Efficient community detection via sampling processes in complex networks

D. R. Amancio, O. N. Oliveira Jr. and L. da F. Costa
Institute of Physics of São Carlos
University of São Paulo, P. O. Box 369, Postal Code 13560-970
São Carlos, São Paulo, Brazil
E-mail: diego.raphael@gmail.com, diego.amancio@usp.br

Abstract. The identification of community structures is essential for characterizing real networks formed by a mesoscopic level of organization where clusters contain nodes with a high internal degree of connectivity. Many methods have been developed to unveil community structures, but only a few studies have probed their suitability in incomplete networks. Here we assess the accuracy of community detection techniques in incomplete networks generated in sampling processes. We show that the walktrap and fast greedy algorithms are highly accurate for detecting the modular structure of incomplete complex networks even if many of their nodes are removed. Furthermore, we implemented an approach that improved the time performance of the walktrap and fast greedy algorithms, while retaining the accuracy rate in identifying the community membership of nodes. Taken together our results show that this new approach can be applied to speed up virtually any community detection method, especially if the network under analysis is tightly connected.
1. Introduction

Many real systems of different natures can be modeled as complex networks, where entities and their relationships are represented as nodes and edges, respectively. Examples of such systems are the Internet [1, 2], the WWW [3], transport [5, 6] and transmission systems [7]. Relevant in this modeling has been the ability of nodes to cluster into communities, defined as groups of strongly connected nodes with a few external links with the other nodes of the network. Various methods for detecting communities have been proposed [8], including waltkrap [9], fast greedy [10, 11], edge-betweenness [12] and leading eigenvector [13]. Unprecedented patterns of topological organization could be unveiled with communities being identified for metabolic, genetic, collaborative and social networks [14, 15, 16, 17].

Major issues for these methods are not only the accuracy but also the efficiency of the algorithm, since some real networks may comprise millions of nodes [18, 19]. Actually, time efficiency is decisive for choosing the method for addressing a given problem as some methods become impractical for very large networks. This is the case of the edge-betweenness method, whose temporal complexity is $O(n^3)$. Perhaps because of the relevance of time efficiency, other important issues have been relatively neglected. An example is the applicability of standard methods in incomplete networks, i.e., networks with imprecise information, such as missing nodes or edges. To our knowledge, only a few studies have probed the efficiency of community detection methods in incomplete networks. In Ref. [20], the authors focus on the predictability of missing edges, which is crucial for real networks resulting from incomplete experiments [21]. In information and social networks, for example, low-degree nodes are usually undiscoverable in crawling systems, while in protein interaction networks many edges may be unknown [22].

In this paper, we evaluate the robustness of two methods in discovering communities in incomplete networks generated from sampling processes. As we shall show, these methods are robust even when many nodes are missing. Furthermore, we found out that the robustness seems weakly dependent on the method evaluated, but there is an important dependence on the network structure. More importantly, we show that robustness in detecting communities allows us to devise a strategy that improves the time performance, while keeping the accuracy. One of the major advantages of the proposed strategy is that it can be applied to virtually all standard methods, since it relies on detecting communities in sampled networks.

This paper is organized as follows. In Section 2 we describe the methods employed to detect communities, whose robustness is evaluated in Section 3.1 using six incomplete networks. In Section 3.2 we argue that our method can improve the time performance, with simulations to study the interplay between accuracy and time performance. Section 4 closes the paper with final remarks and discussing future investigations.
2. Methods

For the description of community detection methods, consider the following notation. A network is defined as \( G = \{V, E\} \), where \( V \) and \( E \) are respectively the set of nodes and edges. The connectivity is represented as an adjacency matrix \( A = \{a_{ij}\} \) with elements

\[
a_{ij} = \begin{cases} 
1, & \text{if } i \text{ and } j \text{ are linked}, \\
0, & \text{otherwise}.
\end{cases}
\]

The degree of node \( i \) is given by \( k_i = \sum_j a_{ij} \). \( D = \{\delta_{ij}\} \) is the diagonal matrix. The element \( \delta_{ij} \) is

\[
\delta_{ij} = \begin{cases} 
k_i, & \text{if } i = j, \\
0, & \text{otherwise}.
\end{cases}
\]

\( P_{ij} = D^{-1}A = \{p_{ij}\} \) is the Markovian adjacency matrix. Each element \( p_{ij} \), defined as \( p_{ij} = a_{ij}/k_i \), represents the probability of a random walker at node \( i \) to reach node \( j \) in the next time step.

