1

Structural Trends in Network Ensembles*

Ulrik Brandes, Jürgen Lerner, Uwe Nagel, and Bobo Nick

Department of Computer & Information Science, University of Konstanz

Summary. A collection of networks is considered a network ensemble if its members originate from a common natural or technical process such as repeated measurements, replication and mutation, or massive parallelism, possibly under varying conditions. We propose a spectral approach to identify structural trends, i. e. prevalent patterns of connectivity, in an ensemble by delineating classes of networks with similar role structure. Formal, experimental, and practical evidence of its potential is given.

1.1 Introduction

Network-analytic studies most frequently are concerned with a small set of networks if not a singleton instance. Indicators employed in such analyses range from properties of individual actors (e. g., centrality and role) and local patterns (e. g., reciprocated ties, stars, and closed triangles) over to global network characteristics (e. g., density, modularity, and degree distributions) [17, 4]. Given the ever increasing availability of data, there is a growing tendency to compare families of networks that, e.g., may be defined on different sets of actors or encode different relations (see, e. g., [10, 7, 9]). Application scenarios for network comparison include examination whether different teams of employees exhibit structural differences [8, p. 81], comparison of networks among different species [9], detection of user-roles in Usenet newsgroups by patterns in egocentric reply-networks [18], and comparison of social integration in personal networks of immigrants [6]. To emphasize the (assumed) existence of an inherent relation in a collection of networks, we will refer to it as a *network ensemble*.

Clearly, the elements of a network ensemble can be compared and categorized based on any global structural property or extrinsic attributes (i. e.,

* Research supported in part by DFG under grant GK 1024 (Research Training Group "Explorative Analysis and Visualization of Large Information Spaces") and University of Konstanz under grant FP 626/08.

"who is in the network"). In this paper we treat networks as similar, if they exhibit the same *role structure*, i. e., if they show the same pattern of connectivity among classes of actors. Actors are said to play the same *role*, or occupy the same *position*, in a network if they are similarly connected to other actors that themselves play the same role [3, 17, 13]. For instance, by this definition university professors would occupy the same structural position if they have identical patterns of ties to students, secretaries, industry contacts, other professors and so on. Such a role assignment on a given network yields a smaller graph, called the *role graph* (compare [14]), whose vertices are the actor classes and whose (weighted) edges encode how actors in one class are on average connected to actors in the other class. In this paper we compare networks by the role graphs they give rise to. Returning to the above example, the networks of two universities might differ in the fact that in one university the professors are differently (stronger or weaker) connected to the students than in the other university.

However, since already the decision problem whether a given graph admits a role structure of a specific type is NP-complete [11], our strategy for network comparison seems to run into serious computational problems. Indeed we do not attempt to design an algorithm that is able to compare any (worst-case) instance of a network ensemble; rather, we propose an efficient heuristic algorithm that provably performs well on networks ensembles arising from certain random graph models. More specifically, if a network ensemble contains subsets of networks that indeed differ sufficiently in their role structure, then our algorithm will correctly distinguish those networks with high probability, i. e., it will detect a good clustering of the network ensemble.

In Sect. 1.2 we define a stochastic model for network ensembles with latent role structures and define the associated clustering problem. We propose an algorithm for clustering network ensembles in Sect. 1.3 and show in Sect. 1.4 that it recovers class-memberships with high probability—given that the stochastic model satisfies certain preconditions. Experimental results on artificially generated networks in Sect. 1.5 and a small case study on Wikipedia edit-networks in Sect. 1.6 provide further evidence of the usefulness of our approach.

1.2 A Network Ensemble Model with Latent Roles

We start by recalling a model for random graphs that exhibit a hidden (latent) class structure; such a model is defined, e. g., in [15] and [12].

Definition 1. *A* planted partition model $\mathcal{G}(n, k, \psi, P)$ *is given by a number of vertices* n*, a number of classes* k*, a partition* $\psi\colon \{1, \dots, n\} \to \{1, \dots, k\}$ *of the* n *vertices into* k *classes and a symmetric* $k \times k$ *matrix* P *of edge probabilities* $P_{ij} \in [0, 1]$ *between classes. The probability of a given graph* $G = (V, E)$ *with* n *vertices given the model* $\mathcal{G}(n, k, \psi, P)$ *is*

$$\mathbb{P}\left(G\middle|\mathcal{G}(n,k,\psi,P)\right) = \prod_{\{u,v\}\in E} P_{\psi(u)\psi(v)} \prod_{\{u,v\}\notin E} 1 - P_{\psi(u)\psi(v)}$$

Alternatively, an instance G of $\mathcal{G}(n,k,\psi,P)$ is drawn by including each edge $\{u,v\}$ into G independently with probability $P_{\psi(u)\psi(v)}$. Thus, the probability of an edge between vertices u and v is only dependent on their class-membership.

A planted partition model $\mathcal{G} = \mathcal{G}(n,k,\psi,P)$ is completely defined by its *expected adjacency matrix* which is the $n \times n$ matrix $\overline{M} = \overline{M}(\mathcal{G})$ whose entries are defined by $\overline{M}_{ij} = P_{\psi(i)\psi(j)}$. Note that $\overline{M}$ is indeed the expectation of the adjacency matrices of graphs drawn from $\mathcal{G}(n,k,\psi,P)$.

