Stability of maximum-likelihood-based clustering methods: exploring the backbone of classifications

Muhittin Mungan[1,2] **and José J Ramasco**[3]

[1] Department of Physics, Boğaziçi University, 34342 Bebek, Istanbul, Turkey
[2] The Feza Gürsey Institute, P.O.B. 6, Çengelköy, 34680 Istanbul, Turkey
[3] Complex Networks Lagrange Laboratory (CNLL), ISI Foundation,
I-10133 Turin, Italy
E-mail: mmungan@boun.edu.tr and jramasco@isi.it

Abstract. Components of complex systems are often classified according to the way they interact with each other. In graph theory such groups are known as clusters or communities. Many different techniques have been recently proposed to detect them, some of which involve inference methods using either Bayesian or maximum likelihood approaches. In this paper, we study a statistical model designed for detecting clusters based on connection similarity. The basic assumption of the model is that the graph was generated by a certain grouping of the nodes and an expectation maximization algorithm is employed to infer that grouping. We show that the method admits further development to yield a stability analysis of the groupings that quantifies the extent to which each node influences its neighbors' group membership. Our approach naturally allows for the identification of the key elements responsible for the grouping and their resilience to changes in the network. Given the generality of the assumptions underlying the statistical model, such nodes are likely to play special roles in the original system. We illustrate this point by analyzing several empirical networks for which further information about the properties of the nodes is available. The search and identification of stabilizing nodes constitutes thus a novel technique to characterize the relevance of nodes in complex networks.

Keywords: random graphs, networks

J. Stat. Mech. (2010) P04028

Contents

1. Introduction

Networks are useful tools to characterize complex systems [1]–[8]. The system components are represented as nodes and their mutual interactions as edges. Finding structures in such networks is therefore of great relevance for understanding the mechanisms that underlie the system evolution. This explains the increasing interest in the topic, particularly in the detection of communities [1, 2, 4], [9]–[16]. Communities are groups of nodes with a high level of group interconnection [4]. They can be seen as relative isolated subgraphs with few contacts with the rest of the network. Communities have an obvious significance for social networks where they correspond to groups of close friends or well-established teams of collaborators [6]. However, they are also important for characterizing other real-world networks such as those coming from biology [3, 4, 7] or from technology and transport [17]–[19]. Communities are not the only meaningful structures in networks: in Ecology, Computer and Social Sciences structurally equivalent nodes have also been considered [5, 6, 8]. These nodes are characterized by similar connectivity patterns and are expected to play similar roles within the system.

There has been a long tradition of applying Bayesian and maximum likelihood methods to structure detection in networks [20]–[28]. These methods have the advantage that, depending on the statistical model used, they can be very general, detecting both communities and structural equivalent sets of nodes. The drawback, shared with many other methods, is that structure detection usually implies computationally expensive exploration of the solutions maximizing the posterior probability or the likelihood. Recently, a maximum likelihood method that considers node clustering as missing information and deals with it using an expectation maximization (EM) approach has

been introduced by Newman and Leicht [1, 2]. This method is computationally less costly to implement and we will denote it by the acronym NL-EM from now on. NL-EM is able to identify network structure relying on three basic assumptions: (i) the actual connectivity of the network is related to a coherent yet *a priori* unknown grouping of the nodes, (ii) the presence or absence of a link is independent from the other links of the network and (iii) the groups are tell-tales of processes that gave rise to the graph. No extra information is assumed except for the network itself and the number of groups. Under these assumptions, the method infers the classification of nodes that most likely generated the graph detecting communities and also structurally equivalent sets of nodes [2]. Here we will show that, due to the simple structure of the NL-EM likelihood, its classifications are based on a subset of nodes which turn out to be responsible for establishing the group memberships of their neighbors. We are able to rank the nodes according to the amount of group-allocation information they transmit to their neighbors and thereby identify those that are essential for establishing each group. These nodes, which we will refer to as stabilizers, constitute the backbone of the classification: the classification would not be viable without them and, conversely, stabilizers turn out to emerge as a result of their distinct connection patterns on the given graph. Given the generality of the NL-EM underlying assumptions and that the resulting classifications can be validated by comparison with other clustering methods, we suggest that the stabilizers have an important inherent value for understanding the processes that generated the given network. Such an expectation is supported by our results on empirical graphs for which additional information regarding the nodes' intrinsic properties is available. We will also briefly discuss the extension of this concept to other inference methods such as Bayesian clustering techniques [22, 24].

2. NL-EM clustering method

We begin with a quick summary of NL-EM as applied to graphs. Labeling the nodes as $i = 1, \ldots, N$, the variables are: π_r, the probability that a randomly selected node is in group r, θ_{rj}, the probability that an edge leaving group r connects to node j, and q_{ir}, the probability that node i belongs to group r. The method is a mixture model where an edge between nodes i and j (expressed as $i \leftrightarrow j$) given the groups of i and j (g_i and g_j) is observed with probability

$$\Pr(i \leftrightarrow j | g_i, g_j) = \theta_{g_i j} \theta_{g_j i}. \tag{1}$$

The edges are considered as independent so the probability that a given grouping realizes an observed network $\mathcal{G}$ can be written as

$$\Pr(\mathcal{G} | \theta, \pi, \{g_i\}) = \prod_i \pi_{g_i} \left[\prod_{j \in \nu_i} \theta_{g_i j} \right], \tag{2}$$

where ν_i is the set formed by the neighbors of node i.

The group assignment captured by the terms q_{ir} is treated as missing information. The expectation step of EM can thus be implemented as an average over the log-likelihood

$$\bar{\mathcal{L}}(\pi, \theta) = \sum_{ir} q_{ir} \left[\ln \pi_r + \sum_{j \in \nu_i} \ln \theta_{rj} \right]. \tag{3}$$

The maximization of $\bar{\mathcal{L}}(\pi, \theta)$ is subject to the normalization constraints

$$\sum_j \theta_{rj} = 1 \qquad \text{and} \qquad \sum_r \pi_r = 1, \tag{4}$$

and leads to

$$\begin{aligned} \theta_{ri} &= \frac{\sum_{j \in \nu_i} q_{jr}}{\sum_{j \in \nu_i} k_j q_{jr}} \\ \pi_r &= \frac{1}{N} \sum_i q_{ir}, \end{aligned} \tag{5}$$

where k_j is the degree of node j. The group assignment probabilities q are determined *a posteriori* from

$$q_{ir} = \frac{\Pr(\mathcal{G}, g_i = r | \theta, \pi)}{\Pr(\mathcal{G} | \theta, \pi)}, \tag{6}$$

as

$$q_{ir} = \frac{\pi_r \prod_{j \in \nu_i} \theta_{rj}}{\sum_s \pi_s \prod_{j \in \nu_i} \theta_{sj}}. \tag{7}$$

The maximization of $\bar{\mathcal{L}}$ can be carried out with different techniques. In order to account for the possible existence of a rough likelihood landscape with many local extrema, we employed an algorithm that alternates between simulated annealing and direct greedy iteration of equations (5) and (7).

3. Stability analysis and stabilizers

The group membership of the nodes is encoded by the probabilities q. It is thus natural to ask for the conditions on a node i and its neighbors to have i crisply classified into a single group r so that $q_{is} = \delta_{rs}$. The answer highlights the role of the neighbors in establishing a node's membership credentials. Looking at the expression for q_{ir}, equation (7),

$$q_{ir} \sim \prod_{j \in \nu_i} \theta_{rj}, \tag{8}$$

where the non-zero prefactors whose sole role is to ensure proper normalization have been suppressed, one finds that for each group $s \neq r$ there must be at least one neighbor j of i whose probability θ_{sj} is zero. However, as seen from equation (5), whether θ_{sj} is zero or not for some group s depends in turn on the group memberships of the neighbors of j. Hence, having a node crisply classified as belonging to a group sets strong constraints on its neighbors and their respective neighborhoods. These constraints propagate throughout the network during the NL-EM iteration until a final configuration for θ and q is established. In this sense, a node j is passing information about group membership to its neighborhood through the probabilities θ_{sj}. This information is negative, of the form 'you do not belong to group X' when θ_{Xj} is zero and we say that node j stabilizes its neighbors against membership in group X. It is worth noting the parallels of this mechanism with message-passing algorithms [29]. In a classification into $\mathcal{N}_C$ groups each crisply classified node i must be stabilized against $\mathcal{N}_C - 1$ groups. Thus one can regard the number of

4

groups a node j stabilizes against as a measure of the amount of information I_j that j passes to its neighbors. If $I_j = \mathcal{N}_C - 1$, node j can stabilize its adjacent nodes alone, thus providing complete information about their group membership. On the other hand, when $I_j < \mathcal{N}_C - 1$, j provides only partial information. The crisp classification of a neighbor i requires then the combined action of other adjacent nodes in order to attain full group membership information. We denote as *stabilizers* of i the union set of neighbors that, alone or in combined action, pass essential information to i, establishing its membership in a single group (a more precise definition will be given below). The above analysis implies that any crisply classified node must be stabilized by one or more stabilizers. Therefore, if the assumptions of the statistical model are justified and the resulting node classification is meaningful, the identification of the corresponding stabilizers may offer useful additional information.

