COMMUNITIES IN COMPLEX NETWORKS: IDENTIFICATION AT DIFFERENT LEVELS

Alex Arenas, Jordi Duch and **Sergi Gómez**
Departament Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Spain

Leon Danon
Mathematics Institute, University of Warwick, Great Britain

Albert Díaz-Guilera
Departament Física Fonamental, Universitat de Barcelona, Spain

Keywords: Communities, hierarchies, overlap, dynamics

Contents

Summary

We present here and compare the most common approaches to community structure identification in terms of sensitivity and computational cost. The work is intended as an introduction as well as a proposal for a standard benchmark test of community detection methods.

1. Introduction

The analysis of complex networks has received a vast amount of attention from the scientific community during the last decade. Statistical physicists in particular have become interested in the study of networks describing the topologies of a wide variety of systems, from biological technological or social networks. Although several questions have been addressed (see the review paper by Costa et al. for a complete set of measurements), many important ones still resist complete resolution. One such problem is the analysis of modular structure found in many networks. Distinct modules or communities within networks can loosely be defined as subsets of nodes which are more densely linked, when compared to the rest of the network. Such communities, as usually called in social sciences, have been observed, using some of the methods we shall go on to describe, in many different contexts, including biological networks, economic networks and most notably social networks. As a result, the problem of identification of communities has been the focus of many recent efforts. As a concrete example we show in Figure 1 the network representing the Spanish research community of Statistical and Nonlinear Physicists (FISES, http://www.fises.es).

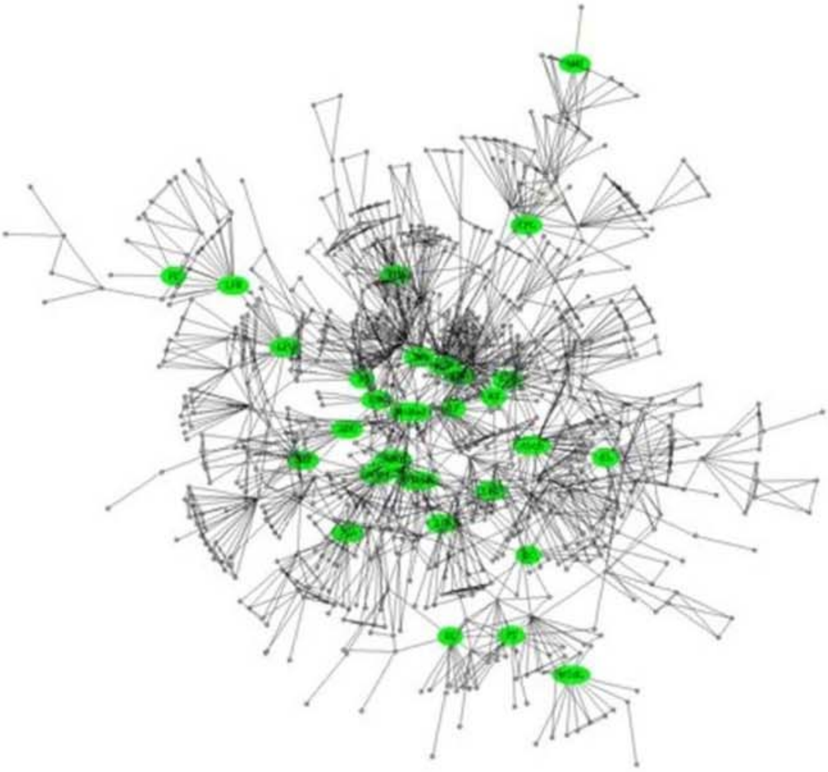

Figure 1. FisEs network. Network of scientists that contributed to the Statistical Physics (Física Estadística) conferences in Spain.

We consider two scientists linked if they have co-authored a panel contribution to any of the conferences. To be able to consider the historical structure of this network we ``accumulate" the network over all the conferences, that is, once a link is created, it remains, even if the authors never collaborated again. The final network (accumulated over all the years) is comprised of 784 nodes with 655 (84%) of those belonging to the giant component. Green nodes denote the member of the scientific committees.

Nodes belonging to the same community are more than likely to have other properties in common and hence community detection in large networks is potentially very useful for instance when trying to understand dynamical properties. In the world wide web, community analysis has uncovered thematic clusters. In biochemical or neural networks, communities may be functional groups, and separating the network into such groups could simplify the functional analysis considerably.

The problem of community detection has been the subject of study in various disciplines. A simpler version of this problem, the graph bi-partitioning problem (GBP) has been the topic of study in the realm of computer science for decades. Here one looks to separate the graph into two equal-size communities, which are connected with the minimum number of links. This is indeed an NP complete problem; however several methods have been proposed to reduce the complexity of the task. In real networks one cannot assume how many communities there are, but in general it is more than two. This makes the process much more costly.

Furthermore communities can be organized in hierarchies, meaning that different organizational levels can be simultaneously important and the question to the best partition has not a single answer. This hierarchical organization strongly affects the dynamical properties of networks. Another additional issue is that sometimes there is not a clear separation among communities and they present a certain degree of overlapping.

In this chapter we would like to present the recent advances made in the field of community identification in networks in a clear and simple fashion. To this end, the sections are organized as follows. In the next section we describe some ways to define communities in a network context. Following this, we present a method to evaluate a particular partition of a network. Then, we go on to describe the various recent methods starting with link removal methods, going on to agglomerative methods, followed by methods optimizing modularity and finally “other” methods. Some of the methods presented do not necessarily fit into just one of these classifications, and there may be some overlap. We finally introduce different structural organizations in networks and dynamical applications of modular networks.

2. Definitions of Communities

There is not a unique definition of what a community is, instead the idea of communities is different and has been evolving depending on the field that defines it. The first definitions of community come from the field of social networks, where the communities are studied and understood according to the effect that an individual player has on the network and vice versa. Some of these ideas have been used and developed by some of the methods we present below, while new approaches have also been adopted from other fields such as physics or mathematics.

The different definitions of what is a community are all based in the concept of a subgraph, that is, groups of nodes and all the connections between them. The definitions can be classified into two main conceptual categories, those who use self-referral information and those based on comparative definitions.

Self referring definitions only use information of the structure of the network to decide what groups of nodes can be considered as a community. The most restricting and simple community structure is a clique, defined as a subgraph that is fully connected (i.e. it has all the possible edges between its nodes). Since this constraint is rarely fulfilled in real sparse networks, there are other approaches that relax it, such as n-cliques, n-clans and n-clubs. Self-referring definitions, while useful in characterizing communities, which are already known, are not the best choice while trying to find them since the methods to find the cliques in a network is very costly.

A second type of definitions use topological information to compare if a group of nodes is a community or not, for instance, counting how many links have the nodes of the subgraph inside of it and how many links have them with nodes outside the subgraph. The strong definition of community requires that all the nodes of a community must have a larger number of links to members of the same community than to members of other communities. A lighter version of this definition is the weak definition of community proposed by Radicchi et al., where it is required that the sum of links inside the community is larger than the total number of links to the outside. This definition and some small variations of it is the most used in the majority of the methods that we will present later, since comparing the internal structure of a community to the external structure gives rise to a measure of how good a particular partition is.

