . This indicates that under no scale the network demonstrates marked modular structures. Therefore, our approach not only resolves the multiscale structure, if there is one, but also determines if at all the network is multiscale.

In the case when the network demonstrates meaningful community organization under multiple scales, we can further use modularity Q to evaluate the communities detected at different scales, inspired by the modularity-landscape surveying [23]. As can be seen from figure 2(c), the Q - h curve shows a staircase form (each stair roughly corresponding to a specific scale, having some fluctuations), and we can choose the reasonable partition (corresponding to a local maximum in a stair) at each resolution scale. In figure 2(c), the maximum modularity is obtained at a mesoscale ($h = 3$) where the network is partitioned into three communities. This is also the most reasonable partition by the human eye. By varying h continuously, we can actually obtain a complete picture on the evolution of the modular structure across all scales. Theoretically, our method can resolve modules of arbitrarily small size, if any, by just

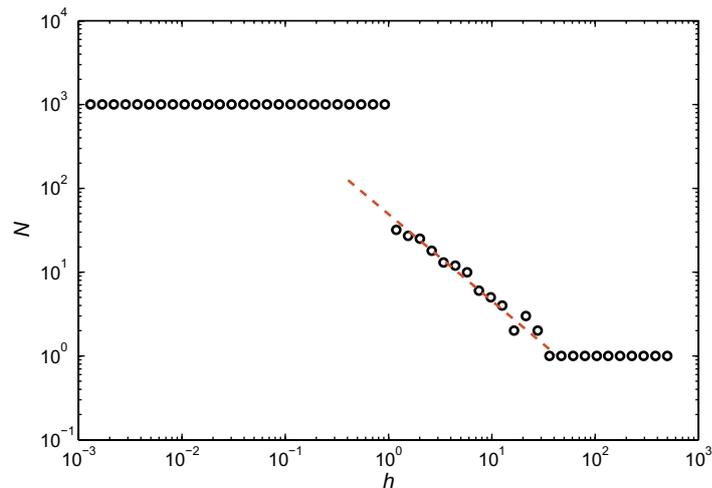


Figure 3. The number of communities (N) identified versus the bandwidth h in a log–log plot for the network (with 1000 nodes) constructed from the chaotic time series [27, 28].

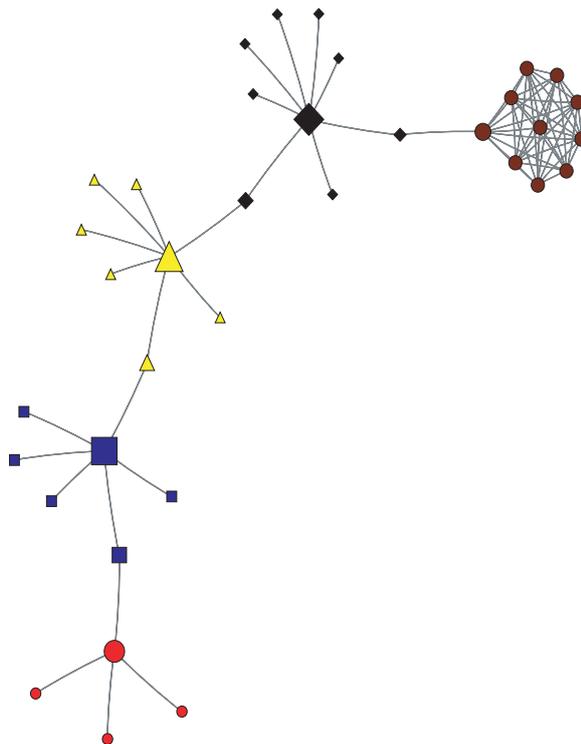


Figure 4. Community detection results for a network with five predefined communities with a Gaussian kernel. The five built-in clusters have different densities and they are correctly identified at a relatively small bandwidth ($h = 1$) by our method.

