

# Classifying Networks with $dk$ -Series

---

**Marija Mitrović Dankulov<sup>1</sup>, Guido Caldarelli<sup>2</sup>, Santo Fortunato<sup>3</sup>,  
and Dmitri Krioukov<sup>4</sup>**

<sup>1</sup>Scientific Computing Laboratory, Center for the Study of Complex Systems, Institute of Physics Belgrade, University of Belgrade, Pregrevica 118, 11080 Belgrade, Serbia

<sup>2</sup>IMT Alti Studi, Lucca, Italy

<sup>3</sup>Center for Complex Networks and Systems Research, School of Informatics and Computing and Indiana University Network Science Institute (IUNI), Indiana University, Bloomington, USA

<sup>4</sup>Department of Physics, Department of Mathematics, Department of Electrical & Computer Engineering, Northeastern University, Boston, MA, USA

MMD was supported by the Ministry of Education, Science, and Technological Development of the Republic of Serbia under project ON171017

## 3.1 Introduction

The theory of complex networks provides powerful tools for studying complex systems in various disciplines such as biology, social sciences, computer sciences, mathematics, and physics [210]. One of the main research directions in network science studies the structural properties of networks and how they affect the dynamical processes and functions of systems represented by these various networks [17]. The standard assumption is that a self-organizing system should evolve to a network structure that makes these dynamical processes, or network functions, efficient [209, 198, 50]. Thus, understanding the network structure also reveals the mechanisms underlying the evolution of the system represented by the network.

Topological structure of networks can be characterized by a great number of various measures describing the system organization at different levels. Measures such as degree, average neighbor degree, clustering coefficient, concentrations of small subgraphs, betweenness, the distribution of shortest paths, and spectral properties have been used to describe in quantitative manner features that are characteristic of wide classes of complex networks. It has been shown that many complex networks have fat-tailed degree distributions [16], possess the small world property [308], and are often organized in

communities [96]. There is a common belief that the evolution of networks with similar structural properties is governed by the same mechanism. For instance, preferential attachment is often used for modeling networks with fat-tailed degree distributions [38]. Thus, classifying networks according to their topological properties is of great importance for identifying these mechanisms and better understanding the evolution, and indirectly the function and dynamics, of various complex systems.

However, classifying network via such graph measures is problematic, as there is no systematic way to determine which of them should be used. Besides, the measures are interdependent, that is, they positively or negatively correlate with each other in a complex way [303, 124, 289, 98, 63]. For these reasons, it is quite difficult to classify the structure of networks in a unique way using these topological measures and thus identify the evolution mechanisms characteristic for each class. For instance, the small world property has been found in many real networks, including social networks and interareal cortical networks in the primate brain. Yet in social networks, which are sparse graphs, this property is due to randomness in the linking patterns between the nodes, whereas in cortical networks it is trivially the consequence of network's high density. Therefore, the evolution of these two networks has been driven by different mechanisms, so that they cannot be assigned to the same network class.

One way to address the problem of interdependence among network properties is to find which of them are significant for a given network, and thus for its function. The standard procedure for the identification of a significant property  $X$  and its dependence on some other property  $Y$  is to generate a set of random graphs that have property  $Y$  but are random in all other respects, and then to check whether the property  $X$  is also characteristic of these graphs. If this is the case, then obviously property  $X$  is not interesting and relevant for the network function and dynamical processes running on it. We conclude that property  $X$  is a statistical consequence of property  $Y$ , and  $Y$  fully describes the structure of the network. Mechanisms generating network property  $Y$  can be thus considered to be relevant for the network evolution and dynamics. If  $X$  is not a typical property of these random graphs, one cannot conclude anything about the relevance of property  $X$ . The only conclusion that would follow from this is that property  $X$  is independent of property  $Y$ , but that does not mean that it is also independent of some other network property.

The identification of significant network features using the procedure described above raises another, equally important, question about the choice of null models. Since there are infinitely many network properties  $Y$ , there are infinitively many null models defined by property  $Y$ , and these can be used to test the statistical significance of any other property  $X$  [297]. For example, for most properties  $X$ , including motifs [195], their significance is tested with respect to random graphs with the same degree distribution. Although the choice of degree distribution as a  $Y$  property can seem natural, given the fundamental role played by it [210], there is no evidence that this choice of null model is less arbitrary than others. In general, there can be some other property which can be explanatory for both  $Y$  (here, the degree distribution) and  $X$ . Thus, one needs to identify the right reference property or properties  $Y$  in the null model that should be used for the testing of the (statistical) significance of property  $X$ .

In a recent work, Orsini *et al.* [219] proposed a way to identify such basic properties, enabling us to do a complete systematic description and unique classification of the structure of real networks. It is based on a set of properties known as the *dk-series* [182], a converging series of basic interdependent degree- and subgraph-based properties that characterize the local network structure at an increasing level of detail. It has been shown [182, 219] that the *dk-series* also defines a corresponding series of null models or random graph ensembles. These random graph models have exactly the same distribution of subgraphs of size  $d$  for all  $d$ -uples of nodes with degree  $(k_1, k_2, k_3, \dots, k_d)$  as in the real network. Or, to be precise, they are random graphs with fixed average degree, degree distribution, degree correlations, clustering, and so on. In Ref. [219], the authors used this methodology to quantify the randomness of six real single-layer networks, of very different function and dynamics. They showed that random graphs with fixed degree distribution, degree-degree correlations, average clustering, and degree-dependent average clustering reproduce all relevant topological properties for most networks. Here, we apply this approach to three networks and show that they differ in the randomness of their structure. We show that although many network properties can be reduced to specific degree- and subgraph-based characteristics, some of them cannot be explained with *dk-series*.

In recent years, a lot of attention in network science has been devoted to networks in which the same set of nodes are connected with multiple links of different types. These networks are referred to as multiplex or multilayer networks, since they consist of correlated single-layer networks composed of links of the same type. Many of the topological measures used to describe the structure of single-layer networks have been adapted in order to characterize the structure of layers and correlations between them [37]. We show how *dk-series* can be extended to describe in a systematic way the structure of multiplex networks using *dk*-annotated series [79].

## 3.2 *dk*-Series for single-layer networks

As indicated in the previous section, one needs to find an ordered set of reference properties of networks  $Y_0, Y_1, \dots$ , satisfying some criteria. The first criterion is *inclusiveness*: every subsequent property provides more details about the network structure than its predecessor. Formally, this is equivalent to the requirement that networks with property  $Y_d$ ,  $d > 0$ , should also have all properties prior to it, that is, all properties  $Y_{d'}$ , where  $0 \leq d' < d$ . The second criterion is *convergence*, that is, the minimal set of properties has to be finite, that is, the last property in series  $Y_D$  should fully characterize the adjacency matrix of any given graph. The  $Y$ -series that satisfies these conditions allows us to claim that, for any property  $X$  that is deemed important in a given real network, we can find a minimal  $d^*$  such that the property  $Y_{d^*}$  explains property  $X$ . The convergence of the series ensures the existence of some  $d^*$ , while the inclusiveness means that random networks with  $Y_d$  ( $d = (d^* + 1), \dots, D$ ) also have property  $X$ , so if we go to higher values of  $d$ , the random network has property  $X$  along with other significant properties. This enables the classification of network structure in a systematic manner. Several

approaches, including motifs [195], graphlets [314], and similar constructions [212], try to fully characterize the structure of networks by using relatively small set of properties, but they all violate the inclusiveness condition. On the other hand, one can still define many  $Y$ -series satisfying both conditions. We chose *dk-series* [182], the most natural choice due to their simplicity and the fact that they are a combination of subgraph- and degree-based characteristics of networks.

In these series, properties  $Y_d$  are *dk-distributions*. Each *dk*-distribution is actually a collection of distributions, stating how the subgraphs of size  $d$  are distributed over nodes with degrees  $k, k', k'', \dots, k^d$  in a graph  $G$ . Note here that the isomorphic subgraphs of  $G$  involving nodes of different degrees are thus counted separately. Specifically, the  $0k$  “distribution” is simply the average degree  $\bar{k} = \frac{2M}{N}$ , where  $N$  and  $M$  are the number of nodes and links in a given graph, while the  $1k$  distribution is the number of subgraphs of size 1, nodes, with the degree  $k$ , that is, the standard degree distribution

$$P(k) = \frac{N(k)}{N}, \quad (3.1)$$

where  $N(k)$  is the number of nodes of degree  $k$ . The  $2k$  distribution counts how many nodes of degrees  $k$  and  $k'$  are forming subgraphs with two nodes, and is known as joint degree matrix  $P(k, k')$

$$P(k, k') = \frac{\mu(k, k')M(k, k')}{2M}, \quad (3.2)$$

where

$$\mu(k, k') = \begin{cases} 2 & \text{if } k = k', \\ 1 & \text{otherwise.} \end{cases} \quad (3.3)$$

The  $3k$  distribution is a set of two distributions corresponding to two non-isomorphic subgraphs of size 3: wedges  $\wedge$  and triangles  $\Delta$ . It characterizes the connectivity patterns between triples of nodes of degrees  $k, k',$  and  $k''$ :

$$P_{\wedge}(k', k, k'') = \mu(k', k'') \frac{N_{\wedge}(k', k, k'')}{2W}, \quad (3.4)$$

$$P_{\Delta}(k, k', k'') = v(k, k', k'') \frac{N_{\Delta}(k, k', k'')}{6T}, \quad (3.5)$$

where  $W$  and  $T$  are the total numbers of wedges and triangles in the network, and

$$v(k, k', k'') = \begin{cases} 6 & \text{if } k = k' = k'', \\ 1 & \text{if } k \neq k' \neq k'', \\ 2 & \text{otherwise,} \end{cases} \quad (3.6)$$

so that both  $P_{\wedge}(k', k, k'')$  and  $P_{\Delta}(k, k', k'')$  are normalized, and  $\sum_{k, k', k''} P_{\wedge}(k', k, k'') = \sum_{k, k', k''} P_{\Delta}(k, k', k'') = 1$ . We could continue in a similar manner to obtain a  $4k$  distribution that consists of six distributions, each corresponding to one of the six non-isomorphic subgraphs of size 4 and so on until we reach  $d = N$ , the  $Nk$ -distribution, which characterizes the whole adjacency matrix of a given graph (see Figure 3.1).

