Modularity with a more accurate baseline model

Chang, Brian L.; Van Mieghem, Piet

DOI
10.1103/PhysRevE.111.044317

Publication date
2025

Document Version
Final published version

Published in
Physical Review E

Citation (APA)
Chang, B. L., & Van Mieghem, P. (2025). Modularity with a more accurate baseline model. *Physical Review E*, *111*(4), Article 044317. https://doi.org/10.1103/PhysRevE.111.044317

Modularity with a more accurate baseline model

Brian L. Chang [*] and Piet Van Mieghem
Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology,
P.O. Box 5031, 2600 GA Delft, The Netherlands

(Received 29 November 2024; accepted 2 April 2025; published 25 April 2025)

We derive an expression for the exact probability $\Pr[i \sim j]$ of a link between a node i with degree d_i and a node j with degree d_j in a graph belonging to the class of Erdős-Rényi $G(N, L)$ random graphs with N nodes and L links. The probability $\Pr[i \sim j]$ is commonly approximated as $\frac{d_i d_j}{2L}$ and appears in the formula of Newman's modularity, which plays a crucial rule in community detection in networks. We show that, when applied to graphs not belonging to the class of Erdős-Rényi random graphs, our formula for $\Pr[i \sim j]$ is considerably more accurate than $\frac{d_i d_j}{2L}$ and leads to the detection of different clusters or partitions than the original modularity formula.

DOI: 10.1103/PhysRevE.111.044317

I. INTRODUCTION

The probability that two nodes i and j (where $i \neq j$) are connected in a random graph with L links is commonly given [[1], Eq. (4.24)] by

$$\Pr[i \sim j] = \frac{d_i d_j}{2L-1}, \tag{1}$$

where d_i and d_j are the degrees of node i and node j, respectively. In the absence of further qualification, (1) is demonstrably false; it is trivial to construct examples where (1) results in a probability greater than 1 as shown in Fig. 1.

In fact, (1) is actually the expected number of links between node i and node j in the configuration model ([2], Chap. 12.1.1). In the configuration model, we start with a degree sequence $(d_1, d_2, \ldots, d_N)$ on N nodes. Each node i has d_i half-links, called stubs, and the total number of stubs is $\sum_{i=1}^{N} d_i = 2L$. To construct the network, each stub is randomly paired with another stub until no stubs remain. Each random pairing of stubs is a link in the network. Importantly, the configuration model allows for self-loops and multilinks and will not necessarily generate a simple graph [a graph in which there can be at most one link between node i and node j and there are no self-loops ([3], Art. 1)].

Consider a pair of nodes i and j with degree d_i and d_j, respectively. Consider any stub of node i; what is the probability that this stub is connected to node j? Excluding the stub we are considering, there are $2L-1$ remaining stubs in the network of which d_j belong to node j; hence, the probability that the chosen stub is connected to node j is $\frac{d_j}{2L-1}$. Since node i has d_i stubs, the expected number of links between node i and node j is

$$\mathbb{E}[a_{i,j}]_{\text{CM}} = \frac{d_i d_j}{2L-1}, \tag{2}$$

where the subscript CM indicates the configuration model. In the configuration model, the entries $a_{i,j}$ of the adjacency matrix are not Bernoulli random variables, because there can be more than one link between node i and node j. Hence, the expected number of links $\mathbb{E}[a_{i,j}]_{\text{CM}}$ upper bounds the probability $\Pr[i \sim j]_{\text{CM}}$ that node i and node j are connected:

$$\begin{aligned}\mathbb{E}[a_{i,j}]_{\text{CM}} &= \sum_{k=0}^{\infty} k \Pr[a_{i,j} = k]_{\text{CM}} \\ &\geqslant \sum_{k=1}^{\infty} \Pr[a_{i,j} = k]_{\text{CM}} = \Pr[i \sim j]_{\text{CM}}.\end{aligned} \tag{3}$$

If the second moment of a random degree D is constant and finite, $\mathbb{E}[D^2] < \infty$, then the probability of observing multi-links and self-loops is of order $O(\frac{1}{N})$, as shown in [2], pp. 374–375. Since $\frac{1}{2L-1} = \frac{1}{2L}[1 + O(\frac{1}{L})]$, for large size N and large number L of links, we find approximately

$$\Pr[i \sim j]_{\text{CM}} \simeq \mathbb{E}[a_{i,j}]_{\text{CM}} \simeq \frac{d_i d_j}{2L}. \tag{4}$$

The asymptotic (4) is conditioned on a degree distribution with a finite second moment and a sufficiently large network. However, in real networks, the degree distribution may follow a power-law distribution in which the second moment diverges, i.e., does not exist. Real networks are also finite in size N. In this work, we compute the exact link probability $\Pr[i \sim j]$ for simple graphs.

II. EXACT PROBABILITY OF A LINK IN A SIMPLE RANDOM GRAPH

Consider the adjacency matrix A of a simple graph G with N nodes. An example adjacency matrix A for $N = 6$ nodes is illustrated in Fig. 2. In a simple graph, there is at most one link between a pair of nodes i and j. The off-diagonal entries $a_{i,j}$ of the adjacency matrix A of a simple random graph are Bernoulli random variables, where $a_{i,j} = 1$ if there is a link

[*] Contact author: b.l.chang@tudelft.nl

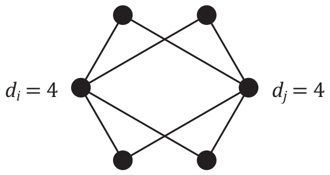


FIG. 1. A graph on $N = 6$ nodes with $L = 8$ links. The degrees of node i and node j are $d_i = 4$ and $d_j = 4$. Applying (1) yields $\Pr[i \sim j] = \frac{16}{15} > 1$. A probability cannot be greater than 1; furthermore, node i and node j are not connected.

between node i and node j, and $a_{i,j} = 0$ otherwise. There are no self-loops in a simple graph, which means that the diagonal entries are always $a_{i,i} = 0$. Because the adjacency matrix A is symmetric, a simple graph G is fully described by the elements of the upper triangle (excluding the main diagonal) of the adjacency matrix. The upper triangle has $L_{\max} = \binom{N}{2} = \frac{N(N-1)}{2}$ entries $a_{i,j}$ that corresponds to the maximum number of links in a simple graph of N nodes.

Suppose the graph G is a realization of the class of Erdős-Rényi $G(N, L)$ random graphs, in which L links are placed uniformly at random in the graph of N nodes. We define the set $\mathcal{G}_{N,L}$ as the set of all possible graphs[1] with N nodes and L links. The graph G is, therefore, chosen uniformly from the set $\mathcal{G}_{N,L}$. The number of possible graphs is $|\mathcal{G}_{N,L}| = \binom{L_{\max}}{L}$, because precisely L entries are $a_{i,j} = 1$ in the upper triangle of the adjacency matrix A.

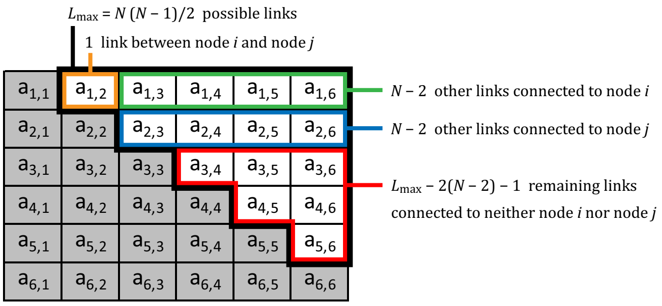
Consider a pair of nodes (i, j) in the graph G. Given the degree d_i of node i and the degree d_j of node j, what is the probability that node i and node j are connected? The set $\mathcal{G}_{N,L,(d_i,d_j)}$ denotes the set of graphs with N nodes and L links, where the node pair (i, j) has the corresponding degree pair (d_i, d_j). We partition the set of graphs $\mathcal{G}_{N,L,(d_i,d_j)}$ based on whether or not there is a link between the node pair (i, j),

$$\mathcal{G}_{N,L,(d_i,d_j)} = \mathcal{G}_{N,L,(d_i,d_j),i\sim j} \cup \mathcal{G}_{N,L,(d_i,d_j),i\nsim j}, \tag{5}$$

where $i \sim j$ denotes that the node pair (i, j) is connected by a link and $i \nsim j$ denotes that the node pair (i, j) is not connected. Since $\mathcal{G}_{N,L,(d_i,d_j)}$ is a subset of $\mathcal{G}_{N,L}$ and every graph in $\mathcal{G}_{N,L}$ occurs with equal probability, the probability $\Pr[i \sim j]$ that the node pair (i, j) is connected is given by

$$\Pr[i \sim j] = \frac{\left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right|}{\left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right| + \left|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}\right|}. \tag{6}$$

The probability $\Pr[i \sim j]$ is defined only if $|\mathcal{G}_{N,L,(d_i,d_j)}| > 0$, which clearly must be true: since the degree pair (d_i, d_j) corresponds to the degrees of a node pair (i, j) in a $G(N, L)$ graph, there must exist at least one graph with the parameters $\{N, L, (d_i, d_j)\}$. In other words, the parameters $\{N, L, (d_i, d_j)\}$ are *graphical*, because they can be realized by a simple graph, which means that they satisfy the constraints described in Appendix A 1. In our derivation, we will assume both $|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}| > 0$ and $|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}| > 0$, which is a stricter constraint that excludes parameters $\{N, L, (d_i, d_j)\}$ for which $\Pr[i \sim j] = 1$ or $\Pr[i \sim j] = 0$. In Appendix A 2, we show that this assumption has no impact on our final result (9), which yields the correct probability for all graphical parameters $\{N, L, (d_i, d_j)\}$.


