

Detecting communities using asymptotical surpriseV. A. Traag,^{1,2,*} R. Aldecoa,³ and J.-C. Delvenne^{4,5}¹*Royal Netherlands Institute of Southeast Asian and Caribbean Studies, Leiden, The Netherlands*²*e-Humanities Group, Royal Netherlands Academy of Arts and Sciences, Amsterdam, The Netherlands*³*Department of Physics, Northeastern University, Boston, Massachusetts 02115, USA*⁴*ICTEAM, Université catholique de Louvain, Louvain-la-Neuve, Belgium*⁵*CORE, Université catholique de Louvain, Louvain-la-Neuve, Belgium*

(Received 2 March 2015; published 24 August 2015)

Nodes in real-world networks are repeatedly observed to form dense clusters, often referred to as communities. Methods to detect these groups of nodes usually maximize an objective function, which implicitly contains the definition of a community. We here analyze a recently proposed measure called surprise, which assesses the quality of the partition of a network into communities. In its current form, the formulation of surprise is rather difficult to analyze. We here therefore develop an accurate asymptotic approximation. This allows for the development of an efficient algorithm for optimizing surprise. Incidentally, this leads to a straightforward extension of surprise to weighted graphs. Additionally, the approximation makes it possible to analyze surprise more closely and compare it to other methods, especially modularity. We show that surprise is (nearly) unaffected by the well-known resolution limit, a particular problem for modularity. However, surprise may tend to overestimate the number of communities, whereas they may be underestimated by modularity. In short, surprise works well in the limit of many small communities, whereas modularity works better in the limit of few large communities. In this sense, surprise is more discriminative than modularity and may find communities where modularity fails to discern any structure.

DOI: [10.1103/PhysRevE.92.022816](https://doi.org/10.1103/PhysRevE.92.022816)

PACS number(s): 89.20.Ff, 89.20.Hh

I. INTRODUCTION

Networks are often used as a model to describe interactions among components of a system [1,2]. In its simplest form, a network is composed of a set of vertices (also called nodes) and a set of edges connecting them. Many real-world systems can be reduced to this scheme, such as social networks establishing relations among individuals, proteins interacting within a cell, or roads connecting different cities [3]. What caught the interest of the scientific community was that most of these real networks share high-order structural patterns and dynamics, such as a wide heterogeneity in the number of neighbors of a node, the presence of many triangles, or a very low network diameter [4,5]. Another feature observed in real networks is the presence of densely connected groups of nodes, known as communities [6]. Nodes in the same group usually share similar characteristics or functions, and, therefore, methods to detect communities in networks are of much interest across different fields [7–12].

Researchers have proposed numerous strategies to detect the community structure of a network [6,13–15]. Ultimately, most methods optimize a given objective function to find a partition into communities. This function contains, either explicitly or implicitly, its own definition of a community. Modularity [16] has been, since its inception, the most extensively used measure for community detection. It belongs to a wider class of functions in which communities are defined by Potts model spin states and the quality of the partition is given by the energy of the system [17,18]. Although this approach based on statistical mechanics may be appealing, empirical evidence shows that in many cases these methods are unable to capture the expected communities of the network [15,19–22].

In fact, numerous studies have pointed out strong theoretical limitations of modularity approaches for community detection [23–29].

A proposed measure based on classical probability, called surprise [30], has been shown to systematically outperform modularity-based methods on different benchmarks [15,21]. Here we demonstrate how surprise can be expressed under an information-theoretic framework by examining its asymptotic formulation. In particular, we describe surprise in terms of the Kullback-Leibler (KL) divergence [31]. This asymptotic formulation allows us to develop, for the first time, an efficient surprise maximization algorithm. Incidentally, this also points to a straightforward extension of surprise to weighted graphs. Additionally, this enables a better analysis of its performance and allows an analytic comparison to other methods.

In particular, we compare surprise to a modularity model and the recently introduced measure of significance, which also detects communities based on the KL divergence [22]. We show that surprise is more discriminative than modularity using an Erdős-Rényi null model and that significance and surprise behave relatively similar. Additionally, we analyze the limitations of community detection, most notably the resolution limit [23] and the detectability threshold [32]. We show that surprise is (nearly) unaffected by the resolution limit and works well in the limit of large number of communities with fixed community sizes. However, in the limit of large community sizes with a fixed number of communities, surprise works worse than ER modularity, as it tends to find smaller subgraphs within those larger communities.

Apart from the choice of the null model, a key component in community detection is how the difference between the actual community structure and the null model is quantified. Relying on the KL divergence to measure such difference results in more discriminative methods. We believe that this fact can improve current and future community detection strategies.

*traag@kitlv.nl

II. SURPRISE

In general, we denote a graph by $G = (V, E)$ consisting of nodes $V = \{1, \dots, n\}$ and edges $E \subseteq V \times V$, which has $n = |V|$ nodes and $m = |E|$ links. The total number of possible links is denoted by $M = \binom{n}{2}$, and the ratio of present links $p = \frac{m}{M}$ is known as the density of the graph.

The general aim is to find a good partition $\mathcal{V} = \{V_1, V_2, \dots, V_r\}$ of the graph, where each $V_c \subseteq V$ is a set of nodes, which we call a community. Such communities are nonoverlapping (i.e., $V_c \cap V_d = \emptyset$ for all $c \neq d$) and cover all the nodes (i.e., $\bigcup V_c = V$). Each community consists of $n_c = |V_c|$ nodes and contains $m_c = |E_c|$ edges. Obviously, then $\sum_c n_c = n$, but the total number of internal edges $m_{\text{int}} = \sum_c m_c$ is smaller than the total number of edges, so that $m_{\text{int}} \leq m$.