The algorithms selected here to detect communities in sampled networks are the walktrap and fast greedy methods, which were chosen because they are suitable for weighted networks that are generated with our approach.

2.1. Walktrap

The walktrap community detection method relies on random walks to split the network in natural partitions. At each time step, a particle moving on the network leaps to a neighboring node, which is chosen randomly. This process is repeated many times so that a Markov chain \([23]\) is generated. Here, the walker is allowed to leap onto a neighbor in fixed, discrete time steps. Random walks are used in walktrap to create a node similarity metric, which in turn is used to cluster nodes into communities. Two nodes \( i \) and \( j \) are considered similar if a random walk starting at \( i \) accesses node \( j \) many times. This similarity can be obtained analytically from the matrix \( P^t \), whose element \( p_{ij}^{(t)} \) quantifies the probability of the walker to reach node \( j \) (from node \( i \)) in \( t \) steps. Each element \( p_{ij}^{(t)} \) of \( P^t \) satisfies the relation \( p_{ij}^{(t)} = k_j k_i^{-1} p_{ji}^{(t)} \). Therefore, if node \( i \) is highly connected, it will reach node \( j \) only a few times. Conversely, the higher the degree of node \( j \) the higher is its probability to be reached from a random walk starting at any other node. In the steady state (i.e., in the limit as \( t \to \infty \)), the stationary probability \( \pi_i \equiv \lim_{t \to \infty} p_{ij}^{(t)} \forall i \) becomes:

\[
\pi_i \equiv \lim_{t \to \infty} p_{ij}^{(t)} = k_j / \sum_l k_l. \tag{1}
\]

Therefore, the parameter \( t \) should not be much higher than the mixing time \([23]\) of \( P \), otherwise the likelihood \( p_{ij}^{(t)} \) would reflect only the degree of connectivity (see figure 1(b)). In addition, \( t \) should not take very low values because far distant nodes would be inaccessible (see figure 1(a)).
Figure 1. Similarity between node “A” and the other nodes. The diameter of the nodes is proportional to their similarity with node “A”. In (a) random walks of length $h = 4$ were used and in (b) we used random walks of infinite length ($h \to \infty$). In panel (a), nodes “B” and “C” are very dissimilar from “A” because their distance from “A” is large. In panel (b), similarities are assigned regardless of the distance from A. Actually, the only factor that matters in this case is the degree.

Given the transition matrix $\mathcal{P}$, the distance $r_{ij}^{(t)}$ between nodes $i$ and $j$ is given by

$$r_{ij}^{(t)} = \sqrt{\sum_t \left( \frac{p_{il}^{(t)} - p_{jl}^{(t)}}{k_l} \right)^2} = \|D^{-\frac{1}{2}}\mathcal{P}_{i}^{(t)} - D^{-\frac{1}{2}}\mathcal{P}_{j}^{(t)}\|,$$

(2)

where $\mathcal{P}_{i}^{(t)}$ is the $i$-th row of $\mathcal{P}^t$. This metric can be generalized to measure the similarity $r_{C_1C_2}$ between two communities $C_1$ and $C_2$. Prior to the definition of $r_{C_1C_2}$, one needs to define the probability $\mathcal{P}_{C_j}^{(t)}$ of a node $i \in C$ to reach node $j \not\in C$ in $t$ steps. This quantity is defined as

$$\mathcal{P}_{C_j}^{(t)} = \frac{1}{\|C\|} \sum_{i \in C} p_{ij}^{(t)}, \quad (3)$$

which represents the average likelihood of a node $i \in C$ to reach a node $j \not\in C$. With this definition, the distance between two communities is

$$r_{C_1C_2} = \sqrt{\sum_t \left( \frac{p_{C_1l}^{(t)} - p_{C_2l}^{(t)}}{k_l} \right)^2} = \|D^{-\frac{1}{2}}\mathcal{P}_{C_1}^{(t)} - D^{-\frac{1}{2}}\mathcal{P}_{C_2}^{(t)}\|.$$

(4)

After computing all pairs of distances between communities, the walktrap method follows an agglomerative approach based on the Wards method [25]. Initially, each node
represents a community. Two communities \( C_1 \) and \( C_2 \) are merged if the new partition minimizes \( \sigma \), the squared distances between nodes and their respective communities:

\[
\sigma_l = \sum_c \sum_{i \in C} r_{iC}^2,
\]

where \( r_{iC} \equiv r_{(i)C}. \) Then, a new community \( C_{n+1} = C_1 \cup C_2 \) arises and the old partition \( P_l \) becomes \((P_l \setminus \{C_1, C_2\}) \cup \{C_{n+1}\})\). Finally, the process is repeated until the expected number of communities is obtained. The detection of two communities using this method is illustrated in figure 2(a).