In this paper we consider random network ensembles that are mixtures of such planted partition models.

Definition 2. *A* (planted partition) network ensemble $\mathcal{E}(N,K,\Psi,\mathcal{G}_1,\ldots,\mathcal{G}_K)$ *is given by a number of graphs N, a number of graph models K, an assignment $\Psi\colon \{1,\ldots,N\} \to \{1,\ldots,K\}$ of the N graphs to the K models and a family of K planted partition models $\mathcal{G}_1,\ldots,\mathcal{G}_K$, where $\mathcal{G}_i = \mathcal{G}(n_i,k_i,\psi_i,P_{(i)})$.*

Thus, a planted partition network ensemble is a set of random graphs drawn from planted partition models. To obtain an instance of $\mathcal{E}(N,K,\Psi,\mathcal{G}_1,\ldots,\mathcal{G}_K)$, the N graphs G_i, $i = 1,\ldots,N$, are independently drawn from the planted partition model $\mathcal{G}_{\Psi(j)}$. For sake of simplicity we will often write in this paper network ensemble instead of planted partition network ensemble. In the major part of this paper we consider network ensembles with the same number of vertices; only in Sect. 1.6, where we analyze real-world networks, do we apply our algorithms to networks of different size.

The *algorithmic problem* associated with a planted partition network ensemble $\mathcal{E} = \mathcal{E}(N,K,\Psi,\mathcal{G}_1,\ldots,\mathcal{G}_K)$ is the following.

> Given an instance $(G_1,\ldots,G_N)$ of $\mathcal{E}$, classify the N graphs such that two graphs are in the same class if and only if they are drawn from the same underlying planted partition model.

Obviously, without any further preconditions this problem is not solvable. (For instance, if two of the underlying planted partition models are identical, the graphs generated from these are not distinguishable.) However, in this paper we propose an efficient algorithm such that, given certain preconditions, we can decide for each given pair of graphs with high probability whether they are drawn from the same underlying model or not. (The term *with high probability* means "with probability that tends to one as the size of the graphs tends to infinity"; this notion is often employed to assess the quality of heuristic algorithms, compare [15].)

1.3 Classification Method

1.3.1 Intuition

Let $(G_1, \ldots, G_N)$ be an instance drawn from a planted partition network ensemble $\mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$. In the following we sketch how we will proceed to determine for any two graphs whether they are drawn from the same underlying planted partition model or not.

A very simple observation is that if we were not given the adjacency matrices $M_1, \ldots, M_N$ of $G_1, \ldots, G_N$ but rather their expectation values $\overline{M_1}, \ldots, \overline{M_N}$, then the problem would be fairly trivial: under the minimal assumption that the planted partition models $\mathcal{G}_1, \ldots, \mathcal{G}_K$ are pairwise different, it follows that their expected adjacency matrices are pairwise different as well. Hence, two graphs G_i, G_j out of $G_1, \ldots, G_N$ are drawn from the same model if and only if their expected adjacency matrices $\overline{M_i}$, $\overline{M_j}$ are equal.

However, our algorithm does not have access to the expected adjacency matrices. Indeed, the adjacency matrix M_i of an instance graph is rather very far from its expectation value $\overline{M_i}$. (Note that M_i is a zero/one-matrix, while the entries of $\overline{M_i}$ are from the real interval $[0, 1]$; thus the expectation is typically not attainable.)

What helps us out of this dilemma is a well-known combination of results from matrix perturbation theory [16] with probabilistic bounds on the eigenvalues of random matrices [1] (also compare [15]). Basically, these results enable us to show that, even if the adjacency matrix M of an instance graph differs entrywise very much from its expectation $\overline{M}$, the spectrum of M is with high probability close to the spectrum of $\overline{M}$. It follows that the adjacency matrices of two graphs drawn from the same model have (with high probability) similar spectra and, under the assumption that the spectra of the expected adjacency matrices differ in at least one value, graphs from different models have a larger difference in their spectra.

1.3.2 Method

The ordered spectrum of a symmetric $n \times n$ matrix is denoted by $\lambda_1 \leq \ldots \leq \lambda_n$ and the vector $\lambda(M) = (\lambda_1, \ldots, \lambda_n)^T$ is referred to as the spectrum vector of matrix M. An instance of our classification problem is created by randomized drawing N adjacency matrices M_i according to some underlying role graphs. Each adjacency matrix M_i provides us with a corresponding graph G_i, which gives us a network ensemble $\mathcal{E} = \{G_1, \ldots, G_N\}$.

We do neither know how many role graphs there are nor which graphs belong to the same role graph. What we do know is that graphs being drawn from the same role graph should have a spectrum much more similar to each other than graphs drawn from different role graphs. As we show in Sec. 1.4 it is suggestive to measure the similarity between two graphs in this context by the supremum norm of their spectrum vectors. So under certain assumptions

$\|\lambda(M_1) - \lambda(M_2)\|_\infty$ should be much greater if the graphs corresponding to M_1 and M_2 are created from different role graphs than if they were from the same role graph.