Surprise is a statistical approach to assess the quality of a partition into communities. Given a graph with n nodes, there are $M = \binom{n}{2}$ possible ways of drawing m edges. Of those, there are $M_{\text{int}} = \sum_c \binom{n_c}{2}$ possible ways of drawing an internal edge. Surprise is then defined as the (minus logarithm of the) probability of observing at least m_{int} successes (internal edges) in m draws *without* replacement from a finite population of size M containing exactly M_{int} possible successes [30,33],

$$\mathcal{S}(\mathcal{V}) = -\ln \sum_{i=m_{\text{int}}}^{\min(m, M_{\text{int}})} \frac{\binom{M_{\text{int}}}{i} \binom{M-M_{\text{int}}}{m-i}}{\binom{M}{m}}, \quad (1)$$

which derives from the hypergeometric distribution.

A. Asymptotic formulation

However, this formulation presents some difficulties. It is not straightforward to work with, nor is it simple to implement in an optimization procedure, mainly due to numerical computational problems. Since we are usually interested in relatively large graphs, an asymptotic approximation may provide a good alternative. The asymptotic expansion we consider here assumes that the graph grows, but that the relative number of internal edges $q = \frac{m_{\text{int}}}{m}$ and the relative number of expected internal edges $\langle q \rangle = \frac{M_{\text{int}}}{M}$ remains fixed. By only considering the dominant term, we obtain a simple and elegant approximation (see Appendix A),

$$\mathcal{S}(\mathcal{V}) \approx mD(q\| \langle q \rangle), \quad (2)$$

where $D(x\|y)$ is the KL divergence,

$$D(x\|y) = x \ln \frac{x}{y} + (1-x) \ln \frac{1-x}{1-y}. \quad (3)$$

The KL divergence measures the distance between two probability distributions (although it is not a proper metric), within this case the Bernoulli probability distributions $x, 1-x$ and $y, 1-y$. Notice that, in general, $D(x\|y) \neq D(y\|x)$. In this case, q and $\langle q \rangle$ denote the probability that a link lies (or is expected to lie) within a community. Whenever $q = \langle q \rangle$, we have that $D(q\| \langle q \rangle) = 0$ and, otherwise, $D(q\| \langle q \rangle) > 0$. Since we are looking for relatively dense communities, we generally have $q > \langle q \rangle$.

The original formulation of surprise in Eq. (1), based on a hypergeometric distribution, can be accurately approximated by a binomial distribution. The only difference between both

approaches is that in the former links are drawn without replacement. Consider again $q = \frac{m_{\text{int}}}{m}$, the fraction of internal edges in the partition, and $\langle q \rangle = \frac{M_{\text{int}}}{M}$, the expected fraction of internal edges. The binomial formulation of surprise would then be

$$\mathcal{S}(\mathcal{V}) = -\ln \sum_{i=m_{\text{int}}}^{\min(m, M_{\text{int}})} \binom{m}{i} \langle q \rangle^i (1-\langle q \rangle)^{m-i}. \quad (4)$$

The asymptotic development for the dominant term of binomial surprise is simpler. We use Stirling's approximation,

$$\ln \binom{n}{k} \approx n H\left(\frac{k}{n}\right), \quad (5)$$

where $H(x) = -x \ln x - (1-x) \ln(1-x)$ is the (binary) entropy and we use that $m_{\text{int}} = qm$. Binomial surprise then becomes

$$\begin{aligned} \mathcal{S}(\mathcal{V}) &\approx -m[H(q) + q \ln \langle q \rangle + (1-q) \ln(1-\langle q \rangle)] \\ &= mD(q\| \langle q \rangle). \end{aligned}$$

Thus, as expected, for large sparse networks the difference between drawing with or without replacement is negligible.

B. Algorithm

Evaluating the quality of a partition using surprise shows excellent results in standard benchmarks. In fact, it has been shown that a meta-algorithm of selecting the partition with the highest surprise, from a set of candidate solutions provided by the best community detection algorithm solutions, outperforms any single algorithm by itself [15,21,34]. However, no algorithm for directly optimizing surprise has been developed yet.

The asymptotic formulation allows a straightforward algorithmic implementation in a similar fashion as the Louvain algorithm [35], which was initially designed to optimize modularity. The basic idea of the Louvain algorithm consists of two steps. We move around nodes from one community to another so as to greedily improve surprise. When surprise can no longer be improved by moving around individual nodes, we aggregate the graph and repeat the procedure on the aggregated graph.

The aggregation of the graph is simply the contraction of all nodes within a community to a single "community node." The multiplicities of the edges are kept as weighted edges, so that $w_{cd} = \sum_{i \in V_c, j \in V_d} w_{ij}$ denotes the weight between the new nodes c and d in the aggregate graph, where initially $w_{ij} = A_{ij}$. Here $A_{ij} = 1$ if there is an edge between i and j and 0 otherwise. We additionally need a node size to keep track of the total size of the communities, similar to [29]. Initially, we set this node size to $n_i = 1$, and upon aggregation the node size $n_c = \sum_{i \in V_c} n_i$ is set to the total number of nodes within the community.

One of the essential elements of the Louvain algorithm is that the surprise of the partition on the aggregated graph is the same as the surprise of the original partition on the original graph. This ensures that moving a node in the aggregated graph corresponds to moving a whole community in the original graph. In other words, if \mathcal{V} denotes the partition of G and

$\mathcal{V}' = \{1, 2, \dots, r\}$ denotes the default partition of the aggregated graph G' , then $\mathcal{S}(\mathcal{V}, G) = \mathcal{S}(\mathcal{V}', G')$. For calculating surprise in the aggregated graph, we then use $m_c = \sum_{i, j \in V_c'} w_{ij}$ as the internal weight and $n_c = \sum_{i \in V_c'} n_i$ as the community size and $n = \sum_c n_c$. With the other definitions remaining the same, it is straightforward to see that $\mathcal{S}(\mathcal{V}, G) = \mathcal{S}(\mathcal{V}', G')$. Notice that the same formulations can also be applied to the original graph, when using $w_{ij} = A_{ij}$ and $n_i = 1$.