tuning h down to an appropriate scale. To demonstrate that our method is not influenced by the size distribution of communities (either broad or not), we further test our algorithm using a synthesized network with clusters of various densities [21]. As can be seen in figure 4, our

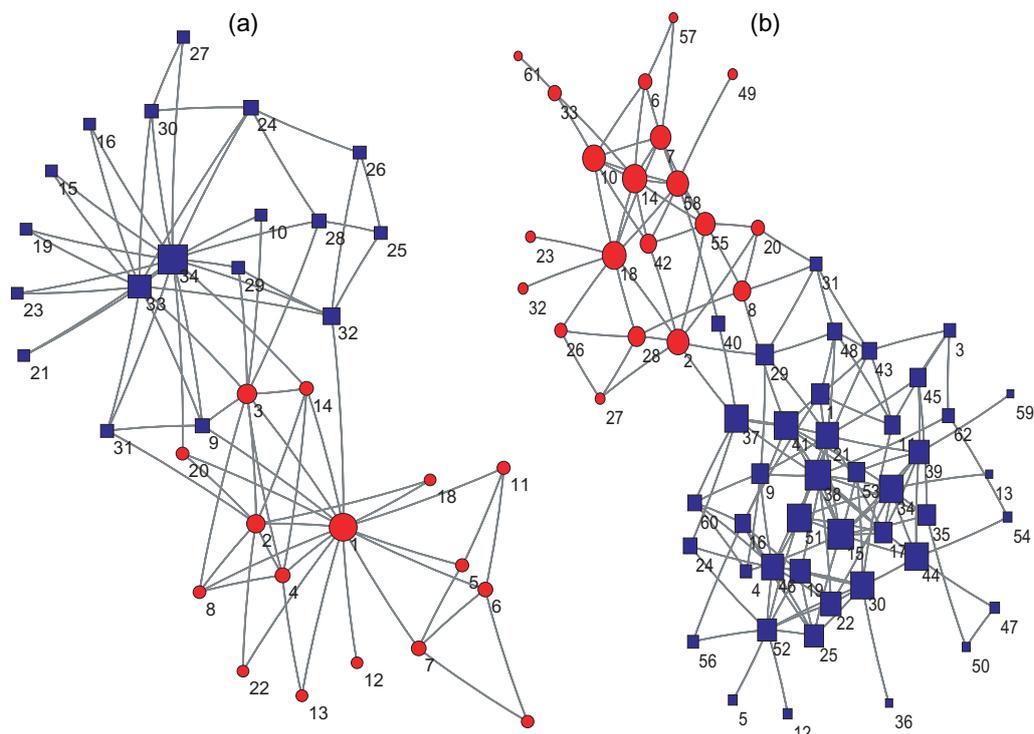


Figure 5. Community structure of (a) the Zachary karate club network and (b) dolphin social network of Lusseau by our algorithm, with Gaussian ($h = 0.8$) and cubic kernel ($h = 3$), respectively.

approach correctly captures all the modular structures from both dense and sparse clusters using a relatively small bandwidth h .

Now we test our algorithm with some real-world networks. First are two benchmark networks whose community structures are known, i.e. the social networks from the Zachary karate club [30] and the bottlenose dolphin network in Doubtful Sound [31]. Figure 5 presents the partition at the scale where the modularity maximizes, and both networks are split into two communities in perfect correspondence with the prior knowledge. The most ‘influential’ nodes that play a vital role in community formation can also be identified by the AI scores, such as nodes 1 (administrator in the club) and 33 in the club network, and nodes 14 and 15 in the dolphin network. We note that different kernels essentially lead to consistent partition results. Theoretically, the degree of smoothness is asymptotically equal for different kernels when they use their corresponding canonical bandwidth parameters.

