

Finding and testing network communities by lumped Markov chains

Carlo Piccardi *

*Department of Electronics and Information, Politecnico di Milano, Piazza Leonardo da Vinci 32, I-20133 Milano, Italy, email: carlo.piccardi@polimi.it

Identifying communities (or clusters), namely groups of nodes with comparatively strong internal connectivity, is a fundamental task for deeply understanding the structure and function of a network. Yet, there is a lack of formal criteria for defining communities and for testing their significance. We propose a sharp definition which is based on a significance threshold. By means of a lumped Markov chain model of a random walker, a quality measure called “persistence probability” is associated to a cluster. Then the cluster is defined as an “ α -community” if such a probability is not smaller than α . Consistently, a partition composed of α -communities is an “ α -partition”. These definitions turn out to be very effective for finding and testing communities. If a set of candidate partitions is available, setting the desired α -level allows one to immediately select the α -partition with the finest decomposition. Simultaneously, the persistence probabilities quantify the significance of each single community. Given its ability in individually assessing the quality of each cluster, this approach can also disclose single well-defined communities even in networks which overall do not possess a definite clusterized structure.

networks | communities | Markov chains | random walks

Complex networks are currently one of the most extensively studied subjects in the field of applied mathematics. In the last fifteen years, a huge number of theoretical results have been put forward, and almost any field of science and technology has benefit from the application of such results to specific problems [1, 2, 3, 4].

One of the most promising but challenging tasks in network science is *community analysis*, which is aimed at revealing possible partitions of a network into subsets of nodes (*communities*, or *clusters*) with dense intra- but sparse inter-group connections. Finding and analyzing such partitions often provides invaluable help in deeply understanding the structure and function of a network, as widely demonstrated by several case studies in social sciences [5, 6], biology [7], economics [8], or information science [9], just to name a few.

Despite the abundance of contributions on this subject (see [10] for a survey), the issue of community analysis cannot be considered satisfactorily solved. First of all, finding communities is a computationally hard task, because the “best” partition must be sought for in a set whose cardinality grows faster than exponentially with the number of nodes. The exhaustive enumeration of the partitions is thus impossible, and heuristic techniques must be employed. Secondly, and perhaps more important, there is no widespread consensus on formal criteria for defining communities and for testing their significance [10]. When a subnetwork can actually be considered to form a community, namely a group of nodes with comparatively strong internal connectivity? Probably the most important attempt to answer this question was put forward by Newman and coworkers [11, 5, 12], who defined a quality index called *modularity* which quantifies, for a given partition of the network into candidate communities, to what extent the distribution of the intra-/inter-community edges is anomalous with respect to a suitably defined random network. Since high modularity values are obtained in presence of groups of nodes with comparatively large intra-community edge density, maximizing modularity should put in evidence

the “best” partition. This method has been proven successful in many circumstances but, on the other hand, it has been widely demonstrated that, due to intrinsic limitations, it does not necessarily always yield a significant partition [13, 14, 10]. And even when it does, it quantifies the quality of a partition but not of each individual community.

This paper introduces a sharp definition of community which is based on a threshold of significance. More precisely, once a level $0 < \alpha < 1$ is specified, a node cluster is defined to be an α -community if the probability that a random walker, which is currently in one of the cluster’s nodes, remains in the cluster in the next step is not smaller than α . Such a probability is obtained from an approximate *lumped Markov chain* model of the random walker (i.e., a reduced-order Markov chain in which the communities of the original network become nodes) which is easily derived from the original (high-order) Markov chain model. Consistently, a partition composed of α -communities is defined to be an α -partition.

If equipped with an effective method for generating a set of “good” candidate partitions, the notions of α -community and α -partition provide a framework for simultaneously finding communities and testing their significance. For that, the desired significance level α is first fixed. Then, a family of partitions is derived and each partition is immediately checked to assess whether it is formed by α -communities. This allows one to identify the α -partitions, and to select one of them. Typically, one searches for communities which are at the same time small (to effectively decompose the network) and significant (with much more internal than external connectivity). For that, a guideline is that of selecting, among the available α -partitions, the one with the largest number of communities.

But the notion of α -community can also be useful in a partially different way. It may happen that, for a given significance level α , no α -partitions are found. Yet, one or a few α -communities could exist. They correspond to strongly connected groups of nodes, even in a network which, overall, does not possess a definite clusterized structure. Or, finally, one can assess the significance of the results of a single-partition method, such as modularity optimization [5], and obtain an immediate assessment of the α -significance of each single community and, consequently, of the entire partition.