Using this formulation of the aggregate graph, it is quite straightforward to calculate the improvement in surprise when moving a node. Before we move node i from community c to community d , assume we have m_{int} internal edges and M_{int} possible internal edges. The total weight between node i and community c is $w_{ic} = \sum_{j \neq i \in V_c} w_{ij}$ and similarly between node i and community d , with a possible self-loop of w_{ii} . The new internal weight after moving node i from community c to community d is then $m'_{\text{int}} = m_{\text{int}} - w_{ic} + w_{id}$. The change in $M_{\text{int}} = \sum_c \binom{n_c}{2}$ is slightly more complicated. After the move, we obtain $n'_c = n_c - n_i$ and $n'_d = n_d + n_i$, so that $M'_{\text{int}} = M_{\text{int}} + n_i(n_i + n_d - n_c)$. Finally, we use $q' = \frac{m'_{\text{int}}}{m}$ and $\langle q' \rangle = \frac{M'_{\text{int}}}{M}$. The difference in surprise for moving node i from community c to community d is then simply

$$\Delta \mathcal{S}(\sigma_i = c \mapsto d) = m[D(q \| \langle q \rangle) - D(q' \| \langle q' \rangle)], \quad (6)$$

where we denote the community of node i by σ_i (i.e., $\sigma_i = c$ if $i \in V_c$). The algorithm can then be simply summarized as follows:

```

function OPTIMIZE SURPRISE (Graph  $G$ )
  while improvement do
     $\sigma_i \leftarrow i$  for  $i = 1, \dots, |V(G)|$ .  $\triangleright$  Initial partition
    while improvement do
      for random  $v \in V(G)$  do
         $\sigma_v \leftarrow \arg \max_d \Delta \mathcal{S}(\sigma_v = c \mapsto d)$ 
      end for
    end while
     $\sigma'_i = \sigma_{\sigma'_i}$   $\triangleright$  Community in original graph.
     $G \leftarrow \text{AGGREGATEGRAPH}(G)$ 
  end while
  return  $\sigma'$ 
end function

```

Incidentally, our formulation for surprise for the aggregated graph yields a weighted version of surprise. While keeping the same formulation of surprise as in Eq. (2), we only need to change the definitions of q and $\langle q \rangle$. Then $q = \sum_c w_c / w$, where $w_c = \sum_{i, j \in V_c} w_{ij}$ is the internal weight and $w = \sum_{ij} w_{ij}$ is the total weight. Assuming then a uniform distribution of weights across the graph in the random graph, the expected weights of an edge would be $\langle w \rangle$, which would not show too much deviation. The total possible internal weight is then $\langle w \rangle M_{\text{int}}$, while the total possible weight would be $\langle w \rangle M$. Hence, $\langle q \rangle = M_{\text{int}} / M$ remains unchanged.

We provide an open-source, fast and flexible C++ implementation of the optimization of Surprise using the Louvain algorithm. It is suitable for use in PYTHON using the IGRAPH package. This implementation is available from GITHUB [36] as LOUVAIN-IGRAPH and from PYPI [37] simply as LOUVAIN and implements various other methods as well.

III. COMPARISON

We now review how surprise compares to some closely related methods. There are many other methods still, and we cannot do all of them justice here. For a more comprehensive review, please refer to [6,38].

A. Modularity

Although relatively recent, modularity has rapidly become an extremely popular method for community detection. The general idea is that we want to find a partition, such that the communities have more internal links than expected. In its original formulation, modularity assumes a null model in which the degree k_i of a node is fixed [16], the so called configuration model [39]. This implies that the expected number of internal edges is

$$\langle m_c \rangle = \frac{K_c^2}{4m}, \quad (7)$$

where $K_c = \sum_{i \in V_c} k_i$ is the total degree of nodes in community c . Modularity compares this value to the observed number of edges m_c within the community and simply sums the difference. The measure is usually normalized by the total number of edges, obtaining

$$\mathcal{Q}_{\text{CM}}(\mathcal{V}) = \frac{1}{m} \sum_c \left(m_c - \frac{K_c^2}{4m} \right). \quad (8)$$

This random graph null model represents the configuration model, where the degree dependency of the nodes is taken into account. We therefore refer to it as the CM modularity.

Alternative derivations of modularity have been proposed, some of them with different null models [17]. Surprise implicitly assumes a null model in which every edge appears with the same probability p , as in an Erdős-Rényi (ER) random graph. The number of expected edges in a community of size c is thus

$$\langle m_c \rangle = p \binom{n_c}{2}. \quad (9)$$

Plugging this null model into modularity, we obtain its ER version [17],

$$\mathcal{Q}_{\text{ER}}(\mathcal{V}) = \frac{1}{m} \sum_c \left[m_c - p \binom{n_c}{2} \right]. \quad (10)$$

There is an interesting relationship between this ER modularity and surprise. Given that $p = m/M$, we can write

$$\mathcal{Q}_{\text{ER}}(\mathcal{V}) = \sum_c \frac{m_c}{m} - \sum_c \frac{\binom{n_c}{2}}{M} \quad (11)$$

$$= q - \langle q \rangle. \quad (12)$$

By Pinsker's inequality this is related to the KL divergence as

$$q - \langle q \rangle \leq \sqrt{\frac{1}{2} D(q \| \langle q \rangle)}, \quad (13)$$

and, therefore,

$$\mathcal{S}(\mathcal{V}) = m D(q \| \langle q \rangle) \geq 2m \mathcal{Q}_{\text{ER}}(\mathcal{V})^2. \quad (14)$$