3. Evaluating Community Identification

Once a partition of the network into communities has been identified, the problem turns on to evaluate how good is the partition. Girvan and Newman proposed a simple approach, based on the intuitive idea of lack of community structure in random networks. Consider an arbitrary partition of a given network into N_c communities. We can define a $N_\mathrm{c} \times N_\mathrm{c}$ size matrix $\mathbf{e}$ where the elements e_{ij} represent the fraction of total links starting at a node in partition i and ending at a node in partition j. Then, the sum of any row of $\mathbf{e}$, $a_i = \sum_j e_{ij}$ corresponds to the fraction of links connected to i.

If there is no community structure in the network the expected value of the fraction of links within partitions can be estimated. It is simply the probability that a link begins at a node in i, a_i, multiplied by the fraction of links that end at a node in i, a_i. Then the expected number of intra-community links is just $a_i a_i$. We also know that the *real* fraction of links exclusively within a partition is e_{ii}. Comparing the two and summing over all the partitions in the graph we get

$$Q = \sum_{i=1}^{c} (e_{ii} - a_i^2)\,. \qquad (1)$$

This is a measure known as *modularity*. As an example, we can consider a network comprised of two disconnected components. If we then have two partitions, corresponding exactly to the two components, modularity will have a value of 1. For particularly “bad” partitions, for example, when all the nodes are in a community of their own, the value of modularity can take negative values.

It is tempting to think that random, Erdos-Renyi networks have little or no community structure. However, as Guimerà *et al.* showed, this in general is not the case. In fact, it is possible to find a partition which not only has a nonzero value of modularity for random networks of finite size, but that this value is quite high. For example a network of 128 nodes and 1024 links has a maximum modularity of 0.208. This suggests that community structure appears in random networks due to fluctuations.

From here on we will look at different methods of community identification presented recently. First we consider methods based on link removal.

4. Link Removal Methods

Divisive methods extract the partition into communities of a network by removing some (or all) of its links until the network is no longer connected or we have a division into communities that meets certain requirements. However, to be able to obtain useful results we need to remove the appropriate links, otherwise the communities will be meaningful. Several methods have been proposed to identify the links that we should remove, which we will revise in this section.

4.1. Shortest Path Centrality

One of the first divisive methods presented in uses the idea of centrality, a measure of how central the node or link is in the network, to decide which links need to be removed. The algorithm uses a particular type of centrality, shortest path centrality, which measures the number of shortest paths between pairs of nodes that pass through a certain node or link. The links with the highest centrality usually act as a bridge between the communities, so if we remove them we can split the network into densely connected communities.

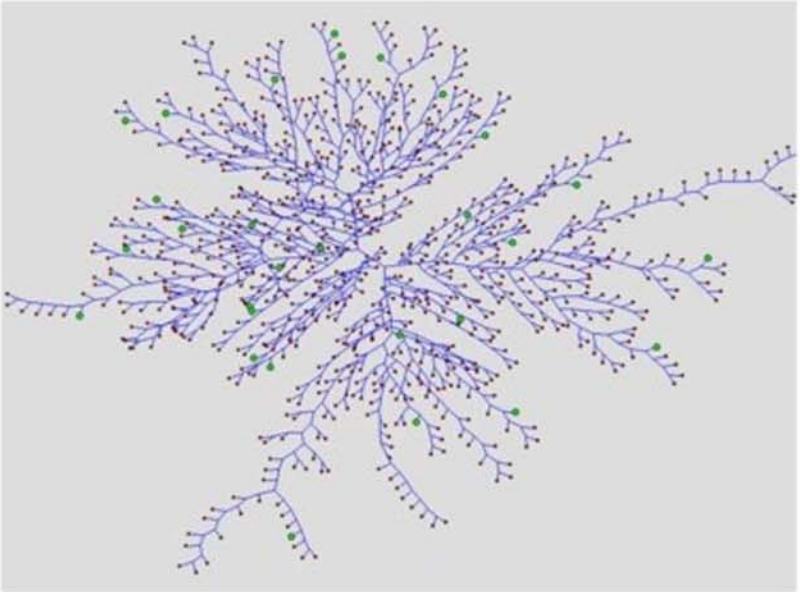

Figure 2. Binary tree showing the result of applying the Girvan-Newman algorithm and our visualization technique to the network of coauthors in FisEs.

The method works recursively eliminating all the links of the network, and stops when there are no more links and all the nodes are isolated. Every time a link is removed, all the centralities are recalculated, otherwise we will obtain an erroneous community detection. This part of the algorithm is the one that requires most computer power and, for a network of size n with m links, using the fastest methods developed independently by Newman and Brandes the speed of calculating all link betweenness-es in one step still remains of $O(m^2n)$ for unweighted networks. This limits the size of the graph that we can process in a reasonable time to a maximum of around 10000 nodes. Figure 2 shows the application of this algorithm to the network depicted in Figure 1.

Each branch corresponds to a real community and the tips of the branches correspond to the people that have played a major role in the different research groups. One can identify here that the members of the scientific committees over the years have indeed played an important role in the development of the community and that they are precisely quite central nodes in the respective local communities.

4.2. Extensions of the Shortest Path Centrality

The same authors of the previous method have also presented two alternative methods to detect community structure by betweenness centrality by calculating this value using two alternative approaches. However, although they are conceptually interesting, both approaches require higher computation than the previous method, and they do not improve the accuracy of it.

The first approach considers the network as a circuit, where links are assigned a unit resistance and we select two nodes that we define as unit voltage source and sink. Using Kirchoff's laws we can calculate the current flow between these two nodes. Adding the flows we will obtain a mesure similar to the centrality, where those links with the lowest resistance (shortest path) carry the most current and, therefore, are the most central. The second approach uses random walks to determine the betweenness centrality of the links. The network is used as a substrate for signals that perform a random walk between pairs of nodes. The link betweenness in this case is simply the rate of flow of random walkers through a particular link summed over all pairs of vertices.

4.3. Information Centrality

Another divisive algorithm available uses the network efficiency measure proposed by Latora and Marchiori. This measure quantifies how efficient is a network in the context of information exchange. If we remove links of the network, its efficiency decreases a certain amount of information centrality.

This method, presented by Fortunato et al., is based on the idea that if we remove the links that act as bridges between communities we should observe the largest drops in network efficiency. from this premise, the method operates similarly to the shortest path centrality method, removing recursively all the links and recalculating the efficiency of all the links at every step. The process is slower than the GN running at $O(n^4)$, but instead the accuracy obtained in the detection is better when the communities to be found are more diffuse.

4.4. Link Clustering

Another approach uses the idea that linked nodes belonging to the same community should have a high clustering coefficient, that is, they share larger number of common neighbors. Based on this idea, the algorithm of Radicchi et al. postulates that the proportion of possible number of loops that go through internal links should be much larger than the proportion of loops for links pointing to outside of the community. The algorithm also works recursively as the previous ones, but in this case by recalculating the *link-clustering coefficient*, which measures the number of loops of a certain length that pass through each link. Longer loops require more computer resources but provide more accurate results.