In the paper, we first introduce the lumped Markov chain model of the random walker and define the notions of α -community and α -partition. Testing the α -significance of a given community or partition turns out to be extremely parsimonious in computational terms. Then we analyze a few examples of application and, for that, we propose an effective algorithm for deriving a meaningful set of partitions. The algorithm, which applies hierarchical cluster analysis, is again based on the Markov chain model of a random walker and, consequently, it involves a notion of similarity/distance among nodes which is consistent with the significance criterion above introduced. We finally compare this approach, which can be applied to fully general networks (i.e., directed and weighted), with other community analysis methods having a similar philosophy.

Networks, α -Communities, and α -Partitions

Consider a network with nodes $\mathbb{N} = \{1, 2, \dots, N\}$ and L edges. We consider the most general case of directed and weighted network, and we denote by $W = [w_{ij}]$ the $N \times N$ weight matrix, where $w_{ij} \geq 0$ is the weight of the edge $i \rightarrow j$. The connectivity matrix $A = [a_{ij}]$ is the $N \times N$ binary matrix where $a_{ij} = 1$ if $w_{ij} > 0$, and $a_{ij} = 0$ otherwise. If the network is actually undirected we have $W = W'$ and $A = A'$, and if it is unweighted we let $W = A$ (i.e., all weights equal to 1). We assume that the network is strongly connected (e.g., [3]), namely there exists an oriented path from any i to any j . If the network is directed, for each node i we define the (total) degree as $k_i = k_i^{in} + k_i^{out} = \sum_j a_{ji} + \sum_j a_{ij}$, whereas $k_i = \sum_j a_{ji} = \sum_j a_{ij}$ for undirected network. The average degree is given by $\langle k \rangle = \sum_i k_i / N$. Similarly, for a directed network the in-, out-, and total strength of node i are given by $s_i^{in} = \sum_j w_{ji}$, $s_i^{out} = \sum_j w_{ij}$, and $s_i = s_i^{in} + s_i^{out}$, respectively, and the total network strength by $s = \sum_{ij} w_{ij}$. If the network is undirected we have instead $s_i = s_i^{in} = s_i^{out} = \sum_j w_{ji} = \sum_j w_{ij}$ and $s = \sum_{ij} w_{ij} / 2$.

A N -state Markov chain $\pi_{t+1} = \pi_t P$, with $\pi_t = (\pi_{1,t}, \pi_{2,t}, \dots, \pi_{N,t})$, can be associated to the N -node network by row-normalizing the weight matrix W , namely by letting the transition probability from i to j equal to

$$p_{ij} = \frac{w_{ij}}{\sum_j w_{ij}} = \frac{w_{ij}}{s_i^{out}}. \quad [1]$$

The quantity p_{ij} is the probability that a random walker which is in node i jumps to node j , and $\pi_{i,t}$ is the probability of being in node i at time t . The transition matrix $P = [p_{ij}]$ is a row-stochastic (or Markov) matrix ($0 \leq p_{ij} \leq 1$ for all i, j , and $\sum_j p_{ij} = 1$ for all i). Furthermore, P is irreducible since the network is connected. This implies that the equation $\pi = \pi P$ has a unique solution π , which is strictly positive ($\pi_i > 0$ for all i) [15] and corresponds to the stationary Markov chain state probability distribution. For undirected networks one can easily check that $\pi = (s_1 s_2 \dots s_N) / (2s)$, whereas for directed networks a general closed form does not exist and π has to be numerically computed.

We denote by \mathbb{P}_q a partition of \mathbb{N} in q subsets (or sub-networks), namely $\mathbb{P}_q = \{\mathbb{C}_1, \mathbb{C}_2, \dots, \mathbb{C}_q\}$ with $\bigcup_c \mathbb{C}_c = \mathbb{N}$ and $\mathbb{C}_c \cap \mathbb{C}_d = \emptyset$ for all c, d . In rough terms, the sub-network \mathbb{C}_c is called a *community* (or *cluster*) if it has a high internal density of weight, i.e., if the total weight of the edges internal to \mathbb{C}_c is much larger than that of the edges connecting \mathbb{C}_c to the rest of the network. The community analysis of a given network consists therefore in finding the “best” partition \mathbb{P} , according to some criterion. Despite a huge amount of contributions, there is however no widespread consensus on formal criteria for defining communities and for testing their significance [10]. As a consequence, in many situations a more fruitful approach is that of searching for a few, “good” partitions $\mathbb{P}', \mathbb{P}'', \dots$, among which selecting with common sense and experience.

