Abstract

Represented as graphs, real networks are intricate combinations of order and disorder. Fixing some of the structural properties of network models to their values observed in real networks, many other properties appear as statistical consequences of these fixed observables, plus randomness in other respects. Here we employ the $dk$-series, a complete set of basic characteristics of the network structure, to study the statistical dependencies between different network properties. We consider six real networks—the Internet, US airport network, human protein interactions,
technosocial web of trust, English word network, and an fMRI map of the human brain—and find that many important local and global structural properties of these networks are closely reproduced by $dk$-random graphs whose degree distributions, degree correlations, and clustering are as in the corresponding real network. We discuss important conceptual, methodological, and practical implications of this evaluation of network randomness.

Introduction

Network science studies complex systems by representing them as networks [1–3]. This approach has proven quite fruitful because in many cases the network representation achieves a practically useful balance between simplicity and realism: while always grand simplifications of real systems, networks often encode some crucial information about the system. Represented as a network, the system structure is fully specified by the network adjacency matrix, or the list of connections, perhaps enriched with some additional attributes. This (possibly weighted) matrix is then a starting point of research in network science.

One significant line of this research studies various (statistical) properties of adjacency matrices of real networks. The focus is often on properties that convey useful information about the global network structure that affects the dynamical processes in the system that this network represents [4, 5]. A common belief is that a self-organizing system should evolve to a network structure that makes these dynamical processes, or network functions, efficient [6–11]. If this is the case, then given a real network, we may “reverse engineer” it by showing that its structure optimizes its function. In that respect the problem of interdependency between different properties becomes particularly important [12–16].

Indeed, suppose that the structure of some real network has property $X$—some statistically over- or under-represented subgraph, or motif [17, 18], for example—that we believe is related to a particular network function. Suppose also that the same network has in addition property $Y$—some specific degree distribution or clustering, for example—and that all networks that have property $Y$ necessarily have property $X$ as a consequence. Property $Y$ thus enforces or “explains” property $X$, and attempts to “explain” $X$ by itself, ignoring $Y$, are misguided. For example, if a network has high density (property $Y$), such as the interarial cortical network in the primate brain where 66% of edges that could exist do exist [19, 20], then it will necessarily have short path lengths and high clustering, meaning it is a small-world network (properties $X$). However, unlike social networks where the small-world property is an independent feature of the network, in the brain this property is a simple consequence of high density.

The problem of interdependencies among network properties has been long understood [21, 22]. The standard way to address it, is to generate many graphs that have property $Y$ and that are random in all other respects—let us call them $Y$-random graphs—and then to check if property $X$ is a typical property of these
Y-random graphs. In other words, this procedure checks if graphs that are sampled uniformly at random from the set of all graphs that have property Y, also have property X with high probability. For example, if graphs are sampled from the set of graphs with high enough edge density, then all sampled graphs will be small worlds. If this is the case, then X is not an interesting property of the considered network, because the fact that the network has property X is a statistical consequence of that it also has property Y. In this case we should attempt to explain Y rather than X. In case X is not a typical property of Y-random graphs, one cannot really conclude that property X is interesting or important (for some network functions). The only conclusion one can make is that Y cannot explain X, which does not mean however that there is no other property Z from which X follows.

In view of this inherent and unavoidable relativism with respect to a null model, the problem of structure-function relationship requires an answer to the following question in the first place: what is the right base property or properties Y in the null model (Y-random graphs) that we should choose to study the (statistical) significance of a given property X in a given network [23]. For most properties X including motifs [17, 18], the choice of Y is often just the degree distribution. That is, one usually checks if X is present in random graphs with the same degree distribution as in the real network. Given that scale-free degree distributions are indeed the striking and important features of many real networks [1–3], this null model choice seems natural, but there are no rigorous and successful attempts to justify it. The reason is simple: the choice cannot be rigorously justified because there is nothing special about the degree distribution—it is one of infinitely many ways to specify a null model.

These observations instruct one to look not for a single base property Y, which cannot be unique or universal, but for a systematic series of base properties Y_0, Y_1, ... . By “systematic” we mean the following conditions:

1. Inclusiveness: the properties in the series should provide strictly more detailed information about the network structure, which is equivalent to requiring that networks that have property Y_d (Y_d-random graphs), d > 0, should also have properties Y_{d'} for all d' = 0, 1, ..., d - 1.

2. Convergence: there should exist property Y_D in the series that fully characterizes the adjacency matrix of any given network, which is equivalent to requiring that Y_D-random graphs is only one graph—the given network itself.

If these Y-series satisfy the conditions above, then whatever property X is deemed important now or later in whatever real network, we can always standardize the problem of explanation of X by reformulating it as the following question: what is the minimal value of d in the above Y-series such that property Y_d explains X? By convergence, such d should exist; and by inclusiveness, networks that have property Y_{d'} with any d' = d, d + 1, ..., D, also have property X. Assuming that properties Y_d are once explained, the described procedure provides an explanation of any other property of interest X.
Yet one can still define many different $Y$-series satisfying the two conditions above. Some further criteria are needed to focus on a particular one. The criteria that we use to select a particular $Y$-series in this study are simplicity and the importance of subgraph- and degree-based statistics in networks. Indeed, in the network representation of a system, subgraphs, their frequency and convergence are the most natural and basic building blocks of the system, among other things forming the basis of the rigorous theory of graph family limits known as graphons [24], while the degree is the most natural and basic property of individual nodes in the network. Combining the subgraph- and degree-based characteristics leads to $dk$-series [25].

**Results**

$dk$-series

In $dk$-series properties $Y_d$ are $dk$-distributions. For any given network $G$ of size $N$, its $dk$-distribution is defined as a collection of distributions of $G$’s subgraphs of size $d = 0, 1, \ldots, N$ in which nodes are labeled by their degrees in $G$. That is, two isomorphic subgraphs of $G$ involving nodes of different degrees—for instance, edges ($d = 2$) connecting nodes of degrees 1, 2 and 2, 2—are counted separately. The $0k$-“distribution” is defined as the average degree of $G$. Figure[1] illustrates the $dk$-distributions of a graph of size 4.

Thus defined the $dk$-series subsumes all the basic degree-based characteristics of networks of increasing detail. The zeroth element in the series, the $0k$-“distribution,” is the coarsest characteristic, the average degree. The next element, the $1k$-distribution, is the standard degree distribution, which is the number of nodes—subgraphs of size 1—of degree $k$ in the network. The second element, the $2k$-distribution, is the joint degree distribution, the number of subgraphs of size 2—edges—between nodes of degrees $k_1$ and $k_2$. The $2k$-distribution thus defines 2-node degree correlations and network’s assortativity. For $d = 3$, the two non-isomorphic subgraphs are triangles and wedges, composed of nodes of degrees $k_1$, $k_2$, and $k_3$, which defines clustering, and so on. For arbitrary $d$ the $dk$-distribution characterizes the ‘$d$’degree ‘$k$’ correlations in $d$-sized subgraphs, thus including, on the one hand, the correlations of degrees of nodes located at hop distances below $d$, and, on the other hand, the statistics of $d$-cliques in $G$. We will also consider $dk$-distributions with fractional $d \in (2, 3)$ which in addition to specifying 2-node degree correlations ($d = 2$), fix some $d = 3$ substatistics related to clustering.

The $dk$-series is inclusive because the $(d+1)k$-distribution contains the same information about the network as the $dk$-distribution, plus some additional information. In the simplest $d = 0$ case for example, the degree distribution $P(k)$ (1k-distribution) defines the average degree $\bar{k}$ (0k-distribution) via $\bar{k} = \sum_k kP(k)$. The analogous expression for $d = 1, 2$ are derived in Supplementary Information, Network Properties Section.

It is important to note that if we omit the degree information, and just count
Figure 1: The $dk$-series illustrated. Panel (a) shows 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. Panel (b) illustrates the inclusiveness and convergence of $dk$-series by showing the hierarchy of $dk$-graphs, which are graphs that have the same $dk$-distribution as a given graph $G$ of size $n$. The black circles schematically shows the sets of $dk$-graphs. The set of $0k$-graphs, i.e., graphs that have the same average degree as $G$, is largest. Graphs in this set may have a structure drastically different from $G$’s. The set of $1k$-graphs is a subset of $0k$-graphs, because each graph with the same degree distribution as in $G$ has also the same average degree as $G$, but not vice versa. As a consequence, typical $1k$-graphs, i.e., $1k$-random graphs, are more similar to $G$ than $0k$-graphs. The set of $2k$-graphs is a subset of $1k$-graphs, also containing $G$. As $d$ increases, the circles become smaller because the number of different $dk$-graphs decreases. Since all the $dk$-graph sets contain $G$, the circles “zoom-in” on it, and while their number decreases, $dk$-graphs become increasingly more similar to $G$. In the $d = n$ limit, the set of $nk$-graphs consists of only one element, $G$ itself.
the number of $d$-sized subgraphs in a given network regardless their node degrees, as in motifs 17–18, graphlets 20–27, or similar constructions 28. Then such degree-agnostic $d$-series (versus $d_k$-series) would not be inclusive (Supplementary Information, Subgraph-based series (d-series) vs. $d_k$-series Section). Therefore preserving the node degree (‘$k$’) information is necessary to make a subgraph-based (‘$d$’) series inclusive. The $d_k$-series is clearly convergent because at $d = N$ where $N$ is the network size, the $N_k$-distribution fully specifies the network adjacency matrix.

A sequence of $d_k$-distributions then defines a sequence of random graph ensembles (null models). The $d_k$-graphs are a set of all graphs with a given $d_k$-distribution, e.g., with the $d_k$-distribution in a given real network. The $d_k$-random graphs are a maximum-entropy ensemble of these graphs 25. This ensemble consists of all $d_k$-graphs, and the probability measure is uniform (unbiased): each graph $G$ in the ensemble is assigned the same probability $P(G) = 1/N_d$, where $N_d$ is the number of $d_k$-graphs. For $d = 0, 1, 2$ these are well studied classical random graphs $G_{N,M}$ 29, configuration model 30–33, and random graphs with a given joint degree distribution 34, respectively. Since a sequence of $d_k$-distributions is increasingly more informative and thus constraining, the corresponding sequence of the sizes of $d_k$-random graph ensembles is non-increasing and shrinking to 1, $N_0 \geq N_1 \geq \ldots \geq N_N = 1$, Fig. 1. At low $d = 0, 1, 2$ these numbers $N$ can be calculated either exactly or approximately 35–38.