Now we apply our method to a collaboration network from the network scientists [32], where nodes are researchers and links represent co-authorship. Although there is no *a priori* knowledge on community structure, the nodes are nicely dispersed into well-defined clusters that possess, to some extent, the hierarchical structure (see figure 6 (a)) which makes it suitable for multiscale analysis. As can be seen, the network is partitioned into 15 local clusters at small h , which contains some information about known groupings. For example, the green box indicates a group of physicists from Boston University. The two clusters with yellow triangles and red circles represent the strong collaboration centered around Newman and Barabási, respectively. By increasing h , these small communities merge into four big ones (indicated by

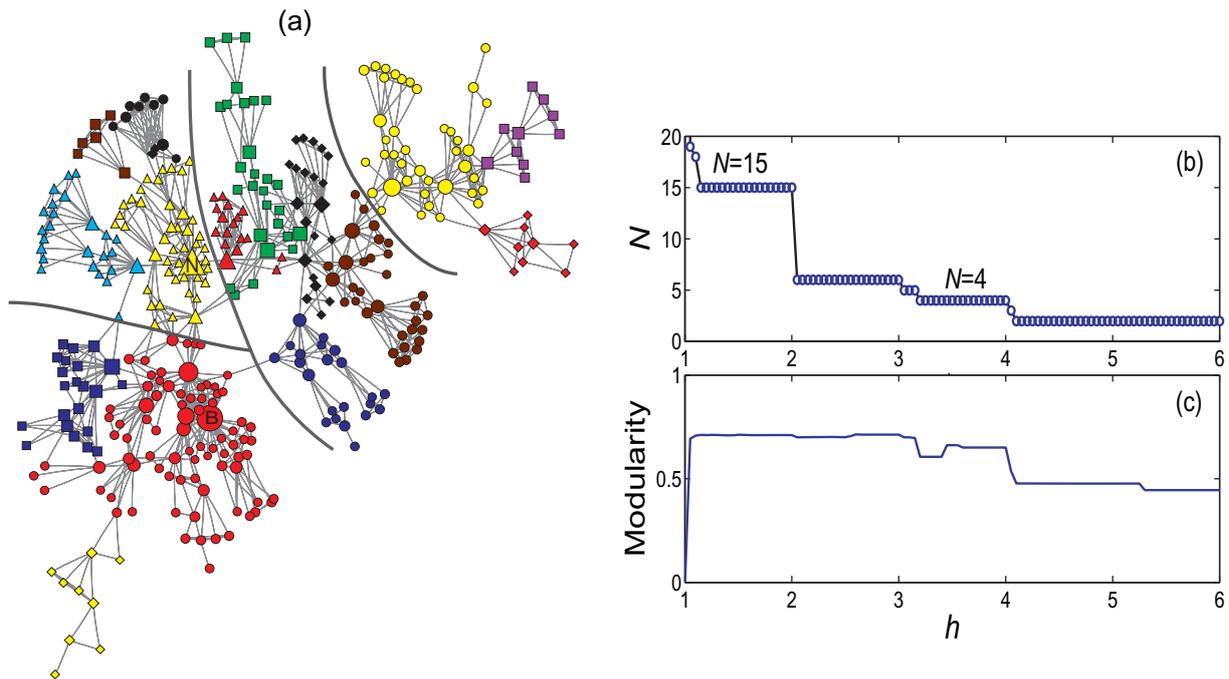


Figure 6. Modular structure of a coauthorship network (we consider the largest component with 379 nodes) (a) Cubic kernel with $h = 1.6$ leads to 15 communities. The curve indicates the partition into four clusters with $h = 3.5$. (b) and (c) show how community number and modularity change with h .

the lines in figure 6 (a) at $h = 3.5$). This large-scale community structure reveals some regional collaboration, for example, the top-right cluster corresponds to researchers mainly from Europe.