Defining a partition \mathbb{P}_q induces a q -state meta-network, where communities become meta-nodes. The rigorous description of the dynamics of the random walker at this scale by a *lumped Markov chain*, however, is not possible if not in special cases [16] - actually, the Markovian property is not even preserved in general. Despite this limitation, a q -state Markov chain can be defined, which correctly describes the random walker at the aggregate level provided the stochastic process is started at the stationary distribution π [17, 18]. This lumped Markov chain is defined by the $q \times q$ row-stochastic matrix

$$U = [\text{diag}(\pi H)]^{-1} H' \text{diag}(\pi) P H, \quad [2]$$

where H (*collecting matrix*) is a $N \times q$ binary matrix coding the partition \mathbb{P}_q , i.e., its entry h_{ic} is 1 if and only if node $i \in \mathbb{C}_c$. The lumped Markov chain $\Pi_{t+1} = \Pi_t U$ shares the stationary distribution with the original one (suitably collected), namely $\Pi = \pi H$ satisfies $\Pi = \Pi U$. On the contrary, starting from an arbitrary π_0 , the lumped Markov chain $\Pi = \Pi U$ started at $\Pi_0 = \pi_0 H$ provides, in general, only an approximate description of the evolution of πH . The difference between the real and approximate Π , however, tends exponentially to zero if the two chains are regular [15], since they converge, by definition, to the same stationary state.

The ability of the lumped Markov chain to describe the random walk dynamics only at stationarity is not a limitation for our purposes, as it will be demonstrated by the examples of application. Note that the entry u_{cd} of U is the probability that the random walker is at time $(t+1)$ in any of the nodes of community d , provided it is at time t in any of the nodes of community c . The diagonal term u_{cc} is defined *peristence probability* of community c . Large values of u_{cc} are expected for significant communities. In fact, the expected escape time from \mathbb{C}_c is $\tau_c = (1 - u_{cc})^{-1}$: the walker will spend long time within the same community if the weights of the internal edges are comparatively large with respect to those pointing outside. Given a value $0 < \alpha < 1$, \mathbb{C}_c is defined *α -community* if $u_{cc} \geq \alpha$. Thus α acts as a selection parameter, as sharply qualifies communities with respect to a given threshold of significance. Consistently, \mathbb{P}_q is defined *α -partition* if it is composed of α -communities, namely $u_{cc} \geq \alpha$ for all $c = 1, 2, \dots, q$.

Consider the simple 12-node network of Fig. 1 [19], which is purposely composed of three clusters. Four partitions are considered, corresponding to finer and finer divisions, and the u_{cc} -s are computed for each candidate community. As long as the latter coincide with the “natural” communities, or with the union of two of them, all the u_{cc} -s are rather large. But as soon as a natural community is broken, some very low persistence probabilities are found. This result can be used in a twofold manner, as extensively shown in the next section. On one hand, if a set of finer and finer partitions is analyzed, the sudden drop of a u_{cc} is the signal that a significant community has been broken: the network decomposition must be stopped before this event. On the other hand, if a single partition is given and its significance has to be assessed, the u_{cc} -s immediately quantify the quality of the partition but also of each individual community.

Applications and Examples

The proposed method is now applied to assess the significance of sets of partitions related to a variety of networks. An algorithm for deriving partitions is first introduced, implementing hierarchical cluster analysis after a random-walk-based node distance is defined. Then the results related to three networks are discussed: a synthetic benchmark network with built-in cluster structure; a real-world network with a rather strong community structure; and another real-world network with weak clustering but with a few well-defined communities.

Deriving Partitions. Cluster analysis can be used to group “similar nodes” into candidate communities. This needs defining a meaningful *similarity/distance* among each pair of nodes. Such a definition is by no means obvious: among the many proposals [10], a few exploit random walks to induce a suitable similarity measure (e.g., [20, 21, 22, 23, 24]). We follow this line by proposing an approach in which, however, we do not explicitly perform random walks in a Monte Carlo

fashion, but derive analytically the global behavior of a large number M of walkers (a “fleet”) started from each node i .

Consider a large number M of repetitions of a random walk started from i . For each repetition, the probability that the walker is in j after t steps is $[P^t]_{ij}$. Thus, if M random walks of length T are performed from i , the expected number of visits to j in any time instant in $1 \leq t \leq T$ is $M \sum_{t=1}^T [P^t]_{ij}$. By averaging with respect to M , we propose a (symmetric) similarity σ_{ij} defined by

$$\sigma_{ij} = \sigma_{ji} = \sum_{t=1}^T \left([P^t]_{ij} + [P^t]_{ji} \right). \quad [3]$$

Note that this is conceptually equivalent to an explicit random walk approach, but with an arbitrarily large number M of repetitions from each starting node instead of one only. Most notably, the results do not depend on the actual stochastic realization of the random walks. We finally define the distance $d_{ij} = d_{ji}$ between nodes (i, j) by complementing the similarity and normalizing the results between 0 and 1:

$$d_{ij} = d_{ji} = 1 - \frac{\sigma_{ij} - \min \sigma_{ij}}{\max \sigma_{ij} - \min \sigma_{ij}}. \quad [4]$$

The rationale underlying the definition of s and d is to assign nodes (i, j) a large similarity if a numerous fleet of random walkers started in i (resp. j) makes a large number of visits to j (resp. i) within a sufficiently small time horizon T . The notion of community induced by this metric, therefore, is that of a subnetwork where a random walker has a large probability of circulating for quite a long time, before eventually leaving to reach another group. This is conceptually consistent with the definition of α -community above introduced. The choice of the time horizon T is potentially critical: if too large, the probability of visiting a given state j becomes independent of the starting state since it tends to π_j , whereas if T is too small the information gathered is insufficient. We will return later to this point.