We emphasize that in $d_k$-graphs the $d_k$-distribution constraints are sharp, i.e., they hold exactly—all $d_k$-graphs have exactly the same $d_k$-distribution. An alternative description uses soft maximum-entropy ensembles belonging to the general class of exponential random graph models 39–44 in which these constraints hold only on average over the ensemble—the expected $d_k$-distribution in the ensemble (not in any individual graph) is fixed to a given distribution. This ensemble consists of all possible graphs $G$ of size $N$, and the probability measure $P(G)$ is the one maximizing the ensemble entropy $S = -\sum_G P(G) \ln P(G)$ under the $d_k$-distribution constraints. Using analogy with statistical mechanics, sharp and soft ensemble are often called microcanonical and canonical, respectively.

As a consequence of the convergence and inclusiveness properties of $d_k$-series, any network property $X$ of any given network $G$ is guaranteed to be reproduced with any desired accuracy by high enough $d$. At $d = N$ all possible properties are reproduced exactly, but the $N_k$-graph ensemble trivially consists of only one graph, $G$sself, and has zero entropy. In the sense that the entropy of $d_k$-ensembles $S_d = \ln N_d$ is a non-increasing function of $d$, the smaller the $d$, the more random the $d_k$-random graphs, which also agrees with the intuition that $d_k$-random graphs are “the less random and the more structured,” the higher the $d$. Therefore the general problem of explaining a given property $X$ reduces to the general problem of how random a graph ensemble must be so that $X$ is statistically significant. In the $d_k$-series context, this question becomes: how much local degree information, i.e., information about concentrations of degree-labeled subgraphs of what minimal size $d$, is needed to reproduce a possibly
global property $X$ with a desired accuracy?

Below we answer this question for a set of popular and commonly used structural properties of some paradigmatic real networks. But to answer this question for any property in any network, we have to be able to sample graphs uniformly at random from the sets of $dk$-graphs—the problem that we discuss next.

**$dk$-random graph sampling**

Soft $dk$-ensembles tend to be more amenable for analytic treatment, compared to sharp ensembles, but even in soft ensembles the exact analytic expressions for expected values are known only for simplest network properties in simplest ensembles [46–48]. Therefore we retreat to numeric experiments here. Given a real network $G$, there exist two ways to sample $dk$-random graphs in such experiments: $dk$-randomize $G$ generalizing the randomization algorithms in [49, 50], or construct random graphs with $G$’s $dk$-sequence from scratch [25, 51], also called direct construction [52–55].

The first option, $dk$-randomization, is easier. It accounts for swapping random (pairs of) edges, starting from $G$, such that the $dk$-distribution is preserved at each swap, Fig. 2. There are many concerns with this prescription [56], two of which are particularly important. The first concern is if this process “ergodic,” meaning that if any two $dk$-graphs are connected by a chain of $dk$-swaps. For $d = 1$ the 2-edge swap is ergodic [49, 50], while for $d = 2$ it is not ergodic. However the so-called restricted 2-edge swap, when at least one node attached to each edge has the same degree, Fig. 2, was proven to be ergodic [57]. It is now commonly believed that there is no edge-swapping operation, of this or other type, that is ergodic for the 3$k$-distribution, although a definite proof is lacking at the moment. If there exists no ergodic 3$k$-swapping, then we cannot really rely on it in sampling $dk$-random graphs because our real network $G$ can be trapped on a small island of atypical $dk$-graphs, which is not connected by any $dk$-swap chain to the main land of many typical $dk$-graphs. Yet we note that in an unpublished work [58] we showed that five out of six considered real networks were virtually indistinguishable from their 3$k$-randomizations across all the considered network properties, although one network (power grid) was very different from its 3$k$-random counterparts.

The second concern with $dk$-randomization is about how close to uniform sampling the $dk$-swap Markov chain is after its mixing time is reached—its mixing time is yet another concern that we do not discuss here, but according to many numerical experiments and some analytic estimates, it is $O(M)$ [25, 31, 49, 51, 57]. Even for $d = 1$ the swap chain does not sample 1$k$-graphs uniformly at random, yet if the 1$k$-distribution is a power law, then the sampling is remarkably close to uniform [53, 59, 60].

A simple algorithm for the second $dk$-sampling option, constructing $dk$-graphs from scratch, is widely known for $d = 1$: given $G$’s degree sequence $\{k_i\}$, build a 1$k$-random graph by attaching $k_i$ half-edges (“stubs”) to node $i$, and then connect random pairs of stubs to form edges [31, 32]. If during this pro-
Figure 2: The $dk$-sampling and convergence of $dk$-series illustrated. 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 $c(k)$ ($2.5k$): $p = \min(1, e^{-\beta \Delta H})$, where $\beta$ 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: $H_{2.1k} = |c_{\text{current}} - c_{\text{target}}|$ and $H_{2.5k} = \sum_i |c_{\text{current}}[k_i] - c_{\text{target}}[k_i]|$. The right column shows LaNet-vi [45] visualizations of the results of these $dk$-rewiring processes (Supplementary Information, Algorithms to sample $dk$-random graphs Section), applied to the PGP network, visualized at the bottom of the left column. The node sizes are proportional to the logarithm of their degrees (left legends), while the color reflects node coreness [45] (right legends). As $d$ grows, the shown $dk$-random graphs quickly become more similar to the real PGP network.
cess a self-loop (both stubs are connected to the same node) or double-edge (two edges between the same pair of nodes) is formed, one has to restart the process from scratch since otherwise the graph sampling is not uniform \[61\]. If the degree sequence is power-law distributed with exponent close to \(-2\) as in many real networks, then the probability that the process must be restarted approaches 1 for large graphs \[62\], so that this construction process never succeeds. An alternative greedy algorithm is described in \[53\], which always quickly succeeds and gives an efficient way of testing if a given sequence of integers is graphical, i.e., if it can be realized as a degree sequence of a graph. The base sampling procedure does not sample graphs uniformly, but then an importance sampling procedure is used to account for the bias, which results in uniform sampling. Yet again, if the degree distribution is a power law, one can show that even without importance sampling, the base sampling procedure is uniform, since the distribution of sampling weights that one can compute for this greedy algorithm approaches a delta function. Extensions of the naive 1\(k\)-construction above to 2\(k\) are less known, but they exist \[25, 34, 55, 63\]. Most of these 2\(k\)-constructions do not sample 2\(k\)-graphs exactly uniformly either \[57\], but importance sampling in \[55\] corrects for the sampling bias.

Unfortunately, to the best of our knowledge, there currently exists no 3\(k\)-construction algorithm that can be successfully used in practice to generate large 3\(k\)-graphs with 3\(k\)-distributions of real networks. The 3\(k\)-distribution is quite constraining and non-local, so that the \(dk\)-construction methods described above for \(d = 1, 2\) cannot be readily extended to \(d = 3\) \[25\]. There is yet another option—3\(k\)-targeting rewiring, Fig. 2. It is 2\(k\)-preserving rewiring in which each 2\(k\)-swap is accepted not with probability 1, but with probability equal to \(\min(1, \exp(-\beta \Delta H))\), where \(\beta\) is the inverse temperature of this simulated-annealing-like process, and \(\Delta H\) is the change in the \(L^1\) distance between the 3\(k\)-distribution in the current graph and the target 3\(k\)-distribution before and after the swap. This probability favors and, respectively, suppresses 2\(k\)-swaps that move the graph closer or farther from the target 3\(k\)-distribution. Unfortunately we report that in agreement with \[51\] this 2\(k\)-preserving 3\(k\)-targeting process never converged for any considered real network—regardless of how long we let the rewiring code run, after the initial rapid decrease, the 3\(k\)-distance, while continuing to slowly decrease, remained substantially large. The reason why this process never converges is that the 3\(k\)-distribution is extremely constraining, so that the number of 3\(k\)-graphs \(N_3\) is infinitesimally small compared to the number of 2\(k\)-graphs \(N_2\), \(N_3/N_2 \ll 1\) \[25, 35, 37\]. Therefore it is extremely difficult for the 3\(k\)-targeting Markov chain to find a rare path to the target 3\(k\)-distribution, and the process gets hopelessly trapped in abundant local minima in distance \(H\).

Therefore, on the one hand, even though 3\(k\)-randomized versions of many real networks are indistinguishable from the original networks across many metrics \[58\], we cannot use this fact to claim that at \(d = 3\) these metrics are not statistically significant in those networks, because the 3\(k\)-randomization Markov chain may be non-ergodic. On the other hand, we cannot generate the corresponding 3\(k\)-random graphs from scratch in a feasible amount of compute time.
The $3k$-random graph ensemble is not analytically tractable either. Given that $d = 2$ is not enough to guarantee the statistical insignificance of some important properties of some real networks, see [58] and below, we, as in [51], retreat to numeric investigations of $2k$-random graphs in which in addition to the $2k$-distribution, some substatistics of the $3k$-distribution is fixed. Since strong clustering is a ubiquitous feature of many real networks [1–3], one of the most interesting such substatistics is clustering.

Specifically we study $2.1k$-random graphs, defined as $2k$-random graphs with a given value of average clustering $\bar{c}$, and $2.5k$-random graphs, defined as $2k$-random graphs with given values of average clustering $\bar{c}(k)$ of nodes of degree $k$ [51]. The $3k$-distribution fully defines both $2.1k$- and $2.5k$-statistics, while $2.5k$ defines $2.1k$. Therefore $2k$-graphs are a superset of $2.1k$-graphs, which are a superset of $2.5k$-graphs, which in turn contain all the $3k$-graphs. $N_2 > N_{2.1} > N_{2.5} > N_3$. Therefore if a particular property is not statistically significant in $2.5k$-random graphs, for example, then it is not statistically significant in $3k$-random graphs either, while the converse is not generally true.

We thus generate 20 $dk$-random graphs with $d = 0, 1, 2, 2.1, 2.5$ for each considered real network. For $d = 0, 1, 2$ we use the standard $dk$-randomizing swapping, Fig. 2. We do not use its modifications to guarantee exactly uniform sampling [59, 60], because: (1) even without these modifications the swapping is close to uniform in power-law graphs, (2) these modifications are non-trivial to efficiently implement, and (3) we could not extend these modifications to the $2.1k$ and $2.5k$ cases. As a consequence, our sampling is not exactly uniform, but we believe it is close to uniform for the reasons discussed above. To generate $dk$-random graphs with $d = 2.1, 2.5$, we start with a $2k$-random graph, and apply to it the standard $2k$-preserving $2.xk$-targeting ($x = 1.5$) rewiring process, Fig. 2. The algorithms that do that, as described in [51], did not converge on some networks, so that we modified the algorithm in [16] to ensure the convergence in all cases. The details of these modifications are in Supplementary Information, Algorithms to sample $dk$-random graphs Section, and the software package implementing these algorithms is freely available at https://github.com/polcolomer/RandNetGen.