The  $dk$ -series is directly related to some of the standard topological measures from complex network theory. Besides the already mentioned average degree and degree distribution, which are directly related to the  $0k$  and  $1k$  distributions, the  $2k$  distribution defines the node degree correlations in networks, or network's assortativity. The average neighbor degree  $\bar{k}_{nn}(k)$  is a projection of  $P(k, k')$  via

$$\bar{k}_{nn}(k) = \frac{\sum_{k'} k' P(k, k')}{\sum_{k'} P(k, k')}.$$

The average clustering coefficient  $\bar{c}$  and degree-dependent clustering coefficient  $\bar{c}(k)$  can be calculated based on the number of triangles in the network. Specifically,

$$\bar{c} = \frac{1}{N} \sum_i \frac{2\Delta_i}{k_i(k_i - 1)},$$

and

$$\bar{c}(k) = \frac{6T}{N} \frac{\sum_{k', k''} P_{\Delta}(k, k', k'')}{k(k-1)P(k)}, \quad (3.7)$$

where  $\Delta_i$  is the number of triangles composed of node  $i$  and its neighbors, while  $T$  is the total number of triangles. In general, the arbitrary  $dk$ -distribution characterizes both degree correlations between nodes at the hop distances  $d' < d$ , and the frequencies of  $d'$ -sized subgraphs,  $d' \leq d$ , in graph  $G$ .

One can easily see that  $dk$ -series is inclusive. The  $(d+1)k$ -distribution contains the same information about the network structure at the level  $d$  as the  $dk$ -distribution, plus some additional information about the degree correlations at the level  $d+1$ . Specifically, the  $1k$ -distribution defines the average degree ( $0k$ -distribution), via

$$\bar{k} = \sum_{k'} k' P(k'),$$

while the  $1k$ -distribution can be obtained from the  $2k$  distribution as

$$P(k) = \frac{\bar{k}}{k} \sum_{k'} P(k, k').$$

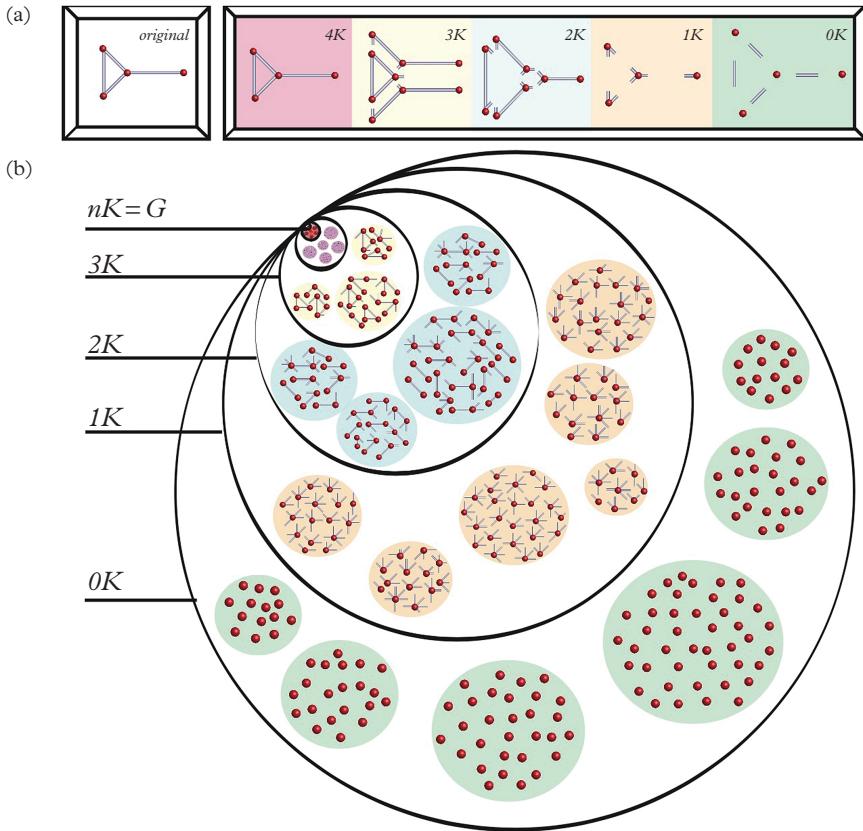
Similarly, the  $3k$ -distribution defines the  $2k$ -distribution by

$$\begin{aligned} P(k, k') = & \frac{1}{k+k'-2} \sum_{k''} \left\{ \frac{6T}{M} P_{\Delta}(k, k', k'') \right. \\ & \left. + \frac{W}{M} [P_{\wedge}(k', k, k'') + P_{\wedge}(k, k', k'')] \right\}. \end{aligned} \quad (3.8)$$

The opposite does not hold, that is, knowing the  $dk$ -distribution will not allow one to infer anything about the  $(d+1)k$ -distribution, meaning that higher values of  $d$  correspond to a greater level of details about the network structure.

The number of non-isomorphic subgraphs, and thus the number of distributions needed to characterize network topology at the level  $d$ , grows exponentially with  $d$ ; hence, the calculation of  $dk$ -series becomes a computationally intensive task for higher values of  $d$ . One could argue that just counting the number of  $d$ -sized subgraphs in a given network regardless of their node degrees should be enough for the description of network structure in a systematic manner [195, 314, 212]. The subgraph-based series obtained from the count of  $d$ -subgraphs without including information about the degree, which we can call a  $d$ -series, satisfies the convergence condition, and the statistics for  $d = N$  subgraphs would also fully describe the topology of a given network, but, unlike  $dk$ -series, they are not inclusive. Careful analysis of the first four elements of a  $d$ -series clearly demonstrates its non-inclusiveness. For the  $d$ -series, the zeroth element is not defined, while the number of nodes  $N$  and number of edges  $M$  are the properties corresponding to  $d = 1$  and  $d = 2$ , respectively. These two quantities are independent of each other, that is, knowing the number of edges does not allow one to tell much about the number of nodes in a network. Similarly, the properties at the level  $d = 3$ , the number of triangles  $T$  and wedges  $W$ , define neither the size nor the density of the network [219]. This analysis demonstrates that the elements in the  $d$ -series are independent of each other and that each of them conveys a different kind of information about network topology.

Figure 3.1(b) illustrates the inclusiveness and convergence of  $dk$ -series, and also suggests that all graphs with  $N$  nodes and  $M$  edges constitute a set of random graphs  $\mathcal{G}_{0k} = \mathcal{G}_{N,M}$  with the same  $0k$  property. Graphs with the same degree sequence, the  $1k$  property, form a smaller set of graphs  $\mathcal{G}_{1k}$ , which is a subset of  $\mathcal{G}_{0k}$ , and so on. Each set of graphs with a given  $dk$ -distribution, known as *dk-graphs*, is at the same time a subset of  $(d-1)k$ - and a superset of  $(d+1)k$ -graphs. It follows from this that a sequence of  $dk$ -distributions defines a sequence of random graph ensembles (null models). In order to compare a real network with the random graphs which have the same  $dk$ -properties, and thus quantify the randomness of its structure of the real network, one needs a maximum entropy ensemble of these graphs or *dk-random graphs* [182]. All graphs in *dk-random graphs* have equal sampling probability  $P(G) = 1/\mathcal{N}_d$ , where  $\mathcal{N}_d$  is the number of *dk-graphs*. Each collection of  $dk$ -distributions is more informative about the network structure and thus more constraining than  $(d-1)k$ -distributions, that is,  $\mathcal{N}_0 \geq \mathcal{N}_1 \geq \dots \geq \mathcal{N}_N = 1$ . The size of the final ensemble at level  $N$  is clearly equal to 1, since it just contains the network with the exact adjacency matrix. The number  $\mathcal{N}_d$  is too



**Figure 3.1** The illustration of dk-series. (a) The dk-distributions for a graph of size 4. The 4k-distribution is the graph itself. The 3k-distribution consists of its three subgraphs of size 3: one triangle connecting nodes of degrees 2, 2, and 3, and two wedges connecting nodes of degrees 2, 3, and 1. The 2k-distribution is the joint degree distribution in the graph. It specifies the number of links (subgraphs of size 2) connecting nodes of different degrees: one link connects nodes of degrees 2 and 2, two links connect nodes of degrees 2 and 3, and one link connects nodes of degree 3 and 1. The 1k-distribution is the degree distribution in the graph. It lists the number of nodes (subgraphs of size 1) of different degree: one node of degree 1, two nodes of degree 2, and one node of degree 3. The 0k-distribution is just the average degree in the graph, which is 2. (b) The inclusiveness and convergence of the dk-series are illustrated via the hierarchy of dk-graphs, which are graphs having the same dk-distribution of a given graph  $G$  of size  $N$ . From Ref. [219].

large, especially for small values of  $d$  (exact or approximate calculations for  $d = 0, 1, 2$  can be found in [32, 20]), making the construction of the whole set of  $dk$ -graphs impossible. Thus, one needs to sample  $dk$ -random graphs uniformly in order to be able to compare them with a given real network and properly test the significance of different topological properties.