This algorithm provides a way to stop the detection process when a certain condition is fulfilled, instead of decomposing the whole network until all the nodes are separated. It is also faster than the previous ones, since to compute the *link-clustering coefficient* we only need local information. However, it is not very useful with networks with a very low clustering coefficient, such as trees, sparse graphs or disassortative networks, where we do not have the necessary loops to compute the *link-clustering coefficient*.

5. Agglomerative Methods

Another approach to identify the communities of a network is to start from all the nodes being in separate communities, and some strategy to join or agglomerate them in larger groups. Here we present some of these methods and their grouping algorithms.

5.1. Hierarchical Clustering

Hierarchical clustering has been used traditionally in social networks analysis to extract the communities of the network. The idea of this method is based on the measurement of the similarity between the elements of the nodes according to some property. Starting from an empty network, the method selects those node (or groups of nodes) that have the highest similarity and joins them. This process is again repeated recursively until all the links are added or when we meet a certain condition. The method is very fast and it can work almost in linear time, being able to analyze networks that cannot be processed otherwise. However, the results are highly dependent on the similarity metric that is used to detect the communities.

5.2. L-Shell Method

A second approach focuses on identifying the community around one node of the network by agglomerating its neighbors until a condition is fulfilled. In particular, the algorithm consists on constructing a *L*-shell around one node, where a *L*-shell is a subset of the nodes with a maximum distance of the shortest path to the node origin is less or equal to L. The algorithm starts from the origin and adds more nodes by increasing the distance L until the emerging degree (number of links to nodes outside the *L*-shell) is lower than a cut-off value, and then it is stopped. Those nodes that fall inside the *L*-shell are grouped within one community.

This algorithm is particularly interesting when one is more interested in finding a single community and not in detecting the entire community structure. If we want to make the algorithm global, the authors suggest that we should repeat the process for each node, and then perform a statistical analysis of the results to detect the communities. Since the method uses local information, it is one of the fastest available.

5.3. K-Clique Method

Another approach introduced the idea that communities can overlap. In their definition of community, one node can belong to various "thematic" communities (i.e. one can belong to a scientific group, a family, a sports team, ...), which usually share a certain amount of nodes. The methodology to detect the overlapped communities is based on the concept of '*k*-clique communities'. A *k*-clique is a group of k nodes that is a complete subgraph, and a '*k*-clique community' is the union of all *k*-clique that are adjacent (two *k*-cliques are adjacent if they share k − 1 nodes).

In terms of accuracy, this method is not comparable with the others presented, since it uses a different definition of community structure. However, it has interesting applications, i.e. it can be used to observe the level of relationship between communities or to determine the communities where a certain node belongs.

6. Maximizing Modularity Methods

Since the modularity measure introduced previously provides a good way to evaluate quantitatively a network partition into communities, many authors have presented methods that focus on optimizing this value to obtain the best partition. The benefit of all these methods is that they do not require extra information about the optimal number of communities, since there is a point where the modularity value cannot be improved further. On the contrary, the optimization process is not straightforward because the partition space of any graph (even relatively small ones) is extremely large. The following approaches present different methods to navigate the space of possible partitions to find the highest possible value of modularity, while balancing between accuracy and speed.

6.1. Greedy Algorithm

The first approach introduced by Newman optimizes the value of Q using a greedy algorithm. Starting from a configuration where each node corresponds to one community, the authors compute all the changes in modularity obtained by joining any possible pair of nodes. The highest increment is selected and the two communities are joined, and the process is repeated until a maximum value of Q is obtained.

This method is really fast, since the recalculation of the increments only uses local information, and can analyze a network in almost linear time. However, the accuracy achieved is the lowest of all the modularity optimizing methods.

The benchmark most commonly used to measure the sensitivity of a particular community identification algorithm does not take into account the fact that networks exhibit highly skewed community size distributions, as this shown in Figure 3, and is potentially flawed. By comparing the results of the greedy algorithm to results obtained from a modified version which takes community size into account, the present authors showed that size heterogeneity can alter the comparative accuracy of community detection.

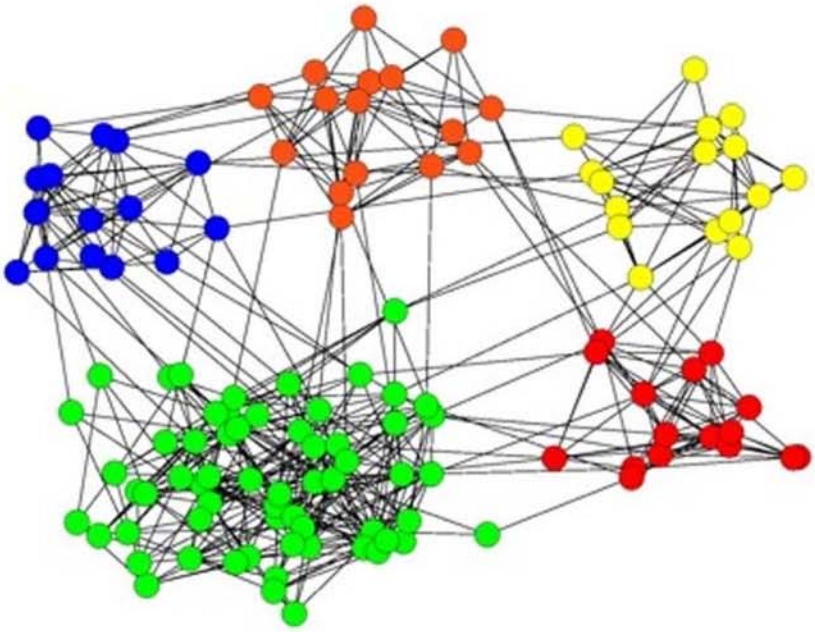

Figure 3. Example of a computer generated network with a heterogeneous distribution of communities where the algorithm proposed in Danon et al 2006 has been applied.

6.2. Extremal Optimization

A third approach, presented by Duch and Arenas, uses a different heuristic search procedure based on extremal optimization to find the best modularity value. The heuristic works at a local scale, by improving the contribution of each node to the global modularity. The nodes are assigned initially to two random partitions, and the local modularity optimization is performed by moving the nodes with the lowest local modularity from one group to another. When the optimization reaches an stationary state where the modularity cannot be improved anymore, the links between the two partitions are removed, and the process is repeated recursively while the total modularity keeps increasing. The algorithm is relatively fast, , scaling as $O\left(n^2\log\left(n\right)\right)$, and it achieves the highest known modularity values for all networks studied.

A modified version of the algorithm has later been introduced by the same authors that adds two new improvements. First, it allows the analysis of weighted and directed networks, using new definitions of modularity. Second, the final results can be fine-tuned using a final bootstrap, which helps correcting small problems that appear due to the recursive process of the divisive part of the algorithm.

6.3. Simulated Annealing Methods

Another approach to optimize the modularity measure is to employ simulated annealing methods. This idea was introduced by Guimerà et al. when they studied modularity in random networks. The method starts with an initial random partition of the nodes into communities, and evolves randomly changing nodes from one community to another. The change is always accepted if the modularity increases, and with a certain probability otherwise. This is also repeated until the modularity cannot be improved anymore for a certain number of steps. The algorithm is slower than some of the other methods, but is one of the most accurate options available.