$L_{\max} = N(N-1)/2$ possible links
1 link between node i and node j
$N-2$ other links connected to node i
$N-2$ other links connected to node j
$L_{\max} - 2(N-2) - 1$ remaining links connected to neither node i nor node j

$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$	$a_{1,6}$
$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{2,6}$
$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$	$a_{3,6}$
$a_{4,1}$	$a_{4,2}$	$a_{4,3}$	$a_{4,4}$	$a_{4,5}$	$a_{4,6}$
$a_{5,1}$	$a_{5,2}$	$a_{5,3}$	$a_{5,4}$	$a_{5,5}$	$a_{5,6}$
$a_{6,1}$	$a_{6,2}$	$a_{6,3}$	$a_{6,4}$	$a_{6,5}$	$a_{6,6}$

FIG. 2. Illustration of an adjacency matrix for $N = 6$ showing the possible links. We define node $i = 1$ and node $j = 2$. The gray entries do not need to be considered because the graph is simple; the main diagonal is 0 and the matrix symmetric.

Consider the adjacency matrix A in Fig. 2, where we have defined node $i = 1$ and node $j = 2$. If the node pair (i, j) is not connected, the entry $a_{i,j} = 0$ in the adjacency matrix A (shown in orange in Fig. 2). We need to connect d_i links to node i and there are $N - 2$ possible entries to choose from (shown in green). Similarly for node j, we need to connect d_j links and there are $N - 2$ possible entries to choose from (shown in blue). There are $L_{\max} - 2(N - 2) - 1$ remaining entries in the adjacency matrix A (shown in red). Since the total number of links is L, we still need to place $L - d_i - d_j$ links in the rest of the graph. Hence, the total number of graphs in which the node pair (i, j) is not connected is given by

$$\left|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}\right| = \binom{N-2}{d_i}\binom{N-2}{d_j}\binom{L_{\max} - 2(N-2) - 1}{L - d_i - d_j}. \tag{7}$$

Suppose now that the pair of nodes (i, j) is connected. Since the entry $a_{i,j} = 1$, we need to connect $d_i - 1$ additional links to node i and $d_j - 1$ additional links to node j. Since the total number of links is L, we still need to place $L - d_i - d_j + 1$ links in the rest of the graph. Hence, the total number of graphs in which the node pair (i, j) is connected is given by

$$\left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right| = \binom{N-2}{d_i-1}\binom{N-2}{d_j-1}\binom{L_{\max} - 2(N-2) - 1}{L - d_i - d_j + 1}. \tag{8}$$

Substituting (7) and (8) into (6) and simplifying (Appendix A 3) yields

$$\Pr[i \sim j] = \frac{d_i d_j \left(L^c - d_i^c - d_j^c + 1\right)}{d_i d_j \left(L^c - d_i^c - d_j^c + 1\right) + d_i^c d_j^c (L - d_i - d_j + 1)}, \tag{9}$$

where $d^c = (N - 1) - d$ and $L^c = L_{\max} - L$ and the superscript "c" refers to the complement of the graph ([3],

[1] We consider each node to be labeled; therefore, isomorphic graphs are different graphs.

Art. 1). Our expression (9) for the probability $\Pr[i \sim j]$ is exact and holds for all random graphs where L links are placed uniformly at random on $N \geqslant 2$ nodes, i.e., for the class of Erdős-Rényi $G(N, L)$ random graphs (Appendix A 2). Increasing the degree d increases the probability of being connected. If either node i or node j has degree $d = 0$, then the numerator becomes zero and $\Pr[i \sim j] = 0$. If either node i or node j has degree $d = N - 1$, then $d^{c} = 0$ and the second term in the denominator becomes zero and $\Pr[i \sim j] = 1$.

Increasing the number of links L decreases the probability of being connected. As derived in Appendix A 1, the minimum number of links given that $d_i, d_j > 0$ is $L = d_i + d_j - 1$; the second term in the denominator becomes zero and $\Pr[i \sim j] = 1$. The maximum number of links given that $d_i, d_j < N - 1$ is $L = L_{\max} - (d_i^{c} + d_j^{c} - 1)$, which means that $L^{c} = d_i^{c} + d_j^{c} - 1$ and that the numerator becomes zero and $\Pr[i \sim j] = 0$.

III. ERROR WHEN USING $\mathbb{E}[a_{i,j}]_{\text{CM}}$ TO ESTIMATE $\Pr[i \sim j]$

We consider the error when using the expected number of links $\mathbb{E}[a_{i,j}]_{\text{CM}}$ in the configuration model (2) as an estimate for the connection probability $\Pr[i \sim j]$ in a simple graph (9). Instead of the relative error, we define an error factor ϵ to quantify the extent to which $\mathbb{E}[a_{i,j}]_{\text{CM}}$ overestimates or underestimates $\Pr[i \sim j]$. The error factor ϵ is defined as

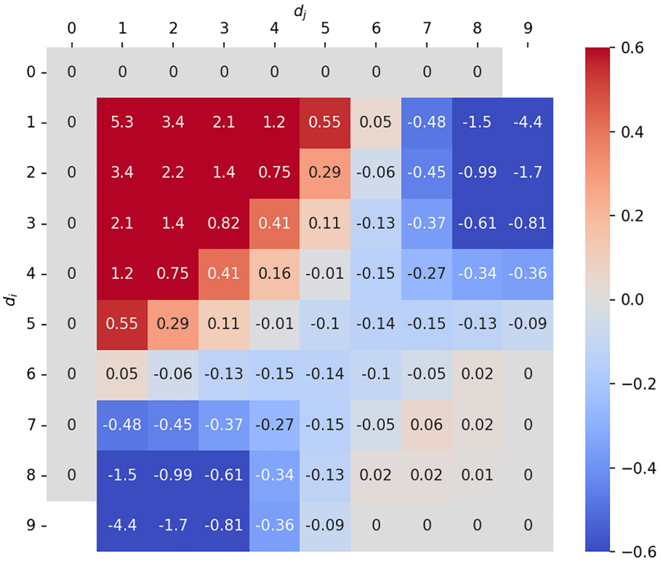
$$\epsilon = \begin{cases} \frac{\min(1, \mathbb{E}[a_{i,j}]_{\text{CM}})}{\Pr[i \sim j]} - 1 & \text{if } \min(1, \mathbb{E}[a_{i,j}]_{\text{CM}}) > \Pr[i \sim j] \\ 1 - \frac{\Pr[i \sim j]}{\min(1, \mathbb{E}[a_{i,j}]_{\text{CM}})} & \text{if } \min\left(1, \mathbb{E}[a_{i,j}]_{\text{CM}}\right) < \Pr[i \sim j] \\ 0 & \text{if } \min(1, \mathbb{E}[a_{i,j}]_{\text{CM}}) = \Pr[i \sim j]. \end{cases} \tag{10}$$

An error factor $\epsilon = +1$ means that the estimate $\mathbb{E}[a_{i,j}]_{\text{CM}}$ is double the true probability $\Pr[i \sim j]$, and $\epsilon = -1$ means that the estimate $\mathbb{E}[a_{i,j}]_{\text{CM}}$ is half of the true probability $\Pr[i \sim j]$. We take the minimum $\min(1, \mathbb{E}[a_{i,j}]_{\text{CM}})$ so that estimates $\mathbb{E}[a_{i,j}]_{\text{CM}} > 1$ are treated as a probability of 1 and are not further penalized.