The  $dk$ -series and  $dk$ -random graphs enable the systematic and full characterization of the structure of any real network by finding the value of  $d$  for which all  $d'k$ -distributions for  $d' > d$  do not contain any additional information of the network structure. This means that any topology metric one can define on network  $G$  is captured with  $dk$ -random graphs. The convergence and inclusiveness properties of  $dk$ -series ensure that this value  $d$  exists, that is, they guarantee that any network property  $X$  of any given network  $G$  can be reproduced with any desired accuracy by high-enough  $d$ . Clearly, all properties are reproduced exactly for  $d = N$ , but the question is whether there is a value  $d < N$  for which all relevant topological properties of a given network are captured with  $dk$ -random graphs. By finding this value  $d$ , one also quantifies the randomness of the structure of a given network. The entropy of  $dk$ -ensembles is  $S_d = \ln \mathcal{N}_d$ , and it is a nonincreasing function of  $d$ , that is, the  $dk$ -random graphs are *less random and more structured*, the higher  $d$  is. In the following section, we demonstrate how one can classify single-layer networks based on their  $dk$ -randomness by applying the procedure described in Ref. [219] to three real networks.

### 3.2.1 Classifying single-layer networks based on their $dk$ -randomness

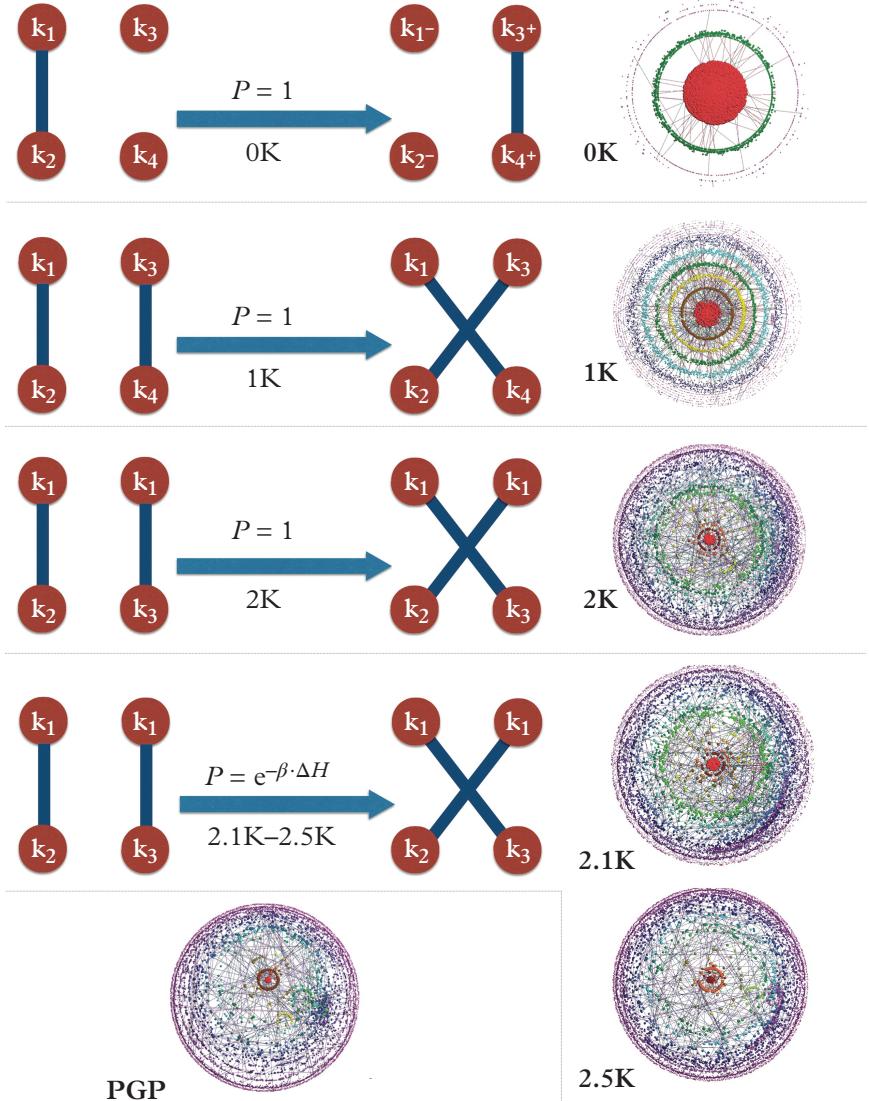
First, we briefly discuss the constructibility of  $dk$ -random graphs, and the problem of sampling graphs uniformly at random from the sets of  $dk$ -graphs. Here, we emphasize that, in  $dk$ -graphs, the  $dk$ -distribution constraints are sharp, that is, all graphs in  $dk$ -graphs set have exactly the same  $dk$ -distribution. Given a real network  $G$ , there exist two ways to sample  $dk$ -random graphs:  $dk$ -randomize  $G$ , generalizing the randomization algorithms in Refs. [186, 187], or construct random graphs with  $G$ 's  $dk$ -sequence from scratch [182, 117], also called direct construction [148, 76, 147, 22]. We chose the first option,  $dk$ -randomization, due to its simplicity and the existence of algorithms that enable the uniform sampling of  $dk$ -random graphs for values of  $d$  greater than 2 (see the detailed discussions about construction algorithms in [182, 219]).

The  $dk$ -randomization is an edge-swapping procedure where pairs of edges are swapped at random, starting from  $G$ , such that the  $dk$ -distribution is preserved at each swap. Figure 3.2 illustrates permitted swaps of edges for each  $dk$ -distribution. Specifically, to preserve  $0k$ -distribution, average degree, we disconnect a pair of nodes and connect two other, non-neighboring, nodes. The graphs obtained in this procedure are Erdős-Rényi graphs  $\mathcal{G}_{N,M}$  of fixed size  $N$  and average degree  $2M/N$ . To preserve the degree sequence ( $1k$ -distribution), we chose at random a pair of edges and swapped their targeting nodes, while, for the  $2k$ -distribution, we swapped edge pairs only if there were at least two nodes of equal degrees adjacent to different edges belonging to this pair. Allowed  $3k$ -swaps are then  $2k$ -swaps that preserve  $3k$ -distribution, the same connectivity patterns between the triplets of nodes with respect to node degrees. From this and the inclusiveness of  $dk$ -series, it follows that  $(d+1)k$ -swaps form a subset of  $dk$ -swaps for  $d > 0$  [182]. During all these rewiring procedures, the edge swapping is only allowed if it does not lead to the creation of multiple edges between the same pair of nodes.

There are many concerns regarding the described rewiring procedure [316], two of which are particularly important: (1) the ergodicity of the rewiring process, that is, whether any two pairs of graphs with the same  $dk$ -properties are connected with a chain of  $dk$ -swaps; (2) the uniformity of the rewiring process, that is, how close to uniform sampling the  $dk$ -swap Markov chain is after its mixing time is reached. It has been shown that  $dk$ -swapping processes for values of  $d = 1, 2$  are ergodic [186, 187, 67], while it is common belief that there is no ergodic edge swapping, of any type, that preserves the  $3k$ -distribution, and thus  $dk$ -distributions for values  $d \geq 4$ , although a rigorous proof of this is lacking at the moment [219]. When it comes to the uniformity of the  $dk$ -swapping process for  $d = 0, 1, 2$ , it has been shown that if the edge-swap process is done correctly, then the sampling is uniform [2, 11].