Later, Massen and Doye proposed two modifications of the simulated annealing approach. First, their algorithm stops periodically, analyzes all the possible node movements, and accepts the move that increases the modularity the most. Second, they use a Basin-Hopping approach, where in each step a group of nodes are moved from one community to another, and this movement is accepted depending on the change of the modularity. The modifications make the process of maximization slower than the original, but are able to find even higher modularity values.

6.4. Information Theoretic Approach

One of the most recent approaches to the community detection problem is based on an information-theoretic framework, where the community detection problem is now treated as an information compression problem. The idea is to reduce the link connectivity of the network (the adjacency matrix) into a more simple description (a module assignment vector and a module matrix). To discover the configuration that provides the best “compression” of the network structure, they maximize the mutual information between the encoded and the global descriptions.

The results presented in their paper show that this method performs better than the others when detecting asymmetric communities. Another advantage is that changing the encoding function we can detect other types of clustering beyond the classical community structure. Similar to the mixture models, the method is also able to identify partitions where the nodes have similar patterns of connection to other nodes.

7. Spectral Analysis Methods

An alternative to the adjacency matrix to represent the information of the connectivity of a graph is the Laplacian matrix. The position n_{ij} of the matrix informs about the existence of a link between i and j, and the diagonal contains the degree of node i, so that the sum of each row and column is equal to zero. The following methods use the algebraic properties of these matrices to identify the nodes that belong to each partition.

7.1. Spectral Bisection

The Laplacian matrix always has an eigenvector with eigenvalue 0, since the sum of elements over each row or column of the Laplacian matrix is equal to $\mathbf{0}$. Also, for each disconnected component of the graph, the Laplacian matrix has a degenerate eigenvector with its corresponding eigenvalue 0. If the components are not completely disconnected (i.e. there are some links between them), the degeneration is no longer present, and we obtain one eigenvalue with value zero and a few eigenvectors with an eigenvalue slightly greater from zero. Therefore, one method to find communities is to find the blocks that give the eigenvalues slightly greater than zero and looking at the components of their eigenvectors.

7.2. Multi Dimensional Spectral Analysis

Another different approach that also uses the properties of the Laplacian matrix was introduced by Donetti and Muñoz. The method consists in extracting the first few non-trivial eigenvectors using the Lanczos method, which is very fast when applied to sparse matrices. They consider the values of the eigenvectors as coordinates in *M*-dimensional space, where M is corresponds to the number of non-trivial eigenvectors extracted. Finally, they measure the distances between the nodes in this space, and cluster them using hierarchical agglomerative methods, obtaining the desired partition into communities. The method is reasonably fast, but the results depend on how many vectors are extracted to separate the communities properly.

7.3. Constrained Optimization

Another method uses the information contained in the spectral properties of the simple adjacency matrix (instead of using the Laplacian as the previous ones). The authors use constrained optimization to extract the eigenvectors much faster, obtaining again a multidimensional space where the eigenvectors contain the coordinates of the nodes. To detect the groups that appear, they use a correlation of the average values of the eigenvectors to measure how close two nodes are in this space. Instead of providing a clear cut community structure, this method gives us an idea of how close any pair of nodes is in the context of communities. The method is able to obtain good results in mid size networks (thousands of nodes).

7.4. Approximate Resistance Networks

Wu et al. presented and extension of the resistance approach method presented before to reduce the time complexity of it. The idea is the same, they select a pair of nodes that act as voltage source and sink, and approximate the voltage of the rest of the nodes. However, instead of using the costly matrix inversion used by Newman, they use an iterative process to approximate the voltage of the other nodes. The accuracy of this approximation is dependent on how many times the iterative step is repeated. It is also dependent on having a good idea of the sizes of communities, which make it difficult to use it in large networks. However, this is one of the few methods that is able to identify the community around one node in linear time.

8. Other Methods

In this last section we include all those methods that do not fit in any of the previous categories.

8.1. Clustering and Curvature

Eckmann and Moses, propose an alternative method based on the concept of curvature of a node. The curvature reflects the average distance between nodes, using the information of the average distance between neighbors of any node (Which is between 1 if they are directly connected and 2 if they do not have any other common neighbor). Since this value is directly related to the clustering, the authors show that finding the connected components that have high curvature gives good insights about the community structure of the graph. The authors have applied successfully this method to study communities in e-mail networks.

8.2. Random Walk Based Methods

Zhou and collaborators have developed different methodologies for community detection based on random walks. Also worthy of note is that the method is applicable to both directed and undirected networks. They also define the concept of 'local' and 'global' community, using the information of attractor nodes (i.e. nodes that are the closest to its neighbors) and a set of rules.

The first approach uses the information contained in the adjacency matrix to determine algebraically the distance between two nodes, so they do not need to actually perform the random walk on the network. This way they obtain faster and more accurate results about which are the attractor nodes, and therefore, about what communities do we obtain.

Later, Zhou and Lipowsky present a different method based on biased random walks. In this modification, each walker has a higher probability to perform jumps from the source node to the node which shares the highest number of neighbors with the source (i.e. biasing the random walker to go down the link with the highest link clustering).

A third different method proposed by Latapy and Pons is based on the idea that a random walker will get trapped for a longer time in a densely connected community. They calculate a distance measure between two nodes, and apply an agglomerative method starting with all nodes in their own community, and joining them two by two.

8.3. Q-Potts Model

Another different approach detects communities by mapping the problem of community detection to the study of a spin system. The authors propose that if the system is in the ground state, communities are identified as groups with equal spin values. To identify the groups, they initialize each node with a random spin state between 1 and q, and determine the energy of the system using a q-Potts Hamiltonian. Then, the system is allowed to evolve using a simple Monte-Carlo method with simulated annealing until it reaches a stationary state.

An interesting feature of this approach is that allows the detection of 'fuzzy' communities, so we can know the level of overlapping between them. The method is reasonably fast, since the calculation of the Hamiltonian only uses local information, and its sensitivity is also good. However, the method needs the input of how many communities we want to find (appropriate values for q are also discussed in the paper).

9. Further Structural Complexity

Most of the methods reported in the previous sections deal with the goal of obtaining the best partition of a network into a set of communities by maximizing the cost function, the modularity. But when looking for the best partition it is implicitly assumed that there is a unique one. It turns out, however, than in some of the networks found there are additional levels of structural complexity, in the sense that there can be communities at different topological levels or there can exist subgroups that belong to more than one community at the same time. We will analyze these two features in the following paragraphs.

9.1. Hierarchical Organization

When dealing with networks that have different scales one can immediately consider what happens with the dynamical evolution of some interaction processes like, for instance, diffusion or synchronization. One expects that interaction at local level develops into a local homogeneity of the variables describing the state of the units at the nodes of the network, and when time goes on this homogenization proceeds into larger and larger scales.