Figure 3 shows a heatmap of the error factor ϵ for the class of graphs with $N = 10$ nodes and $L = 25$ links. A fully red cell indicates an error factor $\epsilon \geqslant 0.6$ and a fully blue cell indicates an error factor $\epsilon \leqslant 0.6$. The degree pairs (0, 9) and (9, 0) are absent in the heatmap because it is impossible to have degree $d = N - 1$ and degree $d = 0$ in the same graph. If node i or node j has degree $d = 0$, then $\mathbb{E}[a_{i,j}]_{\text{CM}} = \Pr[i \sim j] = 0$; hence, there is no error. If node i has close to the maximum degree while node j has a low degree, then $\mathbb{E}[a_{i,j}]_{\text{CM}}$ severely underestimates the probability $\Pr[i \sim j]$. If both d_i and d_j are low, then $\mathbb{E}[a_{i,j}]_{\text{CM}}$ severely overestimates the probability $\Pr[i \sim j]$. Hence, on small networks, $\mathbb{E}[a_{i,j}]_{\text{CM}}$ deviates significantly from the probability $\Pr[i \sim j]$.

IV. LIMIT FOR LARGE N

We define the normalized degree $k = \frac{d}{N-1}$ and its complement $k^{c} = 1 - k$. Expressing the number of links L in terms of the link density $p = \frac{L}{L_{\max}}$ and its complement $p^{c} = 1 - p$,



d_i \ d_j	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	
1	0	5.3	3.4	2.1	1.2	0.55	0.05	-0.48	-1.5	-4.4
2	0	3.4	2.2	1.4	0.75	0.29	-0.06	-0.45	-0.99	-1.7
3	0	2.1	1.4	0.82	0.41	0.11	-0.13	-0.37	-0.61	-0.81
4	0	1.2	0.75	0.41	0.16	-0.01	-0.15	-0.27	-0.34	-0.36
5	0	0.55	0.29	0.11	-0.01	-0.1	-0.14	-0.15	-0.13	-0.09
6	0	0.05	-0.06	-0.13	-0.15	-0.14	-0.1	-0.05	0.02	0
7	0	-0.48	-0.45	-0.37	-0.27	-0.15	-0.05	0.06	0.02	0
8	0	-1.5	-0.99	-0.61	-0.34	-0.13	0.02	0.02	0.01	0
9		-4.4	-1.7	-0.81	-0.36	-0.09	0	0	0	0

FIG. 3. Heatmap of the error factor ϵ for the class of graphs with $N = 10$ nodes and $L = 25$ links.

we rewrite (9) as

$$\begin{aligned} \frac{1}{\Pr[i \sim j]} &= 1 + \frac{k_i^{c} k_j^{c}\left[p L_{\max} - k_i(N-1) - k_j(N-1) + 1\right]}{k_i k_j\left[p^{c} L_{\max} - k_i^{c}(N-1) - k_j^{c}(N-1) + 1\right]} \\ &= 1 + \frac{k_i^{c} k_j^{c}\left(p - \frac{2(k_i + k_j)}{N} + \frac{2}{N(N-1)}\right)}{k_i k_j\left(p^{c} - \frac{2(k_i^{c} + k_j^{c})}{N} + \frac{2}{N(N-1)}\right)}. \end{aligned} \tag{11}$$

Since k and k^{c} are upper bounded by 1, it follows that

$$\begin{aligned} \lim_{N \to \infty} \frac{2(k_i + k_j)}{N} &= 0, \\ \lim_{N \to \infty} \frac{2\left(k_i^{c} + k_j^{c}\right)}{N} &= 0. \end{aligned} \tag{12}$$

Hence, in large networks, the probability that node i and node j are connected tends to

$$\lim_{N \to \infty} \Pr[i \sim j] = \frac{k_i k_j (1 - p)}{k_i k_j (1 - p) + (1 - k_i)(1 - k_j) p}, \tag{13}$$

which is dependent on the link density p, but not the network size N. Similarly, the expected number of links in the configuration model $\mathbb{E}[a_{i,j}]_{\text{CM}}$ in (2) can be rewritten as

$$\mathbb{E}[a_{i,j}]_{\text{CM}} = \frac{d_i d_j}{2L - 1} = \frac{k_i k_j (N-1)^2}{pN(N-1) - 1} = \frac{k_i k_j}{p\frac{N}{N-1} - \frac{1}{(N-1)^2}}. \tag{14}$$

Hence, in large networks, $\mathbb{E}[a_{i,j}]_{\text{CM}}$ tends to

$$\lim_{N \to \infty} \mathbb{E}[a_{i,j}]_{\text{CM}} = \frac{k_i k_j}{p}, \tag{15}$$

which is also only dependent on the link density p. This suggests that when using $\mathbb{E}[a_{i,j}]_{\text{CM}}$ (the expected number of links in the configuration model) as an estimate for $\Pr[i \sim j]$ (the probability of a link in a simple ER graph), given a pair of nodes (i, j) with normalized degrees (k_i, k_j), the approximation error is constant and scale invariant with respect to

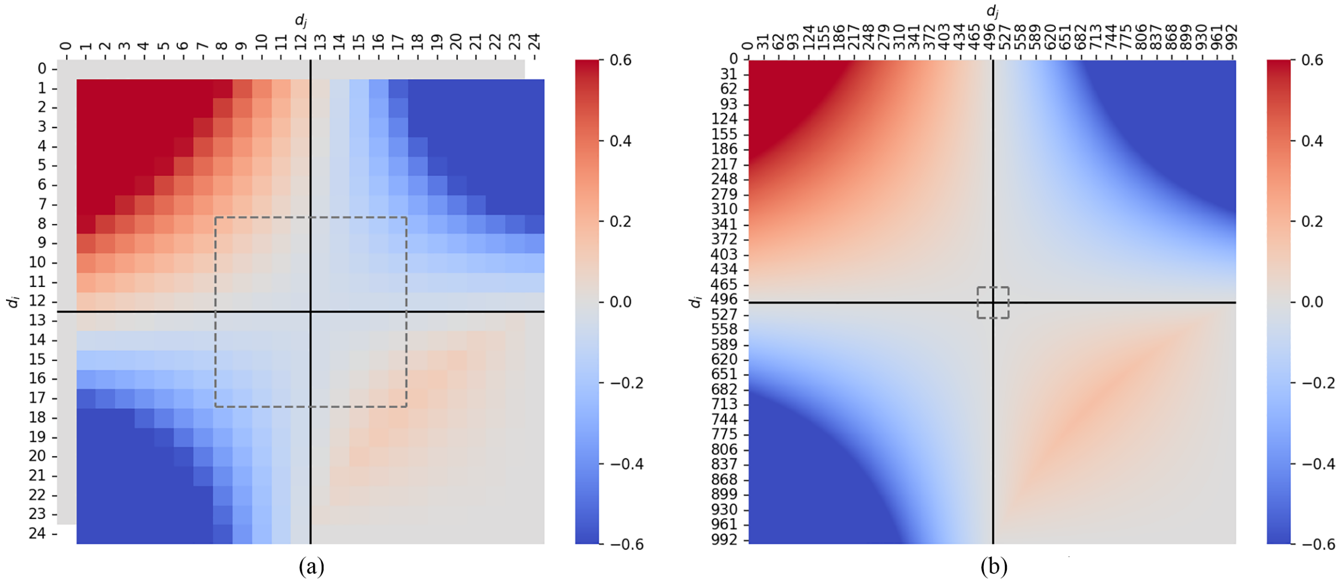


FIG. 4. Comparison of the error factor ϵ of the configuration model approximation $\mathbb{E}[a_{i,j}]_{\text{CM}}$ for $N = 25$ and $N = 1000$ nodes with $p = 0.5$. The solid lines show the average degree, and the dotted lines indicate the area where both d_i and d_j are within two standard deviations of the average degree. (a) $N = 25$ nodes and (b) $N = 1000$ nodes.

the network size N. Figure 4 shows a heatmap of the error factor ϵ for the class of graphs with $N = 25$ nodes and link density $p = 0.5$ ($L = 150$) and for the class of graphs with $N = 1000$ nodes and the same link density $p = 0.5$ ($L = 249\,750$). The exact same pattern in the heatmap is observed in both Figs. 4(a) and 4(b), indicating that the error factor ϵ is constant for the same relative degree k.