LFR benchmark. Lancichinetti, Fortunato, and Radicchi (LFR) [25] proposed a family of synthetically generated graphs, explicitly designed to serve as benchmarks for testing community detection algorithms. They explicitly take into account two properties found in real networks, namely the heterogeneity in the distributions of node degrees and community sizes. Both of the latter are taken as power laws, with prescribed exponents γ and β , respectively. In addition, the network is defined by prescribing the number N of nodes, the average degree $\langle k \rangle$, and a *mixing parameter* μ such that each node shares a fraction $1 - \mu$ of its edges with the other nodes of its own community, and a fraction μ with the rest of the network. The benchmark generating method was later extended to oriented and weighted networks [26] - here we consider an example of an undirected, unweighted network with $N = 1000$, $\langle k \rangle = 20$, $\mu = 0.25$, $\gamma = 2$, and $\beta = 1$. The network we obtained turns out to be formed by 38 communities, with dimensions ranging from 10 to 49 nodes each.

Cluster analysis yields a different dendrogram for each time horizon T , whose choice is thus nontrivial. At the two extremes, setting $T = 1$ restricts the pairs of nodes which are candidate to nonzero similarity to neighboring pairs only, whereas larger and larger values of T tend to make any node equally similar to any other. We found that an effective selection of T can be empirically obtained by maximizing the *cophenetic correlation coefficient* C , which is defined as the linear correlation between the distances d_{ij} and the *cophenetic distances* c_{ij} [27]. The latter are a product of the hierarchi-

cal cluster analysis: for any node pair (i, j) , the cophenetic distance c_{ij} is the height of the link joining (directly or indirectly) nodes (i, j) in the dendrogram. The value of C is generally used to assess whether the adopted distance d_{ij} induces an effective clusterization (notice that C qualifies the entire dendrogram, and not a network partition), although limitations have been observed in specific applications [28]. Figure 2 shows the dependence of C on T : we take $T = 12$, which attains the maximum $C = 0.905$. The related dendrogram is in the same figure.

Horizontal top-down cross-sections of the dendrogram identify a sequence $\mathbb{P}_2, \mathbb{P}_3, \dots$ of partitions with increasing number of candidate communities. For each \mathbb{P}_q we compute U according to (2), and plot its diagonal terms in the *persistence probabilities’ diagram* of Fig. 3. The diagram reveals a sharp discontinuity. For $q \leq 38$, all the u_{cc} -s are rather large ($u_{cc} \geq 0.735$ for all c). This means that significant communities are identified: in rigorous terms, all the proposed partitions \mathbb{P}_q with $2 \leq q \leq 38$ are α -partitions with $\alpha = 0.735$. For $q \geq 39$ significant communities are broken, as revealed by the sudden drop of a larger and larger number of u_{cc} -s. Remind that 38 is exactly the number of communities planted in the synthetically generated network. It is worth mentioning that, if we search for the max-modularity partition (we used the so-called “Louvain algorithm” [29], proved to be one of the most reliable [30]), we obtain a partition with $q = 34$ communities, with modularity $Q = 0.714$. The number of communities of the planted partition is thus not perfectly recovered (small communities tend to be aggregated). Nonetheless, for the obtained \mathbb{P}_{34} partition the persistence probabilities are in the range $0.737 \leq u_{cc} \leq 0.772$, which is qualitatively consistent with the results of Fig. 3.

We have finally compared the built-in partition planted in the LFR benchmark network with the partition \mathbb{P}_{38} obtained with our method, as well as with the “max-modularity” partition. The comparison is in terms of the *normalized mutual information* I , a reliable and often used measure of partition similarity, introduced by [31] to the network research community. Here we only point out that $I = 1$ when the two partitions are identical, whereas I has zero expected value for independent partitions. We obtained $I = 0.992$ for the partition resulting from our method (actually, we checked that as few as 0.08% of the pairs i, j are misclassified), and a slightly smaller $I = 0.987$ for the “max-modularity” partition.

Netscience network. The Netscience network is a weighted, undirected, social network describing the collaborations (up to year 2006) among researchers in network science, the weight of the edge connecting two researchers being proportional to the number of papers they have co-authored [12]. Its giant component has $N = 379$ nodes, and it is generally considered an example of a real network with a rather strong community structure. Many methods for network analysis, included community detection algorithms, have been tested and discussed on this example (e.g., [32, 33, 34]).