In particular, synchronization of phase oscillators in complex networks has been analyzed in many different articles. In early works, it was shown that high densely interconnected sets (motifs) of Kuramoto oscillators synchronize more easily that those with sparse connections. In this model the phase of each oscillator φ_i evolves according to

$$\frac{d}{dt}\phi_i = \omega_i + \sum_j a_{ij} \sin(\phi_j - \phi_i)$$

where ω_i is the natural frequency of the *i*-th oscillator. In the general case in which frequencies are different the evolution is very sensitive to the particular location and the final state is also dependent on this distribution. However, if all oscillators are identical $\omega_i = \omega_j$ then, except for very regular topologies, the only attractor for the dynamics is the completely synchronized state where all phases are identical. And it is precisely the route to reach this attractor what has been exploited to identify communities at all scales in a series of papers.

Then, for a complex network with a non-trivial connectivity pattern, starting from random initial conditions, those highly interconnected units forming local clusters will synchronize first and then, in a sequential process, larger and larger spatial structures also will do it up to the final state where the whole population should have the same phase. This process occurs in a progressive way at different time scales if a clear community structure exists. Thus, the dynamical route towards the global attractor reveals different topological structures, those which represent communities. Therefore, it is the complete dynamical process what unveils the whole organization at all scales, from the microscale at a very early stage up to the macroscale at the end of the time evolution. On the contrary, those systems endowed with a regular topological structure will display a trivial dynamics with a single time scale for synchronization.

It is a normal practice to define, for the Kuramoto and related models, a global "order parameter" to characterize the level of entrainment between oscillators. However, this definition, although suitable for mean-field models, is not efficient to identify local dynamic effects. In particular it does not give information about the route to the attractor (fully synchronization) in terms of local clusters which is so important to identify functional groups or communities. For this reason, instead of considering a global observable, it is better to define a local order parameter measuring the average of the correlation between pairs of oscillators

$$\rho_{ij}(t) = < \cos(\phi_j - \phi_i) > ,$$

where the brackets stand for the average over initial random phases. The main advantage of this approach is that it allows us to trace the time evolution of pairs of oscillators and therefore to identify compact clusters of synchronized oscillators reminiscent of the existence of communities.

The correlation matrix hence contains all the dynamical information of the route towards the final synchronized state. Any element of this matrix is a monotonously increasing function of time. In any case, due to the continuous nature of the phase the completely synchronized state is never reached and the introduction of some sort of threshold above which two oscillators can be considered as synchronized is necessary. For instance, one can measure the time for two oscillators to get entrained or the time needed by the whole system to get completely synchronized.

Under these assumptions the visualization of the dynamic evolution of the correlation matrix of the system helps in elucidating the topology of the network. To extract the quantitative information it is introduced a threshold T to convert the correlation matrix into a binary matrix, that will be used to determine the borders between different groups. This *dynamic connectivity* matrix is defined as

$$D_t(T) = \begin{cases} 1 & \text{if } \rho_{ij}(t) > T \\ 0 & \text{if } \rho_{ij}(t) < T \end{cases}$$

that depends on both the underlying topology and the collective dynamics. For a fixed time t, by moving the threshold T, we obtain different representations of $D_t(T)$ that inform about the structure of the dynamic correlations. When the threshold is large enough the representation of $D_t(T)$ becomes a set of disconnected clumps or communities. Decreasing T a hierarchical structure of communities is devised. Note that since the function $\rho_{ij}(t)$ is continuous and monotonic (because the existence of a unique attractor of the dynamics), we can redefine $D_T(t)$, i.e. fixing the threshold and evolving in time. We obtain the same information about the structure of the dynamic connectivity matrix at different time scales. These time scales unravel the topological structure of the connectivity matrix at different topological scales To give evidence of the aforementioned facts we have analyzed the dynamics towards synchronization in computer-generated networks with a clear community structure.

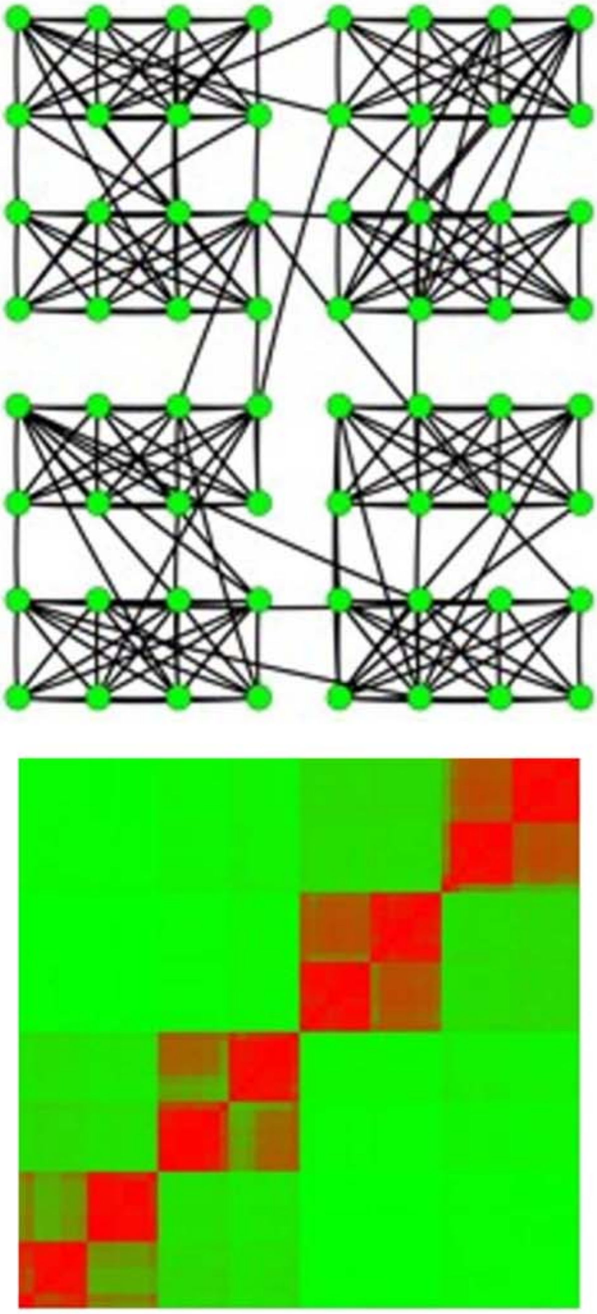

Figure 4. Example of synchronization evolution in a 3-levels hierarchical network. Left: network. Right: time needed for each pair of oscillators of the given network to be synchronized, given a small threshold. Red means shorter synchronization times and green means longer times.

Some of them are homogeneous in degree whereas other networks have special nodes that act as hubs. An example can be seen in Figure 4.