To understand the pattern in the heatmap in Fig. 4, we rewrite (13) as

$$\lim_{N\to\infty} \Pr[i \sim j] = \frac{k_i k_j}{p + \frac{1}{1-p}(p-k_i)(p-k_j)}. \tag{16}$$

Hence, in the limit $N \to \infty$,

$$\mathbb{E}[a_{i,j}]_{\text{CM}} \begin{cases} > \Pr[i \sim j] & \text{if } (p-k_i)(p-k_j) > 0 \\ < \Pr[i \sim j] & \text{if } (p-k_i)(p-k_j) < 0 \\ = \Pr[i \sim j] & \text{if } (p-k_i)(p-k_j) = 0, \end{cases} \tag{17}$$

which explains Fig. 4. The average degree is $d_{\text{av}} = p(N-1)$ and the normalized average degree $k_{\text{av}} = p$. When either node i or node j has the average degree, then $(p-k_i)(p-k_j) = 0$. Therefore, the error factor is (almost) zero along the solid lines in Fig. 4. If both $k_i, k_j > p$, or both $k_i, k_j < p$, then $(p-k_i)(p-k_j) > 0$ and $\mathbb{E}[a_{i,j}]_{\text{CM}}$ overestimates the connection probability $\Pr[i \sim j]$. If $k_i > p$ but $k_j < p$, or $k_i < p$ but $k_j > p$, then $(p-k_i)(p-k_j) < 0$ and $\mathbb{E}[a_{i,j}]_{\text{CM}}$ underestimates the connection probability $\Pr[i \sim j]$.

In summary, if the configuration model expectation $\mathbb{E}[a_{i,j}]_{\text{CM}}$ is used as an estimate for the true connection probability $\Pr[i \sim j]$, the error is constant with respect to the link density p and relative degree k. The error is worse in dense networks because if the link density p is close to 1, then the term $\frac{1}{1-p}$ in the denominator of (16) becomes very large. However, in Erdős-Rényi random graphs, the degree d will be binomially distributed ([4], Sec. 15.7.1) with mean $p(N-1)$ and variance $(N-1)p(1-p)$. Therefore, the relative degree k has mean p and variance $\frac{p(1-p)}{N-1} \to 0$ as $N \to \infty$. The gray dotted lines in Fig. 4 indicate the area where both d_i and d_j are within two standard deviations of the average degree d_{av}. Therefore, the configuration model expectation $\mathbb{E}[a_{i,j}]_{\text{CM}}$ is a good estimate for the connection probability $\Pr[i \sim j]$ if the network belongs to the class of Erdős-Rényi random graphs and N is large, because the probability of observing degrees d that are far away from the average degree d_{av} decreases as N increases.

V. MODULARITY

The modularity m as defined by Newman [5,6] plays a critical role in detecting community structure in networks. The modularity m is given by

$$m = \frac{1}{2L} \sum_{i=1}^{N} \sum_{j=1}^{N} (a_{i,j} - p_{i,j}) \sum_{k=1}^{C} \mathbf{1}_{\{i,j \in C_k\}} \tag{18}$$

and

$$p_{i,j} = \frac{d_i d_j}{2L}, \tag{19}$$

where C is the number of clusters (communities) and C_k denotes cluster k. The indicator function $\mathbf{1}_{\{i,j \in C_k\}}$ means that only nodes belonging to the same cluster C_k contribute to the modularity. The entry $a_{i,j}$ of the adjacency matrix A indicates whether a link exists between node i and node j. The term $p_{i,j}$ represents the probability that a link would exist between node i and node j if "connections are made at random but respecting [node] degrees" [7] and is the baseline or null model with which the existence of a link is compared. Observe that (19) is actually the expected number of links between node i and node j in a large configuration model network (4) and is dependent only on the degrees d_i and d_j; the degrees of the rest of the nodes in the network are not taken into account.

TABLE I. Summary of the modularity values of the clusters found when using different algorithms and objective functions.

		Modularity			
Algorithm	Objective	m	$\widehat{m}$	m_{exact}	Figure
ILP (optimal)	$m, \widehat{m}, m_{\text{exact}}$	0.4198	0.4524	0.4513	Fig. 5
Spectral	$\widehat{m}$	0.4118	0.4455	0.4438	Fig. 6(b)
Spectral	m, m_{exact}	0.3934	0.4216	0.4223	Fig. 6(a)
Greedy	$\widehat{m}, m_{\text{exact}}$	0.3942	0.4206	0.4205	Fig. 7(b)
Greedy	m	0.3807	0.4009	0.4030	Fig. 7(a)

The modularity m provides a measure for evaluating the quality of a given division of a network into communities and is the most commonly used quality function in community detection methods based on optimization [8]. As summarized in a recent review [9], various modifications to the modularity formula (18) have been proposed to address some of its limitations. For example, [10,11] modify the modularity formula to not only consider links present within a community, but also the links that are missing within a community. In [12], the modularity is modified to also take links between communities into account. Here, we consider a simple change where we redefine the probability term $\widehat{p}_{i,j}$ using our exact probability (9) of a link in a simple graph on N nodes and L links,

$$\widehat{p}_{i,j} = \begin{cases} \frac{d_i d_j (L^{\mathrm{c}} - d_i^{\mathrm{c}} - d_j^{\mathrm{c}} + 1)}{d_i d_j (L^{\mathrm{c}} - d_i^{\mathrm{c}} - d_j^{\mathrm{c}} + 1) + d_i^{\mathrm{c}} d_j^{\mathrm{c}} (L - d_i - d_j + 1)}, & i \neq j \\ 0, & i = j. \end{cases} \tag{20}$$

With this change, we still only respect the degrees d_i and d_j, but we additionally account for the fact that the graph must be simple and contain exactly N nodes and L links. We define the adjusted modularity $\widehat{m}$ as

$$\widehat{m} = \frac{1}{2L} \sum_{i=1}^{N} \sum_{j=1}^{N} (a_{i,j} - \widehat{p}_{i,j}) \sum_{k=1}^{C} \mathbf{1}_{\{i,j \in C_k\}}, \tag{21}$$

which is the same as the original modularity formula (18) except $p_{i,j}$ is replaced by $\widehat{p}_{i,j}$.

As an example, we consider partitioning Zachary's karate club network [13]. In Appendix B, we explicitly calculate the probability $\Pr[i \sim j]_{(d_1,\dots,d_N)}$ of a link conditioned on the entire degree sequence of the karate club network, and we verify that our probability term $\widehat{p}_{i,j}$ is more accurate than $p_{i,j}$. We define the modularity calculated using $\Pr[i \sim j]_{(d_1,\dots,d_N)}$ to be the true modularity

$$m_{\text{exact}} = \frac{1}{2L} \sum_{i=1}^{N} \sum_{j=1}^{N} (a_{i,j} - \Pr[i \sim j]_{(d_1,\dots,d_N)}) \sum_{k=1}^{C} \mathbf{1}_{\{i,j \in C_k\}}. \tag{22}$$

We consider two heuristic algorithms (Newman's spectral algorithm [14] and the Clauset-Newman-Moore greedy algorithm [7]) and compare the differences when using Newman's modularity m, our adjusted modularity $\widehat{m}$, and the true modularity m_{exact} as the objective function. We also compare the results with the optimal partitioning obtained through integer linear programming (ILP) [15,16].

A summary of the modularity values of the clusters for the different algorithms and objective functions is presented in Table I. The table is sorted on the true modularity m_{exact} from highest to lowest and our adjusted modularity $\widehat{m}$ agrees with the ordering. However, Newman's modularity m considers the clusters of Fig. 7(b) to have higher modularity than Fig. 6(a). Our adjusted modularity $\widehat{m}$ values are close to the true modularity m_{exact}, but there is a small error, because our probability term $\widehat{p}_{i,j}$ takes only the degrees of node i and j into account. When using integer linear programming to find the optimal partitioning, the same clusters are found for all three objective functions. The clusters are illustrated in Fig. 5 and have been verified against other publications [17,18].

Figure 6 shows the partitioning of the karate club network using Newman's spectral algorithm [14]. As shown in Fig. 6(a), using Newman's modularity m yields the same clusters as the true modularity m_{exact}. Compared to the optimal partitioning (Fig. 5), node 1 and node 12 have been moved to the red cluster. When using the adjusted modularity [Fig. 6(b)], there is only one difference with the optimal partitioning (Fig. 5): node 12 is placed in an isolated blue cluster. As shown in Table I, all three modularity measures indicate that the partitioning in Fig. 6(b) has higher modularity than the partitioning in Fig. 6(a).