This makes our classification problem to a classical clustering problem. Given objects and distances between them, dense clusters of objects are searched. Standard clustering algorithms can be applied as long as they can be parameterized with a distance measure. An example would be a version of k-means, which does not need the number of clusters as an input. We performed some promising experiments with an iterated k-means using the silhouette coefficient to decide for the optimal clustering.

The pseudocode in Alg. 1 summarizes our method for detecting structural trends in network ensembles.

Algorithm 1: Structural Trends in Network Ensembles

Input: network ensemble $\mathcal{E} = \{G_1, \ldots, G_N\}$
Result: clustering $\{C_1, \ldots, C_k\}$ with $\mathcal{E} = \biguplus_i C_i$

for $G \in \mathcal{E}$ **do**
  determine spectrum vector $\lambda(G)$
end
partition $\{\lambda(G) : G \in \mathcal{E}\}$ using supremum norm

In the ideal case this method extracts from an arbitrary ensemble a classification of the graphs into groups having the same role graph and thereby solves the stated algorithmic problem. Taking our results one could also think of classifications of ensembles consisting of differently sized graphs. It would be necessary to restrict the spectrum vector to a size such that it can be determined for all graphs of the ensemble. One would also have to take care of the growth of the eigenvalues which is linear in the number of vertices of the graph. A possible approach is to take the n eigenvalues with maximum absolute value of each graph where n is the size of the smallest graph in the given instance and divide them by the size of the graph. A more efficient method could be inferred by knowledge of the sizes of the underlying role graphs. If the assumptions of the next section are met, the number of eigenvalues used can be limited by the maximum number of vertices of the role graphs without changing the defined distances.

1.3.3 Generalization to Weighted Networks

We restricted the method sketched above to binary (unweighted) graphs only for notational simplification. A model for ensembles of *weighted* networks (i. e., graphs with real edge-weights) could be defined in almost the same way as in Sect. 1.2. A *weighted* planted partition model is defined as in Def. 1 with the difference that when drawing an instance graph one does not include

(unweighted) edges with a given probability but rather the weight of an edge $\{u, v\}$ is drawn from a distribution dependent on the classes of u and v. Examples of such distributions would include the normal distribution where the mean value depends on the vertex classes.

The adjacency matrix of a weighted graph is a real matrix whose entries encode the edge weights. Note that the abovementioned method for network classification via the eigenvalues of graphs can be applied to these weighted matrices without any change. Furthermore, the theorems that will be presented in Sect. 1.4 hold also for the case of weighted matrices. The application to real world data sketched in Sect. 1.6 indeed analyzes an ensemble of weighted networks.

1.4 Evidence from Matrix Perturbation Theory

Let $(G_1, \ldots, G_N)$ be an instance drawn from a planted partition network ensemble $\mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$ whose underlying graph models have a common number of vertices n. Building on results from matrix perturbation theory, we show in this section that for sufficiently large n (and ignoring a small number of outliers) the spectra of graphs drawn from the same model have smaller distance than the spectra of graphs drawn from different models.

We start by associating a planted partition model $\mathcal{G}$ with a matrix $A(\mathcal{G})$ that encodes the relative class-sizes as well as the edge-probabilities between classes of $\mathcal{G}$. It turns out that the eigenvalues of $A(\mathcal{G})$ correspond—up to a multiplicative constant that is related to the size of the classes—to the nonzero eigenvalues of the expected adjacency matrix $\overline{M}(\mathcal{G})$.

Definition 3. *Let $\mathcal{G} = \mathcal{G}(n, k, \psi, P)$ be a planted partition model and denote the proportion of vertices in class $i = 1, \ldots, k$ with*

$$q_i = |\{v\,;\ 1 \le v \le n \text{ and } \psi(v) = i\}|\,/n$$

The structure matrix *associated to $\mathcal{G}$ is the $k \times k$ matrix $A = A(\mathcal{G})$ whose entries are defined by $A_{ij} = \sqrt{q_i q_j} \cdot P_{ij}$.*

To make the notion *with high probability* precise, we define a process by which we can increase the number of vertices in a planted partition model without changing its structure (more precisely: without changing the relative class-sizes nor the edge-probabilities between classes). Let $\mathcal{G}_1 = \mathcal{G}(n_1, k, \psi_1, P)$ be a fixed planted partition model and $t \in \mathbb{N}_{\ge 1}$ an integer. A planted partition model $\mathcal{G}_t$ that has $n_t = t \cdot n_1$ vertices and the same structure matrix as $\mathcal{G}_1$ can be defined by $\mathcal{G}_t = \mathcal{G}(n_t, k, \psi_t, P)$, where $\psi_t \colon \{1, \ldots, n_t\} \to \{1, \ldots, k\}$ with $\psi_t(v) = \psi_1(\lceil v/t \rceil)$. Note that it holds $A(\mathcal{G}_t) = A(\mathcal{G}_1)$.

The next theorem shows that the eigenvalues of a planted partition model with fixed structure matrix grow linearly in the number of vertices.