Based on their classification and the information passed, four types of nodes can be distinguished: nodes can be strong or weak depending on whether they are crisply classified into a single group or not, and they can be stabilizers or not, depending on whether they pass essential information for establishing an adjacent node's group membership. If we consider a node i and denote by $\bar{\sigma}_i = \{r|\theta_{ri} = 0\}$ the set of groups that i does not connect to, and by $\bar{c}_i = \{r|q_{ir} = 0\}$ the set of groups that i does not belong to, the NL-EM equations (5) and (7) relate these sets as follows:

$$\bigcup_{j \in \nu_i} \bar{\sigma}_j = \bar{c}_i \qquad \text{and} \qquad \bigcap_{j \in \nu_i} \bar{c}_j = \bar{\sigma}_i, \tag{9}$$

forming a set of consistency relations with a simple meaning: a node cannot belong to a group to which its neighbors do not connect and the common set of groups to which a node's neighbors do not belong must correspond to the groups that it does not connect to. If we require in particular that a node i is strong, i.e. it is crisply classified as belonging to a particular group A, then $\bar{c}_i = \mathcal{C} \setminus \{A\}$[4].

Given the sets $\bar{\sigma}_j$ associated with the neighbors j of a strong node i, not all adjacent nodes need to contribute to its full stabilization. Likewise, node i can be stabilized by different combinations of its neighbors' sets $\bar{\sigma}_j$. This is best illustrated by an example shown in figure 1. Suppose that the groups are $\mathcal{C} = \{A, B, C, D, E\}$ and let us assume that node i is crisply classified as E. Let i have four neighbors with corresponding sets $\bar{\sigma}_1 = \{A, B, C\}$, $\bar{\sigma}_2 = \{A, D\}$, $\bar{\sigma}_3 = \{B, C\}$ and $\bar{\sigma}_4 = \{A\}$. It is clear that all four nodes together must stabilize i, as otherwise i would not be a strong node. However, the sets of neighbors $\{1, 2\}$ or $\{2, 3\}$ each suffice to stabilize node i. Node 4 is redundant, since it does not contribute a new class against which 2 or 3 are not already stabilizing. In other words, if the set $\{2, 3, 4\}$ is considered, node 4 can be removed without altering the stabilization of i. The same is not true for nodes 2 and 3. The notion of stabilization sets and stabilizer nodes can be defined as follows: a subset of nodes adjacent to i is a stabilization set of i, if the removal of any one of the nodes from the set causes i not to be

---

[4] The question of whether a given graph admits an NL-EM solution that is a (strong) partition of all of its nodes into $\mathcal{C}$ groups can thus be seen as a coloring problem: we seek a partition g of the nodes, i.e. a coloring g_i, corresponding to $c_i = \{g_i\}$, such that

$$\bigcup_{j \in \nu_i} \bigcap_{\ell \in \nu_j} \bar{c}_\ell = \bar{c}_i,$$

is satisfied for all i.
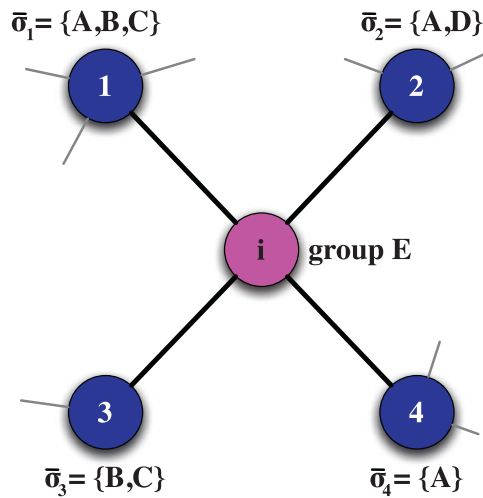
Figure 1. Stabilization of a strong node i. The groups are $\mathcal{C} = \{A, B, C, D, E\}$ and i is crisply classified as E. The four adjacent nodes are shown along with the set of classes $\bar{\sigma}$ to which they have no connections to. In order for i to be classified as E there must be a subset of adjacent nodes such that the union of their corresponding $\bar{\sigma}$ is $\mathcal{C} - \{E\}$, equation (9). All four adjacent nodes must stabilize i, as otherwise i would not be a strong node. However, the sets $\{1, 2\}$ or $\{2, 3\}$ each suffice to stabilize node i. Node 4 is redundant, in the sense that any stabilization of i involving node 4 remains a stabilization of i when 4 is removed. A precise definition of stabilizers is given in the text.

stabilized by that set anymore. A node j is a stabilizer if it is a member of at least one stabilization set. The definition of stabilizer involves thus a stabilization relation with at least one of the node neighbors. In the above example, $1, 2$ and 3 are the only stabilizers of i. Non-stabilizer nodes can be removed without affecting stabilization, while whenever a stabilizer is removed the number of ways in which a given node is stabilized decreases. In the example of figure 1, the removal of node 2 would cause complete loss of stabilization of i, while removal of 3 or 1 would leave i with only a single stabilization. It can be shown that the removal of a stabilizer will never turn a previously non-stabilizer node into a stabilizer, but it might turn some stabilizers into non-stabilizers. Note that, in a sense, stabilizer 2 is more important than 1 or 3, since it is part of every stabilization of i and its removal will thus render i a weak node. In fact, one could attach a strength to each stabilizer by keeping track of the number of stabilizations in which it is involved, but, for the sake of simplicity, we will not pursue this here.