Note that the distance \( r^{(t)}_{ij} \), as defined in equation (2), has a strong relationship with the spectra of \( \mathcal{P} \). More specifically, \( r^{(t)}_{ij} \) can be rewritten as

\[
r^{(t)}_{ij} = \left[ \sum_{\alpha=2}^{\infty} \lambda^2_{\alpha} (v_{\alpha}(i) - v_{\alpha}(j))^2 \right]^{\frac{1}{2}},
\]

where \( \lambda_{\alpha} \) and \( v_{\alpha} \) are respectively the eigenvalues and eigenvectors of \( \mathcal{P} \). In view of this formulation in terms of graph spectra, \( r^{(t)}_{ij} \) in equation (6) may be defined to consider different weighting for distinct eigenvalues, allowing thus the use of continuous random walks [24]. This generalization is achieved with the following relation

\[
r_{ij}^2 = \sum_{\alpha=2}^{N} \left( \sum_{l=0}^{\infty} c_l \lambda^l_{\alpha} \right)^2 (v_{\alpha}(i) - v_{\alpha}(j))^2 = \| \mathcal{D}^{-\frac{1}{2}} \tilde{\mathcal{P}}_{i}^{(t)} - \mathcal{D}^{-\frac{1}{2}} \tilde{\mathcal{P}}_{j}^{(t)} \|,
\]

where

\[
\tilde{\mathcal{P}}_{i} = \sum_{l=0}^{\infty} c_l \mathcal{P}_{i}^{(t)}.
\]

2.2. Fast greedy

Similarly to the walktrap community detection method, the fast greedy algorithm is also based on hierarchical agglomerative clustering. Initially, each node represents a community. As the algorithm is progressively applied, similar nodes are joined into communities (the similarity is established according to a given criterion) until all vertices belong to a same giant community, thus completing the dendrogram. To join two vertices, the algorithm uses the modularity \( Q \), which measures the number of intra-community edges that are higher than the expected by chance. The quantity \( Q \) is

\[
Q = \sum_i (e_{ii} - a_i^2) = \frac{1}{2m} \sum_i \sum_j \left( a_{ij} - \frac{k_i k_j}{2m} \right) \epsilon(g_i, g_j),
\]

where \( e_{ij} \) is the fraction of edges linking nodes in community \( i \) to those in community \( j \), \( a_i = \sum_j e_{ij} \), \( m = 1/2 \sum k_i \), \( g_i \) is the community to which node \( i \) belongs and

\[
\epsilon(g_i, g_j) = \begin{cases} 1, & \text{if } g_i = g_j, \\ 0, & \text{otherwise}. \end{cases}
\]

More specifically, the fast greedy algorithm joins two communities provided that \( \Delta Q = e_{ij} + e_{ji} - 2a_i a_j = 2(e_{ij} - a_i a_j) \) is maximized.
Efficient community detection via sampling processes in complex networks

Figure 2. Example of communities detected with (a) walktrap; and (b) fast greedy methods.

It is worth noting that it is not necessary to check all possible joining possibilities since only the junction of neighboring communities (i.e. communities with at least one edge linking two of their vertices) is able to increase $Q$. An example of a network with two communities identified with the fast greedy method is depicted in figure 2(b).

3. Results and discussion

3.1. Community detection in incomplete networks

The ability of the community detection algorithms to find natural clusters in incomplete networks was tested with the following methodology. We started with toy networks $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, henceforth referred to as original networks, generated according to the procedures described in Refs. [26, 27]. The following parameters were employed: $N$, the number of nodes; $\mu = 0.3$, the mixing parameter (quantifies the fraction of links that are placed outside the community of the node); and $\langle k \rangle = 1/N \sum k$, the average degree. To create an incomplete version $\mathcal{G}' = \{\mathcal{V}', \mathcal{E}'\}$ of $\mathcal{G}$, the nodes in $\mathcal{G}$ were randomly sampled with sampling rate $\mathcal{S}$. The unweighted connectivity matrix $\mathcal{A}$ becomes an weighted matrix $\mathcal{A}'$ such that $a_{ij} \mapsto d_{ij}^{-1}$, $\forall (i, j) \in \mathcal{V}'$, where $d_{ij}$ represents the length of the shortest path linking nodes $i$ and $j$ in $G$. In the experiments, we used the following networks:

- Network $\mathcal{N}_a$: $N = 512$ and $\langle k \rangle = 32$,
- Network $\mathcal{N}_b$: $N = 1,024$ and $\langle k \rangle = 48$,
- Network $\mathcal{N}_c$: $N = 2,048$ and $\langle k \rangle = 96$,
Network $N_d$: $N = 512$ and $\langle k \rangle = 64$,
Network $N_e$: $N = 1,024$ and $\langle k \rangle = 96$,
Network $N_f$: $N = 2,048$ and $\langle k \rangle = 128$.

The ability to detect community structures in figure 3 using the walktrap algorithm is similar for all the networks considered. The performance is very high for incomplete networks with spurious edges (i.e., when the sampling is performed with sampling rate $S = 1$), as revealed by accuracy rates $\Gamma$ above 0.95. When $S$ decreases and therefore less nodes are left in the incomplete network, the ability to detect communities diminishes, as one should expect. Interestingly, for all networks, there exists a threshold $S = \varsigma$ discriminating two regimes. When $S \leq \varsigma$, the organization in communities disappears rapidly as $S$ decreases. In contrast, for $S > \varsigma$, the community structure seems to be maintained in spite of the removal of many nodes. This is apparent for network $N_e$, for example. Even with 70% of the nodes being discarded ($S = 0.30$), the communities are well distinguished from each other. Also, the degree of connectivity $\langle k \rangle$ affects $\varsigma$. The increase in $\langle k \rangle$ causes the network to be more robust so that the modular organization does not disappear at all, an effect that becomes even more evident by comparing $N_a$ ($\langle k \rangle = 32$) and $N_e$ ($\langle k \rangle = 64$). While in the former the threshold is $\varsigma \sim 0.70$, the latter displays a threshold $\varsigma \sim 0.50$.

Figure 4 displays how the accuracy rate varies with the sampling rate for the fast greedy method used to detect communities. The results are essentially similar to those of the walktrap method in figure 3. The fast greedy method performs well when $S = 1$ (i.e., when no node is removed), just as in the walktrap. As nodes are removed with a sampling rate $S < \varsigma$, the accuracy rate $\Gamma$ decreases at a low rate. The values of $\varsigma$ for both methods are similar, suggesting a stronger dependence on network topology. The robustness of the network (in the sense that the community structure is maintained) increases with the average connectivity $\langle k \rangle$, as indicated by comparing $N_a$ and $N_d$ in figure 4.

All in all the results reveal that the community detection methods evaluated are robust for they are able to identify the modular organization even when many nodes from the original network are removed. Hence if we are interested in finding the community to which only a few nodes belong, we can choose to deliberately eliminate the other nodes from the analysis. Provided that the sampling rate is sufficiently large (i.e., $S > \varsigma$), high accuracy can be achieved with a gain in performance, since computation in smaller networks implies a decrease in computational cost. This idea of detecting community in sampled networks with a gain in temporal performance serves as motivation to the proposed method described below.

3.2. Fast community detection via sampling processes

The finding that the community structure is maintained in incomplete networks derived from a random sampling process with a sampling rate $S > \varsigma$ motivated us to devise a method to decrease the computational cost of the walktrap and fast greedy methods.
As we shall show, this gain in time performance does not affect the accuracy. The proposed algorithm initially randomly chooses a set $V' \in V$ such that $\|V'\| \|V\|^{-1} = S$. Then the selected nodes are connected with weights $a_{ij}' = \frac{d_{ij}}{d_{ij}}$, where $d_{ij}$ is the length of the shortest path linking nodes $i$ and $j$ in the complete (not sampled) network. Note that this procedure coincides with the one adopted to form incomplete networks in Section 3. In the next step, communities are discovered using any standard method. Then the membership assigned for each node in the sampled network is mapped to the corresponding node in the original network. To assign the membership of the remaining nodes in $V$, a voting strategy over the neighbors is adopted. If most of the neighbors belong to the community $C$, then $C$ is assigned to that node. In case of ties, the decision is postponed to the next iteration. This process is repeated until all nodes have been classified. The overall process can be summarized in 6 steps:

(i) **Step 1**: Select randomly a set of nodes from the original network
Figure 4. Dependence on the sampling rate for the accuracy using the fast greedy algorithm for network (a) $\mathcal{N}_a$; (b) $\mathcal{N}_b$; (c) $\mathcal{N}_c$; (d) $\mathcal{N}_d$; (e) $\mathcal{N}_e$ and (f) $\mathcal{N}_f$. The dashed line represent the threshold $\varsigma$. The ability to discriminate communities drops as an increasing number of nodes are discarded. Note that all networks are robust to node removal (in the sense that they keep their community structure) provided that the sampling rate is above a given threshold.

(ii) **Step 2**: Create an incomplete network whose edges weights are inversely proportional to the distances in the original network.

(iii) **Step 3**: Identify the communities in the simplified network using any standard community detection method (e.g., walktrap or fast greedy).

(iv) **Step 4**: Transfer the memberships obtained in the incomplete network to the original network.

(v) **Step 5**: Propagate labels according to a voting strategy over neighbors.

(vi) **Step 6**: Repeat step 5 until all nodes have been classified.

The process of detecting communities with the above method is illustrated in the original toy network displayed in figure 5(a). The two communities are divided by a dashed line. Highlighted nodes represent those selected randomly. Initially, an incomplete network comprising the nodes randomly selected from the original network is formed (figure 5(b)). After detecting the communities in the incomplete network (figure 5(c)), the membership of each node is transferred to the original network, giving
rise to the configuration depicted in figure 5(d). Then the label propagation phase takes over until all nodes are classified. The result of the first iteration is displayed in figure 5(e). Note that node X has been classified as belonging to the 'green' community because it is connected to two nodes belonging to the 'green' community and just one belonging to the 'yellow' community. On the other hand, node Y was incorrectly classified as 'green' because it is connected to another 'green' node. The final configuration after the second iteration is shown in figure 5(f).

The efficiency of the proposed technique was verified in the networks $\mathcal{N}_a$ - $\mathcal{N}_f$, with the results obtained using the walktrap method in step 3 displayed in figure 6. In each subplot, the upper curve refers to the accuracy rate $\Gamma$ in assigning communities, while the bottom one shows the normalized processing time (i.e. the time spent in performing the 6 steps divided by the time spent by the community detection method running directly on the original network). Interestingly, the accuracy rates after step 6 are similar to those in figures 3 and 4, thus indicating that the accuracy of our method strongly depends on the ability to detect the communities in the sampled, incomplete networks (step 3). Provided that this detection is correct, the membership labels are propagated with minimum error. The curves of time performance reveal that it is feasible to achieve a high accuracy rate while improving time performance. For instance, in network $\mathcal{N}_b$ our method reaches accuracy greater than 90% and increases time performance in about 50%. The comparison between $\mathcal{N}_b$ and $\mathcal{N}_d$ shows that the proposed method is even more effective when the average connectivity $\langle k \rangle$ of the original network takes high values. While a sampling rate of 15% yields an accuracy rate of 70% in $\mathcal{N}_b$, the same sampling rate yields an accuracy rate of 90% in network $\mathcal{N}_d$. In the latter, our method runs around 10 times faster than the same algorithm running on the original network.

The results with the fast greedy method in step 3 in figure 7 are similar to those for the walktrap method (see figure 6). The most relevant differences are the larger deviations in time and accuracy, which occur mainly for smaller networks (see panels (a) and (d) in figure 7). Similarly to the walktrap algorithm, the processing time for $\Gamma = 1$ in our method exceeds the time in the standard method. This occurs because no node was sampled (i.e., the set of vertices remains the same) and additional edges are added due to the mapping $d_{ij}^{-1} \mapsto a_{ij}$ in step 2. As a decreasing number of nodes are sampled, the accuracy drops slowly, while the processing time drops quickly. Owing to this interplay between time and accuracy, the method proposed here can be used in practical applications.