Since the  $2k$ -random graphs do not capture all topological properties for most of the tested real networks [137, 219], we need algorithms that will allow us to go beyond preserving only  $2k$ -properties. The  $dk$ -targeting  $d'k$ -preserving rewiring, where  $d' < d$ , has proven to be a good choice for generating random graphs with the same  $dk$ -properties as in the considered real network [182, 219]. This procedure incorporates the following modification of the  $d'k$ -rewiring algorithm: the  $d'k$ -swap is accepted with probability  $\min(1, \exp(-\beta\Delta H))$ , where  $\beta$  is the inverse temperature of this simulated annealing process, and  $\Delta H$  is the change in the  $L^1$  distance between the  $dk$ -distribution in the current graph and the targeted  $dk$ -distribution before and after the swap. The numerical experiments with  $3k$ -targeting rewiring have shown that this process does not converge for most real networks [219], due to the extremely constraining nature of the  $3k$ -distribution. Therefore, it is reasonable to retreat to numeric investigations of  $2k$ -random graphs in which, in addition to the  $2k$ -distribution, some substatistics of the  $3k$ -distribution are fixed. In particular, we consider  $2.1k$ -random graphs, which have the same  $2k$ -distribution and value of the average clustering coefficient  $\bar{c}$  as the given real network, and  $2.5k$ -random graphs with the same  $2k$ -properties and average clustering coefficient  $\bar{c}(k)$  of nodes of degree  $k$  [117]. Since  $2.1k$ - and  $2.5k$ -statistics are fully defined by the  $3k$ -distribution, and  $2.1k$  is defined by  $2.5k$ , the  $3k$ -random graphs comprise a subset of  $2.5k$ -random graphs, which are, in turn, subsets of  $2.1k$ -random graphs, that is,  $\mathcal{N}_2 > \mathcal{N}_{2.1} > \mathcal{N}_{2.5} > \mathcal{N}_3$ . As a consequence, if a certain topological property of real networks is captured by  $2.5k$ -random graphs, it will be also captured by  $3k$ -random graphs, while the opposite is not generally true.

The scheme of algorithm(s) that we use for creating a set of  $dk$ -random graphs for  $d = 0, 1, 2, 2.1, 2.5$  is given in Figure 3.2, while their detailed description and a link to a Web page with publicly available software can be found in Ref. [75]. The  $dk$ -random graphs for  $d = 0, 1, 2$  are created using the standard  $dk$ -swapping described above. Although it is known that these procedures for general graphs do not lead to a uniform sampling of  $dk$ -random graphs, unlike their modified versions [2, 11], it has been shown that, for power-law distributions, the obtained sample of uniform graphs is very close to uniform. To generate  $dk$ -random graphs for  $d = 2.1, 2.5$ , we start with a  $2k$ -random graph and apply to it a described  $2k$ -preserving  $2.xk$ -targeting ( $x = 1, 5$ ) rewiring process (see Figure 3.2). For this, we use a modified version of the algorithm [63, 219], which ensures the convergence for all networks.

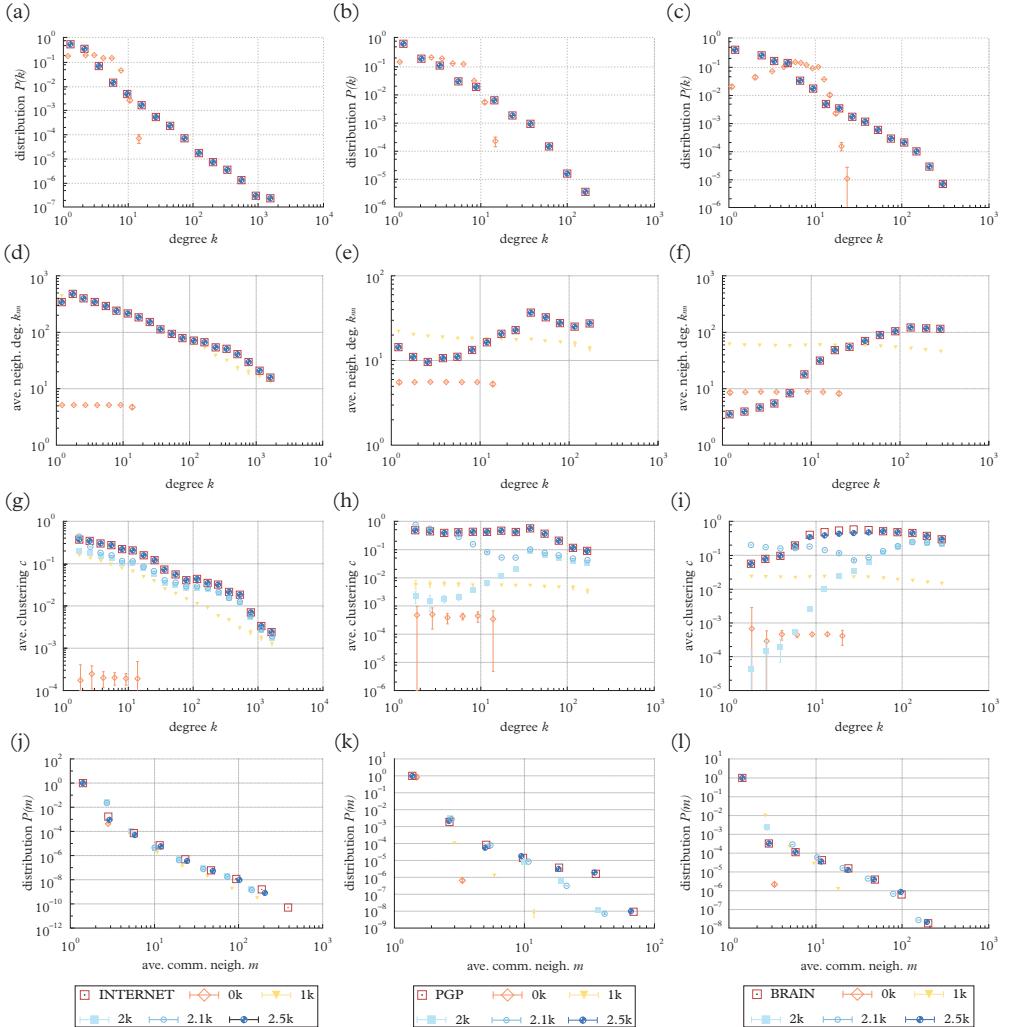


**Figure 3.2** dk-sampling and convergence of dk-series. The left column shows the elementary swaps of dk-randomizing (for  $d = 0, 1, 2$ ) and dk-targeting (for  $d = 2.1, 2.5$ ) rewiring. The nodes are labeled by their degrees, and the arrows are labeled by the rewiring acceptance probability. In dk-randomizing rewiring, random (pairs of) edges are rewired preserving the graph's dk-distribution (and consequently its  $d'K$ -distributions for all  $d' < d$ ). In 2.1k- and 2.5k-targeting rewiring, the moves preserve the 2k-distribution, but each move is accepted with probability  $p$  designed to drive the graph closer to a target value of average clustering  $\bar{c}$  (2.1k) or degree-dependent clustering  $c(k)$  (2.5k):  $p = \min(1, e^{-\beta \cdot \Delta H})$ , where  $\beta$  is the inverse temperature of this simulated annealing process,  $\Delta H = H_a - H_b$ , and  $H_{a,b}$  are the distances, after and before the move, between the current and target values of clustering:  $H_{2.1k} = |\bar{c}_{current} - \bar{c}_{target}|$  and  $H_{2.5k} = \sum_i |\bar{c}_{current}[k_i] - \bar{c}_{target}[k_i]|$ . The right column shows LaNet-vi [29] visualizations of the results of these dk-rewriting processes, applied to the Pretty Good Privacy (PGP) network, visualized at the bottom of the left column. The node sizes are proportional to the logarithm of their degrees, while the color reflects node coreness [29]. As  $d$  grows, the shown dk-random graphs quickly become more similar to the real PGP network. From Ref. [219].

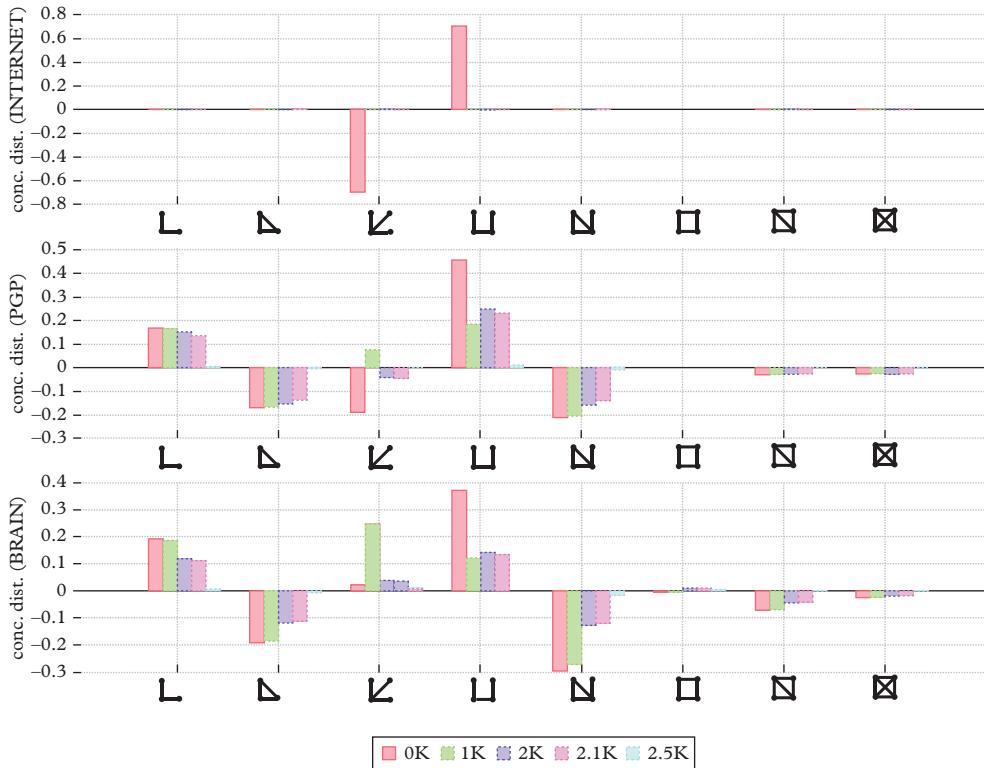
To quantify the randomness of the real network, that is, to determine the value of  $d$  for which  $dk$ -random graphs capture most of its topological properties, we adopt the following procedure. For a given real network, we calculate its average degree, degree distribution, degree correlations, average clustering coefficient, and averaging clustering coefficient of nodes of degree  $k$ ; then, based on this, we generate 20  $dk$ -random graphs, using the methodology described in the previous paragraph, for  $d = 0, 1, 2, 2.1, 2.5$ . Then, for each sample, we compute a variety of network properties and compare their values with the corresponding ones obtained for the real network. The value of  $d$  for which the considered properties of  $dk$ -random graphs are in reasonable agreement with the ones of a real network determines the randomness of its structure. The higher the value of  $d$ , the more structured and less random a given network is.