Given an NL-EM classification with strong nodes, we can immediately identify the stabilizers that are responsible for the crisp classifications. Details on how to implement the identification of stabilizers are provided in appendix A.2. The relation i stabilizes j induces a directed subgraph on the original network and we will refer to this as the stabilizer subgraph. The relation between two stabilizer nodes is not necessarily of mutual stabilization: a necessary condition for adjacent strong nodes i and j to mutually stabilize each other is that both $\bar{\sigma}_i \cap c_j$ and $\bar{\sigma}_j \cap c_i$ are empty. The connections among strong stabilizers capture the relations between groups in the graph. In that sense one can regard the stabilizers as exemplary members of the groups. In the undirected graphs of
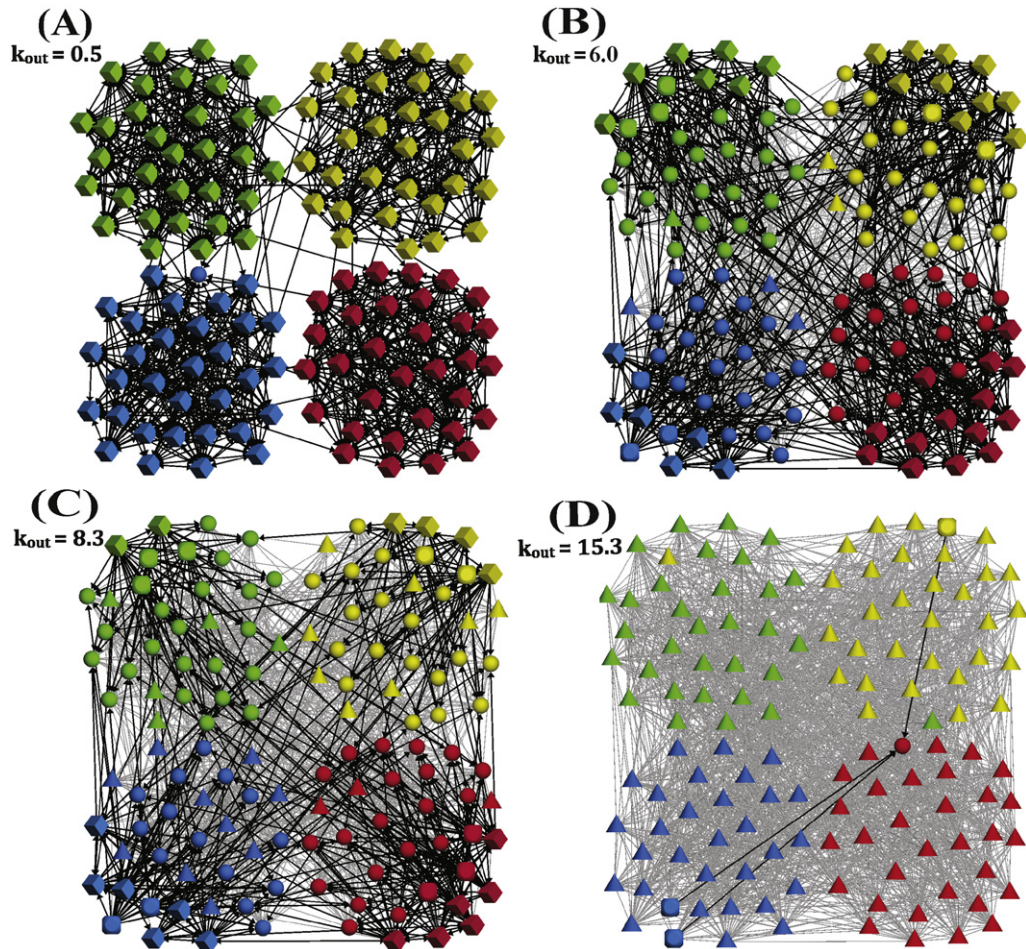
Figure 2. Benchmark with four communities and increasing intra-community connections, k_{out}. (A)–(D): four instances of the graph classification—strong stabilizers, weak stabilizers, strong nodes and weak nodes—are shown as rhomboids, cubes, spheres and cones, respectively. Nodes have been arranged by communities while their color depicts the group to which the NL-EM algorithm assigns them with highest probability. The directed (dark) arcs show the information flow, as captured by the stabilization relation.

figures 2–5 the stabilizer subgraph has been superposed. The extension of these concepts to NL-EM classifications in directed graphs is similar; details are given in appendix B.

The case of NL-EM classifications into two groups is particularly simple. Denoting the groups as A and $\bar{A}$, a crisply classified (strong) node belongs to either A or $\bar{A}$ and a strong node of a given group has to be stabilized against the complementary group. All nodes with non-empty $\bar{\sigma}$ are therefore stabilizers, and if more than one is present all are equivalent, each stabilizing a given node independently from the other stabilizers. Moreover, the strong stabilizers are nodes that are stabilized themselves by some of their neighbors which necessarily are also stabilizers. The conditions of equation (9) permit only two possible configurations of the stabilizer subgraphs. Either strong stabilizers of group A connect to strong stabilizers of their own group or stabilizers of group A connect
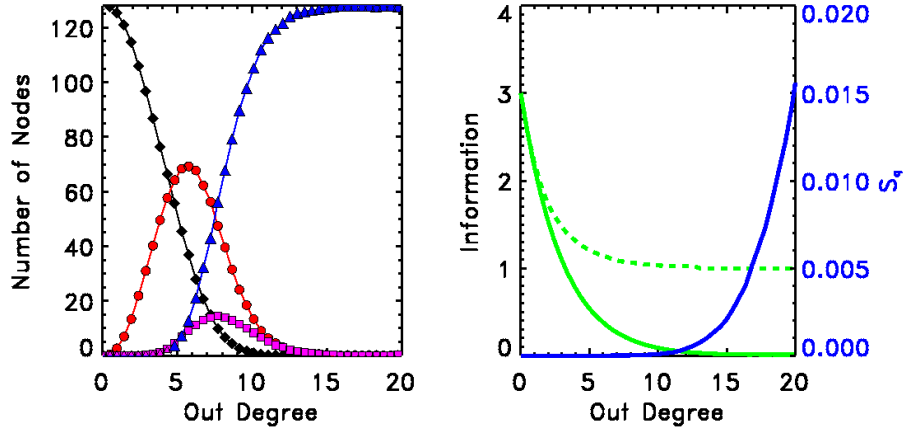
Figure 3. Benchmark with four communities and increasing intra-community connections continued. (Left) Number of nodes versus disorder: strong (diamonds) and weak (boxes) stabilizers, strongly (circles) and weakly classified nodes (triangles). (Right) Average information $\langle I \rangle$ passed by all nodes (continuous green curve) or stabilizers only (green dashed curve, for which $I \geq 1$ by definition). The blue curve shows the entropy of the classification, as defined in equation (10). The values for each data point in the plots have been obtained from averaging over 100 realizations of the random process of edge additions.

to those of the complementary group $\bar{A}$. In the former case we get a disjoint community-like partition (cf figure 4) of the stabilizer graph, whereas in the latter case we obtain a bipartite partition (cf figure 5).

Furthermore, the NL-EM classification into two groups reveals a simple but meaningful hierarchical structure in the way the different types of nodes in the classification relate. Strong (non-stabilizer) nodes are nodes for which $\bar{\sigma} = \emptyset$, so these nodes connect to nodes of both groups (weak or strong). However, in order for them to be strongly classified as in one group, let us say A ($\bar{A}$), they can only connect to those stabilizer nodes with the compatible stabilizer classes $\bar{\sigma} = \{\bar{A}\}$ ($\bar{\sigma} = \{A\}$). In turn, the neighborhood of strong stabilizer nodes with $\bar{\sigma} = \{A\}$ or $\bar{\sigma} = \{\bar{A}\}$ can consist only of nodes strongly classified as $\bar{A}$ or A, respectively. The weak stabilizer nodes are by definition nodes for which $\bar{c} = \emptyset$, but for which $\bar{\sigma} = \{A\}$ or $\bar{\sigma} = \{\bar{A}\}$. Thus weak stabilizer nodes cannot connect to strong stabilizer nodes, but they can stabilize strong (non-stabilizer) nodes. Finally, the weak nodes that are neither strong nor stabilizing can connect to strong non-stabilizing nodes and other weak nodes. In this way the connection rules for the strong stabilizers, weak stabilizers, strong nodes and weak nodes set up a hierarchy of nodes at the core of which are the strong stabilizers.

4. Stabilizers in a benchmark

As we observed in section 3, a node can be stabilized by its neighbors in multiple ways. This redundancy renders classifications robust against disorder introduced by the addition or removal of edges up to a certain point. To illustrate this we consider a benchmark with four communities [11]. The initial network is generated with four disjoint groups
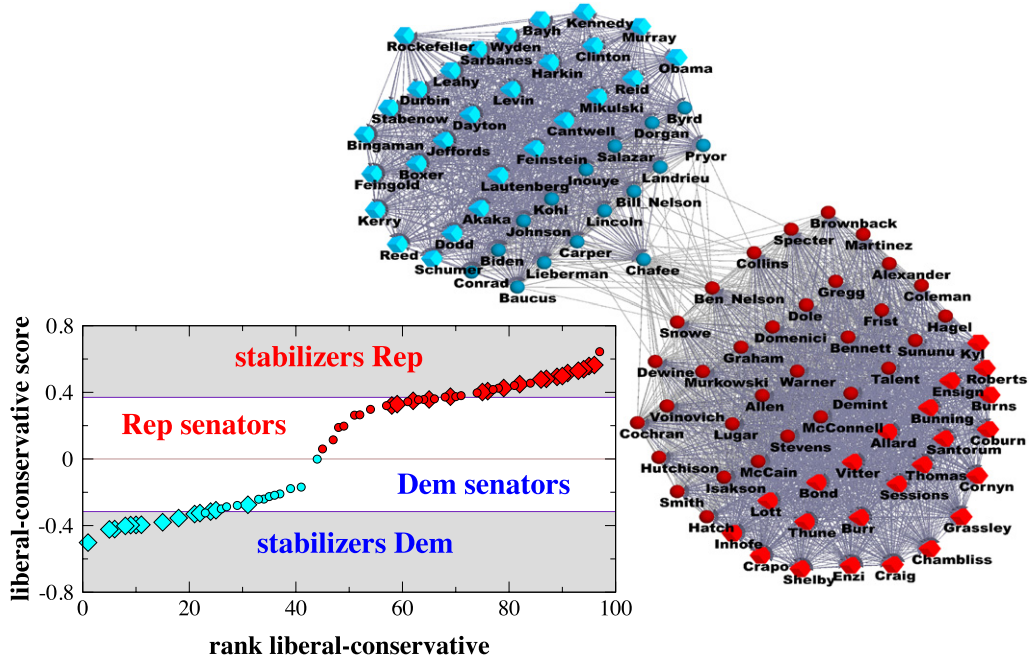
Figure 4. Political affinity network between congressmen of the 109th US Senate. Right: the node shapes are as in figure 1: rhomboids are strong stabilizers and spheres. well-classified nodes not passing essential group information. The colors convey the NL-EM classification which follows the partition into Democrat (blue) and Republican (red) fields. Left: the senators are displayed as ranked according to their liberal–conservative score [32]. The average values of the score in the different sub-groups are: Rep. stabilizers 0.45±0.08, Rep. strong nodes 0.33±0.12, Dem. strong nodes -0.22 ± 0.08 and Dem. stabilizers -0.37 ± 0.06.