Real vs. $dk$-random networks

We performed an extensive set of numeric experiments with six real networks—the US air transportation network, an fMRI map of the human brain, the Internet at the level of autonomous systems, a technosocial web of trust among users of the distributed Pretty Good Privacy (PGP) cryptosystem, a human protein interaction map, and an English word adjacency network (Supplementary Information, Considered networks Section). For each network we compute its average degree, degree distribution, degree correlations, average clustering, averaging clustering of nodes of degree $k$, and based on these $dk$-statistics generate a number of $dk$-random graphs as described above for each $d = 0, 1, 2, 2.1, 2.5$. Then for each sample we compute a variety of network properties, and report their means and deviations for each combination of the real network, $d$, and the
property. Figures 3-6 present the results for the PGP network, while Supplementary Information, the Results Section contains the complete set of results for all the considered real networks. The reason why we choose the PGP network as our main example is that this network appears to be “least random” among the considered real networks, in the sense that the PGP network requires higher values of \( d \) to reproduce its considered properties. The only exception is the brain network. Some of its properties are not reproduced even by \( d = 2.5 \).

Figure 2 visualizes the PGP network and its \( dk \)-randomizations. The figure illustrates the convergence of \( dk \)-series applied to this network. While the 0\( k \)-random graph has very little in common with the real network, the 1\( k \)-random one is somewhat more similar, even more so for 2\( k \), and there is very little visual difference between the real PGP network and its 2.5\( k \)-random counterpart. This figure is only an illustration though, and to have a better understanding of how similar the network is to its randomization, we compare their properties.

We split the properties that we compare into the following categories. The microscopic properties are local properties of individual nodes and subgraphs of small size. These properties can be further subdivided into those that are defined by the \( dk \)-distributions—the degree distribution, average neighbor degree, clustering, Fig. 3—and those that are not fixed by the \( dk \)-distributions—the concentrations of subgraphs of size 3 and 4, Fig. 10. The mesoscopic properties—\( k \)-coreness and \( k \)-density (the latter is also known as \( m \)-coreness or edge multiplicity, Supplementary Information, Network properties Section), Fig. 5—depend both on local and global aspects of network organization. Finally, the macroscopic properties are truly global ones—betweenness, the distribution of hop lengths of shortest paths, and spectral properties, Fig. 6. In Supplementary Information, Section 5 we also report some extremal properties, such as the graph diameter (the length of the longest shortest path), and Kolmogorov-Smirnov distances between the distributions of all the considered properties in real networks and their corresponding \( dk \)-random graphs. The detailed definitions of all the properties that we consider can be found in Supplementary Information, Network properties Section.

In most cases—henceforth by “case” we mean a combination of a real network and one of its considered property—we observe a nice convergence of properties as \( d \) increases. In many cases there is no statistically significant difference between the property in the real network and in its 2.5\( k \)-random graphs. In that sense these graphs, i.e., random graphs whose degree distribution and degree-dependent clustering \( \bar{c}(k) \) are as in the original network, capture many other important properties of the real network.

Some properties always converge. This is certainly true for the microscopic properties in Fig. 5 simply confirming that our \( dk \)-sampling algorithm operates correctly. But many properties that are not fixed by the \( dk \)-distributions converge as well. Neither the concentration of subgraphs of size 3 nor the distribution of the number of neighbors common to a pair of nodes are fully fixed by \( dk \)-distributions with any \( d < 3 \) by definition, yet 2.5\( k \)-random graphs reproduce them well in all the considered networks. Most subgraphs of size 4 are also captured at \( d = 2.5 \) in most networks, even though \( d = 3 \) would not be
enough to exactly reproduce the statistics of these subgraphs. We note that the improvement in subgraph concentrations at \( d = 2.5 \) compared to \( d = 2.1 \) is particularly striking. Fig. 10. The mesoscopic and especially macroscopic properties converge more slowly as expected. Nevertheless, quite surprisingly, both mesoscopic properties (\( k \)-coreness and \( k \)-density) and some macroscopic properties converge nicely in most cases. The \( k \)-coreness, \( k \)-density, and the spectral properties, for instance, converge at \( d = 2.5 \) in all the considered cases other than Internet’s Fiedler value. In some cases a property, even global one, converges for \( d \) lower than 2.5. Betweenness, for example, a global property, converges at \( d = 1 \) for the Internet and English word network.

Finally, there are “outlier” networks and properties of poor or no \( dk \)-convergence. Many properties of the brain network, for example, exhibit slow or no convergence. We have also experimented with community structure inferred by different algorithms, and in most cases the convergence is either slow or non-existent as one could expect.

**Discussion**

In general, we should not expect non-local properties of networks to be exactly or even closely reproduced by random graphs with local constraints. The considered brain network is a good example of that this expectation is quite reasonable. The human brain consists of two relatively weakly connected parts, and no \( dk \)-randomization with low \( d \) is expected to reproduce this peculiar global feature, which likely has an impact on other global properties. And indeed we observe in Supplementary Information, Results Section that its two global properties, the shortest path distance and betweenness distributions, differ drastically between the brain and its \( dk \)-randomizations.

Another good example is community structure, which is not robust with respect to \( dk \)-randomizations in all the considered networks. In other words, \( dk \)-randomizations destroy the original peculiar cluster organization in real networks, which is not surprising, as clusters have too many complex non-local features such as variable densities of internal links, boundaries, etc., which \( dk \)-randomizations, even with high \( d \), are expected to affect considerably.

Surprisingly, what happens for the brain and community structure does not appear representative for many other considered combinations of real networks and their properties. As a possible explanation, one can think of constraint-based modeling as a satisfiability (SAT) problem: find the elements of the adjacency matrix (1/0, True/False) such that all the given constraints in terms of the functions of the marginals (degrees) of this matrix are obeyed. We then expect that the \( 3k \)-constraints already correspond to an NP-hard SAT problem, such as 3-SAT, with hardness coming from the global nature of the constraints in the problem. However, many real-world networks evolve based mostly on local dynamical rules and thus we would expect them to contain correlations with \( d < 3 \), i.e., below the NP-hard barrier. The primate brain, however, has likely evolved through global constraints, as indicated by the dense connectivity.
across all functional areas and the existence of a strong core-periphery structure in which the core heavily concentrates on areas within the associative cortex, with connections to and from all the primary input and subcortical areas [20].

However, in most cases, the considered networks are $dk$-random with $d \leq 2.5$, i.e., $d \leq 2.5$ is enough to reproduce not only basic microscopic (local) properties but also mesoscopic and even macroscopic (global) network properties [12–14]. This finding means that these more sophisticated properties are effectively random in the considered networks, or more precisely, that the observed values of these properties are effective consequences of particular degree distributions and, optionally, degree correlations and clustering that the networks have. This further implies that attempts to find explanations for these complex but effectively random properties should probably be abandoned, and redirected to explanations of why and how degree distributions, correlations, and clustering emerge in real networks, for which there already exists a multitude of approaches [64–71]. On the other hand, the features that we found non-random do require separate explanations, or perhaps a different system of null models.

We reiterate that the $dk$-randomization system makes it clear that there is no a priori preferred null model for network randomization. To tell how statistically significant a particular feature is, it is necessary to compare this feature in the real network against the same feature in an ensemble of random graphs, a null model. But one is free to choose any random graph model. In particular, any $d$ defines a random graph ensemble, and we find that many properties, most notably the frequencies of small subgraphs that define motifs [17, 18], strongly depend on $d$ for many considered networks. Therefore choosing any specific value of $d$, or more generally, any specific null model to study the statistical significance of a particular structural network feature, requires some non-trivial justification before this feature can be claimed important for any network function.

Yet another implication of our results is that if one looks for network topology generators that would veraciously reproduce certain properties of a given real network—a task that often comes up in as diverse disciplines as biology [72] and computer science [73]—one should first check how $dk$-random these properties are. If they are $dk$-random with low $d$, then one may not need any sophisticated mission-specific topology generators. The $dk$-random graph generation algorithms discussed here can be used for that purpose in this case. We note that there exists an extension of a subset of these algorithm for networks with arbitrary annotations of links and nodes [74]—directed or colored (multilayer) networks, for instance.

The main caveat of our approach is that we have no proof that our $dk$-random graph generation algorithms for $d = 2.1$ and $d = 2.5$ sample graphs uniformly at random from the ensemble. The random graph ensembles and edge rewiring processes employed here are known to suffer from problems such as degeneracy and hysteresis [14, 15, 16]. Ideally, we would wish to calculate analytically the exact expected value of a given property in an ensemble. This is currently possible only for very simple properties in soft ensembles with $d = 0, 1, 2$ [46–48]. Some mathematically rigorous results are available for $d = 0, 1$ and for
some exponential random graph models \cite{33, 43}. Many of these results rely
on graphons \cite{24} that are applicable to dense graphs only, while virtually all
real networks are sparse \cite{62}. Some rigorous approaches to sparse networks are
beginning to emerge \cite{77, 78}, but the rigorous treatment of global properties,
which tend to be highly non-trivial functions of adjacency matrices, in random
graph ensembles with $d > 2$ constraints, appear to be well beyond the reach
in the near future. Yet if we ever want to fully understand the relationship
between the structure, function, and dynamics of real networks, this future
research direction appears to be of a paramount importance.

Acknowledgements

We acknowledge financial support by NSF Grants No. CNS-1039646, CNS-
1345286, CNS-0722070, CNS-0964236, CNS-1441828, CNS-144289, CNS-1442999,
and DMR-1206839; by AFOSR and DARPA Grants No. HR0011-12-1-0012 and
FA9550-12-1-0405; by DTRA Grant No. HDTRA-1-09-1-0039; and by Cisco
Systems.

Supplementary Information

Network properties

Here we describe all the network properties measured and discussed in Section \ref{sec:22}
and, where meaningful, their relations to $dk$-series.

Degree distribution

The distribution $P(k)$ of node degrees $k$, i.e., the 1$k$-distribution, is:

$$ P(k) = \frac{N(k)}{N}, \quad (1) $$

where $N(k)$ is the number of nodes of degree $k$ in the network, and $N$ is the
total number of nodes in it, so that $P(k)$ is normalized, $\sum_k P(k) = 1$. The
1$k$-distribution fully defines the 0$k$-distribution, i.e., the average degree $\bar{k}$ in the
network, by

$$ \bar{k} = \sum_k kP(k), \quad (2) $$

but not vice versa.