In Ref. [219] the authors performed an extensive set of numeric experiments with six real, very different networks with respect to their function. Here, we demonstrate the described procedure by applying it to three of these networks: the Internet at the level of autonomous systems (INTERNET) [183], a technosocial web of trust among users of the distributed Pretty Good Privacy (PGP) cryptosystem [39], and a functional MRI (fMRI) map of the human brain (BRAIN) [85]. In the first network, INTERNET, the nodes are so-called autonomous systems (ASs; organizations owning parts of the Internet infrastructure), and there is a link between two ASs if they have a business relationship in which they exchange Internet traffic. The nodes in the second network we consider, PGP, are users' PGP certificates, while the edges denote the existence of trust between two users. We consider here only the largest connected component of the PGP network. The third network considered, BRAIN, is the largest component of an fMRI map of the human brain, where voxels (representing small areas of a resting brain, approximately  $36 \text{ mm}^3$  in volume) are represented with nodes, and an edge exists between two voxels if the correlation coefficient of the fMRI activity of the voxels exceeds 0.7. We chose these three networks because they have different values of  $d$  for which  $dk$ -random graphs capture their structural properties [219]. In particular, most of the considered properties for INTERNET are reproduced with  $2k$ -random graphs, which makes it the most random network among the three networks, while, to reproduce the same properties of the PGP network, we need graphs with preserved  $2.5k$ -distribution. Some of the properties of the BRAIN network are not reproduced even with  $2.5k$ -random graphs, meaning that this network is the least random one among these three networks (Figures 3.3–3.7 and Tables 3.1 and 3.2).

The properties that we use to compare the structure of real networks with the ones of random graphs can be divided into three categories: microscopic, mesoscopic, and macroscopic. The microscopic properties describe the networks' structure at the level of individual nodes and subgraphs of small sizes (see Figures 3.3 and 3.4). Some of these properties, namely, average degree, degree distribution, average degree of nearest neighbors, average clustering coefficient, and average clustering coefficient of nodes of degree  $k$ , are fixed by the corresponding  $dk$ -distributions. On the other hand, the concentration of subgraphs of size 3 and 4 [219], as well as the distribution of the number of common neighbors shared by a pair of nodes, are not fixed by  $dk$ -distributions for  $d < 3$ . The distribution of common neighbors equals the probability that two connected



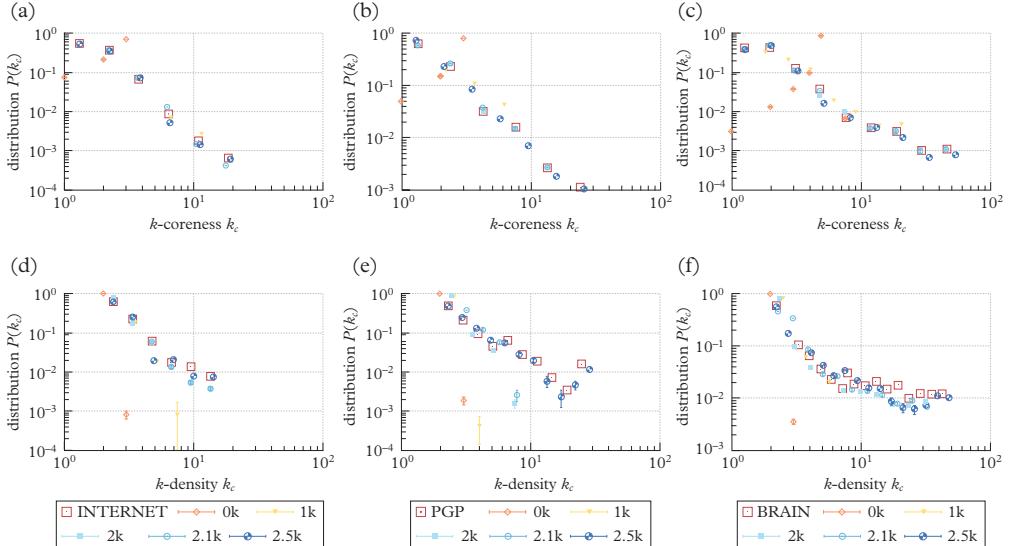
**Figure 3.3** Microscopic properties of real complex networks and their dk-random graphs. The first nine panels show topological properties fixed by dk-distributions: the degree distribution  $P(k)$  for (a) INTERNET, (b) PGP, and (c) BRAIN; the average degree of nearest neighbors  $\bar{k}_{nn}(k)$  for (d) INTERNET, (e) PGP, and (f) BRAIN; and the degree-dependent average clustering coefficient  $\bar{c}(k)$  for (g) INTERNET, (h) PGP, and (i) BRAIN. The last three panels show the distribution of the number of common neighbors  $P(m)$ , which is not fixed by dk-distributions for  $d < 3$ : (j) INTERNET, (k) PGP, and (l) BRAIN. Adapted from Ref. [219].



**Figure 3.4** Density of subgraphs of sizes 3 and 4. The average absolute difference between the subgraph concentration in the  $dk$ -random graphs and its concentration in the real networks: INTERNET (top), PGP (middle), and BRAIN (bottom). Adapted from Ref. [219].

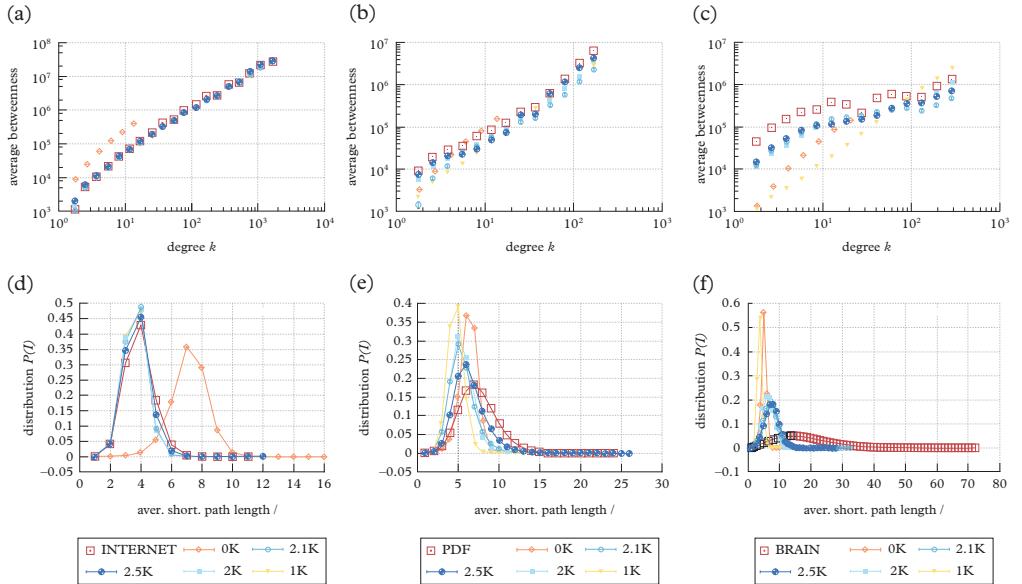
nodes have  $m$  common neighbors and is exactly fixed by the  $3k$ -distribution. Mesoscopic properties depend on both local and global network organization. Here, we consider  $k$ -coreness [9] and  $k$ -density [250] (see Figure 3.5). A node has  $k$ -coreness equal to  $k$  if it belongs to  $k$ -core of the original graph, which is the largest induced subgraph of graph in which every node has degree at least  $k$ . Similarly, an edge has  $k$ -densemess equal to  $k$  if it belongs to the largest induced subgraph of the original graph in which all edges have multiplicity at least  $k$ . Macroscopic properties are truly global: betweenness, the distribution of hop lengths of shortest paths, and spectral properties (see Figure 3.6 and Tables 3.1 and 3.2). We measure the distance between real and  $dk$ -random graphs with Kolmogorov–Smirnov distances between the distributions of all the considered properties (see Figure 3.7).