At $T = 6$ we get the dendrogram with largest C , and the resulting persistence probabilities’ diagram is in Fig. 4. The plot has a less clear structure than that of the LFR network (Fig. 3): the proper q must be selected with a trade-off between a finer decomposition (large q) and a higher significance of the communities (small q). For example, all the partitions with q up to 10 are α -partitions with $\alpha > 0.9$. But, if less stringent significance levels are required, partition with $q \leq 27$, or even $q \leq 35$, seem to be perfectly meaningful.

It is instructive to compare these results with those obtained, on the same case study, by the *graph stability* approach proposed by Delvenne et al. [32] (a detailed compari-

son of the two methods is in the next section). By means of the KVV algorithm [35] (a hierarchical, divisive, non-binary, graph clustering method), they obtain a sequence of six partitions, with $q = 2, 3, 5, 15, 17, 21$. Analyzing and comparing the *stability curve* (i.e., the autocovariance function of a signal emitted by a random walker) of each of them, the authors suggest their partition with $q = 5$ as the more reliable, as it has the largest stability over a longer time span with respect to any other. We created the persistence probabilities' diagram of the six partitions of [32], and compared it with our diagram in Fig. 5. The partition $q = 5$ of [32] confirms to be definitely more significant than those with finer decomposition (i.e., $q = 15, 17, 21$) according to our criterion too. Actually, our and their \mathbb{P}_5 partitions share the same minimal $u_{cc} = 0.952$, due to a common 22-node community. The two partitions are, however, partially different (the normalized mutual information is $I = 0.886$, with about 6% of differently classified node pairs).

The inspection of Fig. 5 also reveals that, for each given q , the partitions obtained with our method are superior than those proposed in [32], provided the criterion put forward in this paper is adopted. In fact, they are α -partitions with an α value which is larger (or at least not smaller) in all six cases. Actually, while the criterion of [32] ranks partitions by “averaging” among the communities, our approach is a “worst-case” one: by selecting an α -partition one guarantees that the “worst” community has a persistence probability not less than α . Finally, note that in the gap from $q = 6$ to 15, where no partition is obtained by the KVV divisive algorithm, our partition generating algorithm provide a set of finer and finer partitions, whose quality only slowly deteriorates as q increases. The analyst of the network can fruitfully select in this interval a proper trade-off between fine granularity and significance of the partition.

Neural network. The third example concerns a directed, weighted network, representing the neural connections of the worm *Caenorhabditis elegans*. Starting from Watts and Strogatz's seminal work [36], different versions of this graph have become a standard benchmark for network analysis. We consider the directed, weighted version (whose largest connected component has $N = 239$ nodes), which does not display a definite community structure. In fact, the maximum modularity (estimated as in [29]) is rather small, namely $Q = 0.486$, if compared to other examples of comparable dimension (e.g., $Q = 0.831$ for the Netscience network). The less clustered structure emerges even visually from the dendrogram of Fig. 6, where only few groups of nodes appear well separated from the rest (compare, e.g., with Fig. 2).

We show that our method is able to detect such groups, namely to isolate well-defined communities even in a network which overall does not possess a definite clustered structure. Consider the persistence probabilities' diagram of Fig. 7. With the exception of the trivial cases $q = 2$ and 3, no α -partition exists with α reasonably large. Nonetheless, a few α -communities with $\alpha \geq 0.8$ appear and are stably detected in a rather wide range of q . More precisely, the same set of five communities with $u_{cc} \geq 0.826$ are revealed in the range $14 \leq q \leq 20$. They are clusters, of dimension ranging from 18 to 29 nodes, with comparatively rather strong internal connectivity. Any other candidate cluster, instead, turns out to have a much smaller u_{cc} value and, therefore, it cannot be considered to be a significant community.

Discussion and Conclusions

In this paper, we have shown that associating a lumped Markov chain to a given network partition (i.e., a set of communities) provides an effective tool for testing the significance of each single community and, consequently, of the entire partition. As a matter of fact, the diagonal terms (called persistence probabilities) of the lumped Markov matrix can be used as quality measures for each individual community. If a threshold level $0 < \alpha < 1$ is fixed, a sharp criterion for defining a community as “significant” is therefore that of requiring that its persistence probability is not less than α .

If an effective method for generating a set of “good” partitions is available, the above criterion can be used to rapidly select one of them among those complying with the prescribed α -significance, typically the one with the finest network decomposition (i.e., the largest number of communities). We have used a generator of partitions based on hierarchical cluster analysis, where the node distance is again defined on the basis of a Markov chain random walk model. Overall, the method has fair computational requirements, and can be applied to fully general networks (i.e., directed and weighted). Its effectiveness has been demonstrated on several medium-scale examples.