Finally, to test the performance of our method, we apply it to a computer-generated benchmark network [4] that is widely adopted to test the effectiveness of the community-detection algorithms. The network has 128 nodes divided into four communities of 32 vertices each. Edges are placed between node pairs independently with a certain probability so as to keep the average degree of each node 16. Each vertex has on average K_{in} edges connecting to those within the same group and K_{out} edges to nodes in other groups, and $K_{\text{out}} + K_{\text{in}} = 16$. Figure 7 shows the fraction of vertices that are correctly classified by our algorithm, as a function of the average number of intercommunity edges K_{out} . Here, the partition result for each network is obtained at scale h where Q maximizes. We can see that our method generally performs better than the traditional GN algorithm [4]. A special case with $K_{\text{out}} = 2$ is demonstrated in figure 8. Although the degree distribution is homogeneous for this network, our method can effectively distinguish the subtle difference among the nodes according to their higher-order neighbors. As can be seen, within each community there are always core nodes (with a larger size) that hold the whole group together.

The computational cost of our approach is low, and time complexity in the worst case is $O(n^2)$. To compute the impact of node i on other nodes, we can perform Breadth-First-Search to find neighbors of each node up to the d th order, and the time is linear with the neighborhood size. In hill climbing, moving each node to the peak takes $O(lc)$ time, where c is the averaged first-order neighborhood size and l is the number of iterations. Therefore,

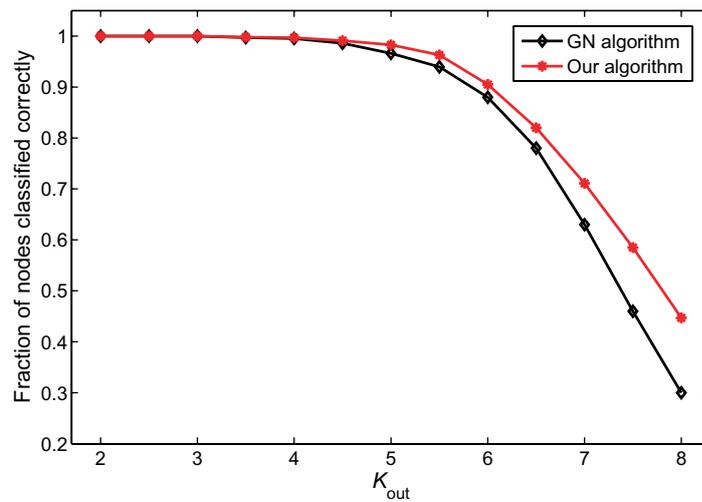


Figure 7. The fraction of vertices correctly identified by our method and the GN algorithm for a network with a known community structure (as described in the text) versus K_{out} . The result is an average over 100 realizations of the graph.