For all three networks and for most of the considered properties, we observe a nice convergence as  $d$  increases, that is, there is no statistically significant difference between the property in the real network and in its  $2.5k$ -random graphs. Although this is expected



**Figure 3.5** Mesoscopic properties:  $k$ -coreness and  $k$ -density distributions. The upper panels show the  $k$ -coreness of (a) INTERNET, (b) PGP, and (c) BRAIN networks, and the lower panels show their  $k$ -density: (e) INTERNET, (f) PGP, and (g) BRAIN. The  $k_c$ -core of a graph  $G$  is the maximal subgraph of  $G$  in which all nodes have degree at least  $k_c$ . A node has  $k$ -coreness  $k_c$  if it belongs to the  $k_c$ -core but not to the  $(k_c + 1)$ -core. The  $k_c$ -dense subgraph is the maximal subgraph of a graph in which all edges have multiplicity  $(k_c - 2)$ ; the multiplicity of an edge is the number of triangles the edge is part of. Adapted from Ref. [219].

for microscopic properties that are fixed with  $dk$ -distributions, there is no reason to expect convergence in the case of small subgraph frequencies, in the distribution of the number of common neighbors, or for mesoscopic or macroscopic properties. Figure 3.4 shows that the relative difference between subgraph frequencies in real and  $2.5k$ -random graphs is very close to zero for subgraphs of sizes 3 and 4. For INTERNET, this property is already captured with  $1k$ -graphs, while it is clear that, for the BRAIN and PGP networks, one needs to fix the degree-dependent clustering coefficient in order to observe the same motif count as in the real systems. Mesoscopic properties are reproduced with  $2.5k$ -graphs for the BRAIN and PGP networks, and with  $2k$ -random graphs for the INTERNET network (see Figure 3.5). While betweenness and average shortest-path distance require  $1k$ - and  $2.5k$ -random graphs for INTERNET and PGP, respectively, such properties are not captured even with  $2.5k$  graphs for BRAIN (see Figure 3.6). Table 3.1 shows that the largest eigenvalue of the adjacency matrix is closely, but not exactly, reproduced by  $d = 2.5$  for all three networks. The spectral gap, the difference between the largest and second largest eigenvalue of the adjacency matrix, given in Table 3.2, shows that  $2k$ - and  $2.1k$ -random graphs are better connected and interlinked, compared to real networks. Figure 3.7 shows that the Kolmogorov–Smirnov distances calculated for the distributions of INTERNET and PGP are either zero or



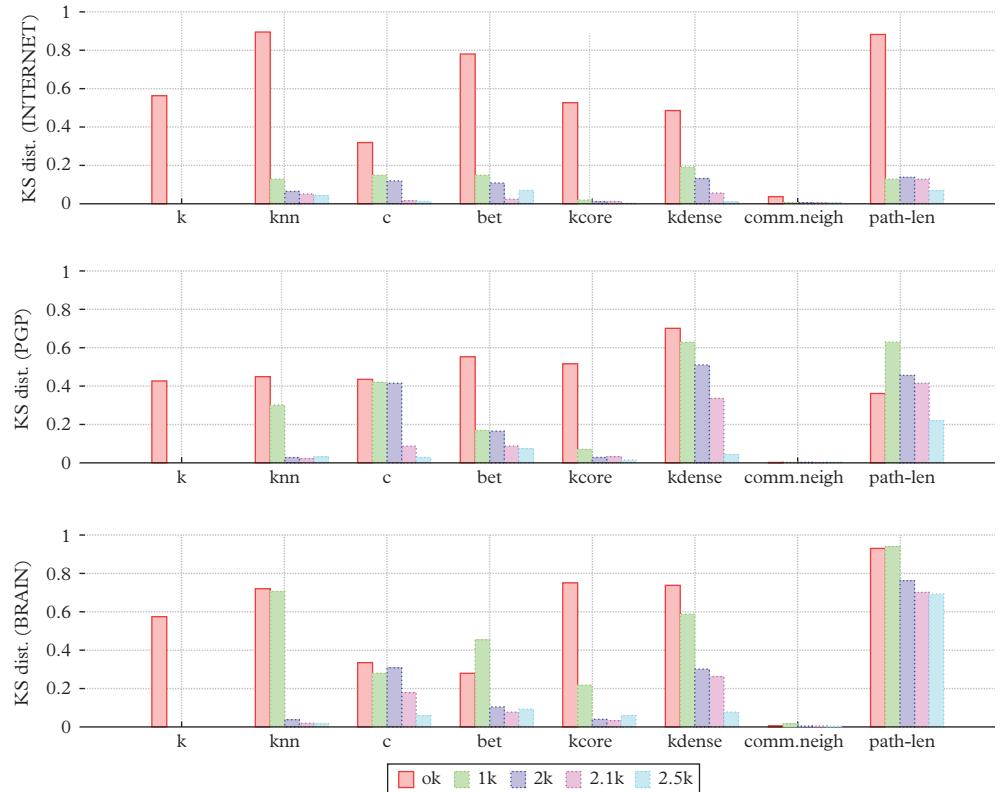
**Figure 3.6** Macroscopic properties: betweenness and average shortest hop distance. The average betweenness  $\bar{b}(k)$  of nodes of degree  $k$  is shown in the upper panels: (a) INTERNET, (b) PGP, and (c) BRAIN. The lower panels show the distribution  $P(l)$  of the length  $l$  of the shortest paths between all pairs of nodes: (d) INTERNET, (e) PGP, and (f) BRAIN. Adapted from Ref. [219].

**Table 3.1** Largest eigenvalues of the adjacency matrix for the three networks considered and their corresponding  $dk$ -graphs. For the latter, we show averages across different realizations for each  $d$ , and their standard deviations in parentheses.

	Original	$0k$	$1k$	$2k$	$2.1k$	$2.5k$
INTERNET	67.17	5.36 (0.01)	56.02 (0.33)	61.15 (0.03)	61.32 (0.06)	65.34 (0.10)
PGP	42.44	5.77 (0.02)	19.50 (0.24)	34.08 (0.03)	34.40 (0.05)	42.95 (0.12)
BRAIN	119.66	8.91 (0.01)	54.89 (0.26)	113.41 (0.02)	114.09 (0.06)	122.27 (0.20)

**Table 3.2** Spectral gap between the largest and the second-largest eigenvalues of the adjacency matrix. For the  $dk$ -graphs, the shown values are the averages across different realizations for each  $d$ , while their standard deviations are reported in parentheses.

	Original	$0k$	$1k$	$2k$	$2.1k$	$2.5k$
INTERNET	17.56	0.70 (0.05)	14.94 (0.53)	18.83 (0.07)	18.55 (0.11)	19.53 (0.25)
PGP	4.25	0.98 (0.04)	5.51 (0.31)	18.01 (0.18)	17.55 (0.21)	4.71 (0.19)
BRAIN	40.97	2.90 (0.06)	35.52 (0.31)	77.53 (0.11)	76.59 (0.27)	42.71 (0.35)



**Figure 3.7** Kolmogorov–Smirnov distance between real networks and their dk-random graphs. The Kolmogorov–Smirnov (KS) distances between the distributions of per-node values of a given property in the real networks, and the same distributions in their dk-random graphs for the following properties: degree ( $k$ ), average degree of nearest neighbors ( $\text{knn}$ ), clustering coefficient ( $c$ ), number of common neighbors ( $\text{comm.neigh}$ ), k-coreness ( $\text{kcore}$ ), k-density ( $\text{kdense}$ ), betweenness ( $\text{bet}$ ), and shortest-path distance ( $\text{path-len}$ ). Adapted from Ref. [219].

very close to zero for higher values of  $d$ , indicating that these two networks can be well approximated with  $2k$ - and  $2.5k$ -random graphs. On the other hand, for BRAIN, the global properties exhibit slow or no convergence at all, so it is an outlier, that is, its properties can be captured with dk-random graphs with  $d \geq 3$ . Although many properties can be reproduced with  $2.5k$ -graphs, we find that community structure is not preserved for any of these networks, regardless of the community detection algorithm.

### 3.3 dk-Series in multilayer networks

In this section, we discuss how  $dk$ -series generalize to multilayer networks in general and to multiplex networks in particular. The key idea behind the generalized  $dk$ -series

is the same as in the case of monolayer networks: a  $dk$ -series is a series of inclusive and convergent statistics based on the frequencies of degree-labeled subgraphs of increasing size in a given network.

### 3.3.1 Multilayer networks

Here, we consider the most general case of multilayer networks with the most detailed form of generalized  $dk$ -statistics.

The structure of a multilayer network is fully specified by the adjacency tensor  $A_{\alpha i, \beta j}$ , where indices  $\alpha, \beta = 1, \dots, L$  indicate layers, while  $i, j = 1, \dots, N$  indicate nodes:  $A_{\alpha i, \beta j} = 1$  if node  $i$  at layer  $\alpha$  is connected to node  $j$  at layer  $\beta$ , and  $A_{\alpha i, \beta j} = 0$  otherwise [72] (see also Chapter 1). To simplify the notation, from now on, we assume that networks are undirected and have no loops— $A_{\alpha i, \beta j} = A_{\beta j, \alpha i}$ , and  $A_{\alpha i, \alpha i} = 0$ , for any combination of  $\alpha, \beta, i$ , and  $j$ —but the generalization to directed networks is straightforward.