The proposed approach has important connections with two recently published community analysis methods. Delvenne et al. [32] show that the autocorrelation function of a signal emitted by a random walker, with value c as long as the walker is in a node $i \in \mathbb{C}_c$, can be expressed in terms of the clustered autocovariance matrix $R_t = H' [\text{diag}(\pi) P^t - \pi' \pi] H$, and they define the stability of the partition H as $r_t^H = \min_{s=0,1,\dots,t} \text{trace}(R_s)$. Given a set of candidate partitions, the *graph stability* function $r_t = \max_H r_t^H$ puts in evidence, for each time instant t , which is the “optimal” partition. It is suggested in [32] that the most relevant partitions are those which are optimal over long time windows. It is straightforward to check that our matrix U is related to the step-1 autocovariance R_1 by $R_1 + \Pi' \Pi = \text{diag}(\Pi) U$. The two methods are thus based on the same ground, but our approach has two advantages: first, for each partition H we do not compute a long time-dependent sequence $R_1, R_2, \dots, R_{t_{\max}}$ (with t_{\max} of the same order as N) of $q \times q$ matrices, but the sole matrix U , with an important reduction in the computational burden. Second, the full list of the persistence probabilities u_{cc} allows one to test the significance of each single community, whereas the stability of the clustering r_t^H averages among all the communities.

Another work with important connections is that of Weinan et al. [37], who suggest to explicitly find the “best” (in a suitable sense) q -state approximated lumped Markov chain. This boils out to the formulation of a minimization problem, after a metric on the space of stochastic matrices is introduced. A drawback of this method is however that q must be *a priori* specified, whereas often identifying the correct number of communities is the main goal of the analysis. For the same reason, it can hardly support the discussion of the significance and convenience of choosing one partition instead of another. We argue that this method could be used, jointly with the one proposed in this paper, to generate a set of partitions with increasing values of $q = 2, 3, \dots$, by repeatedly solving the above problem. Then, their significance could be tested with the tool of the persistence probabilities' diagram. It is not guaranteed, however, that the proposed partitions are “good” in terms of the minimal u_{cc} (i.e., that they are α -partitions with large α). It is therefore a point deserving further study.