of 32 nodes each, with the nodes having on average $\langle k_{in} \rangle = 16$ in-group links. These groups correspond to the four clusters of figures 2(A)–(D). Random links connecting different groups are added to the basic configuration and the number of stabilizers are tracked as a function of the average number of out-group links k_{out}. Figure 2 shows the stabilizers obtained from an NL-EM classification into $\mathcal{N}_C = 4$ groups at disorder level $k_{out} = 0.5, 6.0, 8.3$ and 15.3.

When $k_{out} = 0$ we find a crisp classification where all nodes are strong stabilizers, meaning that all nodes stabilize and are being stabilized. Furthermore, all of them provide complete stabilization information, $I = 3$, with a single stabilizer sufficing to crisply classify a neighbor. Since $\langle k_{in} \rangle = 16$, there is, on average, 16-fold redundancy in the stabilization of each node. As random connections are added to the network, the four clusters become connected with each other. Some of the stabilizers start to stabilize against fewer classes, giving rise to a decrease in the average I. In the right panel of figure 3, we have plotted how the average stabilization information decays when k_{out} increases. In order for nodes with $I < 3$ to be stabilizers they have to act in combined action with other nodes, as in the example of figure 1. Thus an increase of the level of disorder k_{out} causes both a reduction in the redundancy of the stabilization of strong nodes and a shift towards stabilization by the combined action of more than one stabilizer. The increase in disorder eventually leads to a loss of strong nodes, implying that the
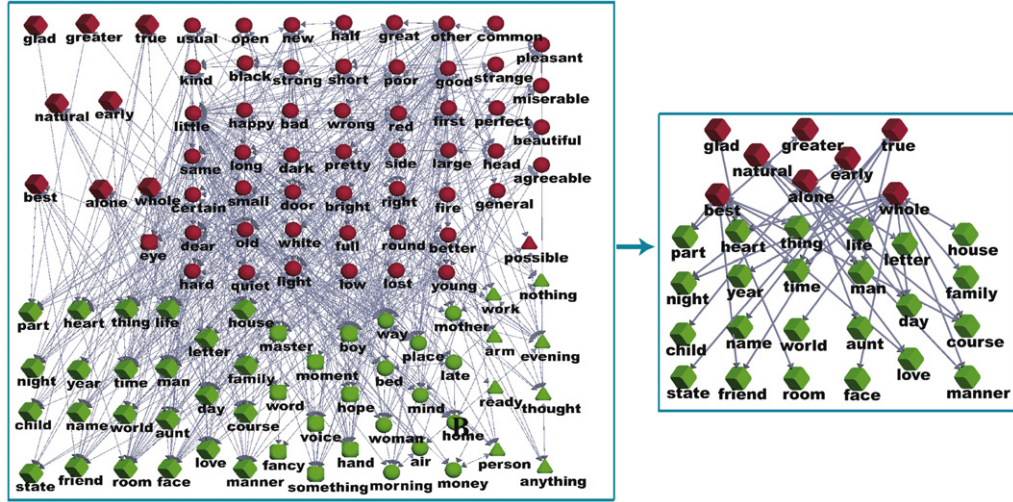
Figure 5. Stabilizer analysis for the noun–adjective network in *David Copperfield* [36]. The links represent words appearing in juxtaposition, while the superposed directed links indicate the stabilization relations. The NL-EM class assignment correlates strongly with the word being a noun (green) or an adjective (red). On the right, the subgraph formed by the strong stabilizers that exhibits a strict bipartite ordering.

classification deteriorates. In order to assess the quality of classifications, we use the entropy S_q, as defined in [2]

$$S_q = -\frac{1}{N} \sum_{ir} q_{ir} \ln q_{ir}. \tag{10}$$

The entropy S_q measures the crispness of a classification. When $S_q = 0$, all the nodes are strong, while $S_q = \ln(\mathcal{N}_C)$ corresponds to a case where the classification of the nodes is maximally uncertain. The right panel of figure 3 displays S_q as a function of k_{out}, showing that the crispness of the classification is lost for large k_{out}.

The increase in entropy is closely related to what happens to the different nodes in the classification as edges are added, particularly to the stabilizers. The variation of the number of the different types of nodes with k_{out} is shown in the left panel of figure 3. As the addition of new edges progresses, some nodes cease to be strong stabilizers. When a node is not a strong stabilizer anymore, it can still remain strong as long as there are other nodes stabilizing it in its neighborhood. As can be seen in the left panel of figure 3, this is what is happening up to $k_{\text{out}} \lesssim 4$: the number of strong stabilizers decreases while the number of strong nodes rises accordingly. Therefore, initially the effect of adding edges is to convert strong stabilizers into strong nodes. Most of the nodes remain strong (stabilizer or not), and the classification is essentially crisp with an entropy $S_q \approx 0$. With the further addition of edges, the number of strong nodes starts to decrease as a result of the loss of stabilization, giving rise to the appearance of weak stabilizing and non-stabilizing nodes at $k_{\text{out}} \gtrsim 4$. Continuing to $k_{\text{out}} \approx 10$, the entropy of the classification remains very low because there is still a sizable number of strong nodes supported by a few weak and strong stabilizers (see panels B and C in figure 2). As further edges are added, the number of

weak stabilizers starts to decrease as well, and eventually most of the nodes are weak and non-stabilizing, accounting for the quick rise in the classification entropy S_q starting around $k_{\mathrm{out}} \approx 10$.

5. Real-world networks

We focus now on some empirical examples to show the special role that the stabilizers play in a classification and the type of information that they convey while also highlighting the versatility of our analysis. As explained, classifications into two groups are particularly simple and in this case the stabilizers can be easily identified once a solution of the NL-EM clustering is given. This simplicity makes them good candidates to illustrate the properties of the stabilizers. We present first two examples of this type that show the role of the stabilizers and the relations between them. We then turn to a directed network with a classification into four groups in order to illustrate a more general situation.