Theorem 1. *Let $\mathcal{G}_1 = \mathcal{G}(n_1, k, \psi_1, P)$ be a planted partition model, $t \in \mathbb{N}_{\geq 1}$ an integer, and set $n_t = t \cdot n_1$. Each eigenvalue λ of $A(\mathcal{G}_1)$ yields an eigenvalue $n_t \cdot \lambda$ of $\overline{M}(\mathcal{G}_t)$. The remaining $n_t - k$ eigenvalues of $\overline{M}(\mathcal{G}_t)$ are equal to zero.*

Proof. Note first that the expected matrix $\overline{M} = \overline{M}(\mathcal{G}_t)$ is (after reordering the vertices such that vertices in the same class are consecutive) an $n_t \times n_t$ block matrix

$$\overline{M} = \begin{pmatrix} B_{11} & \dots & B_{1k} \\ \vdots & & \vdots \\ B_{k1} & \dots & B_{kk} \end{pmatrix} \text{ with blocks } B_{ij} = \begin{pmatrix} P_{ij} & \dots & P_{ij} \\ \vdots & & \vdots \\ P_{ij} & \dots & P_{ij} \end{pmatrix}$$

of dimension $(q_i \cdot n_t) \times (q_j \cdot n_t)$. (Note that $q_i \cdot n_t$ is indeed an integer which follows from the definitions of q_i and n_t.)

Let $x = (x_1, \dots, x_k)^{\mathrm{T}}$ be any eigenvector of $A(\mathcal{G}_1)$ associated to eigenvalue $\lambda \in \mathbb{R}$. Spelling out the equation $A(\mathcal{G}_1) \cdot x = \lambda \cdot x$ yields for $i = 1, \dots, k$

$$\lambda \cdot x_i = \sum_{j=1}^{k} \sqrt{q_i q_j} P_{ij} x_j = \sqrt{q_i} \sum_{j=1}^{k} \sqrt{q_j} P_{ij} x_j \ . \tag{1.1}$$

We claim that the n_t-dimensional vector y defined by

$$y = (\underbrace{x_1/\sqrt{q_1}, \dots, x_1/\sqrt{q_1}}_{n_t \cdot q_1 \text{ times}}, \dots, \underbrace{x_k/\sqrt{q_k}, \dots, x_k/\sqrt{q_k}}_{n_t \cdot q_k \text{ times}})^{\mathrm{T}} \ ,$$

satisfies $\overline{M}(\mathcal{G}_t) \cdot y = n_t \lambda y$ which shows that $n_t \lambda$ is an eigenvalue of $\overline{M}(\mathcal{G}_t)$ and, thus, yields the assertion of the theorem.

To see that this is true let v be any integer satisfying $1 \leq v \leq n_t$ and let $i = \psi_t(v)$ (i. e., i is the index of the class of vertex v.) We get

$$(\overline{M}(\mathcal{G}_t) \cdot y)_v = \sum_{j=1}^{k} n_t q_j P_{ij} x_j / \sqrt{q_j} = n_t \sum_{j=1}^{k} \sqrt{q_j} P_{ij} x_j = n_t \lambda x_i / \sqrt{q_i} = n_t \lambda y_v$$

where the third equation follows from Eq. (1.1). □

Corollary 1. *Let $\mathcal{G}_1$ and $\mathcal{H}_1$ be two planted partition models with the same number of vertices n. Let $t \in \mathbb{N}_{\geq 1}$ and set $n_t = t \cdot n$. Under the assumption that the eigenvalues of $A(\mathcal{G}_1)$ and $A(\mathcal{H}_1)$ differ in at least one value, the distance between the spectrum vectors of the expected adjacency matrices of $\mathcal{G}_t$ and $\mathcal{H}_t$ grows linearly in the number of vertices n_t. More precisely*

$$\|\lambda(\overline{M}(\mathcal{G}_t)) - \lambda(\overline{M}(\mathcal{H}_t))\|_\infty = n_t \cdot \|\lambda(A(\mathcal{G}_1)) - \lambda(A(\mathcal{H}_1))\|_\infty \in \Theta(n_t) \ .$$

All that remains us to do is to bound the difference between the eigenvalues of the adjacency matrix M of an instance graph and its expectation $\overline{M}$. For this purpose define the perturbation matrix $E = M - \overline{M}$ as the difference between the observed adjacency matrix and its expectation. We recall a result from matrix perturbation theory.

Theorem 2 ([16]). *Let $M = \overline{M} + E$ be a symmetric perturbation of a symmetric matrix $\overline{M}$. Then we have*

$$\|\lambda(M) - \lambda(\overline{M})\|_\infty \leq \|E\|_2 \ ,$$

where $\|E\|_2$ denotes the maximal absolute value of an eigenvalue of E.

The second result we need is a probabilistic bound on the maximal eigenvalue of the difference between the observed adjacency matrix and its expectation.

Theorem 3 ([15]). *Let M, $\overline{M}$ and E be defined as above and let n denote their dimension. Let σ^2 be the largest variance of an entry in M. (Note that if the i,j'th entry of $\overline{M}$ equals p, then its variance is $p - p^2$; the variance is non-zero if p is in the open interval from zero to one.) If $\sigma^2 \gg \log^6 n/n$, then $\|E\|_2 \leq 4\sigma\sqrt{n}$ with probability at least $1 - 2e^{-\sigma^2 n/8}$.*

The assumption $\sigma^2 \gg \log^6 n/n$ is satisfied for sufficiently large n if at least one entry of $\overline{M}$ is different from zero and one. For the remainder of this paper we will take this assumption for granted; note that this excludes only uninteresting cases.