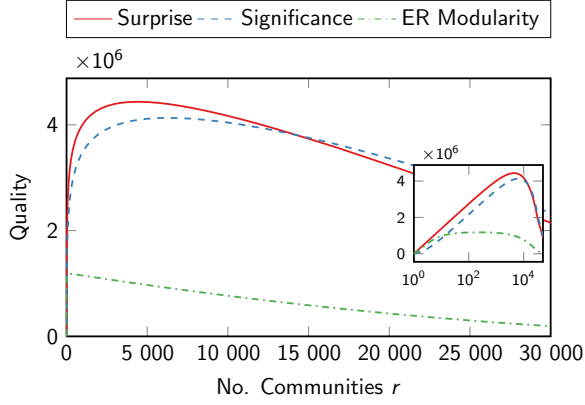


FIG. 1. (Color online) Comparison of bounds. We show the quality of partitions of a lattice in r communities. ER modularity quickly reaches a maximum for few communities (we show $2mQ_{\text{ER}}^2$ rather than Q_{ER} for comparison). Both significance and surprise reach a maximum for many more communities. This illustrates that ER modularity is simply unable to discern partitions with such a high number of communities. The inset shows the same data, but on a logarithmic x axis.

This implies that whenever surprise is low, modularity is also low. Whenever a good partition (in the sense of being different from random) cannot be found by surprise, it is unlikely that modularity will be able to find one. While Eq. (14) is sometimes tight, on some partitions surprise can be much larger than modularity, making it more likely to be selected as optimal, while escaping the scrutiny of modularity optimization. In this sense, surprise is more discriminative than modularity

To illustrate this, consider a one-dimensional circular lattice with neighbors within distance 3. In other words, node i is connected to nodes $i - 3 \bmod n$ to $i + 3 \bmod n$ (excluding the self-loop). We create partitions consisting of r communities by grouping consecutive nodes such that n/r nodes are in the same community. The ER modularity reaches its maximum with just a few communities (Fig. 1). Modularity indeed often detects only few communities, part of the problem of its resolution limit [23,24,29]. Both surprise and significance (see the next section) still increase, whereas ER modularity is already decreasing again. ER modularity may not be able to discern partitions with many communities, whereas surprise and significance can. On the other hand, when surprise goes to 0, we see that ER modularity indeed also goes to 0, showing the upper bound provided by surprise.

B. Significance

Significance [22], a recently introduced objective function to evaluate community structure quality, presents an approach similar to surprise. Surprise describes how likely it is to observe m_{int} internal links in communities. Significance, on the other hand, looks at how likely such dense communities appear in a random graph. Comparing the two measures is not immediately straightforward. On the one hand, if dense communities are unlikely to be present in a random graph (high significance), then a community is also unlikely to contain many links at random (high surprise). On the other hand, if a

community is unlikely to contain many links at random (high surprise), perhaps there are still communities elsewhere in the random graph that contain so many links. Therefore, we should compare the two more formally to make more exact statements.

Asymptotically, significance is defined as

$$\mathcal{Z}(\mathcal{V}) = \sum_c \binom{n_c}{2} D(p_c \| p), \quad (15)$$

where $p_c = m_c / \binom{n_c}{2}$ is the density of community c , p is the density of the graph, and $D(x \| y)$ is again the KL divergence. Significance also showed a great performance in standard benchmarks and helped to determine the proper scale of resolution in multiresolution methods [22].

Both surprise and significance are based on the KL divergence to compare the actual number of internal edges to the expected one. However, they do so in different ways. Whereas surprise compares such difference using global quantities, q and $\langle q \rangle$, significance compares each community density p_c to the average graph density p .

This implies, among other things, that only significance is affected by the actual distribution of edges between communities. In particular, moving edges from a denser community (with a high p_c) to a sparse community (with a low p_c) generally decreases the value of significance. This means that if all communities have the same density, *ceteris paribus*, significance is minimal. This intuition is confirmed by convexity of the KL divergence (see Appendix B), so that significance is lower bounded by

$$\mathcal{Z}(\mathcal{V}) \geq M_{\text{int}} D(\langle p_c \rangle \| p), \quad (16)$$

with the weighted average density

$$\langle p_c \rangle = \sum_c \frac{\binom{n_c}{2}}{M_{\text{int}}} p_c = \frac{m_{\text{int}}}{M_{\text{int}}} = p \frac{q}{\langle q \rangle}. \quad (17)$$

Convexity of the KL divergence also shows that

$$\mathcal{Z}(\mathcal{V}) \geq \mathcal{S}(\mathcal{V}) \quad (18)$$

whenever $\langle q \rangle < p$ (see Appendix B). To gain more insight, we can slightly rewrite $\langle q \rangle$ to obtain

$$\langle q \rangle = \frac{\sum_c \binom{n_c}{2}}{\binom{n}{2}} \approx \frac{\sum_c n_c^2}{n^2} = \frac{1}{r} \frac{\langle n_c^2 \rangle}{\langle n_c \rangle^2}. \quad (19)$$

Then, in general, $\langle q \rangle$ will be inversely proportional to the number of communities and increases with the variance of the community sizes n_c . Hence, if the number of communities is relatively large (small $\langle q \rangle$), or the network is relatively dense (large p), significance is more discriminative than surprise. However, in the case that $\langle q \rangle > p$, Surprise can be more discriminative than significance (see Appendix B). Notice that if $\langle q \rangle = p$, then $p_c = q$, so that $D(\langle p_c \rangle \| p) = D(q \| \langle q \rangle)$ and significance and surprise values are close to each other. Therefore, the two measures are expected to behave relatively similarly, especially for $\langle q \rangle \approx p$. Nonetheless, in dense networks with many communities significance would be more discriminative, whereas for fewer communities or sparse graphs, surprise would show a better performance.