The first example is a network built from the voting records of the 109th US Senate [30]. The nodes represent senators that served the full two-year term (2005–2007) during which 645 issues were voted on. Since our aim is to construct a network based on political affinity, we draw an edge between two senators if they voted in the same way at least once. The edges are weighted by the reciprocal of the number of co-voting senators minus one, a common practice for collaboration networks [31]. In this way, an agreement in minority on an issue has a higher value than that in an unanimous vote, differentiating more clearly close political standings. Due to circumstantial quasi-unanimous votes, the network is initially close to fully connected. A threshold such that edges with lower weights are removed can be introduced, and the resulting networks can be analyzed as the threshold increases. We have applied two-group NL-EM to these networks. Once the threshold is high enough, the clusters found follow well the divide between Democrats and Republicans. The instance in which about half of the senators, either Republicans or Democrats, are stabilizers is displayed in figure 4. Congress roll calls and their derived networks have been extensively studied in the literature [32]–[35]. One of the most interesting results is that single votes of a representative can be understood with a low-dimensional spatial model (DW-NOMINATE [32, 33]) in which a set of coordinates can be assigned to each congressman characterizing his/her political stand on the different issues. Since the 1990s the number of dimensions required has been reduced in good approximation to only one that strongly correlates with the congressman's view on socio-economic questions (liberal versus conservative) [32, 33]. In figure 4, we show the relation between being a stabilizer and the location in the liberal-to-conservative dimension. The stabilizers tend to be the most radical members of the Senate who are probably defining the overall position of their groups. This exercise can be repeated on networks obtained with different thresholds. It can be seen that, as the threshold increases, more and more nodes turn into stabilizers. Keeping track of the senators that become stabilizers at different thresholds allows for a refined exploration of the political spectrum. Note in particular that the above results have been obtained by simply looking at the co-voting relation and without considering the vote records in detail, *i.e. the actual issue put to vote.*

In our second example we show how by extracting the subgraph of stabilizers we can obtain from its structure useful information about what features distinguish a stabilizer node and how the groups relate in a classification. We consider a semantic network in
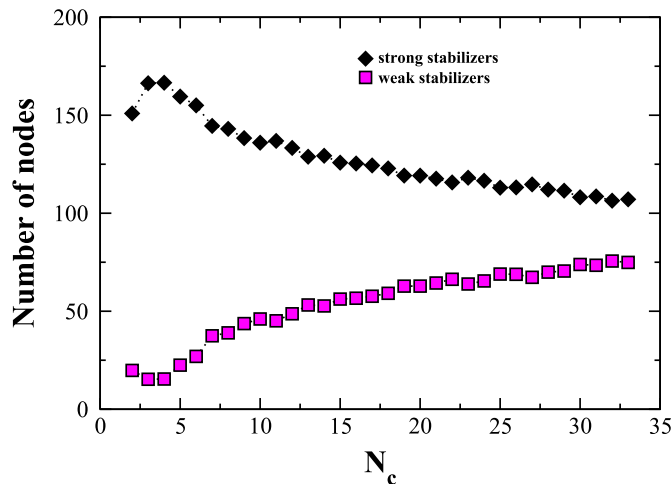
Figure 6. Number of strong and weak stabilizers versus number of groups $\mathcal{N}_C$ as obtained from the NL-EM classification of the Little Rock Lake foodweb [37]. The maximum number of strong stabilizers occurs around $\mathcal{N}_C = 3, 4$. This number is close to the trophic level which is around four, suggesting that a classification into four groups might capture the trophic levels.

which the nodes are adjectives and nouns occurring most frequently in Charles Dickens' novel *David Copperfield* [36]. A relation between any two of these words is established if they occur in juxtaposition. In figure 5, we have represented the network, the best NL-EM partition in two groups and identified the types of nodes. There turn out to be two sub-groups containing nouns or adjectives only that are strong stabilizers. These two sub-groups bear the responsibility for the classification of remaining words by association. Note that the only input to the NL-EM method is the network. We are not introducing any bias for the partition in adjectives and nouns. Most of the remaining words are well classified. The stabilizers, central to establishing the classification, are the words always occurring in strict combinations like *true friends*, never mixing with members of the same group and they form a bipartite subgraph of stabilizers as shown in the right panel of figure 5. Conversely, non-stabilizing nodes are words appearing in mixed roles, such as the word *little* in the adjective–adjective–noun triplet *poor little mother*.

Our final example, showing a more general case with four groups, is the Little Rock foodweb. The vertices of this network are species living in the aquatic environment of Little Rock Lake in Wisconsin [37]. Each directed link represents a predation relation pointing from predator to prey. The number of trophic levels is around four [38] and turns out to be the number of groups for which the NL-EM algorithm produces a partition with the highest abundance of strong stabilizers, as shown in figure 6, where we have plotted the number of stabilizers of an NL-EM solution against the number of groups $\mathcal{N}_C$. A property of the four-group classification depicted in figure 7 is that it keeps basal species (green) in one group, top predators (cyan) in another and assigns the rest to two different groups based on the prey they feed on at the basal level. The species that are not strong stabilizers, for instance nodes 11, 61 or 80, could be related to a missing data problem. In the case of 61 (*Hydroporus*) or 80 (*Lepidoptera Pyralidae*), the species appear only as prey having no connection to lower levels. However, its consumers are not typically feeding

(A)

(B)

(C)

Group	TL	σ_{TL}
cyan	3.49	0.62
red	2.21	0.15
yellow	2.26	0.30
green	1.05	0.09
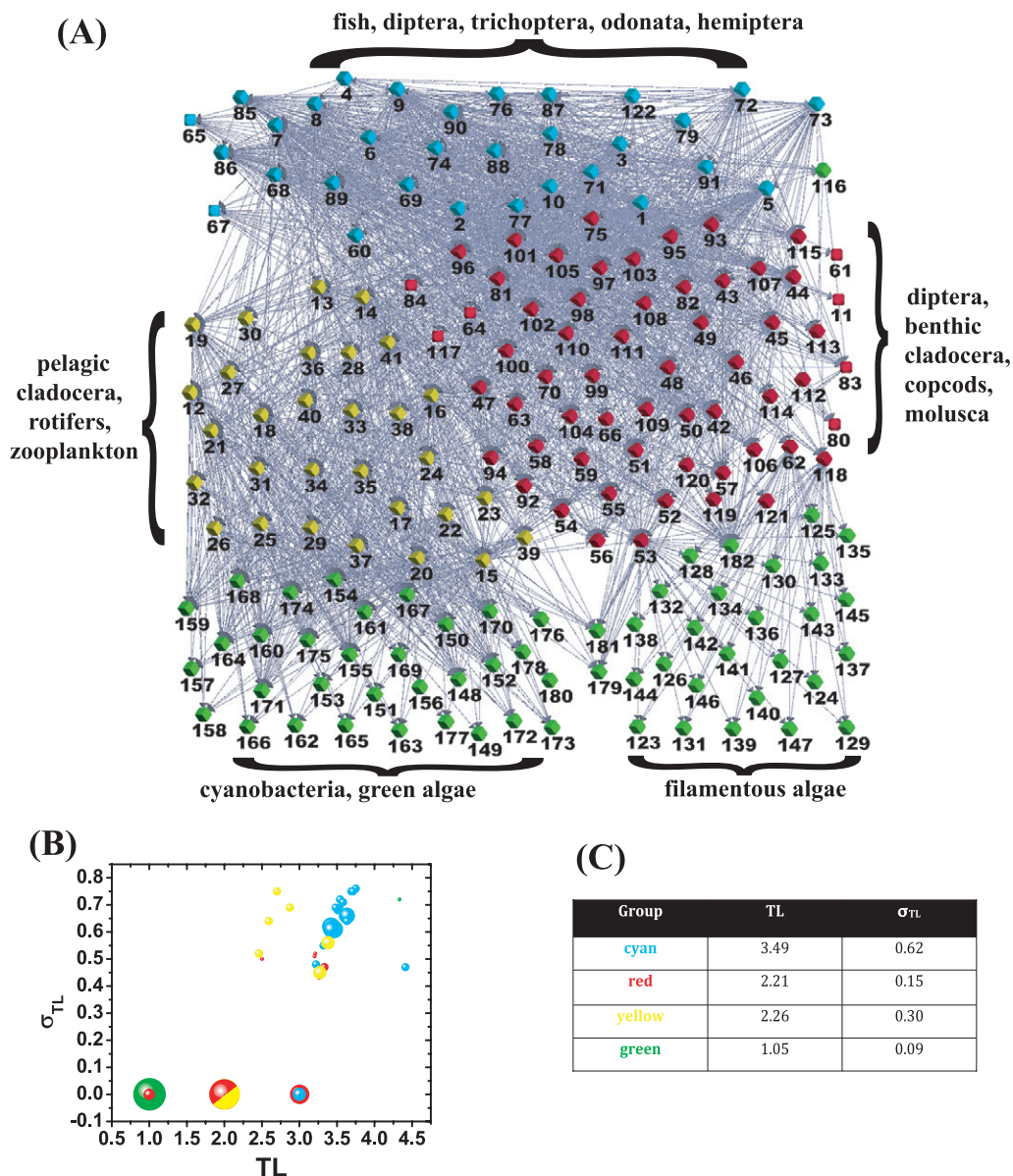
Figure 7. (A): stabilizers for the best four-group NL-EM classification of the Little Rock Lake foodweb [37]. Nodes are species and directed links correspond to predation relations. The node labeling follows [37]. (B): fraction of species belonging to each group plotted against their prey-averaged trophic level (TL) and the standard deviation of TL of their preys, as defined in [38]. The radius of the spheres is proportional to the log of the percentile. Spheres with two colors include species of more than one group (each sphere or half-sphere is independent). (C): averages of TL and σ_{TL} over the species forming each group.

on basal species; they are 'cyan', and this results in an NL-EM classification that assigns them into the 'red' group.