The next corollary follows from Theorems 2 and 3.

Corollary 2. *Let M and $\overline{M}$ be defined as above and let n denote their dimension. It is $\|\lambda(M) - \lambda(\overline{M})\|_\infty \in \mathcal{O}(\sqrt{n})$ with probability in $1 - o(1)$ (i. e., with probability tending to one as n tends to infinity).*

Combining these results enables us to show the following result which indicates that any reasonable clustering on the spectrumvectors will—apart from a small proportion of outliers— correctly assign the networks into clusters according to the underlying graph models.

Theorem 4. *Let $\mathcal{E} = \mathcal{E}(N, K, \Psi, \mathcal{G}_1, \ldots, \mathcal{G}_K)$ be a network ensemble in which the underlying graph models have a common number of vertices n_t. For each $\varepsilon > 0$ there exists $n_0 \in \mathbb{N}$ such that for $n_t \geq n_0$ we have for any instance of $\mathcal{E}$*

$$\|\lambda(G) - \lambda(G')\|_\infty < \varepsilon \cdot \|\lambda(H) - \lambda(H')\|_\infty$$

for any graphs G and G' drawn from the same model and any graphs H and H' drawn from different models, with probability in $1 - o(1)$.

Proof. The following assertions hold with high probability. By Corollary 2 it is $\|\lambda(G) - \lambda(G')\|_\infty \in \mathcal{O}(\sqrt{n_t})$. Let $\overline{M}$ be the expected adjacency matrix of H and $\overline{M'}$ be the expected adjacency matrix of H'. By Corollary 1 it is $\|\lambda(\overline{M}) - \lambda(\overline{M'})\|_\infty \in \Theta(n_t)$ and, again by Corollary 2 we have $\|\lambda(H) - \lambda(\overline{M})\|_\infty \in \mathcal{O}(\sqrt{n_t})$ and $\|\lambda(H') - \lambda(\overline{M'})\|_\infty \in \mathcal{O}(\sqrt{n_t})$. Together if follows $\|\lambda(H) - \lambda(H')\|_\infty \in \Theta(n_t)$ which implies that for sufficiently large n_t the inequality of the theorem is satisfied. □

Note that Theorem 4 makes only assertions for specific numbers of vertices of the form $n = t \cdot n_1$. However, this restriction is only necessary for notational simplification. In the next section we provide evidence by simulation that the spectra of the different graph clusters are well separated for all sufficiently large values of n. Furthermore this simulation indicates which values of n are sufficiently large for the theorems to hold.

1.5 Experimental Evidence

To estimate the tightness and expandability of our results we conducted experiments on artificially generated ensembles. Experiments on these examples split up in two major categories. The first is the case where we specify some simple role graphs on two nodes and try to determine the size needed to distinguish graphs drawn from these models by their spectra. For the second part of the study, role graphs were generated from random edge distributions and random group sizes. In all experiments graphs of different sizes were generated from each model and compared pairwise in terms of the $\| \ \|_\infty$ norm on their spectrum vectors. In the choice of graph sizes we did not restrict ourself to cases that make an exact matching of the group sizes possible but we also integrated graphs where group sizes can only be approximately established.

Although our analytical results apply only to exact matches of the group sizes, our experiments suggest that our method can be used, e.g., in a setting in which the group membership of each node is determined randomly from a distribution where the probability for membership in a class equals the relative class size in the model. This method was used in the experiment on prespecified models and in the second part of the experiment on random models. Here additionally the case of group sizes matched as exact as possible is examined.

The outcome of our experiments is a diagram showing distances between graphs of these examples. Here we distinguish the distances between graphs drawn from different models (points in grey) and those from the same model (in black). What we expect is that distances between graphs from different models grow faster than distances between graphs from identical models with growing graph sizes. The diagrams show the development of distances between graphs in ensembles for growing graph sizes. The size of the graphs is shown on the horizontal axis and the distance between the graphs appears on the vertical axis.

1.5.1 Prespecified Role Graphs

For illustrative purposes we start with some archetypical partition models that have been selected for their simplicity and good separation. Two edge probabilities $p = 0.2$ and $q = 10^{-3}$ are used and every possible symmetric edge distribution for a two-node graph on these values is generated. Excluding

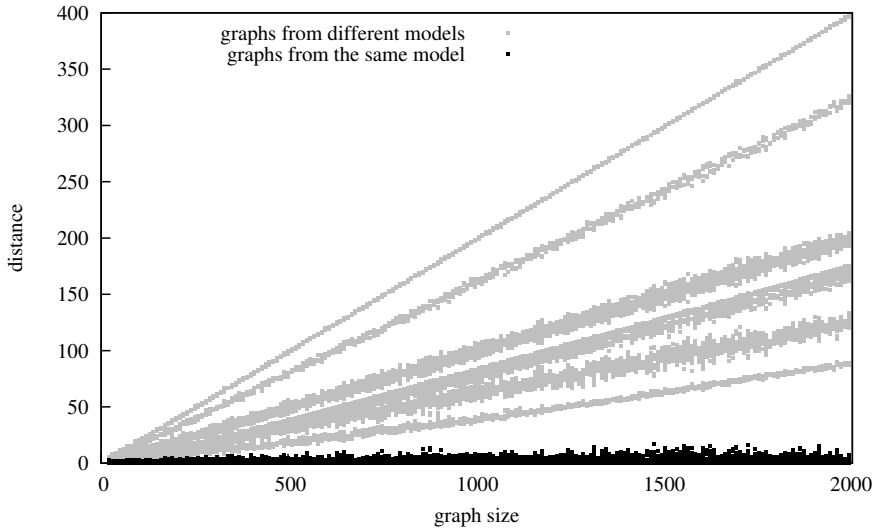


Fig. 1.1. Pairwise distances in ensembles generated for six prespecified role graphs with two nodes each. For ensembles with graphs of 300 vertices, a simple distance threshold separates classes well.