Figure 7 shows the partitioning of the karate club network using the Clauset-Newman-Moore greedy modularity maximization algorithm [7] (as implemented in NetworkX [19]). Figure 7(a) shows the clusters found when using Newman's

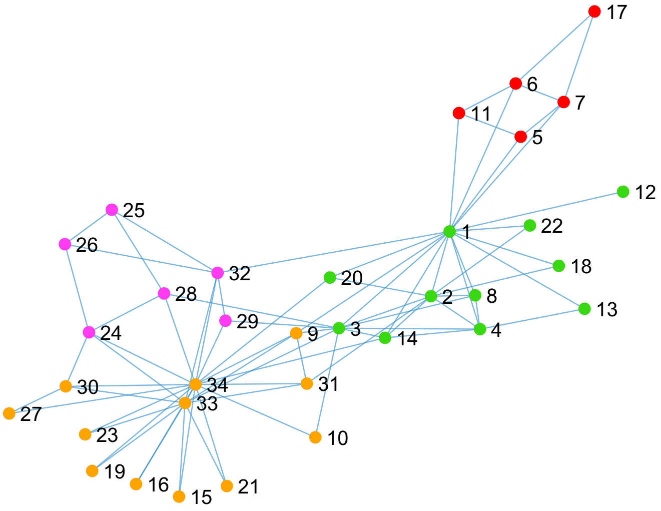


FIG. 5. Optimal partitioning of the karate club network found using integer linear programming. The same partitions are found when using Newman's modularity m, our adjusted modularity $\widehat{m}$, as well as the true modularity m_{exact}.

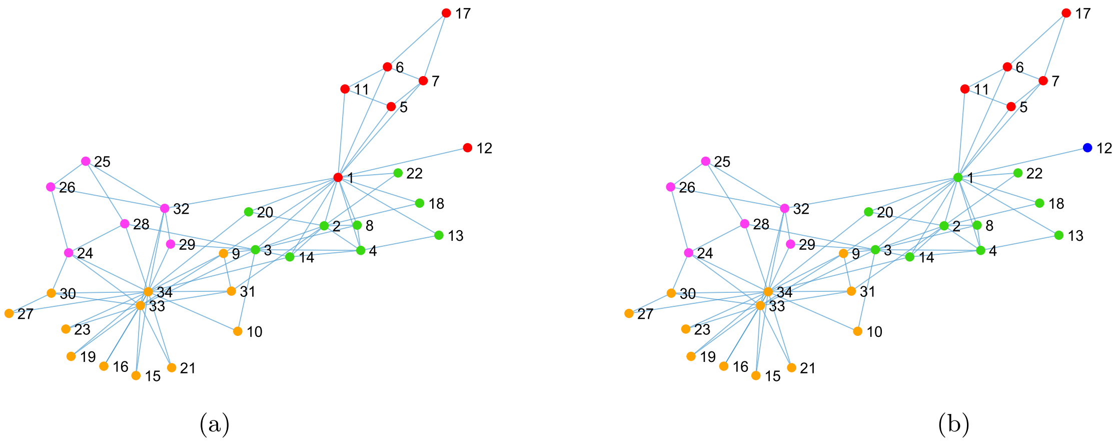


FIG. 6. Partitioning of the karate club network using Newman's spectral algorithm [14] with different objective functions. (a) Newman's modularity m; same clusters as the true modularity m_{exact} and (b) adjusted modularity $\widehat{m}$.

modularity m. There are many differences compared to the optimal partitioning (Fig. 5), most notably the absence of the pink cluster. When using the adjusted modularity $\widehat{m}$, a pink cluster is still detected as shown in Fig. 7(b). Using the true modularity m_{exact} yields the same clusters as the adjusted modularity $\widehat{m}$. As shown in Table I, all three modularity measures indicate that the partitioning in Fig. 7(b) has higher modularity than the partitioning in Fig. 7(a).

VI. CONCLUSION

We have derived an exact formula (9) for the probability $\Pr[i \sim j]$ that two nodes i and j are connected in a simple random graph belonging to the class of Erdős-Rényi $G(N, L)$ random graphs. The expected number of links in the configuration model $\mathbb{E}[a_{i,j}]_{\text{CM}}$ is commonly used as an approximation for the connection probability $\Pr[i \sim j]$. We defined an error factor ϵ to quantify the difference between $\mathbb{E}[a_{i,j}]_{\text{CM}}$ and $\Pr[i \sim j]$, showing that $\mathbb{E}[a_{i,j}]_{\text{CM}}$ severely overestimates the connection probability between two low degree nodes i and j, while severely underestimating the connection probability between a low degree node i and a high degree node j. We show that for constant link density p, the error factor ϵ is scale invariant with respect to the relative degree k. In large Erdős-Rényi graphs, $\mathbb{E}[a_{i,j}]_{\text{CM}}$ becomes a good estimate for $\Pr[i \sim j]$ because the variance of the relative degree k decreases as $O(\frac{1}{N})$.

Many real networks, however, do not belong to the class of Erdős-Rényi random graphs. We consider the application of network partitioning using Newman's modularity m, compared with the adjusted modularity $\widehat{m}$ in which the probability of two nodes being connected is replaced by our formula. Using the karate club network as an example, we showed that our probability term $\widehat{p}_{i,j}$ (20) is a more accurate baseline probability than the original probability term $p_{i,j}$ (19) in the modularity formula.

We tested two heuristic algorithms for modularity maximization and compared the clusters found when using Newman's modularity m with the clusters found when using our adjusted modularity $\widehat{m}$. For both algorithms, we found clusters with higher modularity when using our adjusted modularity $\widehat{m}$ as the objective function. Although our probability term $\widehat{p}_{i,j}$ (20) is a little more complicated than the original probability term $p_{i,j}$ (19), the computational complexity hardly changes. Hence, we believe that it is worth replacing (19) by (20) in the objective function for clustering.

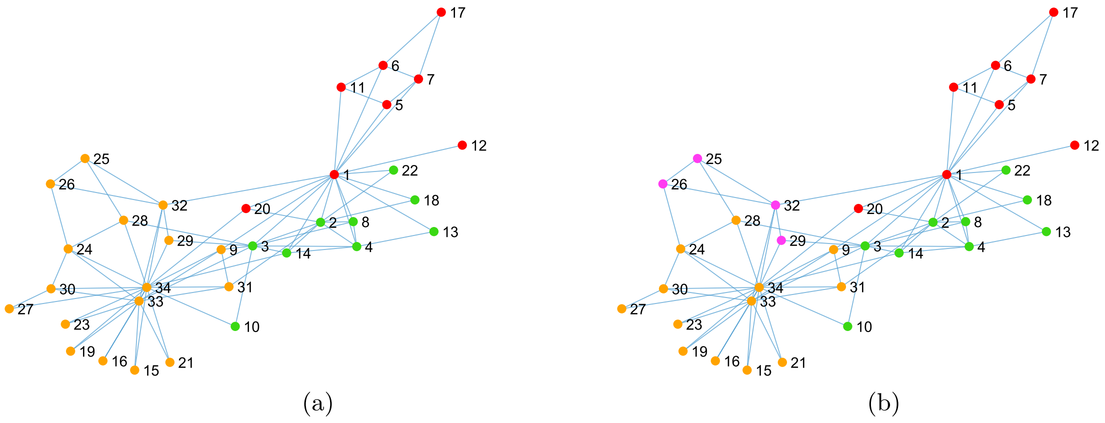


FIG. 7. Partitioning of the karate club network using the Clauset-Newman-Moore greedy modularity maximization algorithm [7] with different objective functions. (a) Newman's modularity m and (b) adjusted modularity $\widehat{m}$; same clusters as the true modularity m_{exact}.

ACKNOWLEDGMENTS

The authors are grateful to Ivan Jokić for his assistance in implementing the clustering algorithms. This research has been funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement No. 101019718).

APPENDIX A: DERIVATIONS

1. Checking whether the parameters are graphical

We derive the conditions under which a degree pair (d_i, d_j) is graphical for a graph of N nodes and L links, meaning there exists at least one simple graph G with the parameters $\{N, L, (d_i, d_j)\}$, which implies

$$\left|\mathcal{G}_{N,L,(d_i,d_j)}\right| = \left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right| + \left|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}\right| > 0. \quad \text{(A1)}$$

The degree d of any node in a simple graph G is bounded by

$$0 \leqslant d \leqslant N - 1. \quad \text{(A2)}$$

Since we are considering a degree pair (d_i, d_j), we must have at least $N \geqslant 2$ nodes. We should also exclude the degree pairs $(0, N-1)$ and $(N-1, 0)$ because degree $d = 0$ means the graph is disconnected while degree $d = N - 1$ means the graph is connected, which cannot occur at the same time.