As seen in figure 7(A), most of the species of the network are strong stabilizers. Their abundance is a direct result of the highly structured organization of the foodweb: similar species have similar prey which, as our analysis shows, is also linked to their trophic levels (see figures 7(B) and (C)). Or, more correctly, the consistent choice of species a predator does not prey on is what renders them stabilizers. The possibility of classifying species in low-dimensional spaces depending on their trophic level and on the intervality of their prey distribution has been extensively discussed in the literature [8], [39]–[41]. Our stability analysis reveals an underlying structure in the connectivity pattern of the foodweb, which is responsible for the success of these low-dimensional models.

6. Discussion and conclusions

The maximum likelihood function upon which the NL-EM inference method is based is rather generic and depends on the assumption that nodes with similar connections should be grouped together. Using this likelihood function we were able to show that a subset of nodes, the stabilizers, associated with a given grouping play a central role as they form the backbone of the classification which could not be attained without them. The mathematical basis behind the concept of stabilizers is rather intuitive and follows from the product form of the group assignment probabilities, q_{ir}, in equation (7), which is in turn a direct consequence of the assumption that the edges are statistically independent (equation (2)). Such an assumption is common to a number of probabilistic clustering methods. We can rewrite equation (7) as

$$q_{ir} = \prod_{j \in \nu_i} \tilde{\theta}_{rj}, \tag{11}$$

where

$$\tilde{\theta}_{rj} = \theta_{rj} \left[\frac{\pi_r}{\sum_s \pi_s \prod_{j \in \nu_i} \theta_{sj}} \right]^{1/k_j}, \tag{12}$$

so that the prefactors are equally absorbed into $\tilde{\theta}_{rj}$. Note that q_{ir} is in the interval $[0,1]$. Written in the above form it is clear that very small values of q_{ir} must arise from very small values of $\tilde{\theta}_{rj}$ dominating the product. Likewise, we see from $\mathrm{d}\ln q_{ir}/\mathrm{d}\tilde{\theta}_{rj} = 1/\tilde{\theta}_{rj}$ that changes in these factors will have the greatest effect on the value of q_{ir}. The stabilizers we have introduced here constitute the extreme case, namely the nodes j for which $\tilde{\theta}_{rj} \equiv 0$. As we have shown, this requirement, together with the fact that $\tilde{\theta}_{rj}$ depends in turn on q_{ir}, has allowed us to extract the stabilization rules for crisply classified nodes, $q_{ir} = \delta_{g_i r}$. However, this concept could be relaxed to define stabilizers more generally by requiring only that $\tilde{\theta}_{rj} < \epsilon$ with ϵ appropriately chosen.

It is possible to apply the notion of stabilizers to other probability models for node classification such as those considered by Airoldi *et al* [24] or Nowicki and Snijders [22]. An inspection of equations (2) and (3), as well as the equation for $\hat{B}(g,h)$ of [24], reveals a similar structure for the interrelation between the edge-based class assignment probabilities ϕ and the class connection probability $\hat{B}(g,h)$, which are analogs of the

probabilities q and θ. The variational expectation maximization approach of [24] can also be applied to a model that was considered by Nowicki and Snijders [22], which is more akin in spirit to the model presented here [42]. For both models, however the resulting rules of stabilization are rather involved due to the inclusion in the likelihood of the absence of edges, as well as due to the non-factorizable form of $\Pr(i \leftrightarrow j|g_i, g_j) \equiv \eta(g_i, g_j)$ as compared to equation (1). The attractiveness of the probability model, equations (1) and (2), is that it delivers meaningful classifications despite its simplicity, while at the same time the corresponding stabilization rules have a rather immediate interpretation, as we have shown in sections 3 and 4.

In summary, we have presented a general method for inferring information about which elements are most relevant in establishing group structures in a complex network. The maximum likelihood function upon which our inference is based is rather generic. This approach does not assume any additional *a priori* knowledge about the network, rendering it attractive in circumstances in which the available information about the nodes is limited. In particular, we have introduced the concept of stabilizers associated with a given NL-EM classification and shown that they play a central role in the network partition. If the stabilizers were removed from the network, the partition would lose its meaning. If on the other hand, the subgraph formed only by stabilizers is considered, the classification remains intact and useful information regarding the interaction between the different groups in the graph can be obtained. The stabilizers represent therefore the gist of a network partition. Their identification is highly useful in understanding the way in which the structure of complex systems form and their elements aggregate in clusters. This technique has a wide applicability as we have shown with three empirical examples of networks of very different origins: social sciences, semantics and ecology. In addition it raises several important questions, such as the role of these special nodes in the evolution of any dynamic process running on the graph such as the spreading of opinions, rumors or diseases, or even in the evolution of the graph itself if the network is dynamic.

Acknowledgments

Appendix A. Numerical implementation details

A.1. NL-EM algorithm

Given a network, the search for classifications of the NL-EM algorithm was carried out using an algorithm that alternates between simulated annealing and a direct greedy iteration of equations (5) and (7). The program was run with a set of 10 000 different initial conditions for θ and π for each value of the number of groups N_c. Once the algorithm converged to a stationary value of the likelihood function, the instance with the best $\bar{\mathcal{L}}$ was selected.

```
Algorithm A1: PickNext(iSelected,iAvailable,arrStab)

   (iAvailable,NavailList) ← WHERE(iAvailable,1)
   for i ← 0 to NavailList-1 do
      iSelectedLocal ← iSelected
      iAvailableLocal ← iAvailable
      iSelectedLocal[iAvailList[i]] ← 1
      GetNextAvailable(iSelectedLocal,iAvailableLocal,iSelectedLocal[iAvailList[i]],arrStab)
      PickNext(iSelectedLocal,iAvailableLocal,arrStab)
   end
   comment: If we reach this stage, there is nothing else to select

   (iUsed,nUsed) ← WHERE(iSelectedLocal,1)
   iBit[*] ← arrStab[iUsed[0],*]
   for i ← 0 to nUsed-1 do
      iBit[*] ← iBit[*] or arrStab[iUsed[i],*]
      (MissingClass,Nmissing) ← WHERE(iBit,0)
   end
   comment: If this is not a stabilization, simply return.

   if Nmissing > 0
      return
   comment: We found a stabilization. Update Lists, etc.
```

Figure A.1. Pseudocode for algorithm **PickNext**.

A.2. Extraction of stabilizers

We outline here the algorithm we have used to extract stabilizers from an NL-EM classification with strong nodes. The problem of determining the set of stabilizers associated with a strong node is related to the set covering problem in Computer Science, which is NP-complete. If a strong node has s adjacent nodes with non-empty $\bar{\sigma}$, there are in principle 2^s combinations that have to be checked for finding the fundamental sets leading to stabilizations. In practice, many of the combinations can be eliminated by observing that if, say, $\bar{\sigma}_1, \bar{\sigma}_2, \ldots, \bar{\sigma}_n$ have been selected as candidates for a stabilization, any $\bar{\sigma}$ that is a subset of the union of the $\bar{\sigma}_i$s is redundant and thus cannot be part of that stabilization. This is the main strategy of our algorithm. Also note that, if there are $\mathcal{N}_C$ numbers of classes the number of possible distinct stabilizer sets $\bar{\sigma}$ is $\Sigma = 2^{\mathcal{N}_C - 1}$. For small $\mathcal{N}_C$, s can be larger than Σ so that there are duplicates which can be removed beforehand.