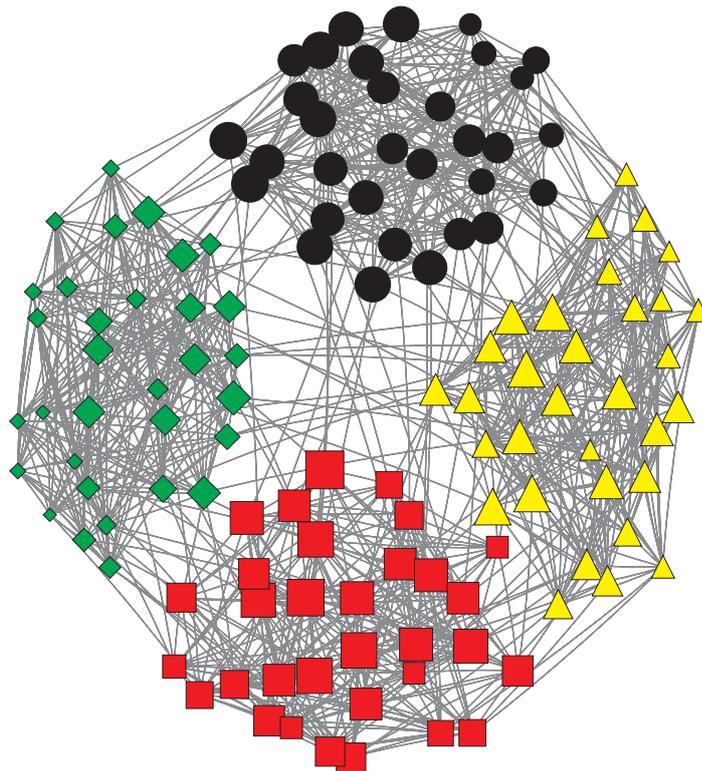


Figure 8. Community structure found for the synthesized network with intra-cluster degree being 14 and inter-cluster degree being 2 by our method. The four colors indicate four different communities. Here we use the Gaussian kernel of $h = 3$, and we successfully assign every node to its corresponding community.

the overall complexity is strictly bounded by $O(n^2)$. When the scale parameter h is small, node interactions become sparse and only a small neighborhood size needs to be considered. In practice, h can be chosen from a geometric series $\{1, 2, 4, \dots, R\}$, where $R = O(\max(G_{ij}))$. Namely, we only need $\log(R)$ trials (and $O(\log(R)n^2)$ time) to achieve a complete view of how the community structure evolves over scale.

4. Conclusion and discussion

In summary, we have proposed a multi-resolution approach for community detection, capable of resolving the full spectrum of modular structures of a complex network in a precise and systematic way. The key idea is to build a node-interaction system via the introduction of kernels, wherein the modular structure is visualized from the emergent patterns in the interaction landscape. The range of node interactions is controlled by the bandwidth of the kernel, the variation of which provides a reliable, multiscale screening of the modular structure.

Intuitively, our community detection approach can also be deemed as a *clustering around the nodes with high AI scores* at various scales. The AI score can be considered as a weighted version of degree, but takes into account the high-level topologies, i.e. the higher-order neighbors. Therefore, the high AI score nodes correspond to the cores that brace the whole network by ‘attracting’ the nodes pertaining to them. By increasing the kernel bandwidth h , these high AI score nodes turn from the local cores to the global centers of the network, which lead to the multi-resolution partition of the network in a convenient and efficient manner. Since our approach utilizes both the core nodes (which sustain the communities) and the borderline nodes (which separate the communities) in the ‘hill climbing’ process, it avoids the high computational cost usually present in those optimization-based schemes. Our method can also be extended to the general weighted networks. In this case, the AI scores of the nodes can similarly be obtained from the shortest path distance G_{ij} between nodes. The G_{ij} can be conveniently calculated by the Dijkstra’s algorithm, which takes into account the weights of the edges in this case.

Our approach can be applied to a wild variety of real networks, especially those emergent from biological and social fields, where networks with hierarchical and multiscale modular structures abound [16, 17]. Compared with traditional methods that usually detect community at a specific organization level, our multiscale approach promises a more comprehensive characterization of the community structure across all scales. This multiscale community profile is expected to provide new insights into the functional organization and evolution of a network.

Acknowledgments

This research was funded by the Hong Kong Polytechnic University Postdoctoral Fellowships Scheme 2007–2008 (G-YX0N).

References

- [1] Newman M E J 2004 Detecting community structure in networks *Eur. Phys. J. B* **38** 321
- [2] Danon L, Diaz-Guilera A, Duch J and Arenas A 2005 Comparing community structure identification *J. Stat. Mech.* P09008
- [3] Newman M E J and Girvan M 2004 Finding and evaluating community structure in networks *Phys. Rev. E* **69** 026113