isomorphic role graphs this yields six edge distributions for us to cluster. For this experiment we chose a uniform class distribution and while generating instances from our models we do not try to match class sizes as exact as possible but rather assign vertices to classes uniformly at random. Ensembles consisting of graphs with size $10i$ where examined for $i = 2, \ldots, 100$. The main result can be seen in Fig. 1.1, which suggests that a clear separation by spectrum vectors should be possible for graphs having about 300 vertices. For random 200-node graphs we give three samples for each of the six models in Fig. 1.2; structural trends are clearly recognizable.

1.5.2 Random Role Graphs

For the next experiments we generated five role graphs with seven nodes each. For each partition model the desired class sizes and the edge distribution were drawn randomly and independent from a uniform distribution over $[0, 1]$. Basically each role graph consists of two random matrices, a $n \times 1$ matrix for node distribution to classes and a $n \times n$ matrix for the edge distribution. The graphs drawn from the different models are distinguished by their spectrum vectors and those again derive from the corresponding model.

As can be seen from the comparison in Fig. 1.3, the corresponding models do not differ much. A table with pairwise distances in supremum norm and an overview of these distances in a two dimensional layout obtained via multidimensional scaling quantify their relative shapelessness.

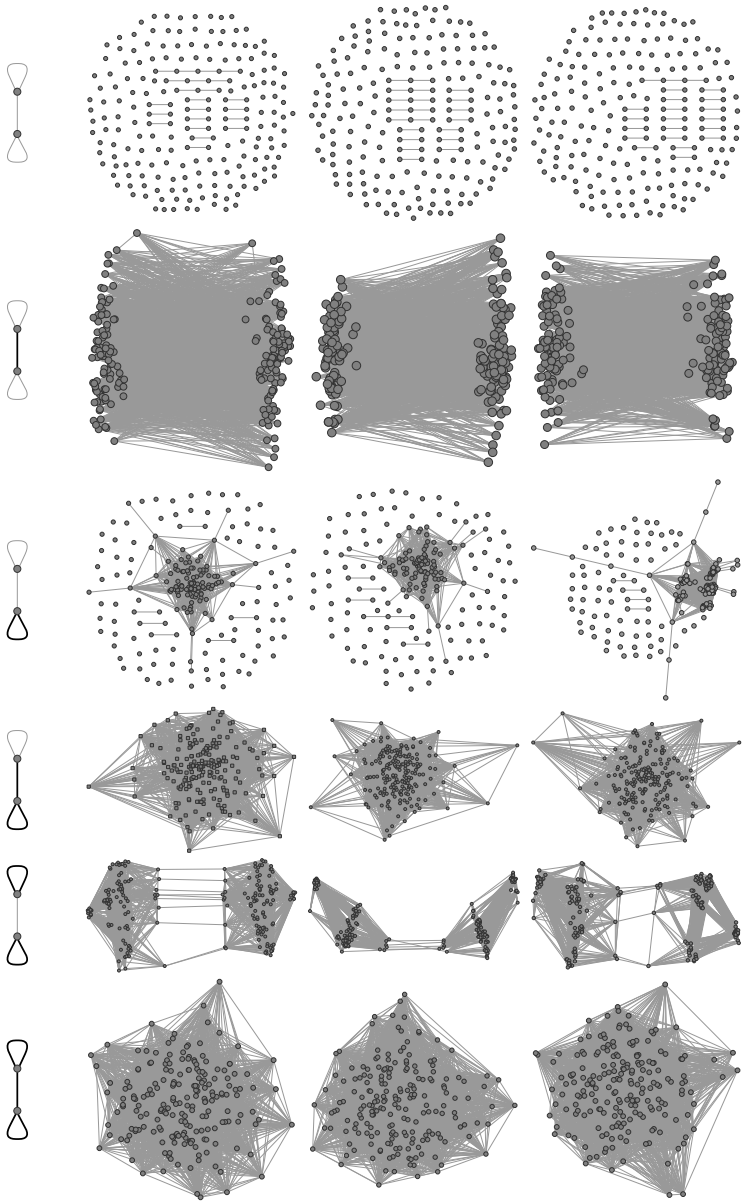
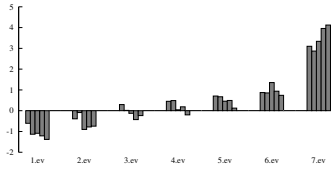


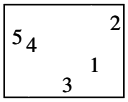
Fig. 1.2. Sample graphs with 200 vertices each from six prespecified two-node role graphs.