Finally it is important to emphasize the role of the magnitude that has been widely used in the analysis of networks with community structure, the modularity. Given a network structure the best partition into communities is the one that maximizes the modularity. We showed that meta-stable patterns of synchronization in the path towards complete synchronization are closely related to the partitions obtained optimizing modularity on complex networks. In particular, in networks with homogeneous degree distributions the correspondence is very precise and the larger the modularity of a given partition the more stable the synchronized cluster is. However, in networks where there are hubs, the correspondence fails. Hubs take longer to synchronize and hence they tend to be isolated in our synchronization route. This fact makes a big difference in the way synchronization evolves and the way modularity is heuristically optimized. Of course, modularity optimization is not a dynamical process and hence it does not take into account the relative stability of patterns. If two configurations are equivalent, in the sense that the modularity gain is the same at some point in the optimization process then one chooses at random one of the two configurations. This is what happens for instance when considering isolated nodes, since there will never be isolated nodes in an optimal modularity partition. But from a dynamical point of view such a node, or group of nodes, that has to decide in joining one larger group or another can stay longer in this isolation because it is dynamically stable. Finally it will join one larger group or the other but it can be along a slow process in which the merging into the group is followed by a subsequent merging of the full group of nodes.

One of the most attractive aspects of modularity is its non-parametric nature, the absence of parameters to be tuned before the optimization process starts. In contrast, traditional data clustering algorithms (e.g. *k*-means) always depend on initial parameters such as the number of clusters, their minimum size, or the minimum density needed to become a cluster. Although it seems an advantage of the modularity approach, this absence means that it is not possible to find community structures at different scales of description. As discussed in previous paragraphs, in networks with a well-defined hierarchical structure, modularity can only provide insight in just one of the hierarchical levels, without any control of the level to be analyzed. Another consequence is the existence of a resolution limit, pointed out by Fortunato and Barthélemy, beyond which no modular structure can be detected even though modules might have their own entity.

There are several alternatives, within the modularity framework, for the scanning of the whole mesoscale of complex networks, from the macroscale in which all nodes are together in a single community, to the microscale in which each node forms its own isolated module. Reichardt and Bornholdt, following their Potts model approach to modularity, suggested the introduction of a parameter r multiplying the null-case term of modularity. This parameter controls the tradeoff between the tendency of links to join nodes in larger communities, and the trend of the null-case term to avoid the formation of modules. For $r=1$ the standard modularity is recovered. When r is increased, the null-case becomes more important and the found communities are smaller, whereas for values of $r<1$ the attractive term enforces the appearance of larger communities.

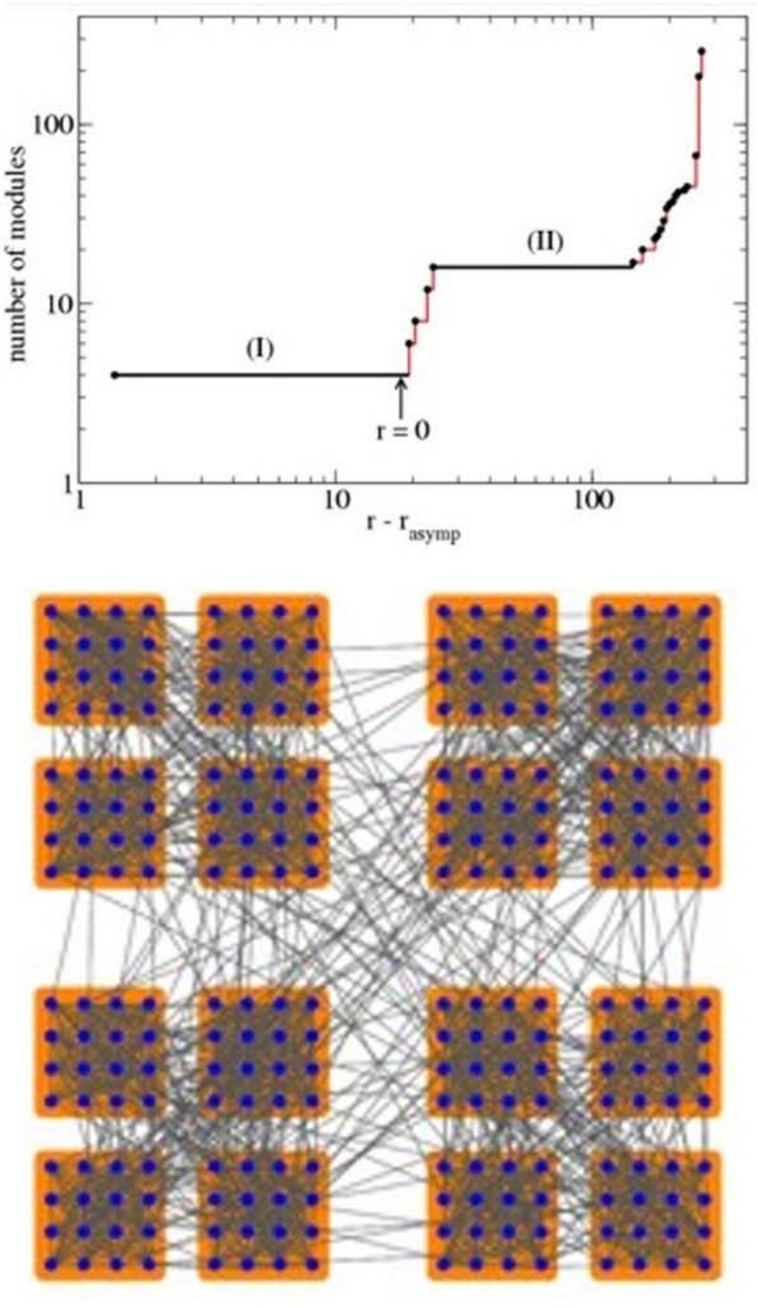

Figure 5. Analysis of the whole mesoscale of a random homogeneous complex network with two hierarchical levels. The plateaus in the plot of the number of modules correspond to the two most important or *stable* partitions of the network.

Arenas et al. avoided the modification of the expression of modularity with the introduction of a new parameter r, which controls the resistance of nodes to form communities. This resistance takes the form of a self-loop of weight r added to each vertex. In this approach only the topology of the network is changed, but in such a way that all the structural properties remain the same, since they only depend on the non-diagonal elements of the adjacency or weights matrices. Scanning the resistance in a well-defined interval and optimizing modularity for each value of r it is possible to go from the macroscale ($r < r_{\min}$) to the microscale ($r > r_{\max}$), including the standard

Newman and Girvan scale at $r = 0$. They also showed that optimal partitions which span across larger intervals of r are the most relevant of the whole mesoscale (see Fig. 5).

A completely different approach is that of Sales-Pardo et al., where it is proposed the analysis of the local maxima of the modularity landscape to find a hierarchical organization of communities. Their main idea is the definition of node affinities from the coexistence of nodes in the same communities of the partitions which are local maxima of Newman and Girvan modularity. These affinities are then used to group nodes in a hierarchical structure.

Finally, a local clustering technique involving a resolution parameter was introduced. In this case modularity is replaced by a local fitness function, which is used to find the community in the neighbourhood of any random vertex of the network. These communities may overlap, and form a hierarchical structure when different values of the resolution parameter are considered.

9.2. Overlap

Another case in which the assignment of some nodes to a single partition has no sense at all is when these nodes can belong simultaneously to two or more groups at the same time (within the same scale, to distinguish from the previously discussed case of hierarchical organization), in such a case we say that communities overlap. Several methods are able to deal with this issue and even some of them have been created with this goal in mind.