Given a degree pair (d_i, d_j), we derive the minimum L_- and maximum L_+ number of links L,

$$L_- \leqslant L \leqslant L_+. \quad \text{(A3)}$$

There are d_i links connected to node i and d_j links connected to node j. If $\min(d_j, d_j) > 0$, the minimum number of links $L_- = d_i + d_j - 1$ because we can place a link between node i and node j. If $\min(d_j, d_j) = 0$, then it is not possible to place a link between node i and node j and the minimum number of links is $L_- = d_i + d_j$. Hence, the minimum number of links L_- is

$$L_- = \begin{cases} d_i + d_j - 1 & \text{if } \min(d_i, d_j) > 0 \\ d_i + d_j & \text{if } \min(d_i, d_j) = 0. \end{cases} \quad \text{(A4)}$$

We derive the maximum number of links L_+ in the same way by considering the complement graph G^{c}. In the complement graph G^{c}, there are $L^{\text{c}} = L_{\max} - L$ links, node i has degree $d_i^{\text{c}} = (N-1) - d_i$, and node j has degree $d_j^{\text{c}} = (N-1) - d_j$. The minimum number of links L_-^{c} in the complement graph G^{c} is

$$L_-^{\text{c}} = \begin{cases} d_i^{\text{c}} + d_j^{\text{c}} - 1 & \text{if } \min\left(d_i^{\text{c}}, d_j^{\text{c}}\right) > 0 \\ d_i^{\text{c}} + d_j^{\text{c}} & \text{if } \min\left(d_i^{\text{c}}, d_j^{\text{c}}\right) = 0. \end{cases} \quad \text{(A5)}$$

When the number of links in the graph G is maximal, $L = L_+$, the number of links in the complement graph G^{c} is minimal, $L^{\text{c}} = L_-^{\text{c}}$. Hence, $L_+ = L_{\max} - L_-^{\text{c}}$.

2. Constraints on parameters such that $0 < \Pr[i \sim j] < 1$

During the derivation of (9), we assumed

$$\begin{aligned} \left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right| &> 0, \\ \left|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}\right| &> 0, \end{aligned} \quad \text{(A6)}$$

which is a stricter condition than graphicality (A1) and excludes parameters $\{N, L, (d_i, d_j)\}$, which yield $\Pr[i \sim j] = 0$ or $\Pr[i \sim j] = 1$. We derive the constraints on the parameters $\{N, L, (d_i, d_j)\}$ in order to satisfy (A6).

TABLE II. All possible values of L and (d_i, d_j) for $N = 2$ and $N = 3$ nodes.

N	L	(d_i, d_j)	L^{c}	$(d_i^{\text{c}}, d_j^{\text{c}})$	$\Pr[i \sim j]$
2	0	(0,0)	1	(1,1)	0
2	1	(1,1)	0	(0,0)	1
3	0	(0,0)	3	(2,2)	0
3	1	(0,1)	2	(2,1)	0
3	1	(1,1)	2	(1,1)	1
3	2	(1,1)	1	(1,1)	0
3	2	(1,2)	1	(1,0)	1
3	3	(2,2)	0	(0,0)	1

A node with degree $d = 0$ is not connected to any other node, which implies $\Pr[i \sim j] = 0$. A node with degree $d = N - 1$ is connected to every other node, which implies $\Pr[i \sim j] = 1$. To satisfy (A6), the degree d must be strictly bounded by

$$0 < d < N - 1. \quad \text{(A7)}$$

The bound (A7) implies that $\min(d_j, d_j) > 0$ and $\min(d_j^{\text{c}}, d_j^{\text{c}}) > 0$. From (A4) and (A5), it follows that the minimum L_- and maximum L_+ number of links L is

$$\begin{aligned} L_- &= d_i + d_j - 1, \\ L_+ &= L_{\max} - \left(d_i^{\text{c}} + d_j^{\text{c}} - 1\right). \end{aligned} \quad \text{(A8)}$$

If the number of links is minimal, $L = L_-$, then $\Pr[i \sim j] = 1$. If the number of links is maximal, $L = L_+$, then $\Pr[i \sim j] = 0$. To satisfy (A6), the number of links must be strictly bounded by

$$L_- < L < L_+. \quad \text{(A9)}$$

The inequality (A9) cannot be satisfied for $N = 2$ and $N = 3$ nodes. Hence, the number of nodes N is at least

$$N \geqslant 4. \quad \text{(A10)}$$

Indeed, the constraints (A7), (A9), and (A10) ensure the binomial coefficients in (7) and (8) are always valid.

We verify that our expression (9) holds for *all* graphical parameters $\{N, L, (d_i, d_j)\}$. In the main text, we have already shown that our expression correctly yields $\Pr[i \sim j] = 0$ when $d = 0$ or $L = L_+$; we have also shown that our expression correctly yields $\Pr[i \sim j] = 1$ when $d = N - 1$ or $L = L_-$. In Table II, we summarize all graphical values of L and (d_i, d_j) for $N = 2$ and $N = 3$ nodes; substituting these values into (9) yields the correct probability. Hence, our expression (9) for the probability $\Pr[i \sim j]$ holds for all Erdős-Rényi $G(N, L)$ graphs on $N \geqslant 2$ nodes.

3. Simplifying binomial coefficients

The binomial coefficient $\binom{n}{r}$ is given by

$$\binom{n}{r} = \frac{n!}{r!\,(n-r)!}. \quad \text{(A11)}$$

For $r > 0$, it follows that

$$\frac{\binom{n}{r}}{\binom{n}{r-1}} = \frac{(r-1)!\,(n-r+1)!}{r!\,(n-r)!} = \frac{n-r+1}{r}. \tag{A12}$$

For $r < n$, it follows that

$$\frac{\binom{n}{r}}{\binom{n}{r+1}} = \frac{(r+1)!\,(n-r-1)!}{r!\,(n-r)!} = \frac{r+1}{n-r}. \tag{A13}$$

The probability $\Pr[i \sim j]$ is given by

$$\begin{aligned}\Pr[i \sim j] &= \frac{\left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right|}{\left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right| + \left|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}\right|}\\ &= \frac{1}{1 + \frac{\left|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}\right|}{\left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right|}}.\end{aligned} \tag{A14}$$

Using the identities (A12) and (A13), we simplify the second denominator term

$$\begin{aligned}\frac{\left|\mathcal{G}_{N,L,(d_i,d_j),i\nsim j}\right|}{\left|\mathcal{G}_{N,L,(d_i,d_j),i\sim j}\right|} &= \frac{\binom{N-2}{d_i}\binom{N-2}{d_j}\binom{L_{\max}-2(N-2)-1}{L-d_i-d_j}}{\binom{N-2}{d_i-1}\binom{N-2}{d_j-1}\binom{L_{\max}-2(N-2)-1}{L-d_i-d_j+1}}\\ &= \frac{(N-1-d_i)}{d_i}\frac{(N-1-d_j)}{d_j}\\ &\quad\times \frac{(L-d_i-d_j+1)}{(L_{\max}-L-2(N-2)+d_i+d_j-1)}\\ &= \frac{d_i^{\mathrm{c}} d_j^{\mathrm{c}}(L-d_i-d_j+1)}{d_i d_j\left(L^{\mathrm{c}}-d_i^{\mathrm{c}}-d_j^{\mathrm{c}}+1\right)},\end{aligned} \tag{A15}$$

where $d^{\mathrm{c}} = (N-1) - d$ and $L^{\mathrm{c}} = L_{\max} - L$.

APPENDIX B: PROBABILITY OF A LINK CONDITIONED ON THE DEGREE SEQUENCE

Consider a simple graph G with degree sequence $(d_1, \ldots, d_N)$ on N nodes with L links. The probability that a pair of nodes i and j are connected if connections are made at random while respecting *all* node degrees is given by

$$\Pr[i \sim j]_{(d_1,\ldots,d_N)} = \frac{\left|\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}\right|}{\left|\mathcal{G}_{(d_1,\ldots,d_N)}\right|}, \tag{B1}$$

where $\mathcal{G}_{(d_1,\ldots,d_N)}$ is the set of all labeled graphs with degree sequence $(d_1, \ldots, d_N)$ and $\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}$ is the subset of those graphs in which there is a link between node i and node j. Unfortunately, there is no closed-form solution for (B1); an exact numerical calculation is possible by counting the number of graphs with a given degree sequence but this quickly becomes intractable as the network size increases.