Average nearest neighbor degree (ANND)

The average degree $\bar{k}_{nn}(k)$ of nearest neighbors of nodes of degree $k$ is a com-
monly used projection of the joint degree distribution (JDD) $P(k, k')$, i.e., the
2k-distribution. The JDD is defined as

\[ P(k, k') = \mu(k, k') \frac{N(k, k')}{2M}, \tag{3} \]

where \( N(k, k') = N(k', k) \) is the number of links between nodes of degrees \( k \) and \( k' \) in the network, \( M \) is the total number of links in it, and

\[ \mu(k, k') = \begin{cases} 2 & \text{if } k = k', \\ 1 & \text{otherwise}, \end{cases} \tag{4} \]

so that \( P(k, k') \) is normalized, \( \sum_{k,k'} P(k, k') = 1 \). The 2k-distribution fully defines the 1k-distribution by

\[ P(k) = \bar{k}_{nn} P(k), \tag{5} \]

but not vice versa. The average neighbor degree \( \bar{k}_{nn}(k) \) is a projection of the 2k-distribution \( P(k, k') \) via

\[ \bar{k}_{nn}(k) = \frac{\bar{k}}{kP(k)} \sum_{k'} k' P(k, k') = \frac{\sum_{k'} k' P(k, k')}{\sum_{k'} P(k, k')} \tag{6} \]

**Clustering**

Clustering of node \( i \) is the number of triangles \( \Delta_i \) it belongs to, or equivalently the number of links among its neighbors, divided by the maximum such number, which is \( k(k-1)/2 \), where \( k \) is \( i \)'s degree, \( \deg(i) = k \). The average clustering coefficient of the network is

\[ \bar{c} = \frac{1}{N} \sum_i \frac{\Delta_i}{k_i(k_i - 1)/2} \tag{7} \]

Averaging over all nodes of degree \( k \), the degree-dependent clustering is

\[ \bar{c}(k) = \frac{2\Delta(k)}{k(k - 1)N(k)}, \text{ where } \Delta(k) = \sum_{i: \deg(i) = k} \Delta_i. \tag{8} \]

The degree-dependent clustering is a commonly used projection of the 3k-distribution. (See [79, 80] for an alternative formalism involving three point correlations.) The 3k-distribution is actually two distributions characterizing the concentrations of the two non-isomorphic degree-labeled subgraphs of size 3, wedges and triangles:
Let \(N_\Lambda(k', k, k'') = N_\Lambda(k'', k, k')\) be the number wedges involving nodes of degrees \(k\), \(k'\), and \(k''\), where \(k\) is the central node degree, and let \(N_\Delta(k, k', k'')\) be the number of triangles consisting of nodes of degrees \(k\), \(k'\), and \(k''\), where \(N_\Delta(k, k', k'')\) is assumed to be symmetric with respect to all permutations of its arguments. Then the two components of the 3\(K\)-distribution are

\[
P_\Lambda(k', k, k'') = \mu(k', k'') \frac{N_\Lambda(k', k, k'')}{2W},
\]

\[
P_\Delta(k, k', k'') = \nu(k, k', k'') \frac{N_\Delta(k, k', k'')}{6T},
\]

where \(W\) and \(T\) are the total numbers of wedges and triangles in the network, and

\[
\nu(k, k', k'') = \begin{cases} 
6 & \text{if } k = k' = k'', \\
1 & \text{if } k \neq k' \neq k'', \\
2 & \text{otherwise},
\end{cases}
\]

so that both \(P_\Lambda(k', k, k'')\) and \(P_\Delta(k, k', k'')\) are normalized, \(\sum_{k, k', k''} P_\Lambda(k', k, k'') = \sum_{k, k', k''} P_\Delta(k, k', k'') = 1\). The 3\(K\)-distribution defines the 2\(k\)-distribution (but not vice versa), by

\[
P(k, k') = \frac{1}{k + k' - 2} \left[ \frac{6T}{M} P_\Delta(k, k', k'') \right] + \frac{W}{M} \left[ P_\Lambda(k', k, k'') + P_\Lambda(k, k', k'') \right].
\]

The normalization of 2\(k\)- and 3\(k\)-distributions implies the following identity between the numbers of triangles, wedges, edges, nodes, and the second moment of the degree distribution \(k^2 = \sum_k k^2 P(k)\):

\[
23T + W + M = \bar{k}^2.
\]

The degree-dependent clustering coefficient \(\bar{c}(k)\) is the following projection of the 3\(k\)-distribution

\[
\bar{c}(k) = \frac{6T}{N} \frac{\sum_{k', k''} P_\Delta(k, k', k'')}{k(k-1)P(k)}.
\]

**Subgraph frequencies**

The concentration of subgraphs of size 3 is exactly fixed only by the 3\(k\)-distribution, or by the 3-distribution, Section . There are two non-isomorphic connected graphs of size 3 (triangles and wedges), and their concentrations are defined as

\[
C_\Lambda = \frac{\wedge}{N_3}, \quad C_\Delta = \frac{\Delta}{N_3},
\]

where \(\wedge\) is the number of wedges in the graph, \(\Delta\) is the number of triangles in the graph, and \(N_3 = \wedge + \Delta\) is the total number of connected subgraphs of size 3 in the graph.
The concentration of subgraphs of size 4 is exactly fixed only by the 4k-distribution, or by the 4-distribution. There are six non-isomorphic connected graphs of size 4,

and their concentrations are defined as the number of subgraphs of a particular type divided by the total number of connected subgraphs of size 4.

In our comparisons of real networks and their dk-randomizations in Section 22 we choose to compare the subgraph concentrations directly, versus computing z-scores, as common in the motif literature. The reasons for this decision is that z-scores are tailored for a fixed null model, while we are considered a series of null models parameterized by \( d \) in \( dk \)-series. There is nothing in the z-score and \( dk \)-series definitions that could easily provide any estimates of how fast the subgraph frequency means and standard deviations in the z-score definition converge as functions of \( d \). Therefore the comparisons of z-scores for different values of \( d \) would be meaningless.

**Common neighbors**

The number \( m_{ij} \) of common neighbors between two connected nodes \( i \) and \( j \) is the number of nodes to which both \( i \) and \( j \) are connected, or equivalently the multiplicity of edge \((i, j)\):

\[
m_{ij} = \sum A_{il} A_{lj} A_{ij},
\]

where \( \{A_{ij}\} \) is the adjacency matrix of the graph. The distribution \( P(m) \) of the number of common neighbors \( m \) is then

\[
P(m) = \frac{\sum_{i<j} \delta_{m_{ij},m}}{N(N-1)/2},
\]

where \( \delta \) is the Kronecker delta. The common neighbor distribution is thus the probability that two connected nodes in the graph have \( m \) common neighbors. This property is exactly fixed only by the \( 3k \)-distribution.

**\( k \)-coreness and \( k \)-denseness**

The \( k \)-core decomposition [81] of a graph is a set of nested subgraphs induced by nodes of the same \( k \)-coreness. A node has \( k \)-coreness equal to \( k \) if it belongs to a maximal connected subgraph of the original graph, in which all nodes have degree at least \( k \), i.e., in which each node is connected to at least \( k \) other nodes in the subgraph.

Similarly, the \( k \)-dense decomposition [82] of a graph is a set of nested subgraphs induced by edges of the same \( k \)-denseness. An edge has \( k \)-denseness
equal to $k$ if it belongs to a maximal connected subgraph of the original graph, in which all edges have multiplicity at least $k - 1$, i.e., in which each pair of connected nodes has at least $k - 1$ common neighbors in the subgraph.

Both the $k$-core and $k$-dense decompositions rely on the analysis of local properties of nodes and edges. However, due to the recursive nature of these decompositions, the $dk$-distributions with $d = 0, 1, 2, 1.2, 2.5$ do not exactly fix either the $k$-core or $k$-dense distributions.

**Betweenness**

Betweenness $b(i)$ of node $i$ is a measure of how “important” $i$ is in terms of the number of shortest paths passing through it. Formally, if $\sigma_{st}(i)$ is the number of shortest paths between nodes $s \neq i$ and $t \neq i$ that pass through $i$, and $\sigma_{st}$ is the total number of shortest paths between the two nodes $s \neq t$, then betweenness of $i$ is

$$b(i) = \sum_{s,t} \frac{\sigma_{s,t}(i)}{\sigma_{s,t}}.$$  \hfill (18)

Averaging over all nodes of degree $k$, degree-dependent betweenness $\bar{b}(k)$ is

$$\bar{b}(k) = \sum_{i: \text{deg}(i)=k} \frac{b(i)}{N(k)}.$$  \hfill (19)

**Shortest path distance**

The distance distribution is the distribution of hop-lengths of shortest path between nodes in a network. Formally, if $N(h)$ is the number of node pairs located at hop distance $h$ from each other, then the distance distribution $P(h)$ is

$$P(h) = \frac{N(h)}{N(N-1)/2},$$  \hfill (20)

where $N(N-1)/2$ is the total number of nodes pairs in the network. The average distance is:

$$\bar{h} = \sum_{h} h P(h).$$  \hfill (21)

Finally, the network diameter, i.e., the maximum hop distance between nodes in the network, is

$$d = \max(h).$$  \hfill (22)

**Spectral properties**

The adjacency matrix of graph $A$ gives the full information on the structure of the graph. The largest eigenvalue of $A$ and the spectral gap, which is defined as the difference between the largest and second largest eigenvalue $A$, play important roles in the dynamic processes on networks. For instance, the largest
Subgraph-based series \( (d\text{-series}) \) vs. \( dk\text{-series} \)

We compare \( dk\text{-series} \) with the series based on subgraph frequencies, and show that the latter cannot form a systematic basis for topology analysis.

The difference between \( dk\text{-series} \) and subgraph-based-series, which we can call \( d\text{-series} \), is that the former is the series of distributions of \( d\)-sized subgraphs labeled with node degrees in a given network, while the \( d\text{-series} \) is the distributions of such subgraphs in which this degree information is ignored. This difference explains the mnemonic names for these two series: ‘\( d\)’ in ‘\( dk\)’ refers to the subgraph size, while ‘\( k\)’ signifies that they are labeled by node degrees—‘\( k\)’ is a standard notation for node degrees.

This difference between the \( dk\text{-series} \) and \( d\text{-series} \) is crucial. The \( dk\text{-series} \) are inclusive, in the sense that the \( (d+1)k\)-distribution contains the full information about the \( dk\)-distribution, plus some additional information, which is not true for \( d\text{-series} \).