IV. LIMITATIONS

Although modularity was lauded by the possibility to detect communities without specifying the number of communities, this came at a certain price. One of the best known problems in community detection is the resolution limit [23], which prevents modularity from detecting small communities. It thus tends to underestimate the number of communities in a graph and lumps together several smaller communities in larger communities. Moreover, this depends on the scale of the graph, so that modularity has a problem of scale. It was shown that this is the case for both ER and CM modularity and that other null models also suffer from the same drawbacks [24]. In fact, most methods are expected to suffer from this problem, and only few methods are able to avoid it completely [29]. Additionally, there is also a lower counterpart to the resolution limit, leading to unnecessary splitting of cliques [40,41]. Finally, modularity is also myopic, cutting across long dendrites [42]. Another fundamental limit in community detection is called the detectability threshold [32], which also has some counterintuitive effects [43]. This prevents any method from correctly detecting communities beyond this threshold. The asymptotic formulation of surprise enables us to understand better how it performs with respect to these limitations.

A. Resolution limit

The resolution limit is traditionally studied through the ring of cliques [23]. This is a graph consisting of r cliques (i.e., completely connected subgraphs) connected only by one link between two cliques to form a ring. This is one of the most modular structure possible: We cannot delete more than one link between communities and still keep it connected, while we cannot add any more links within the cliques. When a method starts to join the cliques, it can no longer detect the smaller cliques, and so, *a fortiori*, cannot detect less well defined subgraphs either. We denote by q_1 (and $\langle q_1 \rangle$) the (expected) proportion of edges within communities for the partition where each community contains a single clique and use q_2 (and $\langle q_2 \rangle$) for the partition where each community contains two cliques. To facilitate the derivation, we work with self-loops (and directed edges), so that the total number of edges is n_c^2 within communities, respectively. Let r denote the number of cliques. Then, obviously, $n = rn_c$ and $m = rn_c^2 + 2r$. For the partition of each clique in its own community we then obtain

$$q_1 = \frac{n_c^2}{n_c^2 + 2}, \quad \langle q_1 \rangle = \frac{1}{r}, \quad (20)$$

while for the partition with two cliques merged we obtain

$$q_2 = \frac{n_c^2 + 1}{n_c^2 + 2}, \quad \langle q_2 \rangle = \frac{2}{r}. \quad (21)$$

Hence, $q_2 = q_1 + \epsilon$ with $\epsilon = \frac{1}{n_c^2 + 2}$ and $\langle q_2 \rangle = 2\langle q_1 \rangle$. The difference of surprise is

$$\Delta S = \frac{S_2 - S_1}{m} = D(q_2 \| \langle q_2 \rangle) - D(q_1 \| \langle q_1 \rangle), \quad (22)$$

which works out to

$$\begin{aligned} \Delta S = & q_1 \ln \frac{q_2}{\langle q_2 \rangle} \frac{\langle q_1 \rangle}{q_1} + (1 - q_1) \ln \frac{1 - q_2}{1 - \langle q_2 \rangle} \frac{1 - \langle q_1 \rangle}{1 - q_1} \\ & + \epsilon \ln \frac{q_2}{\langle q_2 \rangle} \frac{1 - \langle q_2 \rangle}{1 - q_2}. \end{aligned} \quad (23)$$

Approximating $r - 2 \approx r - 1 \approx r$, we obtain

$$\Delta S \approx -D(q_1 \| q_2) - q_1 \ln 2 + \epsilon \ln \frac{r}{2} \frac{q_2}{1 - q_2}. \quad (24)$$

Solving for r at the point at which $\Delta S = 0$ yields

$$r = 2 \frac{1 - q_2}{q_2} \exp \left[\frac{1}{\epsilon} D(q_1 \| q_2) \right] 2^{\frac{q_1}{\epsilon}}, \quad (25)$$

which scales as $r \sim \frac{2n_c^2}{n_c^2}$ so that for larger r surprise starts to merge cliques.

Working out the inequality for both CM and ER modularity we obtain that $r \sim n_c^2$. Hence, the number of cliques r at which modularity starts to merge cliques lies considerably lower than for surprise and grows linearly with the square of community sizes rather than exponentially. So, although surprise shows a similar problem as modularity, it only starts to show at really large graphs, so is unlikely to be a problem in any empirical graph. Indeed, this demonstrates exactly the key difference between modularity and surprise: The first is unable to detect relatively small communities in large graphs, whereas the latter has (nearly) no such difficulties.

B. Detectability threshold

In order to study the detectability threshold, we first introduce the planted partition model. This means that we build a graph such that it will contain a specified partition: We plant it in the graph. We create n nodes and assign each node to a certain community. An edge within a community is created with probability p_{in} , whereas an edge between two communities is created with probability p_{out} . We define the probability of an internal edge p_{in} and the probability of an external edge to be, respectively,

$$p_{\text{in}} = \frac{(1 - \mu)k}{n_c - 1}, \quad p_{\text{out}} = \frac{\mu k}{n - n_c}, \quad (26)$$

so that the average degree is k and μ is the probability that an edge is between communities. When $\mu = 0$ all links are thus placed within the planted communities, whereas for $\mu = 1$ all links are placed between the planted communities. Uncovering the planted communities correctly is trivial for $\mu = 0$ but becomes increasingly more difficult for higher μ . The average degree within a cluster is $k_{\text{in}} = (1 - \mu)k$, while the average degree between clusters is $k_{\text{out}} = \mu k$. We denote community sizes by n_c for the r different communities.

Notice that, most conveniently, $q = 1 - \mu$, while $\langle q \rangle = \frac{1}{r} \frac{\langle n_c^2 \rangle}{\langle n_c \rangle^2}$. We can thus easily calculate \mathcal{S}_{plt} the surprise for the planted partition. Since $\mathcal{S} > 0$ by definition, communities can thus only be detected when $1 - \mu > \frac{1}{r} \frac{\langle n_c^2 \rangle}{\langle n_c \rangle^2}$. This yields the rather trivial detectability threshold of

$$\mu < \frac{r - \frac{\langle n_c^2 \rangle}{\langle n_c \rangle^2}}{r}. \quad (27)$$

In the case of equisized communities, this reduces to the familiar trivial threshold $\mu < \frac{r-1}{r}$ [19].