The  $0k$ -statistics, that is, the number of edges in the network, and the  $0k$ -“distribution,” that is, the average degree, are no longer scalar  $M$  and  $\bar{k}$  as in the monolayer case, but the  $L \times L$ -matrices

$$\hat{M} \equiv M_{\alpha\beta} = \sum_{i \leq j} A_{\alpha i, \beta j}, \quad (3.9)$$

$$\bar{k} \equiv \bar{k}_{\alpha\beta} = \frac{2M_{\alpha\beta}}{N}, \quad (3.10)$$

specifying the number of interlayer edges between layers  $\alpha$  and  $\beta$  if  $\alpha \neq \beta$ , or the number of intralayer edges at layer  $\alpha$  if  $\alpha = \beta$ . Similarly, the degree of node  $i$  is no longer scalar but the  $L \times L$ -matrix

$$\hat{k}_i \equiv (k_i)_{\alpha\beta} = \sum_j A_{\alpha i, \beta j} \quad (3.11)$$

specifying the number of node's connections to other nodes in the same layer if  $\alpha = \beta$ , or to nodes in other layers if  $\alpha \neq \beta$ . If node  $i$  is not present at layer  $\alpha$ , then  $(k_i)_{\alpha\beta} = 0$ . Node degrees are thus also matrices  $\hat{k} = k_{\alpha\beta}$ . A node that has this degree has  $k_{\alpha\beta}$  connections from layer  $\alpha$  to other nodes at layer  $\beta$ , which can be equal to  $\alpha$ . We note that the degree matrices are not, in general, symmetric, even if the network is undirected.

The  $1k$ -statistics, that is, the number of nodes  $N(\hat{k})$  with degree  $\hat{k}$ , and the  $1k$ -distribution are then

$$N(\hat{k}) = \sum_i \delta(\hat{k}_i, \hat{k}), \quad (3.12)$$

$$P(\hat{k}) = \frac{N(\hat{k})}{N}, \quad (3.13)$$

where  $\delta$  stands for the Kronecker delta, and the distribution is properly normalized:  $\sum_{\hat{k}} P(\hat{k}) = 1$ . In contrast with the monolayer case, the degree distribution is no longer a univariate distribution but a multivariate joint distribution of  $L^2$  variables. In particular, this distribution contains all the information on the correlation of degrees of the same node at different layers. As in the monolayer case, the  $1k$ -distribution fully defines the  $0k$ -distribution via

$$\bar{\hat{k}} = \sum_{\hat{k}} \hat{k} P(\hat{k}). \quad (3.14)$$

The  $2k$ -statistics, that is, the number of links  $N_{\alpha\alpha'}(\hat{k}, \hat{k}')$  between nodes of degrees  $\hat{k}$  and  $\hat{k}'$  at layers  $\alpha$  and  $\alpha'$ , and the corresponding  $2k$ -distribution are given by the matrices

$$\hat{N}(\hat{k}, \hat{k}') \equiv N_{\alpha\alpha'}(\hat{k}, \hat{k}') = \sum_{i \leq i'} A_{\alpha i, \alpha' i'} \delta(\hat{k}_i, \hat{k}) \delta(\hat{k}_{i'}, \hat{k}'), \quad (3.15)$$

$$\hat{P}(\hat{k}, \hat{k}') = \mu(\hat{k}, \hat{k}') \frac{\hat{N}(\hat{k}, \hat{k}')}{2\hat{M}}, \text{ where} \quad (3.16)$$

$$\mu(\hat{k}, \hat{k}') = \begin{cases} 2, & \text{if } \hat{k} = \hat{k}', \\ 1, & \text{otherwise,} \end{cases} \quad (3.17)$$

is the factor taking care of proper normalization  $\sum_{\hat{k}, \hat{k}'} P_{\alpha\alpha'}(\hat{k}, \hat{k}') = 1$  for any  $\alpha, \alpha'$ . Here and below, all vector, matrix, and tensor multiplication and divisions are element-wise, for example,  $(\hat{N}/\hat{M})_{\alpha\alpha'} = \hat{N}_{\alpha\alpha'}/\hat{M}_{\alpha\alpha'}$ . Instead of a joint distribution of two variables  $P(k, k')$  in the monolayer case, we deal with  $L(L - 1)$  joint distributions of  $2L^2$  variables  $P_{\alpha\alpha'}(\hat{k}, \hat{k}')$ . These distributions contain strictly more information about degree correlations than the  $1k$ -distribution does. In particular, in addition to capturing the degree correlations of the same node ( $i = j$ ) across layers, they also encompass all the degree correlations of distinct connected nodes ( $i \neq j$ ) across both intralayer ( $\alpha = \alpha'$ ) and interlayer ( $\alpha \neq \alpha'$ ) connections. The  $2k$ -distributions define the  $1k$ -distribution similarly to the monolayer case:

$$P(\hat{k}) = \frac{\bar{\hat{k}}}{\hat{k}} \sum_{\hat{k}'} \hat{P}(\hat{k}, \hat{k}'). \quad (3.18)$$

It is evident from the expressions above that the  $dk$ -series in multilayer networks are different from the  $dk$ -series in monolayer networks only in that the scalar number of edges and node degrees are replaced by the  $L \times L$ -matrices  $M \rightarrow \hat{M}$  and  $k \rightarrow \hat{k}$ , while their  $dk$ -distributions form tensors of rank  $d$  whose indices are layers. If  $d = 1$ , this tensor is trivial:  $\mathbf{P}(\hat{k}) \equiv P_\alpha(\hat{k}) = P(\hat{k})$  for any layer  $\alpha$ ; but, starting with  $d = 2$ , any two elements of these  $dk$ -distribution tensors can, in general, be different, specifying in the  $d = 2$  case,

for instance, the degree correlations across pairs of different layers if  $\alpha \neq \alpha'$ , or within the same layer if  $\alpha = \alpha'$ . Each element of these  $dk$ -tensors is a joint distribution of  $d$  degrees, that is, of  $dL^2$  variables.

All higher-order statistics and distributions are then defined exactly as in the monolayer case, albeit with these two modifications. For instance, the  $3k$ -distribution is defined by Eqs (3.4)–(3.5) and determines the  $2k$ -distributions via Eq. (3.8), except that scalar degrees in these equations are replaced by degree matrices, and the numbers of wedges and triangles have three indices specifying to which layers the three nodes forming these two subgraphs belong. As in the monolayer case, higher- $d$   $dk$ -distributions determine the degree correlations of nodes at distance  $d - 1$ , the frequencies of  $d$ -cliques, including clustering at  $d = 3$ , and so on.

### 3.3.2 Multiplex networks

In node-aligned multiplex networks, all nodes are present in all layers, and all interlayer connections are trivial: every node is connected only to all its copies in all other layers, so that interlayer connections form  $N$  disjoint  $L$ -cliques:  $A_{\alpha i, \beta i} = 1$  for any combination of  $i$  and  $\alpha \neq \beta$ . The  $dk$ -series can therefore be excused from keeping track of statistics of interlayer connections, which somewhat simplifies the formalism in the previous section, as described below. This simplification boils down to per-layer projections of the most detailed  $dk$ -statistics discussed in the previous section.

Since in multiplex networks  $k_{\alpha\beta} = L - 1$  for all nodes and all  $\beta \neq \alpha$ , all the off-diagonal components of degree matrices  $\hat{k}$  are not informative and can thus be dropped, mapping degree  $L \times L$ -matrices to degree  $L$ -vectors composed of the diagonal elements of  $\hat{k}$ :  $\hat{k} \mapsto \mathbf{k} \equiv \text{diag}(\hat{k})$ . The  $\alpha$ -component  $k_\alpha$  of this vector  $\mathbf{k}$  specifies the number of intralayer connections of a node at layer  $\alpha$ . Similarly, the number-of-edges matrix  $\hat{M}$  maps to the vector  $\mathbf{M} \equiv \text{diag}(M)$ , whose components  $M_\alpha$  are the numbers of intralayer edges within layer  $\alpha$ .

Similarly, it is convenient to project the  $dk$ -distribution tensors per layer, forming vectors of distributions  $\mathbf{P}(\mathbf{k}, \mathbf{k}', \dots) \equiv P_\alpha(\mathbf{k}, \mathbf{k}', \dots)$ , consisting of the diagonal elements of the full distribution tensor  $P_{\alpha, \alpha', \dots}(\mathbf{k}, \mathbf{k}', \dots)$ , that is,  $P_\alpha(\mathbf{k}, \mathbf{k}', \dots) = P_{\alpha, \alpha, \dots}(\mathbf{k}, \mathbf{k}', \dots)$ , thus keeping track only of intralayer correlations of degrees of different nodes. The correlations of the degrees of the same node at different layers are still contained in  $P(\mathbf{k})$ .