(a) spectra

model	2	3	4	5
1	0.528	0.509	0.859	1.022
2		0.810	1.089	1.252
3			0.619	0.782
4				0.388

(b) distances



(c) plot

Fig. 1.3. Spectra and pairwise distances of five randomly selected role graphs.

Since there are no pointed differences in these spectra, the sampled models can be considered quite typical. In particular, they form a classification instance much harder than the prespecified models used in the previous section. This is supported by additional experiments on different role graphs created in the same way and giving similar results, but not reported here.

For $i = 2, \ldots, 200$ ensembles were created consisting of five graphs with $10i$ vertices for each model, which gives us a sample ensemble with 25 graphs for every i. The difference in the two experiments lies in the assignment of vertices to classes. While in the first part it was tried to match partition sizes as exact as possible, in the second part the approach described above was used where desired partition sizes are used as a distribution.

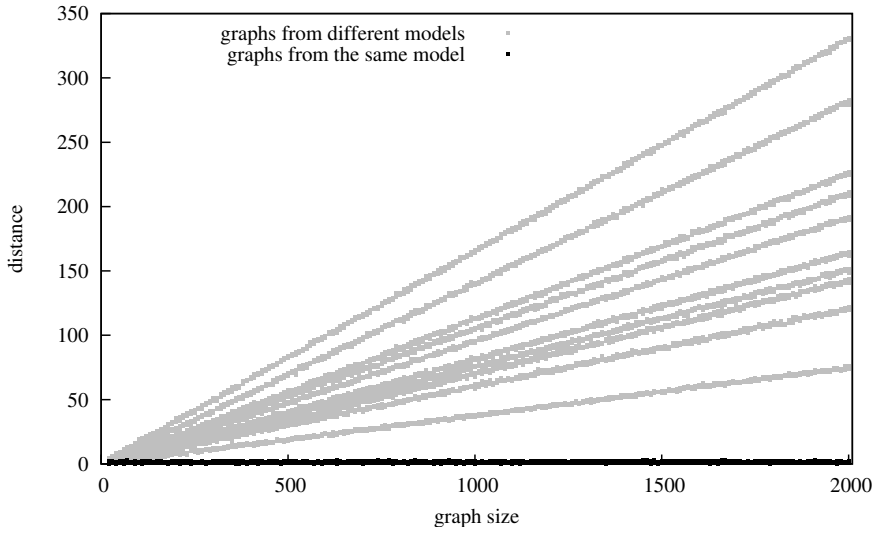


Fig. 1.4. Distance development with class sizes matched as accurate as possible.

Figure 1.4 shows how the distances between graphs drawn from different role graphs diverge from each other such that 10 different rays of dots can

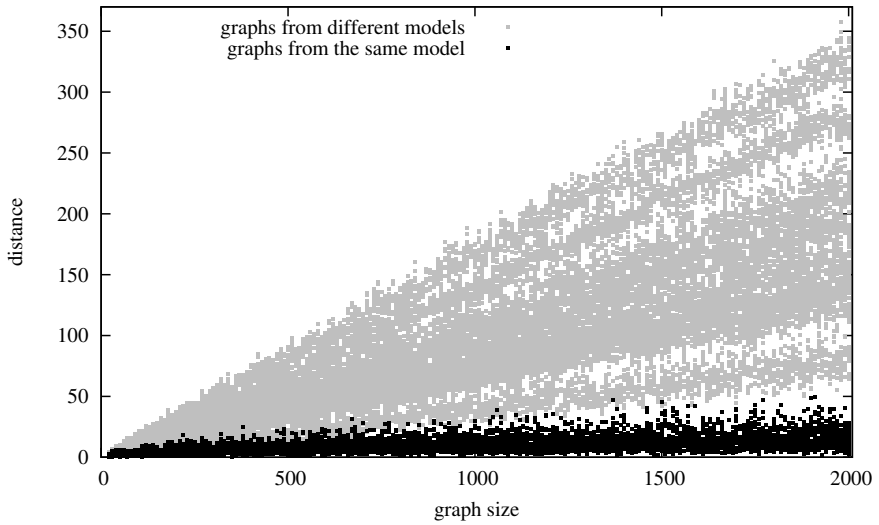


Fig. 1.5. Distance development with class sizes as distribution.

be seen, which is expected when the distances between the role graphs differ pairwise. Consider a graph with a node for each role graph and edge weights defined by the distance between adjacent role graphs measured as described above. The edge weights of this graph growing linearly in the number of vertices of the graphs the ensemble contains plus some random noise are the rays that can be seen in the diagram. The bottom line in black consists of distances between graphs drawn from the same role graph.

As an unexpected result the distances between graphs corresponding to the same role graph seem to be constant which could be a hint that the established borders are not tight.

The diagram in Fig. 1.5 shows how the divergence is weakened by inexact partition sizes. Compared to Fig. 1.4 a clear distinction between graphs drawn from different and those drawn from equal role graphs is achieved only with graphs having significantly more vertices, even though a trend towards clear separation can be observed.