However, due to stochastic fluctuation, the communities become already ill defined prior to the threshold. Indeed, $\mathcal{S} = 0$ provides a rather naive bound, since $\mathcal{S} > 0$ also in random graphs. In general, $\mathcal{S} = 0$ for both trivial partitions of one large community and n small communities (since then $q = \langle q \rangle$), so that optimizing surprise in a random graph will yield some partition with strictly positive surprise. This implies that at some (lower) critical μ^* the community structure is essentially no longer discernible from the community structure in a random graph. Hence, we should not consider when $\mathcal{S}_{\text{plt}} > 0$ but when $\mathcal{S}_{\text{plt}} > \mathcal{S}_{\text{rnd}}$, where \mathcal{S}_{rnd} is the surprise attainable in a random graph. We first examine the case with $r = 2$ and $n_c = n/2$. Previous literature found a detectability threshold for $k_{\text{in}} - k_{\text{out}} \leq \sqrt{k_{\text{in}} + k_{\text{out}}}$ [32,44,45]. Beyond this threshold, the optimal bisection becomes indiscernible from an optimal bisection in a random graph. This threshold thus coincides with the expected number of internal edges for an optimal bisection in a random graph. We can use this to calculate $\mathcal{S}_{\text{rnd}}(2)$, the maximum surprise for a bisection in a random graph. Let us denote by $q_{\text{rnd}}(2)$ the probability an edge is within a community in the best bisection for a random graph. Substituting $k_{\text{in}} = q_{\text{rnd}}(2)k$ and $k_{\text{out}} = [1 - q_{\text{rnd}}(2)]k$ and solving for $q_{\text{rnd}}(2)$ yields

$$q_{\text{rnd}}(2) = \frac{1}{2} \left(1 + \sqrt{\frac{1}{k}} \right). \quad (28)$$

We thus obtain $\mathcal{S}_{\text{rnd}}(2) = mD[q_{\text{rnd}}(2)\|\frac{1}{2}]$ for the maximum surprise for a bisection in a random graph. If $\mathcal{S}_{\text{plt}}(2) < \mathcal{S}_{\text{rnd}}(2)$, the planted partition is no longer optimal, and we will likely find an alternative partition with surprise equal to $\mathcal{S}_{\text{rnd}}(2)$. The threshold is then $\mu^* = 1 - q_{\text{rnd}}(2)$, congruent with previous results. So, in general, surprise is expected to show similar behavior concerning the detectability threshold as other methods.

However, this analysis restricts itself to finding the same number of communities (i.e., two in this case), while it is possible that an optimal partition would split the graph in more communities. In other words, we need to compare the surprise of the planted partition to the maximum surprise in a random graph, while allowing more than two communities. Although the expected value of the maximum surprise in a random graph is not easy to find, a random graph is likely to contain a near perfect matching. Using that, we can derive a lower bound on the expected surprise in a random graph. In such a perfect matching there are $r = \frac{n}{2}$ communities which contain one link each. For a graph that contains $m = nk$ edges, then $q = \frac{1}{2k}$, while $\langle q \rangle = \frac{2}{n}$. This leads to a surprise of approximately $\mathcal{S}_{\text{rnd}}(\frac{n}{2}) \sim \frac{n}{2} \ln \frac{n}{4k}$. Hence, whenever we obtain that $\mathcal{S}_{\text{plt}} \leq \mathcal{S}_{\text{rnd}}(\frac{n}{2})$, optimization should find another partition than the planted one. In the case of two planted communities, we require that $D(1 - \mu\|\frac{1}{2}) \geq \frac{\ln \frac{n}{2k}}{2k}$ to make sure that we still detect the two clusters. Although we cannot solve explicitly for μ , this inequality shows that n is bounded above by

$$n \leq 4ke^{2kD(1-\mu\|\frac{1}{2})}. \quad (29)$$

If n grows large, there is likely some structure arising from random fluctuations within the planted communities. Notice

that there are likely better partitions than a perfect matching. We can therefore expect the actual critical n for which the planted partition is no longer optimal to be lower.

We can similarly derive such thresholds for ER modularity. For a perfect matching the ER modularity is $Q_{\text{rnd}}(\frac{n}{2}) = \frac{1}{2k} - \frac{2}{n}$. Then solving $Q_{\text{plt}} \leq Q_{\text{rnd}}(\frac{n}{2})$ gives us an estimate of when ER modularity is likely to find an alternative partition (i.e., a perfect matching in this case). The critical μ^* can in this case be explicitly derived and yields $\mu^* = \frac{1}{2}(1 - \frac{1}{k} + \frac{4}{n})$. However, the detectability threshold is already reached before that point at $\mu^* = \frac{1}{2}(1 - \sqrt{\frac{1}{k}})$, leaving n essentially unbounded. Again, there will be better partitions than a perfect matching, so that n may still be bounded to some extent. Nonetheless, this shows that ER modularity is less affected by the size of the communities than surprise and is less likely to find substructure within the planted communities.

In summary then, surprise does not tend to suffer from the resolution limit, but does quickly find substructure due to random fluctuations. ER modularity, on the other hand, suffers from a resolution limit, but tends to ignore substructure in communities. Stated differently, for a planted partition model with r communities and $n = rn_c$ nodes, surprise and ER modularity work well in different limits. Whenever $r \rightarrow \infty$ with n_c fixed, surprise works well, but ER modularity works poorly. Whenever r is fixed but $n_c \rightarrow \infty$, ER modularity works well, but surprise works poorly. An interesting question would concern which method would work well for both limits.

V. EXPERIMENTAL RESULTS

We here confirm our theoretical results experimentally. We first show numerically that the asymptotic formulation of surprise provides an excellent approximation. Second, we validate the inequalities between surprise, significance, and ER modularity. Third, we show the different limitations on surprise and modularity. Finally, we demonstrate that the asymptotic formulation of surprise performs very well in LFR benchmarks [46].