One of the methods that have become more popular in the last years is the clique-percolation method, developed by Palla et al. The details of this method are developed previously but here it is worth to say that overlapping communities are in this case found by construction when nodes belong to more than one clique at the same time.

We can also find several methods which have been developed for detecting overlapping communities. The first attempt searched the optimization of a given function and admitting partitions where some nodes can belong to more than one group simultaneously. Later on, a new method based on spectral clustering was developed and nodes are assigned probabilities to belong to any of the prescribed groups. Another similar approach was based on vertex similarity and on maximizing a cost function and on the concept of bridge-ness.

We should also mention a couple of methods which are designed to detect community overlapping but are not based on absolute maximization of some cost function. Some authors have proposed an ensemble of networks, that have overlapping groups by construction, to be used as the right benchmark for comparison of different methods and their ability to detect communities that overlap. The main finding is that in order for the overlapping to be detectable its size must be significantly smaller than the size of the modules involved. Three different community detection methods were used which have been already reported before in this paper: modularity maximization, k-clique percolation and modularity-landscape surveying. Only the last method is able to detect unambiguously the heterogeneities in the group membership of the nodes that form the overlapping region.

Finally, special mention is deserved to methods that incorporate some dynamical properties, as for instance synchronization as being discussed earlier in the current section. It was proposed to identify the topological overlap of communities in complex networks with the interface between synchronized groups of Kuramoto oscillators. The nodes that belong to this interface have clear different dynamical properties from these in the bulk of the large clusters, since their effective frequencies jump from time to time between the effective frequencies (homogeneous in the groups because the internal synchronization) of the non-overlapping groups.

As a final conclusion we can say that identifying different levels of structural organization is not an easy task and probably new methods will be necessary in the future and these methods will probably need to be brand new and not just implementation of existing ones based mainly on the absolute optimization of some magnitude.

10. Applications: Search and Congestion

Community structure is also found to have an effect on the progression of dynamical processes on networks. An example of particular importance can be seen in information transfer or communication processes. In many studies it has been shown that congestion occurs when the demand for transfer (packet injection rate) becomes high, greatly reducing the efficiency of information flow.

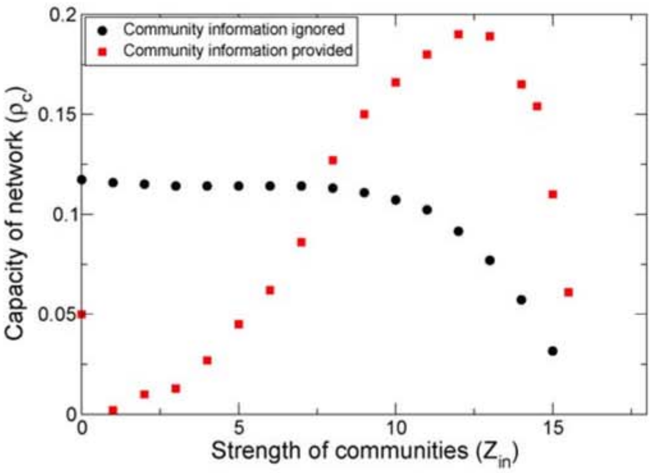

Figure 6. The impact of community structure on the carrying capacity of networks.

A pronounced community structure is detrimental to the efficient flow of information if information routing is oblivious to the presence of communities. Once information on community structure is used to define targeted routing strategies, this efficiency can be greatly improved as is shown in Figure 6.

However, the accuracy at which community structure information is provided turns out to play an important role. In a hierarchical network setting, providing knowledge of communities at the level of highest modularity will improve the transfer capacity of the network by the largest amount. This finding suggests that designing routing protocols which utilize the best community information, as found by the most accurate detection algorithms would have the greatest impact on the efficiency of communication networks.

For a stylized model of search and congestion, on networks with ad-hoc community structure, we consider two routing strategies. Black circles denote a routing strategy that ignores the communities, and red squares denote the alternative where community information is utilized.

11. Conclusions

In this work we have attempted to give an overview of the modern approaches to community identification in complex networks. A large amount of knowledge has been collected in the field, and real progress has been made, both in the identification of communities and their characterization. Some questions do remain open, and it is these that we would suggest for further study. Despite these efforts, computational cost involved in computing communities in complex network remain significant. At present, the fastest method for finding an unknown number of communities of unknown sizes has a cost which scales as $O\left(n\log^2 n\right)$ with network size. While this makes the analysis of extremely large networks feasible this algorithm does not guarantee that the partition found will be the best possible one. Other algorithms which give better partitions are more expensive. The challenge, then, is to come up with a method which is both fast and accurate.

Another major challenge is to understand the mechanisms which are responsible for the characteristic scale free distributions of community sizes observed. Such distributions often suggest an underlying optimization is responsible, but this remains to be shown.

Acknowledgements

The authors are grateful to Luca Donetti, Haijun Zhou, Mark Newman, Santo Fortunato, Jörg Reichardt, Claudio Castellano, Matthieu Latapy and Jean-Pierre Eckmann for providing their data.

Glossary

Community: From a network perspective, a community is a group of nodes more densely linked that with the rest of the network.

Modularity: Mathematical function that quantifies the community structure, depends on the network and on the proposed partition into groups, clusters, modules, or communities.

Centrality: Set of properties of a node that account for its role in the global network. Several measures are degree (local centrality), betweenness, closeness

Shortest path Number of steps (links or edges) to reach one node from another one

distance: following the shortest path.

Network spectrum: Set of eigenvalues and eigenvectors of the Laplacian matrix of the network (related by a simple transformation to the adjacency matrix) .

Greedy: A greedy algorithm is any_algorithm that solves the problem by making the locally optimal choice at each stage with the hope of finding the global optimum.

Bibliography

A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, C. Zhou (2008) Synchronization in complex networks. *Physics Reports* 469:93. [Review paper on synchronization in complex networks]

A. Arenas, A. Dáz-Guilera, and C. J. Pérez-Vicente. (2006) Synchronization Reveals Topological Scales in Complex Networks. *Phys. Rev. Lett.*, 96:114102 [First attempt in detecting hierarchical structures in complex networks based on dynamical methods]

A. Arenas, A. Fernández, and S. Gómez. (2008) Analysis of the structure of complex networks at different resolution levels. *New J. Phys.*, 10:053039+. [Multi-resolution at multiple scales]

Albert Laszlo Barabási and Reka Albert. (2002) Statistical mechanics of complex networks. *Review of Modern Physics*, 74:47–97. [The first review on complex networks].

Alex Arenas, Leon Danon, Albert Diaz-Guilera, Pablo M Gleiser, and Roger Guimerà. (2004) Community analysis in social networks. *European Physical Journal B*, 38:373–380 [Community analysis in some examples of social networks]

C. Bron and J. Kerbosch. (1973) Finding all cliques in an undirected graph. *Communications of the ACM*, pages 575–577 [A paper on the determination of communities in a simple case].