To see this, let us consider the first few elements of both series in Table 1. In Section \[ \] we show explicitly how the \( (d+1)k\)-distributions define the \( dk\)-distribution for \( d = 0, 1, 2 \). The key observation is that the \( d\text{-series} \) does not have this property. The 0'th element of \( d\text{-series} \) is undefined. For \( d = 1 \) we have the number of subgraphs of size 1, which is just \( N \), the number of nodes in the network. For \( d = 2 \), the corresponding statistics is \( M \), the number of links, subgraphs of size 2. Clearly, \( M \) and \( N \) are independent statistics, and the former does not define the latter. For \( d = 3 \), the statistics are \( W \) and \( T \), the total number of wedges and triangles, subgraphs of size 3, in the network. These do not define the previous element \( M \) either. Indeed, consider the following two

| \( d \) | \( dk\text{-statistics} \) | \( d\text{-statistics} \) |
|---|---|---|
| 0 | \( k \) | - |
| 1 | \( N(k) \) | \( N \) |
| 2 | \( N(k,k') \) | \( M \) |
| 3 | \( N_\wedge(k,k',k'') \) | \( W \) |
| \( N_\triangle(k,k',k'') \) | \( T \) |
networks of size $N$—the chain and the star:

There are no triangles in either network, $T = 0$. In the chain network, the number of wedges is $W = N - 2$, and in the star $W = (N - 1)(N - 2)/2$. We see that even though $W$ ($d = 3$) scales completely differently with $N$ in the two networks, the number of edges $M = N - 1$ ($d = 2$) is the same.

In summary, $d$-series is not inclusive. For each $d$, the corresponding element of the series reflects a different kind of statistical information about the network topology, unrelated or only loosely related to the information conveyed by the preceding elements. At the same time, similar to $dk$-series, the $d$-series is also converging since at $d = N$ it specifies the whole network topology. However, this convergence is much slower than in the $dk$-series case. In the two networks considered above, for example, neither $W = N - 2$, $T = 0$ nor $W = (N - 1)(N - 2)/2$, $T = 0$, fix the network topology as there are many non-isomorphic graphs with the same $(W, T)$ counts, whereas the $3k$-distributions $N_k(1, 2, 2) = 2$, $N_k(2, 2, 2) = N - 4$ and $N_k(1, N - 1, 1) = (N - 1)(N - 2)/2$ define the chain and star topologies exactly.

The node degrees thus provide necessary information about subgraph locations in the original network, which significantly speeds up convergence as a function of $d$, and more importantly makes the $dk$-series basis inclusive and systematic.

**Algorithms to sample $dk$-random graphs**

The methods that we use to sample $dk$-random graphs for a given graph representing a real network are based on two different rewiring processes: $dk$-randomizing rewiring ($d = 0, 1, 2$) and $p$-targeting $dk$-preserving rewiring ($p = 2.1k, 2.5k$).

The first method ($dk$-randomization) consists of swapping random pairs of edges in the original network preserving its $dk$-distribution, Algorithm 1. The following three input parameters are required: $G_T$ the original graph, $R$ the number of rewirings to apply, and $d$ index that indicates the $dk$-distribution to preserve. The random edge selection function on line 4 and the rewiring function on line 5 depend on $d$ as follows:

- if $d = 0$, random edge $(i, j)$ and non-edge $(a, b)$ (disconnected nodes $a$ and $b$) are selected, and the rewiring consists of removing edge $(i, j)$ and adding edge $(a, b)$.

- if $d = 1$, two random edges $(i, j)$ and $(a, b)$ are selected and discarded if either edge $(i, b)$ or edge $(j, a)$ exists; if neither edge $(i, b)$ nor edge $(j, a)$ exists, the rewiring consists of removing edges $(i, j)$ and $(a, b)$, and adding edges $(i, b)$ and $(j, a)$. 

20
• if $d = 2$, two random edges $(i, j)$ and $(a, b)$ such that degrees $k_i = k_a$
are selected and discarded if either edge $(i, b)$ or edge $(j, a)$ exists; if nei-
ther edge $(i, b)$ nor edge $(j, a)$ exists, the associated rewiring consists of
removing edges $(i, j)$ and $(a, b)$ and adding edges $(i, b)$ and $(j, a)$.

Algorithm 1: $dk$-randomization process.

| Line | Description |
|------|-------------|
| 1    | $i = 0$; // Rewiring counter |
| 2    | $G_0 = G_T$; // Graph to rewire |
| 3    | /* 1. Initialize process variables */ |
| 4    | $i = R$; // Number of rewirings |
| 5    | Input: $d$; // $d$ value: 0, 1, or 2 |
| 6    | /* 2. Apply $R$ $dk$-rewirings */ |
| 7    | while $i < R$ do |
| 8    | /* Select a random pair of edges (see the text) */ |
| 9    | rew = random(edges $\in G_i$); |
| 10   | /* Apply the rewiring to $G_i$ */ |
| 11   | apply rewiring($G_i$, rew); |
| 12   | $i = i + 1$; |
| 13   | Output: $G_i$; // $dk$-randomized graph |

The second method of ($p$-targeting $dk$-preserving rewiring) is based on
simulated annealing, and consists of two phases: randomization and targeting
rewiring, Algorithm 2. The following input parameters are required: $G_T$ the
original graph, $p_{G_T}$ the property to target, $R$ the number of $dk$-rewirings to
apply at each value of temperature, $\beta_0$ the initial inverse temperature, $\beta_{factor}$
the rate of temperature decrease, and $\alpha$ the acceptance threshold. In the first
phase the original graph is 2-k-randomized by Algorithm 1. In the second phase,
the obtained 2-k-random graph is 2-k-rewired, but each rewiring is accepted with
probability $min[exp(-\beta H), 1]$ which depends on current values of energy $H$ and
temperature $1/\beta$. Energy is defined as the distance between the values of prop-
ty $p$ in the original and current rewired graphs. Temperature is high initially,
but each round of $R$ rewirings (line 9), it decreases by factor $\beta_{factor}$, thus de-
creasing the probability of accepting a rewiring that increases energy. This
second phase terminates when either energy is zero, meaning that the value of
$p$-property in the rewired graph $p_{G_i}$ is equal to its value in the original graph
$p_{G_T}$, or when the percentage of accepted rewirings during the last round falls
below a user-specified threshold $\alpha$. Function compute property($G$) appear-
ing on lines 3 and 12 returns average clustering $c$ or average degree-dependent
clustering $c(k)$ of $G$ depending on whether $d = 2.1$ or $d = 2.5$, respectively.
Energy function distance($p_{G_i}$, $p_{G_T}$) appearing on lines 4 and 13 depends on $d$
as follows:

- if $d = 2.1$, distance($p_{G_i}$, $p_{G_T}$) = $|\tau_{G_i} - \tau_{G_T}|$,
- if $d = 2.5$, distance($p_{G_i}$, $p_{G_T}$) = $\sum_k |\tau_{G_i}(k) - \tau_{G_T}(k)|$. 

21
Algorithm 2: \(p\)-targeting \(dk\)-preserving rewiring process.

Input: \(G_T\); // Original graph
Input: \(R\); // Number of rewirings
Input: \(p_{G_T}\); // Target value of the property
Input: \(\beta_0\); // Initial inverse temperature
Input: \(\beta_{\text{factor}}\); // Temperature decrease rate
Input: \(\alpha\); // Acceptance threshold

/* 1. \(2k\)-randomize the original graph using Alg. 1 */
1 \(M = \text{num.edges}(G_T)\);
2 \(G_0 = \text{dk-randomize}(G_T, R = 100M, d = 2)\);
/* 2. Compute the current value of the property */
3 \(p_{G_0} = \text{compute-property}(G_0)\);
/* 3. Compute the current energy, the distance between \(p_{G_0}\) and \(p_{G_T}\) */
4 \(H_0 = \text{distance}(p_{G_0}, p_{G_T})\);
/* 4. Initialize process variables */
5 \(i = 0\); // Accepted rewirings counter
6 \(\text{step} = 0\); // Counter associated with current temperature
7 \(\beta_{\text{step}} = \beta_0\);
8 \(A_{\text{step}} = 1\); // Acceptance ratio, percentage of accepted rewirings that modified energy at current temperature
/* 5. \(2k\)-rewire until energy goes to zero or the acceptance ratio falls below the threshold */
9 while \(H_i > 0\) and \(A_{\text{step}} > \alpha\) do
  /* Apply \(R \ p\)-targeting \(2k\)-rewirings at constant temperature \(1/\beta_{\text{step}}\) */
  for \(r \leftarrow 1\) to \(R\) do
    /* Apply a single \(2k\)-rewiring to the current graph, using Alg. 4 */
    10 \(G_{\text{hyp}} = \text{dk-randomize}(G_i, 1, 2)\);
    11 \(p_{\text{hyp}} = \text{compute-property}(G_{\text{hyp}})\);
    /* Compute current energy */
    12 \(H_{\text{hyp}} = \text{distance}(p_{\text{hyp}}, p_{G_T})\);
    /* Compute energy change */
    13 \(\Delta H = H_{\text{hyp}} - H_i\);
    /* Accept graph \(G_{\text{hyp}}\) with the following probability */
    with probability \(\min(e^{-\beta_{\text{step}} \cdot \Delta H}, 1)\) do:
      \(r = r + 1\);
      \(i = i + 1\);
      \(G_i = G_{\text{hyp}}\);
      \(H_i = H_{\text{hyp}}\);
  15 \(\beta_{\text{step}+1} = \beta_{\text{step}} \cdot \beta_{\text{factor}}\); // Update \(\beta\)
  16 \(A_{\text{step}+1} = \frac{\#\text{rewirings accepted with } \Delta H \neq 0}{\#\text{rewirings with } \Delta H \neq 0}\); // Update the acceptance ratio
  17 \(\text{step} = \text{step} + 1\);
Output: \(G_i\)
Table 2: The considered networks, their abbreviations, and the numbers of nodes and links in them.

| Network                                      | Abbr. | $N$  | $M$  |
|----------------------------------------------|-------|------|------|
| US air transportation network [87]           | AIR   | 500  | 2,980|
| Brain network [88]                           | BRAIN | 17,455| 67,895|
| English word network [89]                    | WORDS | 7,377| 44,205|
| Internet AS-level [90]                       | INTERNET | 20,906| 42,994|
| PGP web of trust [68]                        | PGP   | 10,680| 24,316|
| Protein interaction network [91]             | PPI   | 4,099| 13,355|

Considered networks

We apply the $dk$-series analysis to the following six social, biological, language, communication, and transportation networks, Table 2:

- **AIR.** The US air transportation network [87]. The nodes are airports, and there is a link between two airports if there is a direct flight between them.