1.6 Practical Evidence

In this section we want to demonstrate the performance of our graph distance in an application on real world data. We analyzed the edit networks of Wikipedia articles (see [5] for the definition of edit networks) and retrieved the expected result that average distances between networks with a supposed common structure are smaller than those with an expected difference in struc-

ture. For the analysis we randomly chose 60 articles with at least 1000 edits and 60 networks that were labeled 'featured' by the Wikipedia community.

From the edit logs of these articles a complete graph with a node for each author was created. Each edge was weighted by sums of negative edits between the adjacent authors. A negative edit occurs if either one author deletes words written by the other or if he restores words that were deleted by the other and is valued by the logarithm of the number of words deleted/restored. Since the edit graphs have in general different sizes we had to restrict the comparison to graphs having at least 500 vertices and vectors consisting of the 500 eigenvalues with biggest absolute value divided by the number of vertices. The number 500 was chosen since the differences are not expressed that clear with smaller values. For greater values the number of graphs being left is not meaningful for class comparison since noise and outliers could dominate the results.

The distance between classes was computed as the average of the pairwise distances between all graphs of the corresponding classes, while the distances between two graphs was measured as the above described distance on the spectrumvector of their weighted adjacency matrices.

The computations yield average distances of $21.7 \cdot 10^{-3}$ within the arbitrary chosen articles, $15.7 \cdot 10^{-3}$ in the class of featured articles and an average distance of $20.9 \cdot 10^{-3}$ between the two classes. As expected the featured articles tend toward a structure in their edit graphs that is common among this class and distinguishable from those of arbitrary articles. The fact that the inner class difference of arbitrary articles is higher than the distance to the featured articles can be easily explained by the fact that featured articles are a subclass.

This example represents an even more general case than the one where class memberships are a distribution. Here we have differently sized graphs and a statement on class sizes is impossible. Additionally we are not dealing with unweighted graphs anymore but with graphs having weighted edges. We tried to drop the weights by applying a threshold. Unfortunately in this scenario too much of the original information is lost and no separation between the classes can be seen at all. This drove us to use our method beyond proved effectiveness, on adjacency matrices of weighted graphs of different sizes. The obtained results support our decisions and encourage further examination of possible applications in this direction.

1.7 Conclusion

We introduced a spectral approach to identify groups of networks with similar role structure, i. e., networks that show the same pattern of connectivity among actor-classes, in network ensembles. We provided evidence for the usefulness of this method by probabilistic arguments (Sect. 1.4), by simulation results (Sect. 1.5), and by analyzing an ensemble of empirical networks

generated from the edit-history of sampled Wikipedia articles (Sect. 1.6). In previous work, network ensembles have often been described by other indicators such as density, degree sequences, or nearest neighbor connectivity (see e. g., [2]). Note that such approaches are not in competition, but orthogonal to our method, since they are based on different assumptions about the underlying ordering principle.

References

1. N. Alon, M. Krivelevich, and V. H. Vu. On the concentration of eigenvalues of random symmetric matrices. *Israel Journal of Mathematics*, 131(1):259–267, 2002.
2. G. Bianconi. The entropy of randomized network ensembles. *Europhysics Letters*, 81:28005, 2008.
3. S. P. Borgatti and M. G. Everett. Notions of position in social network analysis. *Sociological Methodology*, 22:1–35, 1992.
4. U. Brandes and T. Erlebach, editors. *Network Analysis*. Springer, 2005.
5. U. Brandes, P. Kenis, J. Lerner, and D. van Raaij. Network analysis of collaboration structure in Wikipedia. In *Proc. 18th Intl. World Wide Web Conf. (WWW2009)*, to appear.
6. U. Brandes, J. Lerner, M. J. Lubbers, C. McCarty, and J. L. Molina. Visual statistics for collections of clustered graphs. In *Proc. IEEE Pacific Visualization Symp. (PacificVis'08)*, pages 47–54, 2008.
7. C. T. Butts and K. M. Carley. Some simple algorithms for structural comparison. *Computational & Mathematical Organization Theory*, 11(4):291–305, 2005.
8. K. M. Carley, J.-S. Lee, and D. Krackhardt. Destabilizing networks. *Connections*, 24(3):79–92, 2002.
9. K. Faust. Comparing social networks: Size, density, and local structure. *Metodološki zvezki*, 3(2):185–216, 2006.
10. K. Faust and J. Skvoretz. Comparing networks across space and time, size and species. *Sociological Methodology*, 32(1):267–299, 2002.
11. J. Fiala and D. Paulusma. The computational complexity of the role assignment problem. In *Proc. Intl. Colloquium on Automata, Languages, and Programming (ICALP'03)*, pages 817–828, 2003.
12. B. Golub and M. O. Jackson. How homophily affects communication in networks. `http://arxiv.org/abs/0811.4013`, 2008.
13. E. A. Leicht, P. Holme, and M. E. J. Newman. Vertex similarity in networks. *Physical Review E*, 73, 2006.
14. J. Lerner. Role assignments. In Brandes and Erlebach [4], pages 216–252.
15. F. McSherry. Spectral partitioning of random graphs. In *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science (FOCS'01)*, pages 529–537, 2001.
16. G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Academic Press, 1990.
17. S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.
18. H. T. Welser, E. Gleave, D. Fisher, and M. Smith. Visualizing the signatures of social roles in online discussion groups. *Journal of Social Structure*, 8, 2007.