We have used a recursive algorithm for detecting the stabilizers. We partially order the s sets $\bar{\sigma}_i$ by their size. Two binary arrays of size s, *iSelected* and *iAvailable* indicate the candidates already selected and those available for contribution to a stabilization, respectively. The classes against which the s nodes stabilize are coded in an $s \times (\mathcal{N}_C - 1)$ binary array *arrStab*, where the non-zero elements of *arrStab*[$j, *$] indicate the classes against which node j is stabilizing. A recursively called subroutine **PickNext**(), given in figure A.1 as pseudocode, performs the task of determining all stabilizations of a strong node, given the sets *arrStab*. In the algorithms we have assumed that there is already defined a procedure Where(*List, Value*) = (*Pointer, NFound*), which takes a list and returns the indices where the list element equals *Value* along with the number of elements found, *NFound*. Also in our notation, when two lists are operated on term by term we

```
Algorithm A2: GetNextAvailable(iSelected,iAvailable,iPointer,arrStab)

    for i ← 0 to iPointer do
       iAvailable[i] ← 0
    end

    comment: Make sure we have something to select.

    (iNext,nNext) ← WHERE(iAvailable,1)
    if nNext = 0
       return

    comment: Form the union of sets already selected.

    (iUsed,nUsed) ← WHERE(iSelected,1)
    iBit[*] ← arrStab[iUsed[0],*]
    for i ← 0 to nUsed-1 do
       iBit[*] ← iBit[*] or arrStab[iUsed[i],*]
    end

    comment: Now determine the stabilizers that contribute.

    for i ← 0 to nNext-1 do
       iDiff[*] ← iBit[*] - arrStab [iNext[i],*]
       (Diff,nDiff) ← WHERE(iDiff,-1)
       if nDiff = 0
          iAvailable[iNext[i]] ← 0
    end
```

Figure A.2. Pseudocode for algorithm **getNextAvailable**.

denote this as $NewList[*] \leftarrow ListOne[*]\langle\text{Operator}\rangle ListTwo[*]$, avoiding having to write out explicitly a loop over the operation on individual elements.

Initially **PickNext**$(iSelected, iAvailable, arrStab)$ is called with the binary arrays $iSelected$ and $iAvailable$ initialized to zero and one, respectively. The algorithm **getNextAvailable** (see figure A.2) updates $iAvailableLocal$, the set of available stabilizers that can contribute to a stabilization after $iSelectedLocal[i]$ has been added.

Appendix B. Extension of NL-EM to directed graphs and stabilization

A generalization of NL-EM to directed graphs that preserves structural equivalence [5, 6, 8] was recently provided in our earlier work [2]. We assume that, given a node i, a link to a node j can be either out-going, in-going or bidirectional. We thus introduce the probabilities:

- $\overrightarrow{\theta}_{rj}$ that a directed link leaving a vertex of group r connects to node j,

- $\overleftarrow{\theta}_{rj}$ that a directed link pointing to a node in group r exists from j, and

- $\overleftrightarrow{\theta}_{rj}$ that a bidirectional link exiting from group r connects to j,

and construct the probability of realizing a directed graph $\mathcal{G}$ as

$$\Pr(\mathcal{G}, g | \pi, \overleftarrow{\theta}, \overrightarrow{\theta}, \overleftrightarrow{\theta}) = \prod_i \left[\pi_{g_i} \prod_{j \in \overleftarrow{\nu}_i} \overleftarrow{\theta}_{g_i, j} \prod_{j \in \overrightarrow{\nu}_i} \overrightarrow{\theta}_{g_i, j} \prod_{j \in \overleftrightarrow{\nu}_i} \overleftrightarrow{\theta}_{g_i, j} \right], \tag{B.1}$$

$\overleftarrow{\nu}_i$, $\overrightarrow{\nu}_i$ and $\overleftrightarrow{\nu}_i$ are the set of adjacent nodes of i from which i receives an in-coming, out-going and bidirectional link, respectively.

The likelihood can now be written as

$$\bar{\mathcal{L}}(\pi, \theta) = \sum_{ir} q_{ir} \left[\ln \pi_r + \sum_{j \in \overleftarrow{\nu}_i} \ln \overleftarrow{\theta}_{r,j} + \sum_{j \in \overrightarrow{\nu}_i} \ln \overrightarrow{\theta}_{r,j} + \sum_{j \in \overleftrightarrow{\nu}_i} \ln \overleftrightarrow{\theta}_{r,j} \right], \qquad \text{(B.2)}$$

which has to be maximized under the following constraint on the probabilities θ_{rj}:

$$\sum_i \left(\overleftarrow{\theta}_{r,i} + \overrightarrow{\theta}_{r,i} + \overleftrightarrow{\theta}_{r,i} \right) = 1, \qquad \text{(B.3)}$$

implying that there is no isolated node. The probability π_r, that a randomly selected node belongs to group r, is again given by $\sum_r \pi_r = 1$. The final result is [2]

$$\pi_r = \frac{1}{N} \sum_i q_{ir}, \qquad \text{(B.4)}$$

$$\begin{aligned}
\overleftarrow{\theta}_{rj} &= \frac{\sum_{i \in \overrightarrow{\nu}_j} q_{ir}}{\sum_i q_{ir}(\bar{k}_i^{\mathrm{i}} + \bar{k}_i^o - \bar{k}_i^{\mathrm{b}})}, \\
\overrightarrow{\theta}_{rj} &= \frac{\sum_{i \in \overleftarrow{\nu}_j} q_{ir}}{\sum_i q_{ir}(\bar{k}_i^{\mathrm{i}} + \bar{k}_i^o - \bar{k}_i^{\mathrm{b}})}, \\
\overleftrightarrow{\theta}_{rj} &= \frac{\sum_{i \in \overleftrightarrow{\nu}_j} q_{ir}}{\sum_i q_{ir}(\bar{k}_i^{\mathrm{i}} + \bar{k}_i^o - \bar{k}_i^{\mathrm{b}})},
\end{aligned} \qquad \text{(B.5)}$$

where $\bar{k}_i^{\mathrm{i}}$, $\bar{k}_i^o$ and $\bar{k}_i^{\mathrm{b}}$ are the in-degree, out-degree and bidirectional degree of node i, respectively.

These expressions have to be again supplemented with the self-consistent equation for q_{ir} which now is

$$q_{ir} = \frac{\pi_r \prod_{j \in \overleftarrow{\nu}_i} \overleftarrow{\theta}_{rj} \prod_{j \in \overrightarrow{\nu}_i} \overrightarrow{\theta}_{rj} \prod_{j \in \overleftrightarrow{\nu}_i} \overleftrightarrow{\theta}_{rj}}{\sum_s \left\{ \pi_s \prod_{j \in \overleftarrow{\nu}_i} \overleftarrow{\theta}_{sj} \prod_{j \in \overrightarrow{\nu}_i} \overrightarrow{\theta}_{sj} \prod_{j \in \overleftrightarrow{\nu}_i} \overleftrightarrow{\theta}_{sj} \right\}}. \qquad \text{(B.6)}$$

Note that, when we have only bidirectional links so that $\overleftarrow{\nu}_i = \overrightarrow{\nu}_i = \emptyset$ for all i, it follows from equation (B.5) that $\overleftarrow{\theta}_{rj} = \overrightarrow{\theta}_{rj} = 0$. Thus we recover the undirected EM equations (5) and (7) under the identification $\theta_{rj} = \overleftrightarrow{\theta}_{rj}$.

B.1. Stabilization rules for directed graphs

The case of directed graphs is similar to the undirected case with a few minor modifications. Given an NL-EM classification of a directed graph $\mathcal{G}$, we associate with each node i the following four sets:

- $\overleftarrow{\sigma}_i = \{r | \overleftarrow{\theta}_{ri} = 0\}$, the set of groups that i does not have an out-going connection to,
- $\overrightarrow{\sigma}_i = \{r | \overrightarrow{\theta}_{ri} = 0\}$, the set of groups that i does not have an in-going connection to,

- $\overline{\overset{\leftrightarrow}{\sigma}}_i = \{r|\ \overset{\leftrightarrow}{\theta}_{ri} = 0\}$, the set of groups that i does not have a bidirectional connection to,

- $\overline{c}_i = \{r|q_{ir} = 0\}$, the set of groups that i does not belong to,

along with their complements, $\overset{\leftarrow}{\sigma}_i$, $\overset{\rightarrow}{\sigma}_i$, $\overset{\leftrightarrow}{\sigma}_i$ and c_i.