- **BRAIN.** The largest connected component of an fMRI map of the human brain [88]. The nodes are voxels (small areas of a resting brain of approximately $36\text{mm}^3$ volume each), and two voxels are connected if the correlation coefficient of the fMRI activity of the voxels exceed 0.7.

- **WORDS.** The largest connected component of the network of adjacent words in Charles Darwin’s “The Origin of Species” [89]. The nodes are words, and two words are connected if they are adjacent in the text.

- **INTERNET.** The topology of the Internet at the level of Autonomous Systems (ASes) [90]. The nodes are ASs (organizations owning parts of the Internet infrastructure), and there is a link between two ASs if they have a business relationship to exchange Internet traffic.

- **PGP (considered in the main text).** The largest strongly connected component of the technosocial web of trust relationships among people extracted from the Pretty Good Privacy (PGP) data [68]. The nodes are PGP certificates of users, and there is a link between two certificates if their users mutually trust each other’s certificate/user associations.

- **PPI.** The largest connected component of the human protein interaction network [91]. The nodes are proteins, and there is a link between two proteins if they interact.

Table 3 reports the parameters used for each network in the $dk$-randomization and $p$-targeting $dk$-preserving rewiring processes.
Table 3: Parameters used for the $dk$-randomization (left) and $2.1k/2.5k$-targeting $2k$-preserving (right) rewiring processes ($M$ the number of edges in the real network, $\bar{c}$ average clustering, $\bar{c}(k)$ average clustering of nodes of degree $k$).

| $G_T$ | $dk$-randomization | $p$-targeting $dk$-preserving rewiring |
|-------|--------------------|---------------------------------------|
|       | $R$    | $d$ | $\rho_{G_T}$ | $d$ | $R$ | $\beta_0$ | $\beta_{factor}$ | $\alpha$ |
| AIR   | $100M$ | 1/2 | $\bar{c}$, $\bar{c}(k)$ | 2  | $10M$ | $10^{-2}$ | 1.4 | $5 \cdot 10^{-4}$ |
| BRAIN | $100M$ | 1/2 | $\bar{c}$, $\bar{c}(k)$ | 2  | $10M$ | $10^{-2}$ | 1.1 | $5 \cdot 10^{-5}$ |
| WORDS | $100M$ | 1/2 | $\bar{c}$, $\bar{c}(k)$ | 2  | $10M$ | $10^{-2} \cdot 10^{-4}$ | 1.4 | $5 \cdot 10^{-4}$ |
| INTERNET | $100M$ | 1/2 | $\bar{c}$, $\bar{c}(k)$ | 2  | $10M$ | $10^{-2}$ | 1.4 | $5 \cdot 10^{-4}$ |
| PGP   | $100M$ | 1/2 | $\bar{c}$, $\bar{c}(k)$ | 2  | $100M$ | $10^{-2}$ | 1.1 | $5 \cdot 10^{-7}$ |
| PPI   | $100M$ | 1/2 | $\bar{c}$, $\bar{c}(k)$ | 2  | $200M$ | $10^{-2}$ | 1.1 | $5 \cdot 10^{-7}$ |

Results

Degree distribution. We observe in Fig. 7 that while $0k$-randomizations are way off, the $dk$-random graphs with $d \geq 1$ reproduce the degree distributions in the real networks exactly, which is by definition: the $1k$-distribution is the degree distribution, and $dk$-random graphs with $d \geq 1$ have exactly the same degree distributions as the real networks.

Average nearest neighbor degree (ANND). We observe in Fig. 8 that while $0k$-randomizations are way off, the $1k$-random graphs tend to be closer to the real networks in terms of ANND, whereas the $dk$-random graphs with $d \geq 2$ have exactly the same average neighbor degrees as the real networks, which is again by definition: the $dk$-random graphs with $d \geq 2$ have exactly the same JDD $P(k,k')$ as the real networks. In the WORDS, INTERNET, and PPI cases, the ANNDs $k_{nn}(k)$ even in the $1k$-random graphs do not noticeably differ from the ANNDs in the real networks.

Clustering. We observe in Fig. 9 that degree-dependent average clustering in the $2.5k$-random graphs matches the one in the real networks, which is again by definition. For $d < 2.5$, degree-dependent clustering differs sensibly in many cases. However, degree-dependent clustering in the AIR network does not exhibit noticeable differences with its $2.1k$-randomizations, while in the WORDS case, even the $1k$-random graphs reproduce degree-dependent clustering nearly exactly.

Subgraph frequencies. We observe in Fig. 10 that the $2k$-random graphs reproduce the subgraphs frequencies in most cases, but the BRAIN and PGP require $d = 2.5$ to reproduce these frequencies.

Common neighbors. We observe in Fig. 11 that the $1k$-random graphs reproduce the common neighbor distributions in all the cases except the BRAIN, which requires $d = 2$, and PGP, which requires $d = 2.5$. 24
Table 4: Largest eigenvalues, averaged across different realizations for each $d$, and their standard deviations in parentheses.

|        | Original | $0k$      | $1k$      | $2k$      | $2.1k$ | $2.5k$          |
|--------|----------|-----------|-----------|-----------|--------|-----------------|
| AIR    | 48.07    | 12.97(0.08)| 42.41(0.24)| 47.46(0.01)| 47.51(0.02)| 47.82(0.03)    |
| BRAIN  | 119.66   | 8.91(0.01)| 54.89(0.26)| 113.41(0.02)| 114.09(0.06)| 122.27(0.20)  |
| WORDS  | 109.44   | 13.06(0.02)| 104.12(0.28)| 108.82(0.03)| 108.80(0.04)| 108.92(0.02)  |
| 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)   |
| PPI    | 38.56    | 8.05(0.05)| 32.47(0.17)| 34.07(0.04)| 34.05(0.04)| 35.56(0.10)   |

$k$-coreness and $k$-denseness. We observe in Fig. 12 that the $2k$-random graphs reproduce the $k$-coreness distributions in all the networks except the PGP and BRAIN that require $d = 2.5$. We observe in Fig. 13 that the $2.5k$-random graphs reproduce the $k$-denseness distributions in all the networks. The $k$-denseness distributions in the AIR and WORDS networks are reproduced even by their $2k$-random graphs.

Betweenness. We observe in Fig. 14 that betweenness in the BRAIN network cannot be approximated even by its $2.5k$-random graphs. The INTERNET lies at the other extreme: even the $1k$-random graphs reproduce its betweenness. The PGP network requires all the constraints imposed by the $2.5k$-distribution, while betweenness in all the other networks is similar to betweenness in their $2k$-random graphs.

Shortest path distance. We observe in Fig. 15 that the distance distributions in the INTERNET and WORDS networks are correctly reproduced by their $1k$-random graphs. Even $d = 2.5$ is not enough for the BRAIN, while the same value of $d = 2.5$ suffices for all the networks.

Spectral properties. We observe in Table 4 that the largest eigenvalue of the adjacency matrix is closely, although not exactly, reproduced by $d = 2.5k$-random graphs for all six networks. Furthermore, we observe that the largest eigenvalues for $2k$-random graphs of AIR and WORDS networks are very close to the eigenvalues of the original networks. The values of the spectral gaps for $2.5k$-random graphs shown in Table 5 are relatively close to the values observed for the original networks, with relative difference for AIR, BRAIN and WORDS networks around 5%. The large values of the spectral gaps for $2k$ and $2.1k$-random graphs indicate that they are more robust, in the sense of being better connected and interlinked, compared to the original networks.

Kolmogorov-Smirnov distance. In Fig. 16 we quantify the convergence of $dk$-series in terms of Kolmogorov-Smirnov (KS) distances between the distribu-
Table 5: Spectral gaps, averaged across different realizations for each $d$, and their standard deviations in parentheses.

|          | Original | 0K       | 1K       | 2K       | 2.1K      | 2.5K      |
|----------|----------|----------|----------|----------|-----------|-----------|
| AIR      | 29.34    | 6.04(0.09)| 32.61(0.46)| 37.86(0.25)| 37.21(0.21)| 30.93(0.29)|
| BRAIN    | 40.97    | 2.90(0.06)| 35.52(0.31)| 77.53(0.11)| 76.59(0.27)| 42.71(0.35)|
| WORDS    | 65.31    | 5.86(0.02)| 65.28(0.51)| 68.53(0.14)| 68.47(0.12)| 68.21(0.15)|
| 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)|
| PPI      | 11.69    | 2.25(0.07)| 15.75(0.27)| 16.44(0.19)| 16.28(0.20)| 10.76(0.17)|

The Kolmogorov-Smirnov distance between two cumulative distribution functions (CDFs) $F_1(x)$ and $F_2(x)$ is

$$D = \sup_x |F_1(x) - F_2(x)|.$$  \hspace{1cm} (23)

In our case, $F_1(x)$ is the per-node CDF of a given property in a real network, and $F_2(x)$ is the per-node CDF for the same property computed across all different $dk$-random graph realizations for the network with a given $d$. We note that the KS distances provides more detailed statistics than the $dk$-distributions, because the latter do not differentiate between nodes of the same degree, while the former do. For example, even if the $2k$-distributions and consequently ANNDs $\bar{k}_{nn}(k)$ in two different networks are exactly the same, the distributions of average degrees $\bar{k}_{i,nn}$ of neighbors of each individual node $i, i = 1, \ldots, N$, are in general different, so that the KS distance between the two per-node ANND CDFs is in general greater than zero.
References

[1] S N Dorogovtsev and J F F Mendes. *Evolution of Networks: From Biological Nets to the Internet and WWW.* Oxford University Press, Oxford, 2003.

[2] S Boccaletti, V Latora, Y Moreno, M Chavez, and D.-U. Hwanga. Complex Networks: Structure and Dynamics. *Phys Rep,* 424:175–308, 2006. doi: 10.1016/j.physrep.2005.10.009.

[3] M E J Newman. *Networks: An Introduction.* Oxford University Press, Oxford, 2010.

[4] Alain Barrat, Marc Barthélemy, and Alessandro Vespignani. *Dynamical Processes on Complex Networks.* Cambridge University Press, Cambridge, 2008.

[5] Guido Caldarelli and Alessandro Vespignani. *Large Scale Structure and Dynamics of Complex Networks.* World Scientific, Singapore, 2007.

[6] M E J Newman. The Structure and Function of Complex Networks. *SIAM Rev,* 45(2):167–256, 2003. doi:10.1137/S003614450342480.

[7] Z. Toroczkai and K.E. Bassler. Jamming is limited in scale-free systems. *Nature,* 428(6984):716, April 2004. doi:10.1038/428716a.