Given this simplified representation, the  $dk$ -statistics, distributions, and their relations are exactly as in the general multilayer case, except that all matrices and tensors are replaced by vectors whose components are layers. For  $d = 0, 1, 2$ , for instance, we have the following expressions:

$$\bar{\mathbf{k}} = \frac{2\mathbf{M}}{N}, \quad (3.19)$$

$$N(\mathbf{k}) = \sum_i \delta(\mathbf{k}_i, \mathbf{k}), \quad (3.20)$$

$$P(\mathbf{k}) = \frac{N(\mathbf{k})}{N}, \quad (3.21)$$

$$\bar{k} = \sum_k k P(k), \quad (3.22)$$

$$N(\mathbf{k}, \mathbf{k}') \equiv N_\alpha(\mathbf{k}, \mathbf{k}') = \sum_{i \leq i'} A_{\alpha i, \alpha i'} \delta(\mathbf{k}_i, \mathbf{k}) \delta(\mathbf{k}'_{i'}, \mathbf{k}'), \quad (3.23)$$

$$\mathbf{P}(\mathbf{k}, \mathbf{k}') = \mu(\mathbf{k}, \mathbf{k}') \frac{N(\mathbf{k}, \mathbf{k}')}{2M}, \quad (3.24)$$

$$P(\mathbf{k}) = \frac{\bar{k}}{k} \sum_{\mathbf{k}'} \mathbf{P}(\mathbf{k}, \mathbf{k}'), \quad (3.25)$$

which all are lists of  $L$  per-layer standard monolayer expressions, except that degrees are vectors. Compared to general multilayer networks, the  $dk$ -distributions in multiplex networks with these simplifications are all  $L$ -vectors, for  $d > 1$ , whose components consists of joint distributions of  $d$  degrees, that is, of  $dL$  variables.

### 3.3.3 Application to real networks

The general methodology behind the application of  $dk$ -series generalized to multilayer networks is the same as in the monolayer case discussed in Section 3.2. Yet, one has to keep in mind that multilayer  $dk$ -statistics tend to be extremely sparse and thus extremely constraining, even in the multiplex-projected case. This is because, compared to monolayer networks, the  $dk$ -distributions in multilayer networks contain much more detailed information about degrees, which are no longer scalars but matrices or vectors, and about their correlations within subgraphs of different sizes. Therefore, it is usually convenient to consider summary statistics of these distributions and define graph randomization procedures based on those. These procedures may depend on a particular choice of real network, on its specifics, and on particular questions one is to answer about the network.

For instance, in Ref. [79], generalized  $dk$ -series were applied to the Internet at the AS level. This network is a multiplex network with two layers. One layer consists of directed customer-provider links, for which, in order to send traffic over them, customer ASs must pay provider ASs, while the other layer consists of undirected peer-to-peer links connecting mostly large Internet service provider ASs, which exchange traffic free of charge over these links, based on bilateral agreements. The specific question addressed for this network was how to generate synthetic random graphs of varying sizes that reproduce specific types of degree correlations that reflect realities of business relationships between ASs in the Internet. For instance, peer-to-peer links tend to exist only between large Internet providers of large AS degree. Large providers tend to have large number of customers, a handful of peers, and few or no providers. Small customer ASs have no customers, no peers, and a small number of providers, and so on.

To properly capture these correlations, three joint distributions were considered. One was the full  $1k$ -distribution  $P(\mathbf{k})$  specifying the correlations among the numbers of customer, provider, and peer connections that nodes have. Since the

customer-provider layer is a directed network, this distribution is a joint distribution of three variables: in- and out-degrees  $k_{1,in}$  and  $k_{1,out}$  in the customer-provider layer, and degrees  $k_2$  in the peer-to-peer layer. The other two distributions were the  $2k$ -distributions of the total degrees  $k = \sum_\alpha k_\alpha = k_{1,in} + k_{1,out} + k_2 = |\mathbf{k}|_1$  in the two layers. These distributions are projections of the multiplex  $2k$ -distributions:  $\mathbf{P}(k, k') = \sum_{\mathbf{k}, \mathbf{k}'} \mathbf{P}(\mathbf{k}, \mathbf{k}') \delta(|\mathbf{k}|_1, k) \delta(|\mathbf{k}'|_1, k')$ . After these distributions were determined from data from the real Internet data, the  $dk$ -series were used to generate synthetic graphs of any size reproducing all the degree correlations contained in these distributions by first computing the marginals of these distributions  $P(k_{1,in})$ ,  $P(k_{1,out})$ ,  $P(k_2)$ , and  $P(k)$ , and the three copulas [206] representing their correlations in their joint distributions  $P(\mathbf{k}) = P(k_{1,in}, k_{1,out}, k_2)$ ,  $P_1(k, k')$ , and  $P_2(k, k')$ . Joint degree sequences of varying lengths were then sampled from these copulas, and random graphs were constructed using stub-matching procedures. As expected, the degree correlations in these random graphs reproduced the degree correlations in the real Internet. Many other important structural properties of the Internet, including properties specific to the Internet, were reproduced by these  $2k$ -random graphs as well [79], corroborating the finding that the Internet is nearly  $2k$ -random with respect to many important properties [182, 219].

We conclude this section by reiterating that, when classifying, that is, determining “how random” a given multilayer or multiplex network is, the full matrix-degree-based  $dk$ -series provides a rich set of inclusive and convergent statistics, which contain a variety of summary statistics as different projections of the full  $dk$ -distributions. Any combination of these statistics constrained to their values observed in a given network defines a null random graph model, in which any structural property of the network can be tested on its typicality in the model. There seems to be no good-for-all-networks rule of what these projections are, as different multilayer networks may require different projections. Yet, the full joint  $1k$ -distribution should most likely be always considered, while per-layer projections of  $dk$ -distributions with  $d > 1$  are likely to be good projection choices for many networks, especially multiplex ones.

### 3.4 Discussion and conclusion

Topological measures commonly used for characterizing the structure of complex networks are interdependent, but the relation and the extent of their mutual correlations are often unknown. For this reason, systematic classification of networks via standard topological measures is not feasible. Here, we show how the problem of interdependence can be overcome by finding the set of base properties that can be used to explain all other relevant topological features of a network’s structure. We describe the methodology proposed in Ref. [219] and show that most topological properties, which are deemed relevant for dynamics and function of networks, can be reproduced by random graphs with fixed degree distribution, degree-degree correlations, an average clustering coefficient and a degree-dependent average clustering coefficient, as in a given real network.

There is no reason to expect that non-local properties, namely, mesoscopic and macroscopic properties, cannot be reproduced by random graphs with local constraints.

And, for some networks and some properties, this is true. Our numerical experiments show that global features of brain networks, for example, the shortest-path length and betweenness distributions, differ drastically between the original network and  $dk$ -random graphs. This suggests that brain network evolution was subjected to some global constraints, which is reflected in its structure. The human brain consists of two weakly connected parts, corresponding to two brain hemispheres, a feature that cannot be reproduced with  $dk$ -random graphs with small  $d$ . In general,  $dk$ -random graphs with fixed local properties can not reproduce the community structure of all complex networks studied in Ref. [219], that is, the cluster organization is not robust to  $dk$ -randomization.

On the other hand, INTERNET and PGP, and other networks considered in Ref. [219], are clearly  $dk$ -random, with  $d \leq 2.5$ . Our analysis shows that the most basic properties of these networks, including microscopic, mesoscopic, and macroscopic ones, are a consequence of several local  $dk$ -properties: degree distribution, degree-degree correlations, and global and degree-dependent average clustering coefficients. This implies that the evolution of these networks was dominated by local dynamical rules and that it can be explained to a certain extent by mechanisms that are responsible for the manifestation of specific  $dk$ -properties. There already exists a multitude of approaches [80, 151, 302, 10, 225, 34] proposing different mechanisms to explain the emergence of these local topological properties. Clearly, the features that cannot be reproduced by  $dk$ -random graphs require separate explanations, or maybe some other set of base properties and different systems of null models.

The most basic topological features considered in this work can be considered non-significant, that is, there exists a  $dk$ -property captured by the corresponding  $dk$ -random graphs. In general, to tell how statistically significant a particular feature is, one needs to compare this feature in a real network with the same feature in an ensemble of random graphs, that is, a null model. The choice of the null model is free, but one should be careful when choosing the null model, since the significance of a certain feature is strongly dependent on it. The  $dk$ -random graphs discussed in this chapter can be used for determining the right network topology generator. One should first check whether most topological features of networks can be reproduced in  $dk$ -random graphs with a low value of  $d$ . If this is the case, then one may not need any sophisticated mission-specific topology generators. The proposed extension of  $dk$ -series to multilayer graphs enables the use of similar procedures on wider classes of real networks.

There are certain drawbacks of our approach that have to be mentioned. First, we do not have a proof that the proposed  $dk$ -random graph generation algorithms for  $d = 2.1$  and  $d = 2.5$  sample graphs uniformly at random from the ensemble. Second, it is known that the random graph ensembles and edge-rewiring processes employed here suffer from problems such as degeneracy and hysteresis [97, 239, 133]. The ideal solution would be to calculate analytically the expected value of given property in an ensemble. For this, we need an analytical description of null models which is currently only available for soft  $d = 0, 1, 2$ -random graph models [281, 282, 59, 58]. Unfortunately, the null models for generation of random graphs ensembles with constraints  $d > 2$  are still not feasible, and they appear to be beyond the reach in the near future. We also

lack algorithms for the generation of  $dk$ -random graphs for multiplex networks which would allow us to apply similar procedures and quantify the randomness of multilayer networks. Clearly, the solution of these problems will be of great importance for a full understanding of the relationship between the structure, function, and dynamics of real networks.