In the light of the behavior displayed in figures 6 and 7, the average connectivity $\langle k \rangle$ seems to play a crucial role on the curves for accuracy versus sampling rates. A more detailed analysis of the relationship between the sampling rate $S$ and accuracy rate $\Gamma$ was conducted on networks $\mathcal{N}_a$ and $\mathcal{N}_b$, with the results for the walktrap and fast greedy methods being shown in figures 8 and 9 respectively. It is clear that the tuning of $\langle k \rangle$ affects the threshold $S_c$. Whenever $\langle k \rangle$ takes sufficiently low values (e.g. $\langle k \rangle = 10$ in figure 8(a)), the community structure fades away even with high sampling rates. These results suggest that the strategy developed here is especially useful when the original
Figure 5. Evolution of the community detection method based on the analysis of incomplete networks. (a) (step 1) Sampling in the original network. (b) (step 2) Construction of the incomplete network (the edge thickness is proportional to the strength of the links). (c) (step 3) Community detection in the incomplete network. (d) (step 4) Transference of the memberships obtained in the incomplete network to the original network. (e) (step 5) Label propagation in the original network. (f) (step 6) Repetition of step 5 until all nodes are classified.
network is very connected. Actually, our method is most suitable to detect communities in weighted, complete networks, for the sampling process ensures that both the number of nodes and edges decreases, thus assuring an enhancement in time performance.

The methodology proposed in this paper was also assessed in networks with four communities, and the results with the walktrap and fast greedy algorithms are summarized in figure 10. Apart from the fast greedy method applied to network $N_b$, there is a significant gain in performance without a minimal loss in accuracy. The best results were found for the walktrap method, with our method running 50% faster than the standard method with no practical loss in performance. This result suggests that our methodology can be applied in networks with many communities, which turns out to be the most prevalent scenario of networks representing real systems, as in social networks [28, 29, 30, 31].
Efficient community detection via sampling processes in complex networks

Figure 7. Dependence of the accuracy rate $\Gamma$ and computing time with the sampling rate $S$ in (a) $N_a$; (b) $N_b$; (c) $N_c$; (d) $N_d$; (e) $N_e$; (f) $N_f$. The communities were identified in step 3 with the fast greedy algorithm. The dashed lines delimit the point up to which our method runs faster than the standard method. There are no dashed lines in (a), (c), (d) and (f) because our algorithm is always faster than the standard one within the conditions in the graph.

4. Conclusion

We have demonstrated that the walktrap and fast greedy algorithms are suitable to identify communities with high accuracy even if many nodes of the real network were missing, which is a key issue in network theory for the many cases of incomplete information. Inspired by this robust behaviour, we devised a technique to detect the modular structure of networks that is based on the application of standard methods in sampled networks. Our method provided high accuracy rates while improving the time performance in networks made up of 2 or 4 communities.

As for future work, we are planning to devise an approach to identify automatically the best sampling rate that provides optimized gain in temporal complexity, given a fixed margin of error in accuracy. We also intend to conceive novel ways to propagate the memberships of nodes in step 5 of our method through techniques similar to those used in semi-supervised pattern recognition [32]. Another possibility is to investigate the applicability of novel sampling techniques to further improve the accuracy and time performance. Finally, one could verify the effect of sampling in multi-resolution
Figure 8. Dependence of $\Gamma$ obtained with the walktrap method as the sampling rate varies for networks comprising (a) $N = 512$ nodes; and (b) $N = 1024$ nodes.

Figure 9. Dependence of $\Gamma$ with the fast greedy method on the sampling rate for networks comprising (a) $N = 512$ nodes; and (b) $N = 1024$ nodes.

community analysis [33] and in networks with overlapping community structure [34].

5. Acknowledgments

The authors are grateful to FAPESP (grant numbers 2010/00927-9, 2011/50761-2 and 2013/06717-4) and CNPq (Brazil) for financial support.

References

[1] Faloutsos M, Faloutsos P and Faloutsos C 1999 On power-law relationships of the internet topology Computer Communication Review 29 (4) 251-262
Figure 10. Dependence of the accuracy rate and processing time with the sampling rate $S$ in (a-b) $N_a$ with 4 communities; (c-d) $N_b$ with 4 communities; (e-f) $N_c$ with 4 communities. The walktrap and fast greedy methods were employed to identify communities (step 3 of the proposed method) in (a-c-e) and (b-d-f), respectively. The dashed lines delimit the point up to which our method runs faster than the standard method.