ALGORITHM 1. NUMGRAPHSWITHDEGSEQUENCE. Count the number of labeled graphs with the degree sequence $(d_1, \ldots, d_N)$. We remove the node with the smallest degree d from the degree sequence; we iterate over all possible ways that d links can be connected to the remaining nodes $N-1$; for each of the possible ways we compute the number of graphs of the corresponding degree sequence recursively and sum them.

```
Inputs:
$D_N = (d_1, \ldots, d_N)$: degree sequence of $N$ nodes
Outputs:
$c = |\mathcal{G}_{(d_1,\ldots,d_N)}|$: number of labeled graphs with degree sequence $D_N$
NUMGRAPHSWITHDEGSEQUENCE($D_N$):
if $N = 2$ then
    if $D_N = (0, 0)$ or $D_N = (1, 1)$ then
        return 1
    else
        return 0
    end if
end if
$d \leftarrow$ smallest degree in $D_N$
$D_{N-1} \leftarrow$ COPY($D_N$) and delete $d$ from $D_{N-1}$
$c \leftarrow 0$
for each way to choose $d$ indexes from $N-1$ total indexes do
    $D'_{N-1} \leftarrow$ COPY($D_{N-1}$)
    subtract 1 from the elements at the corresponding indexes
      in $D'_{N-1}$
    $c \leftarrow c +$ NUMGRAPHSWITHDEGSEQUENCE($D'_{N-1}$)
end for
return $c$
```

The karate club network contains $N = 34$ nodes and $L = 78$ links. The degree sequence of the karate club network (sorted from smallest to largest) is

$$(1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 5, 5, 5, 6, 6, 9, 10, 12, 16, 17).$$

The total number of graphs with the same degree sequence as the karate club network is

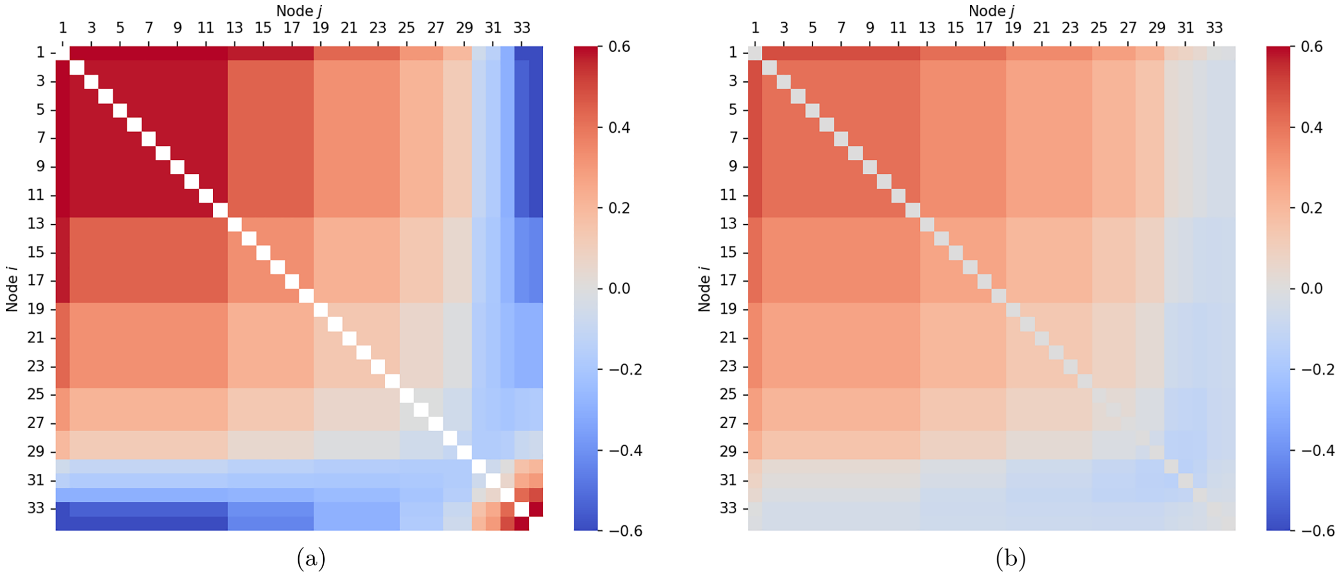
$$\left|\mathcal{G}_{(d_1,\ldots,d_N)}\right| = 27\,425\,053\,479\,717\,264\,361\,406\,133\,594\,918\,792\,062\,198\,598\,516\,534\,680,$$

which is approximately 2.74×10^{52}. We compute the number of graphs with a given degree sequence using a recursive algorithm, which is described in [20]; the pseudocode is given in Algorithm 1. NUMGRAPHSWITHDEGSEQUENCE. As in [20], we make use of dynamic programming in our solution, which we implemented in Python; the computation took 1.7 h using an Intel i7-1265U CPU at 1.80 GHz and 16 GB of RAM on a machine running Windows 10.

In Table III, we computed the number of graphs $|\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}|$ in which a node pair (i, j) with degrees (d_i, d_j)

ALGORITHM 2. numGraphsWhereConnected. Count the number of labeled graphs with the degree sequence $(d_1, \ldots, d_N)$ in which a pair of nodes i and j is connected. We remove node i and node j from the degree sequence; we iterate over all possible ways that node i and node j can be connected to the remaining $N-2$ nodes (while also being connected to each other); for each of the possible ways we compute the number of graphs of the corresponding degree sequence (using Algorithm 1. numGraphsWithDegSequence) and sum them.

```
Inputs:
$D_N = (d_1, \ldots, d_N)$: degree sequence of $N$ nodes
$i$: index of node
$j$: index of node
Outputs:
$c = |\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}|$: number of labeled graphs with degree sequence $D_N$ in which nodes $i$ and $j$ are connected
numGraphsWhereConnected($D_N$, $i$, $j$):
$d_i \leftarrow$ $i$th element of $D_N$
$d_j \leftarrow$ $j$th element of $D_N$
$D_{N-2} \leftarrow$ copy($D_N$) and delete $i$th and $j$th elements
$c \leftarrow 0$
for each way to choose $d_i - 1$ indexes from $N-2$ total indexes do
    $D'_{N-2} \leftarrow$ copy($D_{N-2}$)
    subtract 1 from the elements at the corresponding indexes in $D'_{N-2}$
    for each way to choose $d_j - 1$ indexes from $N-2$ total indexes do
        $D''_{N-2} \leftarrow$ copy($D'_{N-2}$)
        subtract 1 from the elements at the corresponding indexes in $D''_{N-2}$
        $c \leftarrow c +$ numGraphsWithDegSequence($D''_{N-2}$)
    end for
end for
return $c$
```



(a)

(b)

FIG. 8. Heatmaps of the error factor ϵ comparing the modularity probability term with the probability of a link $\Pr[i \sim j]_{(d_1,\ldots,d_N)}$ conditioned on the degree sequence $(d_1, \ldots, d_N)$ of the karate club network. (a) Original probability term $p_{i,j}$ and (b) our probability term $\widehat{p}_{i,j}$.

TABLE III. Number of graphs $|\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}|$ in which a node pair (i, j) with degrees (d_i, d_j) is connected for degree sequence $(d_1, \ldots, d_N)$ of the karate club network.