1. Strogatz SH (2001) Exploring complex networks. *Nature* 410:268–276.
2. Boccaletti S, Latora V, Moreno Y, Chavez M, Hwang DH (2006) Complex networks: Structure and dynamics. *Phys Rep* 424:175–308.
3. Barrat A, Barthlemy M, Vespignani A (2008) *Dynamical Processes on Complex Networks* (Cambridge University Press).
4. Newman MEJ (2010) *Networks: An Introduction* (Oxford University Press).
5. Newman MEJ (2006) Modularity and community structure in networks. *Proc Natl Acad Sci USA* 103:8577–8582.
6. Guimera R, Sales-Pardo M, Amaral LAN (2007) Module identification in bipartite and directed networks. *Phys Rev E* 76:036102.
7. Jonsson P, Cavanna T, Zicha D, Bates P (2006) Cluster analysis of networks generated through homology: automatic identification of important protein communities involved in cancer metastasis. *BMC Bioinformatics* 7.
8. Piccardi C, Calatroni L, Bertoni F (2010) Communities in Italian corporate networks. *Physica A* 389:5247–5258.
9. Flake G, Lawrence S, Giles C, Coetzee F (2002) Self-organization and identification of web communities. *Computer* 35:66–71.
10. Fortunato S (2010) Community detection in graphs. *Phys Rep* 486:75–174.
11. Newman MEJ, Girvan M (2004) Finding and evaluating community structure in networks. *Phys Rev E* 69:026113.
12. Newman MEJ (2006) Finding community structure in networks using the eigenvectors of matrices. *Phys Rev E* 74:036104.
13. Reichardt J, Bornholdt S (2006) When are networks truly modular? *Physica D* 224:20–26.
14. Fortunato S, Barthlemy M (2007) Resolution limit in community detection. *Proc Natl Acad Sci USA* 104:36–41.
15. Meyer C (2000) *Matrix Analysis and Applied Linear Algebra* (SIAM).
16. Kemeny JG, Snell JL (1976) *Finite Markov Chains* (Springer-Verlag).
17. Buchholz P (1994) Exact and ordinary lumpability in finite Markov-chains. *J Appl Probab* 31:59–75.
18. Hoffmann KH, Salamon P (2009) Bounding the lumping error in Markov chain dynamics. *Appl Math Lett* 22:1471–1475.
19. Fortunato S, Castellano C (2009) Community structure in graphs. In Meyers, RA, ed., *Encyclopedia of Complexity and System Science* (Springer-Verlag Berlin), pp. 1141–1163.
20. Zhou H (2003) Distance, dissimilarity index, and network community structure. *Phys Rev E* 67.
21. Pons P, Latapy M (2005) Computing communities in large networks using random walks. In Yolum, P and Gungor, T and Gurgun, F and Ozturan, C, ed., *Computer and Information Sciences - ISCIS 2005, Proceedings* (Springer-Verlag Berlin), vol. 3733 of *Lecture Notes In Computer Science*, pp. 284–293.
22. Fouss F, Pirotte A, Renders JM, Saerens M (2007) Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Trans Knowl Data Eng* 19:355–369.
23. Rosvall M, Bergstrom CT (2008) Maps of random walks on complex networks reveal community structure. *Proc Natl Acad Sci USA* 105:1118–1123.
24. Steinhilber K, Chawla NV (2010) Identifying and evaluating community structure in complex networks. *Pattern Recognit Lett* 31:413–421.
25. Lancichinetti A, Fortunato S, Radicchi F (2008) Benchmark graphs for testing community detection algorithms. *Phys Rev E* 78.
26. Lancichinetti A, Fortunato S (2009) Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys Rev E* 80.
27. Everitt BS, Landau S, Leese M, Stahl D (2011) *Cluster Analysis*, 5th ed. (John Wiley & Sons).
28. Holgersson M (1978) The limited value of cophenetic correlation as a clustering criterion. *Pattern Recognit* 10:287–295.
29. Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. *J Stat Mech-Theory Exp* :P10008.
30. Lancichinetti A, Fortunato S (2009) Community detection algorithms: A comparative analysis. *Phys Rev E* 80:056117.
31. Danon L, Diaz-Guilera A, Duch J, Arenas A (2005) Comparing community structure identification. *J Stat Mech-Theory Exp* :P09008.
32. Delvenne JC, Yaliraki SN, Barahona M (2010) Stability of graph communities across time scales. *Proc Natl Acad Sci USA* 107:12755–12760.
33. Narayanam R, Narahari Y (2011) A Shapley Value-Based Approach to Discover Influential Nodes in Social Networks. *IEEE Trans Autom Sci Eng* 8:130–147.
34. Cafieri S, Hansen P, Liberti L (2011) A locally optimal heuristic for modularity maximization of networks. *Phys Rev E* :, in press.
35. Kannan R, Vempala S, Vetta A (2004) On clusterings: Good, bad and spectral. *J ACM* 51:497–515.
36. Watts DJ, Strogatz SH (1998) Collective dynamics of 'small-world' networks. *Nature* 393:440–442.
37. Weinan E, Li T, Vanden-Eijnden E (2008) Optimal partition and effective dynamics of complex networks. *Proc Natl Acad Sci USA* 105:7907–7912.

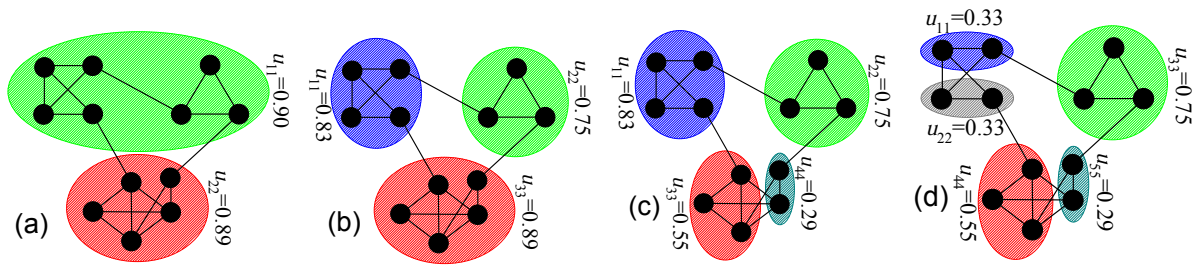


Fig. 1. Four different partitions (with increasing number q of communities) of the same network. The persistence probabilities u_{cc} remain rather large as long as the network is partitioned into “natural” communities. Passing from (b) to (c), and from (c) to (d), significant communities are broken, with a sudden drop of the relevant persistence probabilities.