For comparing the asymptotic formulation with the exact hypergeometrical and binomial formulation, we used regular rooted trees with three children. To create such trees, we first create the root node and add three children to this root node. We then keep on adding children to the leaves of the tree until we obtain the desired number of nodes. We use trees to minimize the number of edges to prevent numerical problems with the hypergeometrical and binomial formulation. Using relatively large numbers results in numerical issues, preventing a comparison to the asymptotic formulation. We optimize asymptotic surprise using the Louvain algorithm to find a partition on this graph. As can be seen in Fig. 2, the approximation is quite good, and the approximation ratio tends to 1. Notice that the number of nodes in these graphs is limited to 200, whereas complex networks are usually much larger. Hence, we expect the approximation to be accurate for any real network.

To demonstrate the limitations on surprise and (ER) modularity, we create some test networks with a planted partition. We generate networks with average degree $\langle k \rangle = 10$ and set $\mu = 0.1$. In the first test, we create networks with fixed

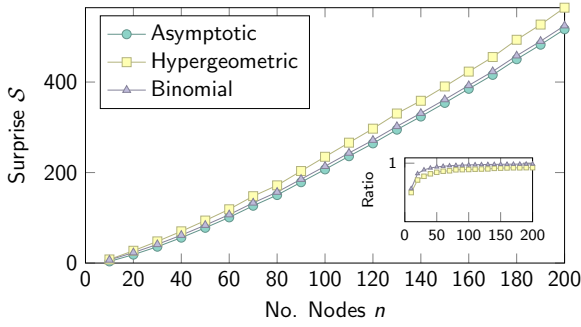


FIG. 2. (Color online) Approximation of surprise. The asymptotic formulation of surprise, using the KL divergence, approximates well both the binomial and the hypergeometric surprise. The inset shows the approximation ratio $S_{\text{asym}}/S_{\text{hyper}}$ and $S_{\text{asym}}/S_{\text{binom}}$, both going to 1 for large graphs.

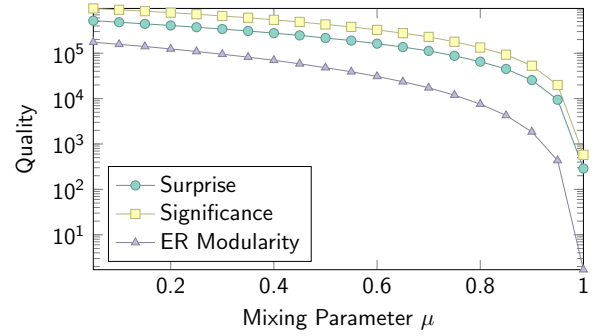


FIG. 4. (Color online) Inequalities. In most cases significance is more discriminative than surprise, which is more discriminative than the ER modularity, so that $\mathcal{Z} > \mathcal{S} > \mathcal{Q}_{\text{ER}}$. These inequalities clearly hold over the whole range of the mixing parameter μ for LFR benchmarks ($n = 10^4$). For ER modularity we display $2mQ_{\text{ER}}^2$, as used in Eq. (14).

community sizes $n_c = 10$ and vary the number of communities r . In the second test, we have fixed the number of communities to 2 but vary the community size n_c from 10 to 500. We consider whether the planted partition remains optimal by analyzing the quality of the planted partition \mathcal{S}_{plt} (or \mathcal{Q}_{plt} for modularity) and the partition found through optimization \mathcal{S} (or \mathcal{Q} for modularity). Whenever $\mathcal{S}_{\text{plt}} < \mathcal{S}$ we thus know that the planted partition remains no longer optimal. The results shown in Fig. 3 clearly confirm our theoretical analysis. In the case that $r \rightarrow \infty$ with fixed n_c , surprise does well, whereas (ER) modularity suffers from the resolution limit. In the case where r is fixed to $r = 2$, but $n_c \rightarrow \infty$, surprise does less well, as it tends to find subgraphs within the two large communities. Modularity also has problems identifying the optimal bisection. Indeed, the uncovered partitions do

not coincide exactly with the planted partition, even though the modularity value remains rather similar. Such partitions are likely to occur because of the degeneracy of modularity [20]. Nonetheless, our results show that the modularity of the planted partition remains (nearly) optimal, whereas surprise for the planted partition clearly diminishes compared to surprise of the uncovered partitions.

We also tested the various methods more extensively using benchmark graphs with a more realistic community size and degree distribution [46]. We set the average degree $\langle k \rangle = 20$ while the maximum degree is 50 and follows a power law degree distribution with exponent 2. Planted community sizes range from 10 to 50 for the “small” communities and from 20 to 100 for “large” communities. The planted community