[8] U Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits.* Chapman & Hall/CRC, Boca Raton, 2006.

[9] Adilson E Motter and Zoltán Toroczkai. Introduction: optimization in networks. *Chaos,* 17(2):026101, July 2007. doi:10.1063/1.2751266.

[10] Marián Boguñá, Dmitri Krioukov, and Kc Claffy. Navigability of Complex Networks. *Nat Phys,* 5:74–80, 2009. doi:10.1038/NPHYS1130.

[11] Z Burda, A Krzywicki, O C Martin, and M Zagorski. Motifs emerge from function in model gene regulatory networks. *Proc Natl Acad Sci,* 108(42):17263–8, October 2011. doi:10.1073/pnas.1109435108.

[12] A Vázquez, R Dobrin, D Sergi, J-P Eckmann, Z N Oltvai, and A-L Barabási. The topological relationship between the large-scale attributes and local interaction patterns of complex networks. *Proc Natl Acad Sci,* 101(52):17940–5, December 2004. doi:10.1073/pnas.0406024101.

[13] Roger Guimerà, Marta Sales-Pardo, and Luis A. Amaral. Classes of complex networks defined by role-to-role connectivity profiles. *Nat Phys,* 3(1):63–69, January 2007. doi:10.1038/nphys489.

[14] K Takemoto, C Oosawa, and T Akutsu. Structure of n-clique networks embedded in a complex network. *Physica A,* 380:665–672, July 2007. doi:10.1016/j.physa.2007.02.042.
[15] David V. Foster, Jacob G. Foster, Peter Grassberger, and Maya Paczuski. Clustering drives assortativity and community structure in ensembles of networks. *Phys Rev E*, 84(6):066117, December 2011. doi:10.1103/PhysRevE.84.066117.

[16] Pol Colomer-de Simón, M. Ángeles Serrano, Mariano G Beiró, José Ignacio Alvarez-Hamelin, and Marián Boguñá. Deciphering the global organization of clustering in real complex networks. *Sci Rep*, 3:2517, January 2013. doi:10.1038/srep02517.

[17] R Milo, S Shen-Orr, S Itzkovitz, N Kashtan, D Chklovskii, and U Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–7, October 2002. doi:10.1126/science.298.5594.824.

[18] U Alon. Network Motifs: Theory and Experimental Approaches. *Nat Rev Genet*, 8:450–461, 2007. doi:10.1038/nrg2102.

[19] M. Ercsey-Ravasz, N.T. Markov, C. Lamy, D.C. Van Essen, K. Knoblauch, Z. Toroczkai, and H. Kennedy. A predictive network model of cerebral cortical connectivity based on a distance rule. *Neuron*, 80(1):184–197, 2013. doi:10.1016/j.neuron.2013.07.036.

[20] N.T Markov, M. Ercsey-Ravasz, D.C. Van Essen, K. Knoblauch, Z. Toroczkai, and H. Kennedy. Cortical high-density counterstream architectures. *Science*, 342(6158):1238406, November 2013. doi:10.1126/science.1238406.

[21] Luis A. Amaral and Roger Guimera. Complex networks: Lies, damned lies and statistics. *Nat Phys*, 2(2):75–76, February 2006. doi:10.1038/nphys228.

[22] Vittoria Colizza, Alessandro Flammini, M. Ángeles Serrano, and Alessandro Vespignani. Detecting rich-club ordering in complex networks. *Nat Phys*, 2(2):110–115, January 2006. doi:10.1038/nphys209.

[23] J. Trevino III, A. Nyberg, C.I. Del Genio, and K.E. Bassler. Fast and accurate determination of modularity and its effect size. *J Stat Mech*, 15:P02003, February 2015. doi:10.1088/1742-5468/2015/02/P02003.

[24] László Lovász. *Large Networks and Graph Limits*. American Mathematical Society, Providence, RI, 2012.

[25] Priya Mahadevan, Dmitri Krioukov, Kevin Fall, and Amin Vahdat. Systematic Topology Analysis and Generation Using Degree Correlations. *Comput Commun Rev*, 36(4):135–146, 2006. doi:10.1145/1151659.1159930.

[26] Natasa Pržulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):e177–83, January 2007. doi:10.1093/bioinformatics/btl301.
[27] Ömer Nebil Yaveroğlu, Noël Malod-Dognin, Darren Davis, Zoran Levnajic, Vuk Janjic, Rasa Karapandza, Aleksandar Stojmirovic, and Nataša Pržulj. Revealing the hidden language of complex networks. Sci Rep, 4:4547, January 2014. doi:10.1038/srep04547

[28] B Karrer and M E J Newman. Random graphs containing arbitrary distributions of subgraphs. Phys Rev E, 82(6):066118, December 2010. doi:10.1103/PhysRevE.82.066118

[29] P Erdős and A Rényi. On Random Graphs. Publ Math, 6:290–297, 1959.

[30] E Bender and E Canfield. The Asymptotic Number of Labeled Graphs with Given Degree Distribution. J Comb Theory A, 24:296–307, 1978.

[31] M Molloy and B Reed. A Critical Point for Random Graphs With a Given Degree Sequence. Random Struct Algorithms, 6:161–179, 1995.

[32] M E J Newman, S H Strogatz, and D J Watts. Random Graphs with Arbitrary Degree Distributions and Their Applications. Phys Rev E, 64:26118, 2001. doi:10.1103/PhysRevE.64.026118

[33] Sourav Chatterjee, Persi Diaconis, and Allan Sly. Random graphs with a given degree sequence. Ann Appl Probab, 21(4):1400–1435, August 2011. doi:10.1214/10-AAP728

[34] Isabelle Stanton and Ali Pinar. Constructing and sampling graphs with a prescribed joint degree distribution. J Exp Algorithmics, 17(1):3.1, July 2012. doi:10.1145/2133803.2330086

[35] Ginestra Bianconi. The entropy of randomized network ensembles. Eur Lett, 81(2):28005, January 2008. doi:10.1209/0295-5075/81/28005

[36] Kartik Anand and Ginestra Bianconi. Entropy Measures for Networks: Toward an Information Theory of Complex Topologies. Phys Rev E, 80:045102(R), 2009. doi:10.1103/PhysRevE.80.045102

[37] Ginestra Bianconi, Paolo Pin, and Matteo Marsili. Assessing the relevance of node features for network structure. Proc Natl Acad Sci USA, 106(28):11433–8, July 2009. doi:10.1073/pnas.0811511106

[38] A Barvinok and J A Hartigan. The number of graphs and a random graph with a given degree sequence. Random Struct Algorithms, 42(3):301–348, May 2013. doi:10.1002/rsa.20409

[39] Paul W Holland and Samuel Leinhardt. An Exponential Family of Probability Distributions for Directed Graphs. J Am Stat Assoc, 76(373):33–50, 1981.

[40] Marián Boguñá and Romualdo Pastor-Satorras. Class of Correlated Random Networks with Hidden Variables. Phys Rev E, 68:36112, 2003. doi:10.1103/PhysRevE.68.036112
[41] J Park and M E J Newman. Statistical Mechanics of Networks. *Phys Rev E*, 70:66117, 2004. [doi:10.1103/PhysRevE.70.066117]

[42] Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguñá. Hyperbolic Geometry of Complex Networks. *Phys Rev E*, 82:36106, 2010. [doi:10.1103/PhysRevE.82.036106]

[43] Sourav Chatterjee and Persi Diaconis. Estimating and understanding exponential random graph models. *Ann Stat*, 41(5):2428–2461, October 2013. [doi:10.1214/13-AOS1155]

[44] Sz. Horvát, É. Czabarka, and Z. Toroczkai. Reducing Degeneracy in Maximum Entropy Models of Networks. *Phys Rev Lett*, 114(15-17):158701, April 2015. [doi:10.1103/PhysRevLett.114.158701]

[45] M G Beiró, J I Alvarez-Hamelin, and J R Busch. A low complexity visualization tool that helps to perform complex systems analysis. *New J Phys*, 10(12):125003, December 2008. [doi:10.1088/1367-2630/10/12/125003]

[46] Diego Garlaschelli and Maria Loffredo. Maximum likelihood: Extracting unbiased information from complex networks. *Phys Rev E*, 78(1):015101, July 2008. [doi:10.1103/PhysRevE.78.015101]

[47] Tiziano Squartini and Diego Garlaschelli. Analytical maximum-likelihood method to detect patterns in real networks. *New J Phys*, 13(8):083001, August 2011. [doi:10.1088/1367-2630/13/8/083001]

[48] Tiziano Squartini, Rossana Mastrandrea, and Diego Garlaschelli. Unbiased sampling of network ensembles. *New J Phys*, 17(2):023052, 2015. [doi:10.1088/1367-2630/17/2/023052]

[49] Sergei Maslov, Kim Sneppen, and Uri Alon. *Handbook of Graphs and Networks*, chapter 8. Wiley-VCH, Berlin, 2003.

[50] Sergei Maslov, Kim Sneppen, and Alexei Zaliznyak. Detection of topological patterns in complex networks: Correlation profile of the Internet. *Physica A*, 333:529–540, February 2004. [doi:10.1016/j.physa.2003.06.002]

[51] Minas Gjoka, Maciej Kurant, and Athina Markopoulou. 2.5K-graphs: From sampling to generation. In *2013 Proc IEEE INFOCOM*, pages 1968–1976. IEEE, April 2013. [doi:10.1109/INFCOM.2013.6566997]

[52] H. Kim, Z. Toroczkai, P.L. Erdős, I. Miklós, and L.A. Székely. Degree-based graph construction. *J Phys A Math Theor*, 42(39):392001, October 2009. [doi:10.1088/1751-8113/42/39/392001]

[53] C.I Del Genio, H. Kim, Z. Toroczkai, and K.E. Bassler. Efficient and exact sampling of simple graphs with given arbitrary degree sequence. *PLoS One*, 5(4):e10012, January 2010. [doi:10.1371/journal.pone.0010012]
[54] H. Kim, C.I. Del Genio, K.E. Bassler, and Z. Toroczkai. Constructing
and sampling directed graphs with given degree sequences. New J Phys,
14:023012, 2012. doi:10.1088/1367-2630/14/2/023012.