d_i	d_j	$\lvert\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}\rvert$	Exact $\lvert\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}\rvert$
1	2	2.03×10^{50}	203465676733748493823662233969243882674397849632503
1	3	3.35×10^{50}	335138986101163740931904899352277411443028413324199
1	4	4.90×10^{50}	490353935594975119965653979099935224307525406605949
1	5	6.72×10^{50}	671798464251910119246043264689493290899435959443732
1	6	8.82×10^{50}	881775728102236344291700215299113498989196381473411
1	9	1.69×10^{51}	1685767821671073255424245879820309811859665373689287
1	10	2.01×10^{51}	2009360839076345283569699839619042763673439227858611
1	12	2.75×10^{51}	2745894612135793202070993415145338189132399644229636
1	16	4.69×10^{51}	4693834043189311700657709644029053657146370670506418
1	17	5.32×10^{51}	5320169340436471275916316747263382245788323693434289
2	2	4.45×10^{50}	445095241323276849325505021982568939186793845378089
2	3	7.30×10^{50}	729993800191351101921545006288111082132453924834841
2	4	1.06×10^{51}	1062625740607953717497882955963886053628936388499767
2	5	1.45×10^{51}	1447029791830398928611895717488337261040603223612944
2	6	1.89×10^{51}	1885812392252473208556921463056088161291368266224879
2	9	3.50×10^{51}	3503822708470880559350374909649443735331229771660844
2	10	4.13×10^{51}	4126848764640108113550634133633223638559702008285383
2	12	5.48×10^{51}	5484141483525387404861315735936476347199877170906072
2	16	8.66×10^{51}	8660536347009646891373628362136647616756600322414654
2	17	9.55×10^{51}	9551908161030016647131503742597688591734563373092776
3	3	1.19×10^{51}	1191100267530147667163656341126617260023961562192674
3	4	1.72×10^{51}	1723260233252800426308130989713762996113199740244059
3	5	2.33×10^{51}	2329755965823245953660594852275146243362364299765816
3	6	3.01×10^{51}	3010643248436718662572873216626434985406691781212357
3	9	5.41×10^{51}	5412184657175496734417360391506052538311807149751084
3	10	6.29×10^{51}	6290269332389720899473100026192217555390251696731113
3	12	8.11×10^{51}	8113252617344239607547241262399602288977734808710341
3	16	1.19×10^{52}	11913464836517078749571110295320286766033092695990615
3	17	1.29×10^{52}	12875301076008515151346090206822744745283404897763551
4	4	2.48×10^{51}	2475151903476364527391202752444586762144693552727671
4	5	3.32×10^{51}	3317891880575692998052925905399916001422120364243917
4	6	4.25×10^{51}	4245442323368255490031478310240936599183578949432001
4	9	7.35×10^{51}	7354112767566938328508422490660603837938375254076011
4	10	8.43×10^{51}	8426853024659940838269078073900290341576626398513687
4	12	1.05×10^{52}	10547349953441384694048223289366792689090031840690176
4	16	1.45×10^{52}	14548379947369957807973371573545753151931270128921983
4	17	1.55×10^{52}	15484399938186154595356538420311913423996079567135015
5	5	4.40×10^{51}	4403752283515778516415915671623152655548456766855770
5	6	5.57×10^{51}	5571547350930337620783508797670037616624679085614632
5	9	9.27×10^{51}	9270725926130121978978194136851504370216110261045575
5	10	1.05×10^{52}	10471520360091141637141071203509697349573503047214017
5	12	1.27×10^{52}	12738015252748238951135745052151472933904128356968663
5	16	1.67×10^{52}	16675995748522801050951400619535802050536743186378709
5	17	1.75×10^{52}	17543397589421853870167387320847526431363256622881118
6	6	6.96×10^{51}	6960677624784803516885235745505914514304196857099457
6	9	1.11×10^{52}	11106196308116579707096929773034718827131614300369818
6	10	1.24×10^{52}	12372093838941429204325096324574595143460358474865705
6	12	1.47×10^{52}	14665312099435453619864891131232377231066457887456716
6	16	1.84×10^{52}	18392663450306698558578728765841803301483760933182692
6	17	1.92×10^{52}	19176510030218322145291447966192917725135293695576568
9	10	1.69×10^{52}	16926346855540913249381569674429189748377803200410354
9	12	1.89×10^{52}	18926054909459425851679100828077069467450466204573390
9	16	2.18×10^{52}	21785310643451594830019076439869278886772046610315603
9	17	2.23×10^{52}	22337596351074550184613946276305510534271223964715271
10	12	1.99×10^{52}	19928901936845786888052921954340319425021826688267038
10	16	2.25×10^{52}	22518715584627096475819129914161964042411938444298200
10	17	2.30×10^{52}	23010390269587058721654564236438614900901760850531123
12	16	2.36×10^{52}	23636658413130309966263997511496487984095611768926346

TABLE III. (*Continued.*)

d_i	d_j	$\lvert\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}\rvert$	Exact $\lvert\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}\rvert$
12	17	2.40×10^{52}	24029290184427225072622311605147506729232328270230435
16	17	2.53×10^{52}	25316054324467781389349761422153512378065076848970020

is connected; the pseudocode is given in Algorithm 2: NUMGRAPHSWHERECONNECTED. The computation took 11.3 h (using the same machine). The values in Table III are exact, which can be easily verified by checking the sum

$$\sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} \left|\mathcal{G}_{(d_1,\ldots,d_N),i\sim j}\right| = 2L|\mathcal{G}_{(d_1,\ldots,d_N)}|. \tag{B2}$$

We compare the probability term $p_{i,j}$ of the modularity formula, given in (19), with the connection probability $\Pr{[i \sim j]}_{(d_1,\ldots,d_N)}$ conditioned on the degrees of all nodes. Similar to Sec. III, we define an error factor ϵ to quantify the difference between $p_{i,j}$ and $\Pr{[i \sim j]}_{(d_1,\ldots,d_N)}$,

$$\epsilon = \begin{cases} \frac{p_{i,j}}{\Pr{[i\sim j]}_{(d_1,\ldots,d_N)}} - 1 & \text{if } p_{i,j} > \Pr{[i \sim j]}_{(d_1,\ldots,d_N)} \\ 1 - \frac{\Pr{[i\sim j]}_{(d_1,\ldots,d_N)}}{p_{i,j}} & \text{if } p_{i,j} < \Pr{[i \sim j]}_{(d_1,\ldots,d_N)} \\ 0 & \text{if } p_{i,j} = \Pr{[i \sim j]}_{(d_1,\ldots,d_N)}. \end{cases} \tag{B3}$$

We do the same for the probability term $\widehat{p}_{i,j}$ of our adjusted modularity, given in (20). In Fig. 8(a), we plot the heatmap of the error factor for the original probability term $p_{i,j}$, and in Fig. 8(b) our probability term $\widehat{p}_{i,j}$. As seen in the figure, our probability term $\widehat{p}_{i,j}$ is considerably more accurate than $p_{i,j}$ when applied to a network that is not Erdős-Rényi.

[1] A.-L. Barabási, *Network Science* (Cambridge University Press, Cambridge, 2016).

[2] M. E. J. Newman, *Networks*, 2nd ed. (Oxford University Press, New York, 2018).

[3] P. Van Mieghem, *Graph Spectra for Complex Networks*, 2nd ed. (Cambridge University Press, Cambridge, 2023).

[4] P. Van Mieghem, *Performance Analysis of Complex Networks and Systems*, 2nd ed. (Cambridge University Press, Cambridge, 2014).

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

[6] M. E. J. Newman, Fast algorithm for detecting community structure in networks, Phys. Rev. E **69**, 066133 (2004).

[7] A. Clauset, M. E. J. Newman, and C. Moore, Finding community structure in very large networks, Phys. Rev. E **70**, 066111 (2004).

[8] S. Fortunato and M. E. J. Newman, 20 years of network community detection, Nat. Phys. **18**, 848 (2022).

[9] V. Blondel, J.-L. Guillaume, and R. Lambiotte, Fast unfolding of communities in large networks: 15 years later, J. Stat. Mech. (2024) 10R001.

[10] R. Campigotto, P. Conde Céspedes, and J.-L. Guillaume, A generalized and adaptive method for community detection, arXiv:1406.2518.

[11] P. Conde-Céspedes and J. F. Marcotorchino, Comparing different modularization criteria using relational metric, in *Geometric Science of Information*, edited by F. Nielsen and F. Barbaresco (Springer, Berlin, Heidelberg, 2013), pp. 180–187.

[12] Z. Zhou, W. Wang, and L. Wang, Community detection based on an improved modularity, in *Pattern Recognition*, edited by C.-L. Liu, C. Zhang, and L. Wang (Springer, Berlin, Heidelberg, 2012), pp. 638–645.

[13] W. W. Zachary, An information flow model for conflict and fission in small groups, J. Anthropological Res. **33**, 452 (1977).

[14] M. E. J. Newman, Modularity and community structure in networks, Proc. Natl. Acad. Sci. USA **103**, 8577 (2006).

[15] A. Ferdowsi and A. Khanteymoori, Discovering communities in networks: A linear programming approach using max-min modularity, in *Proceedings of the 16th Conference on Computer Science and Intelligence Systems*, edited by M. Ganzha, L. Maciaszek, M. Paprzycki, and D. Ślęzak (IEEE, New York, 2021), pp. 329–335.

[16] T. N. Dinh and M. T. Thai, Finding community structure with performance guarantees in complex networks, arXiv:1108.4034.

[17] G. Agarwal and D. Kempe, Modularity-maximizing graph communities via mathematical programming, Eur. Phys. J. B **66**, 409 (2008).

[18] D. Kosmas, J. E. Mitchell, T. C. Sharkey, and B. K. Szymanski, Optimizing edge sets in networks to produce ground truth communities based on modularity, Networks **80**, 152 (2022).

[19] A. A. Hagberg, D. A. Schult, and P. J. Swart, Exploring network structure, dynamics, and function using NetworkX, in *Proceedings of the 7th Python in Science Conference*, edited by G. Varoquaux, T. Vaught, and J. Millman (SciPy, Pasadena, CA, 2008), pp. 11–15.

[20] A. Kaygun, Enumerating labeled graphs that realize a fixed degree sequence, arXiv:2101.02299.