The NL-EM equations, equations (B.5) and (B.6), relate the sets $\overline{\sigma}_i$ and $\overline{c}_i$ to each other as follows:

$$\bigcup_{j\in\overset{\leftarrow}{\nu}_i} \overline{\overset{\leftarrow}{\sigma}}_j \bigcup_{j\in\overset{\rightarrow}{\nu}_i} \overline{\overset{\rightarrow}{\sigma}}_j \bigcup_{j\in\overset{\leftrightarrow}{\nu}_i} \overline{\overset{\leftrightarrow}{\sigma}}_j = \overline{c}_i, \tag{B.7}$$

$$\bigcap_{i\in\overset{\rightarrow}{\nu}_j} \overline{c}_i = \overline{\overset{\leftarrow}{\sigma}}_j, \tag{B.8}$$

$$\bigcap_{i\in\overset{\leftarrow}{\nu}_j} \overline{c}_i = \overline{\overset{\rightarrow}{\sigma}}_j, \tag{B.9}$$

$$\bigcap_{i\in\overset{\leftrightarrow}{\nu}_j} \overline{c}_i = \overline{\overset{\leftrightarrow}{\sigma}}_j. \tag{B.10}$$

Defining the set of all stabilizer classes associated with a node, irrespective of the directionality, as

$$\overline{\sigma}_i \equiv \overline{\overset{\leftarrow}{\sigma}}_i \cup \overline{\overset{\rightarrow}{\sigma}}_i \cup \overline{\overset{\leftrightarrow}{\sigma}}_i, \tag{B.11}$$

the stabilization condition for a node i becomes identical to the one for the undirected case:

$$\bigcup_{j\in\nu_i} \bar{\sigma}_j = \bar{c}_i. \tag{B.12}$$

References

[1] Newman M E J and Leicht E A, *Mixture models and exploratory analysis in networks*, 2007 *Proc. Nat. Acad. Sci.* **104** 9564

[2] Ramasco J J and Mungan M, *Inversion method for content-based networks*, 2008 *Phys. Rev. E* **77** 036122

[3] Rives A and Galitski T, *Modular organization of cellular networks*, 2003 *Proc. Nat. Acad. Sci.* **100** 1128

[4] Girvan M and Newman M E J, *Community structure in social and biological networks*, 2002 *Proc. Nat. Acad. Sci.* **99** 7821

[5] Doreian P, Batagelj V and Ferligoj A, 2005 *Generalized Blockmodeling* (Cambridge: Cambridge University Press)

[6] Lorrain F and White H C, *Structural equivalence of individuals in social networks*, 1971 *J. Math. Sociol.* **1** 49

[7] Babu M M, Luscombe N M, Aravind L, Gerstein M and Teichmann S A, *Structure and evolution of transcriptional regulatory networks*, 2004 *Curr. Opini. Struct. Biol.* **14** 283

[8] Williams R J and Martinez N, *Simple rules yield complex food webs*, 2000 *Nature* **404** 180

[9] Radicchi F, Castellano C, Cecconi F, Loreto V and Parisi D, *Defining and identifying communities in networks*, 2004 *Proc. Nat. Acad. Sci.* **101** 2658

[10] Fortunato S and Barthélemy M, *Resolution limit in community detection*, 2007 *Proc. Nat. Acad. Sci.* **104** 36

[11] Newman M E J and Girvan M, *Finding and evaluating community structure in networks*, 2004 *Phys. Rev. E* **69** 026113

[12] Newman M E J, *Modularity and community structure in networks*, 2004 *Phys. Rev. E* **69** 066133

[13] Newman M E J, *Modularity and community structure in networks*, 2006 *Proc. Nat. Acad. Sci.* **103** 8577

[14] Arenas A, Díaz-Guilera A and Pérez-Vicente C J, *Synchronization reveals topological scales in complex networks*, 2006 *Phys. Rev. Lett.* **96** 114102

[15] Fortunato S, *Community detection in graphs*, 2010 *Phys. Rep.* **486** 75

[16] Clauset A, Moore C and Newman M E J, *Hierarchical structure and the prediction of missing links in networks*, 2008 *Nature* **453** 98

[17] Flake G W, Lawrence S, Giles C L and Coetzee F M, *Self-organization and identification of Web communities*, 2002 *Computer* **35** (3) 66

[18] Guimerà R, Mossa S, Turtschi A and Amaral L A N, *The worldwide air transportation network: anomalous centrality, community structure, and cities' global roles*, 2005 *Proc. Nat. Acad. Sci.* **102** 7794

[19] Capocci A, Rao F and Caldarelli G, *Taxonomy and clustering in collaborative systems: the case of the on-line encyclopedia Wikipedia*, 2008 *Europhys. Lett.* **81** 28006

[20] Holland P W and Leinhardt S, *Local structure in social networks*, 1976 *Sociol. Methodol.* **7** 1

[21] Wang Y J and Wong G Y, *Stochastic blockmodels for directed graphs*, 1987 *J. Am. Stat. Assoc.* **82** 8

[22] Nowicki K and Snijders T A B, *Estimation and prediction for stochastic blockstructures*, 2001 *J. Am. Stat. Assoc.* **96** 1077

[23] Hastings M B, *Community detection as an inference problem*, 2006 *Phys. Rev. E* **74** 035102(R)

[24] Airoldi E M, Blei D M, Fienberg S E and Xing E P, *Mixed membership stochastic blockmodels*, 2008 *J. Mach. Learn. Res.* **9** 1823

[25] Vazquez A, *Population stratification using a statistical model of hypergraphs*, 2008 *Phys. Rev. E* **77** 066106

[26] Vazquez A, *Bayesian approach to clustering real value, hypergraph and bipartite graph data: solution via variational methods*, 2008 arXiv:0805.2689

[27] Zanghi H, Ambroise C and Miele V, *Fast online graph clustering via Erdös-Rényi mixture*, 2008 *Pattern Recognit.* **41** 3592

[28] Hofman J M and Wiggins C H, *Bayesian approach to network modularity*, 2008 *Phys. Rev. Lett.* **100** 258701

[29] Mackay D J C, 2003 *Information Theory, Inference and Learning Algorithms* (Cambridge: Cambridge University Press)

[30] Data available at http://www.voteview.com

[31] Newman M E J, *Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality*, 2001 *Phys. Rev. E* **64** 016132

[32] Poole K T and Rosenthal H, 2007 *Ideology and Congress* (New Jersey: Transaction Publishers)

[33] Poole K T, 2005 *Spatial Models of Parliamentary Voting* (Cambridge: Cambridge University Press)

[34] Porter M A, Mucha P J, Newman M E J and Warmbrand C M, *A network analysis of committees in the US House of Representatives*, 2005 *Proc. Nat. Acad. Sci.* **102** 7057

[35] Porter M A, Mucha P J, Newman M E J and Friend A J, *Community structure in the United States House of Representatives*, 2007 *Physica A* **386** 414

[36] Newman M E J, *Finding community structure in networks using the eigenvectors of matrices*, 2006 *Phys. Rev. E* **74** 036104

[37] Martinez N, *Artifacts or attributes? Effects of resolution on the little rock lake food web*, 1991 *Ecol. Monogr.* **61** 367

[38] Williams R J and Martinez N, *Limits to trophic levels and omnivory in complex food webs: theory and data*, 2004 *Am. Nat.* **163** 458

[39] Cattin M F, Bersier L F, Banašek-Richter C, Baltensperger R and Gabriel J P, *Phylogenetic constraints and adaptation explain food-web structure*, 2004 *Nature* **427** 835

[40] Stouffer D B, Camacho J and Amaral L A N, *A robust measure of food web intervality*, 2006 *Proc. Nat. Acad. Sci.* **103** 19015

[41] Allesina S, Alonso D and Pascual M, *A general model for food web structure*, 2008 *Science* **320** 658

[42] Mungan M, 2010 unpublished