[2] Siganos G, Faloutsos M, Faloutsos P and Faloutsos C 2003 Power laws and the AS-level internet topology IEEE/ACM Transactions on Networking 11 (4) 514524
Efficient community detection via sampling processes in complex networks

[3] Albert R, Jeong H and Barabsi A-L 1999 Diameter of the World Wide Web Nature 401 130-131
[4] Guimer R, Mossa S, Turtschi A and Amaral L A N 2005 The worldwide air transportation network: anomalous centrality, community structure, and cities’ global roles Proceedings of the National Academy of Science USA 102 7794-7799
[5] Porta S, Crucitti P and Latora V 2006 The network analysis of urban streets: a dual approach Physica A 369 (2) 853-866
[6] Schadschneider A 2002 Traffic flow: a statistical physics point of view 2002 Physica A 313 153-187
[7] Carreras B A, Lynch V E, Dobson I and Newman D E 2004 Complex dynamics of blackouts in power transmission systems Chaos 14 (3) 643-652
[8] Fortunato S 2010 Community detection in graphs Physics Reports 486 75-174
[9] Pons P and Latapy M 2008 Computing communities in large networks using random walks Journal of graph algorithms and applications 10 191-218
[10] Newman M E J 2004 Fast algorithm for detecting community structure in networks Physical Review E 69 066133
[11] Clauset A, Newman M E J and Moore C 2004 Finding community structure in very large networks Physical Review E 70 066111
[12] Girvan M and Newman M E J 2002 Community structure in social and biological networks Proceedings of the National Academy of Science USA 99 7821-7826
[13] Newman M E J 2006 Finding community structure in networks using the eigenvectors of matrices Physical Review E 74 036104
[14] Wilkinson D M and Huberman B A 2004 A method for finding communities of related genes Proceedings of the National Academy of Science USA 101 5241-5248
[15] Holme P, Huss M and Jeong H 2003 Subnetwork hierarchies of biochemical pathways Bioinformatics 19 (4) 532-538
[16] Viana M P, Amancio D R and Costa L da F 2013 On time-varying collaboration networks Journal of Informetrics 7 (2) 371-378
[17] Gisle P M and Danon L 2003 Community structure in jazz Advances in Complex Systems 6 565
[18] Kleinberg J and Lawrence S 2001 The structure of the Web Science 294 1849-1850
[19] Redner S 1998 How popular is your paper? An empirical study of the citation distribution European Physical Journal B 4 131-134
[20] Yan B and Gregory S 2011 Finding missing edges and communities in incomplete networks Journal of Physics A 44 495102
[21] Sprinzak E, Sattath S and Margalit H 2003 How reliable are experimental protein-protein interaction data? Journal of Molecular Biology 327 919
[22] Guimerà R and Sales-Pardo M 2009 Missing and spurious interactions and the reconstruction of complex networks Proceedings of the National Academy of Science USA 106 22073
[23] Norris J R 1998 Markov chains (Cambridge series in statistical and probabilistic mathematics)
[24] Montroll E W and Weiss G H 1965 Random walks on lattices Journal of Mathematical Physics 6 167-181
[25] Ward J H 1963 Hierarchical grouping to optimize an objective function Journal of the American Statistical Association 58 236-244
[26] Lancichinetti A and Fortunato S 2009 Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities Physical Review E 80 016118
[27] Lancichinetti A, Fortunato S and Radicchi F 2008 Benchmark graphs for testing community detection algorithms Physical Review E 78 046110
[28] Daraganova G, Pattison P, Koskinen J, Mitchell B, Bill A, Watts M and Baum S 2012 Networks and geography: modelling community network structures as the outcome of both spatial and network processes Social Networks 34 6-17
[29] Gregory S 2010 Finding overlapping communities in networks by label propagation New Journal of Physics 12 103018
[30] Kitchovitch S, Liò P 2011 Community structure in social networks: applications for epidemiological
modelling *PLoS ONE* 6 (7) e22220

[31] REX Consortium 2007 Structure of the scientific community modelling the evolution of resistance *PLoS ONE* 2 (12) e1275

[32] Silva T C and Zhao L 2012 Network-based stochastic semi-supervised learning *IEEE Transactions on Neural Networks and Learning Systems* 23 385-398

[33] Zhang J, Zhang K, Xu X, Tse C K and Small M 2009 Seeding the kernels in graphs: toward multi-resolution community analysis *New Journal of Physics* 11 113003

[34] Lancichinetti A, Fortunato S and Kertesz J 2009 Detecting the overlapping and hierarchical community structure in complex networks *New Journal of Physics* 11 033015