[55] K.E. Bassler, C.I. Del Genio, P.L. Erdős, I. Miklós, and Z. Toroczkai. Exact
sampling of graphs with prescribed degree correlations, 2015. arXiv:1503.06725

[56] V. Zlatic, G. Bianconi, A. Díaz-Guilera, D. Garlaschelli, F. Rao, and
G. Caldarelli. On the rich-club effect in dense and weighted networks.
Eur Phys J B, 67(3):271–275, 2009. doi:10.1140/epjb/e2009-00007-9

[57] Éva Czabarka, Aaron Dutle, Péter L. Erdős, and István Miklós. On real-
zizations of a joint degree matrix. Discret Appl Math, 181:283–288, January
2015. doi:10.1016/j.dam.2014.10.012

[58] Almerima Jamakovic, Priya Mahadevan, Amin Vahdat, Marián Boguñá,
and Dmitri Krioukov. How small are building blocks of complex networks,
August 2009. arXiv:0908.1143

[59] A C C Coolen, A Martino, and A Annibale. Constrained Markovian Dynamics
of Random Graphs. J Stat Phys, 136(6):1035–1067, September
2009. doi:10.1007/s10955-009-9821-2

[60] A Annibale, A C C Coolen, L Fernandes, F Fraternali, and J Kleinjung.
Tailored graph ensembles as proxies or null models for real networks I: tools
for quantifying structure. J Phys A-Math Gen, 42(48):485001, December
2009. doi:10.1088/1751-8113/42/48/485001

[61] R. Milo, N. Kashtan, S. Itzkovitz, M E J Newman, and U. Alon. On the
uniform generation of random graphs with prescribed degree sequences,
December 2003. arXiv:0312028

[62] Charo Del Genio, Thilo Gross, and Kevin E Bassler. All Scale-Free Net-
works Are Sparse. Phys Rev Lett, 107(17):1–4, October 2011. doi:
10.1103/PhysRevLett.107.178701

[63] Minas Gjoka, Balint Tillman, and Athina Markopoulou. Construction of
Simple Graphs with a Target Joint Degree Matrix and Beyond. In 2015
Proc IEEE INFOCOM, page (to appear). IEEE, 2015.

[64] S N Dorogovtsev, J. Mendes, and A. Samukhin. Size-dependent degree
distribution of a scale-free growing network. Phys Rev E, 63(6):062101,
May 2001. doi:10.1103/PhysRevE.63.062101

[65] Konstantin Klemm and Víctor Eguíluz. Highly clustered scale-free net-
works. Phys Rev E, 65(3):036123, February 2002. doi:10.1103/PhysRevE.65.036123
[66] Alexei Vázquez. Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. *Phys Rev E*, 67(5):056104, May 2003. doi:10.1103/PhysRevE.67.056104.

[67] Alexei Vázquez, Marián Boguñá, Yamir Moreno, Romualdo Pastor-Satorras, and Alessandro Vespignani. Topology and correlations in structured scale-free networks. *Phys Rev E*, 67(4):046111, April 2003. doi:10.1103/PhysRevE.67.046111.

[68] Marián Boguñá, Romualdo Pastor-Satorras, Albert Díaz-Guilera, and Alex Arenas. Models of social networks based on social distance attachment. *Phys Rev E*, 70(5):056122, November 2004. doi:10.1103/PhysRevE.70.056122.

[69] M. Ángeles Serrano and Marián Boguñá. Tuning clustering in random networks with arbitrary degree distributions. *Phys Rev E*, 72(3):036133, September 2005. doi:10.1103/PhysRevE.72.036133.

[70] Fragkiskos Papadopoulos, Maksim Kitsak, M. Ángeles Serrano, Marián Boguñá, and Dmitri Krioukov. Popularity versus similarity in growing networks. *Nature*, 489:537–540, September 2012. doi:10.1038/nature11459.

[71] Ginestra Bianconi, Richard K. Darst, Jacopo Iacovacci, and Santo Fortunato. Triadic closure as a basic generating mechanism of communities in complex networks. *Phys Rev E*, 90:042806, 2014. doi:10.1103/PhysRevE.90.042806.

[72] P Kuo, W Banzhaf, and A Leier. Network Topology and the Evolution of Dynamics in an Artificial Genetic Regulatory Network Model Created by Whole Genome Duplication and Divergence. *Biosystems*, 85:177–200, 2006. doi:10.1016/j.biosystems.2006.01.004.

[73] A Medina, A Lakhina, I Matta, and J Byers. BRITE: An approach to universal topology generation. In *MASCOTS 2001, Proc Ninth Int Symp Model Anal Simul Comput Telecommun Syst*, pages 346–353, 2001. doi:10.1109/MASCOT.2001.948886.

[74] Xenofontas Dimitropoulos, Dmitri Krioukov, George Riley, and Amin Vahdat. Graph Annotations in Modeling Complex Network Topologies. *ACM T Model Comput S*, 19(4):17, 2009. doi:10.1145/1596519.1596522.

[75] David Foster, Jacob Foster, Maya Paczuski, and Peter Grassberger. Communities, clustering phase transitions, and hysteresis: Pitfalls in constructing network ensembles. *Phys Rev E*, 81(4):046115, April 2010. doi:10.1103/PhysRevE.81.046115.

[76] E S Roberts and A C C Coolen. Random Graph Ensembles with Many Short Loops. In *ESAIM Proc Surv*, volume 47, pages 97–115, 2014. doi:10.1051/proc/201447006.
[89] R Milo, S Itzkovic, N Kashtan, R Levitt, S Shen-Orr, I Ayzenshtat, M Sheffer, and U Alon. Superfamilies of Evolved and Designed Networks. *Science*, 303:1538–1542, 2004. [doi:10.1126/science.1089167](https://doi.org/10.1126/science.1089167).

[90] Priya Mahadevan, Dmitri Krioukov, Marina Fomenkov, Bradley Huffaker, Xenofontas Dimitropoulos, Kc Claffy, and Amin Vahdat. The Internet AS-Level Topology: Three Data Sources and One Definitive Metric. *Comput Commun Rev*, 36(1):17–26, 2006. [doi:10.1145/1111322.1111328](https://doi.org/10.1145/1111322.1111328).

[91] Thomas Rolland, Murat Tas, Nidhi Sahni, Song Yi, Irina Lemmens, Celia Fontanillo, Roberto Mosca, Atanas Kamburov, Susan D Ghiassian, Xiping Yang, Lila Ghamsari, Dawit Balcha, Bridget E Begg, Pascal Braun, Marc Brehme, Martin P Broly, Anne-ruxandra Carvunis, Dan Converyzupan, Roser Corominas, Changyu Fan, Eric Franzosa, Jasmin Coulombehuntington, Elizabeth Dann, Matija Dreze, Fana Gebreab, Bryan J Gutierrez, Madeleine F Hardy, Mike Jin, Shuli Kang, Ruth Kiros, Guan Ning Lin, Ryan R Murray, Alexandre Palagi, Matthew M Poulin, Katja Luck, Andrew Macwilliams, Xavier Rambout, John Rasla, Patrick Reichert, Viviana Romero, Elien Ruyssinck, Julie M Sahalie, Annemarie Scholz, Akash a Shah, Amitabh Sharma, Yun Shen, Kerstin Spirohn, Stanley Tam, Alexander O Tejeda, Shelly a Trigg, Jean-claude Twizere, Kerwin Vega, and Jennifer Walsh. A Proteome-Scale Map of the Human Interactome Network. *Cell*, 159:1212–1226, 2014. [doi:10.1016/j.cell.2014.10.050](https://doi.org/10.1016/j.cell.2014.10.050).
Figure 3: Basic microscopic (local) properties of the PGP network and its dk-random graphs. The figure shows the degree distribution $P(k)$, average degree $\bar{k}_{nn}(k)$ of nearest neighbors of nodes of degree $k$, average clustering $\bar{c}(k)$ of nodes of degree $k$, the distribution $P(m)$ of the number $m$ of common neighbors between all connected pairs of nodes, and the means and standard deviations of the corresponding distributions. The error bars in this and subsequent figures indicate the standard deviations of the metrics across different graph realizations.
Figure 4: The densities of subgraphs of size 3 and 4 in the PGP network and its \(dk\)-random graphs. The two different graphs of size 3 and six different graphs of size 4 are shown on each panel. The numbers on top of panels are the concentrations of the corresponding subgraph in the PGP network, while the histogram heights indicate the average absolute difference between the subgraph concentration in the \(dk\)-random graphs and its concentration in the PGP network. The subgraph concentration is the number of given subgraphs divided by the total number of subgraphs of the same size.
Figure 5: Basic mesoscopic properties, the $k$-coreness and $k$-density distributions, in the PGP network and its $dk$-random graphs. The figure shows the distributions $P(k_{c,d})$ of node $k$-coreness $k_c$ and edge $k$-density $k_d$, and their means and standard deviations. The $k_c$-core of a graph is its maximal subgraph in which all nodes have degree at least $k_c$. The $k_d$-core of a graph is its maximal subgraph in which all edges have multiplicity at least $k_d - 2$; the multiplicity of an edge is the number of common neighbors between the nodes that this edge connects, or equivalently the number of triangles that this edge belongs to. A node has $k$-coreness $k_c$ if it belongs to the $k_c$-core but not to the $k_c + 1$-core. An edge has $k$-density $k_d$ if it belongs to the $k_d$-core but not to the $k_d + 1$-core.
Figure 6: **Basic macroscopic (global) properties of the PGP network and its dk-random graphs.** The figure shows the average betweenness $b(k)$ of nodes of degree $k$, the distribution $P(l)$ of hop lengths $l$ of the shortest paths between all pairs of nodes, the means and standard deviations of the corresponding distributions, the largest eigenvalues of the adjacency matrix $A$, and the Fiedler value, which is the spectral gap (the second largest eigenvalue) of the graph’s Laplacian matrix $L = D - A$, where $D$ is the degree matrix, $D_{ij} = \delta_{ij} k_i$, $\delta_{ij}$ the Kronecker delta, and $k_i$ the degree of node $i$. 
Figure 7: Degree distributions in real networks and their $dk$-randomizations.
Figure 8: Average nearest neighbor degrees (ANNDs) of nodes of a given degree in real networks and their $dk$-randomizations.
Figure 9: Average clustering of nodes of a given degree in real networks and their $dk$-randomizations.
Figure 10: Subgraph concentration differences between $dk$-randomizations and real networks.
Figure 11: Common neighbor distributions in real networks and their $dk$-randomizations.
Figure 12: $k$-coreness distributions in real networks and their $dk$-randomizations.
Figure 13: $k$-denseness distributions in real networks and their $dk$-randomizations.
Figure 14: Average betweenness of nodes of a given degree in real networks and their $dk$-randomizations.
Figure 15: Shortest path distance distributions in real networks and their $dk$-randomizations.
Figure 16: Kolmogorov-Smirnov distances between real networks and their $dk$-randomizations.