- [4] Girvan M and Newman M 2002 Community structure in social and biological networks *Proc. Natl Acad. Sci. USA* **99** 7821
- [5] Newman M E J 2006 *Proc. Natl Acad. Sci. USA* **103** 8577
- [6] Guimera R and Amaral Nunes L A 2005 Functional cartography of complex metabolic networks *Nature* **433** 895
- [7] Clauset A, Newman M E J and Moore C 2004 Finding community structure in very large networks *Phys. Rev. E* **70** 066111
- [8] Duch J and Arenas A 2005 Community detection in complex networks using extremal optimization *Phys. Rev. E* **72** 027104
- [9] Leicht E A and Newman M E J 2008 Community structure in directed networks *Phys. Rev. Lett.* **100** 118703
- [10] Pothen A, Simon H and Liou K P 1990 Partitioning sparse matrices with eigenvectors of graphs *SIAM J. Matrix Anal. Appl.* **11** 430
- [11] Wasserman S and Faust K 1994 *Social Networks Analysis* (Cambridge: Cambridge University Press)
- [12] Reichardt J and Bornholdt S 2004 Detecting fuzzy community structures in complex networks with a Potts model *Phys. Rev. Lett.* **93** 218701
- [13] Hofman J M and Wiggins C H 2008 Bayesian approach to network modularity *Phys. Rev. Lett.* **100** 258701
- [14] Rosvall M and Bergstrom C T 2008 An information-theoretic framework for resolving community structure in complex networks *Proc. Natl Acad. Sci. USA* **104** 7327
- [15] Brandes U, Delling D, Gaertler M, Goerke R, Hofer M, Nikoloski Z and Wagner D 2006 Maximizing Modularity is hard arXiv:physics/0608255
- [16] Ravasz E, Somera A L, Mongru D A, Oltvai Z N and Barabasi A L 2002 Hierarchical organization of modularity in metabolic networks *Science* **297** 1551
- [17] Ravasz E and Barabasi A L 2003 Hierarchical organization in complex networks *Phys. Rev. E* **67** 026112
- [18] Clauset A, Moore C and Newman M E J 2008 Hierarchical structure and the prediction of missing links in networks *Nature* **453** 98
- [19] Song C, Havlin S and Makse H A 2005 Self-similarity of complex networks *Nature* **433** 392
- [20] Fortunato S and Barthelemy M 2007 Resolution limit in community detection *Proc. Natl Acad. Sci. USA* **104** 36
- [21] Arenas A, Fernandez A and Gomez S 2008 Analysis of the structure of complex networks at different resolution levels *New J. Phys.* **10** 053039
- [22] Serrano M A, Bogun M and Vespignani A 2009 Extracting the multiscale backbone of complex weighted networks *Proc. Natl Acad. Sci. USA* **106** 6483
- [23] Sales-Pardo M, Guimera R, Moreira A A and Amaral L A N 2007 Extracting the hierarchical organization of complex systems *Proc. Natl Acad. Sci. USA* **104** 15224
- [24] Wand M P and Jones M 1995 *Kernel Smoothing* (London: Chapman and Hall)
- [25] Zhang K and Kwok T J 2006 Simplifying mixture models through function approximation *NIPS2006* (Vancouver, Dec. 2006)
- [26] Zhang K, Tang M and Kwok J T 2005 Applying neighbourhood consistency for fast clustering and kernel density estimation *IEEE Comp. Soc. Conf. Comp. Vision and Pattern Recognition* **2** 1001–7
- [27] Zhang J and Small M 2006 Complex network from pseudoperiodic time series: topology vs dynamics *Phys. Rev. Lett.* **96** 238701
- [28] Zhang J, Sun J, Luo X, Zhang K, Nakamura T and Small M 2008 Characterizing topology of pseudoperiodic time series via complex network approach *Physica D* **237** 2856
- [29] Hunt B and Ott E 1996 Optimal periodic orbits of chaotic systems *Phys. Rev. Lett.* **76** 2254
- [30] Zachary W W 1977 An information flow model for conflict and fission in small groups *J. Anthropol. Res.* **33** 452
- [31] Lusseau D 2003 The emergent properties of a dolphin social network *Proc. R. Soc. B* **270** S186
- [32] Park J and Newman M E J 2003 Origin of degree correlations in the Internet and other networks *Phys. Rev. E* **68** 026112