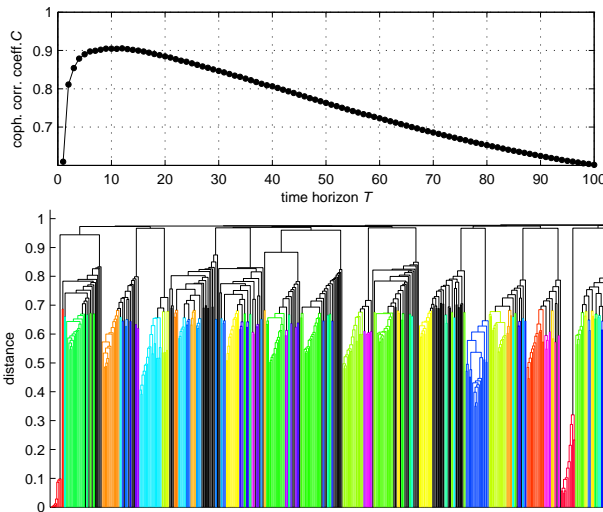


Fig. 2. LFR benchmark network. Above: The cophenetic correlation coefficient C as a function of T . The maximum is attained at $T = 12$. Below: The dendrogram obtained with $T = 12$ (only half of the plot is presented for readability).

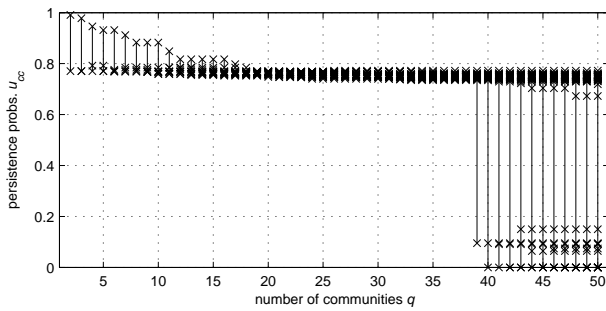


Fig. 3. The persistence probabilities' diagram of the LFR benchmark network. For a partition with q clusters, crosses denote the values of the q diagonal terms u_{cc} of the matrix U . Vertical straight lines are only for visual aid.

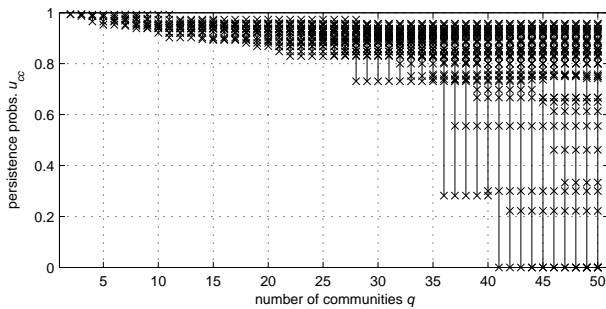


Fig. 4. The persistence probabilities' diagram of the Netscience network.

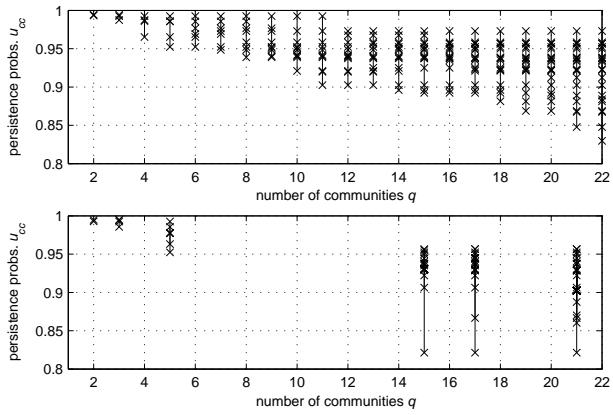


Fig. 5. Comparison of two persistence probabilities' diagrams for the Netscience network (the two plots are in the same scale). Above: blow-up of the diagram of Fig. 4 (our results). Below: the diagram related to the six partitions proposed in [32].

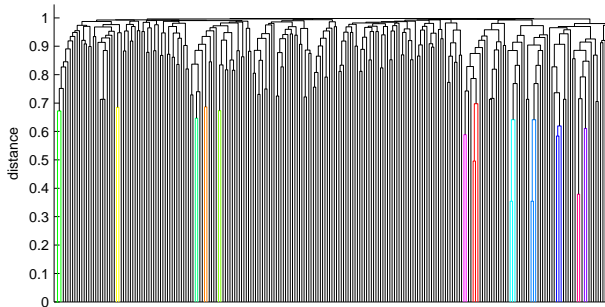


Fig. 6. Neural network. The dendrogram obtained with $T = 3$.

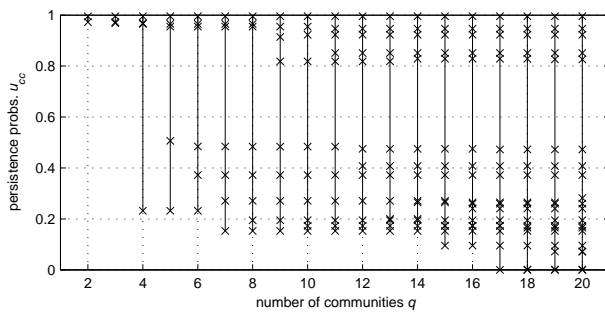


Fig. 7. The persistence probabilities' diagram of the neural network.