E. N. Sawardecker, M. Sales-Pardo, and Amaral. (2009) Detection of node group membership in networks with group overlap. *The European Physical Journal B - Condensed Matter and Complex Systems*, 67:227 [Overlapping communities]

E. Oh, K. Rho, H. Hong, and B. Kahng. (2005) Modular synchronization in complex networks. *Phys. Rev. E*, 72:047101. [Dynamical properties of structured networks]

Erzsébet Ravasz and Albert-László Barabási. (2003) Hierarchical organization in complex networks. *Phys. Rev. E*, 67:026112. [Hierarchical networks]

F. Wu and B.A. Huberman. (2004) Finding communities in linear time: a physics approach. *Eurpean Physics Journal B*, 38:331–338. [Method based on dynamical properties]

Filippo Radicchi, Claudio Castellano, Frederico Cecconi, Vittorio Loreto, and Domenico Parisi. (2004) Defining and identifying communities in networks. *Publications of the National Academy of Sciences*, 101(9):2658–2663 [Weak and strong definitions of community]

John Hopcroft, Omar Khan, Brian Kulis, and Bart Selman. (2004) Tracking evolving communities in large networks. *Publications of the National Academy of Sciences USA*, 101(Suppl. 1):5249–5253. [Dynamic evolution of communities]

Jordi Duch and Alex Arenas, (2005) "Community detection in complex networks using extremal optimization", Phys. Rev. E **72**, 027104. [Method based on extremal optimization]

JReichardt and Stefan Bornholdt. (2004) Detecting fuzzy community structure in complex networks with a q-state potts model. *Phys. Rev. Lett.*, 93:218701. [Method based on spin models]

L. da F. Costa, F. A. Rodrigues, G. Travieso, and P. R. Villas Boas. (2007) Characterization of complex networks: A survey of measurements. *Adv. Phys.*, 56:167–242. [A proposal for the characterization of networks].

Leon Danon, Albert Díaz-Guilera, and Alex Arenas. (2006) The effect of size heterogeneity on community identification in complex networks. *J. Stat. Mech.*, 2006:P11010+ [Newman’s fast algorithm adapted to inhomogeneous networks]

Leon Danon, Alex Arenas, and Albert D. Guilera. (2008) Impact of community structure on information transfer. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 77. [Effect of the community structure on the dynamics of information transfer]

Luca Donetti and Miguel A. Muñoz. (2004) Detecting network communities: a new systematic and efficient algorithm. *Journal of Statistical Mechanics: Theory and Experiment*, (P10012). [Method based on spectral properties]

Mark E. J. Newman. (2003) The structure and function of complex networks. *SIAM Review*, 45:167–256. [A more recent review on complex networks].

Mark E. J. Newman. (2004) Detecting community structure in networks. *European Physics Journal B*, 38:321–330. [Early review on community detection in complex networks]

Mark E. J. Newman and Michelle Girvan. (2004) Finding and evaluating community structure in networks. *Physical Review E*, 69:026113. [Computation of communities through betweenness].

Mark E. J. Newman. (2001) Scientific collaboration networks. i. network construction and fundamental results. *Physical Review E*, 64:016132. [A paper on the properties of scientific networks].

Mark E. J. Newman. (2004) Fast algorithm for detecting community structure in networks. *Physical Review E*, 69(066133). [Newman's fast algorithm]

Marta Sales-Pardo, Roger Guimera, Andre A. Moreira, and Luis A. Amaral. (2007) Extracting the hierarchical organization of complex systems. *Proceedings of the National Academy of Sciences*, 104:15224–15229 [Hierarchical detection of communities]

Matthieu Latapy and Pascal Pons. (2004) Computing communities in large networks using random walks. *cond-mat/0412568*. [Method based on random walks properties]

Michelle Girvan and Mark E.J. Newman. (2002) Community structure in social and biological networks. *Publications of the National Academy of Sciences USA*, 99(12):7821–7826 [Definition of modularity]

R. Pastor-Satorras, M. Rubi, and A. Díaz-Guilera, (eds.) (2003) *Proceedings of the Conference "Statistical Mechanics of Complex Networks"*. Springer. [A series of contributions on complex networks].

Roger Guimera, Leon Danon, Albert Diaz-Guilera, Francesc Giralt, and Alex Arenas. (2003) Self-similar community structure in a network of human interactions. *Physical Review E*, 68(065103) [Community analysis of the email network of Universitat Rovira I Virgili]

Roger Guimerà, Marta Sales, and Luìs N. A. Amaral. (2004) Modularity from fluctuations in random graphs and complex networks. *Physical Review E*, 70:025101. [Computation of modularity in a null case].

S. Bornholdt and H. G. Schuster, (eds.) (2002) *Handbook of Graphs and Networks - From the Genome to the Internet*. Wiley-VCH, Berlin. [First edited book on complex networks]

Santo Fortunato and Marc Barthelemy. (2007) Resolution limit in community detection. *PNAS*, 104:36–41. [Discussion about the resolution limit in communities' identification]

Santo Fortunato, Vito Latora, and Massimo Marchiori. (2004) Method to find community structures based on information centrality. *Physical Review E*, 70(056104). [A betweenness related method].

Sergei N. Dorogovtsev and J. F. F. Mendes. (2003) *Evolution of Networks: From biological nets to the internet and WWW*. Oxford University Press, Oxford. [A textbook on complex networks with mathematical approach].

Stefan Boettcher and Allon G. Percus. (2001) Extremal optimization for graph partitioning. *Physical Review E*, 64 [Application of extremal optimization]

Stefan Boettcher and Allon G. Percus. (2001) Optimization with extremal dynamics. *Physical Review Letters*, 86(23):5211–5214 [Basic reference for extremal optimization methods]

Steven H. Strogatz. (2001) Exploring complex networks. *Nature*, 410:268–276. [a review on small-world effect and dynamical properties of complex networks].

Ulrik Brandes. (2001) A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(2):163–177, [The fastest method to compute betweenness].

Biographical Sketches

Alex Arenas: is associate professor at the University Rovira i Virgili. He doctorates in physics in 1996 at the University of Barcelona. His research covers aspects of statistical physics and computer science. He has published more than 80 papers and participated in 24 research projects.

Leon Danon: is a Research Fellow at the Harvard School of Public and the University of Warwick. He has worked on problems such as the nature of earthquake patterns, community detection in complex network and epidemic dynamics in structured populations (both human and animal). He has extensive experience in the analysis of large complex datasets and development of stochastic models of infectious diseases at multiple scales.

Albert Diaz-Guilera: Associate professor in Condensed Matter Physics at Universitat de Barcelona. PhD in Physics in 1987 from Universitat Autonoma de Barcelona. Expert in Statistical Physics, in the last years he has been specialized in the analysis of complex networks, mainly their dynamical properties.

Jordi Duch: He studied Computer Science at University Rovira Virgili. PhD in the Department of Physics, at Universitat de Barcelona under the supervision of Dr. Alex Arenas. Currently postdoctoral fellow in Luis Amaral's Group, in the Department of Chemical and Biological Engineering of Northwestern University.

Sergio Gómez: is associate professor at Universitat Rovira i Virgili, Tarragona (Spain). He has degrees in physics (1990) and mathematics (1995), and PhD in physics (1994) at Universitat de Barcelona. His research is concentrated in two main fields, artificial neural networks and complex networks, and covers both theory and application to real world phenomena.