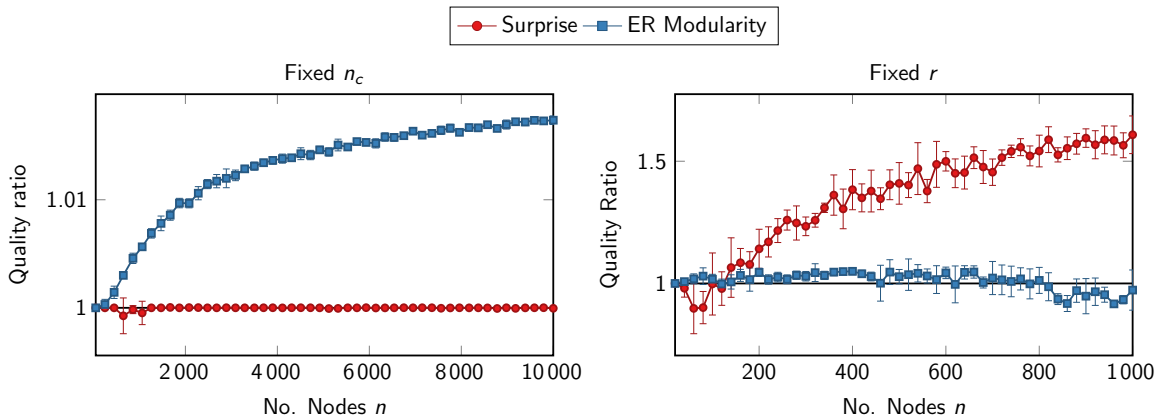


FIG. 3. (Color online) Limitations on community detection. We construct graphs with a planted partition, with a probability of an edge between communities of $\mu = 0.1$. We show the quality ratio $\frac{\mathcal{S}}{\mathcal{S}_{\text{plt}}}$ between the quality of the partition found by optimization \mathcal{S} and the quality of the planted partition \mathcal{S}_{plt} (and similarly for ER modularity). Hence, if the quality ratio $\frac{\mathcal{S}}{\mathcal{S}_{\text{plt}}} > 1$, the planted partition is no longer optimal. In the figure on the left we consider the case for fixed community size $n_c = 10$, but increase the number of communities r . The results show that in this case surprise finds the planted partitions, whereas ER modularity has more difficulties, in line with our analysis. This is mostly due to the resolution limit in modularity, which underestimates the number of communities. In the figure on the right we consider the case of a fixed number of communities $r = 2$ but an increasing community size n_c . In this case, surprise quickly finds other partitions than the planted partition, whereas modularity remains closer to the planted partition, consistent with our analysis. This is mostly because surprise tends to find substructure in the rather large communities arising from random fluctuations, which thus overestimates the number of communities. However, modularity also has some difficulty in finding the planted partition. This figure shows the average over five replications for each setting, and the error bars show the standard deviation.

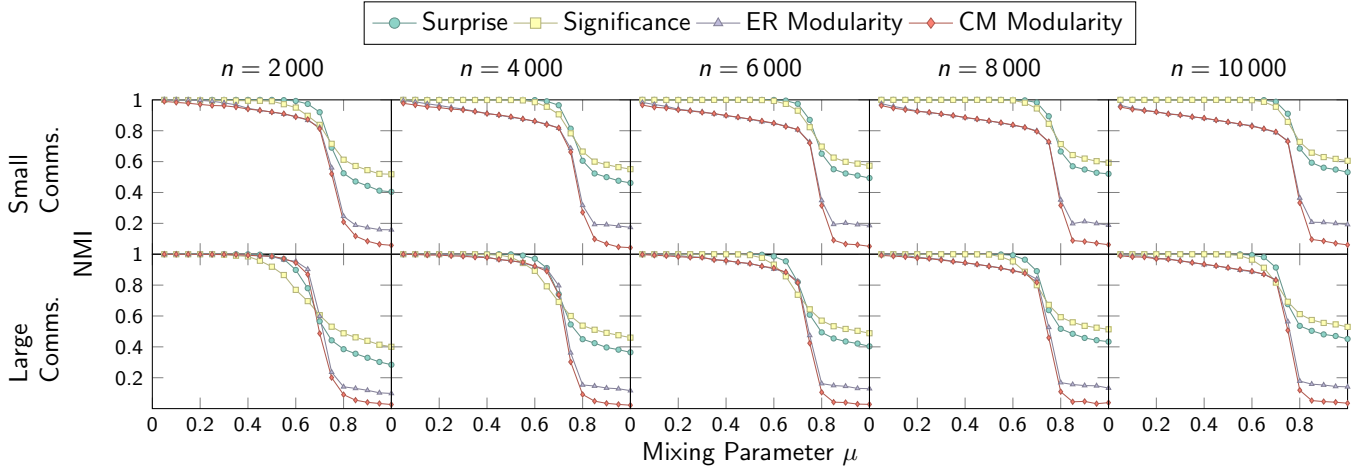


FIG. 5. (Color online) Benchmark results. The first row shows results for “small” communities, which range from 10 to 50, while the second row contains results for “large” communities, ranging from 20 to 100. The community sizes are power-law distributed with exponent 1. We set the average degree $\langle k \rangle = 20$ and the maximum degree is 50, which follows a power-law degree distribution with exponent 2. Both surprise and significance perform very well, especially for relatively large graphs, where ER and CM modularity fail. This difference is more notable for smaller communities, for which both ER and CM modularity have difficulties. This is, in part, due to the well-known resolution limit and in line with our earlier analysis.

sizes are also distributed according to a power law, but with an exponent of 1. The parameter μ again controls the probability of internal links.

In Fig. 4 we show the function values for surprise, significance, and ER modularity. This clearly shows that the inequalities hold over the whole range of mixing parameters. At the same time, they show very similar behavior to each other. Although this could indicate a relatively similar performance, we next show that this is not the case.

In Fig. 5 we show the benchmark results for the four different methods. Surprise and significance performances are very good and clearly much better than both modularity models. Notice that surprise and ER modularity use the same global quantities. However, the use of the KL divergence gives the former a much greater advantage, as expected from Eq. (14).

LFR benchmark graphs have a clearer community structure for larger graphs. The critical mixing parameter at which the inner community density equals the outer community density is roughly $\mu \approx 1 - \frac{n_c}{n}$, so that with growing n this threshold goes to 1. Both surprise and significance start to work better for somewhat larger graphs, consistent with the clearer community structure. This is in a sense the opposite of both ER and CM modularity. Their performance is worse for larger graphs, consistent with our earlier analysis of the limitations of community detection.

VI. CONCLUSION

Community detection is an important topic in the field of complex networks, as it can give us a better understanding of real-world networks. Here we analyzed a recent measure known as surprise. We developed an accurate asymptotic approximation, based on the KL divergence which we use to develop a competitive new algorithm. Applying this algorithm to standard benchmarks, we show its great potential. Significance, another quality measure also based on the KL divergence, performs similar to surprise.

We showed analytically that surprise is more discriminative than modularity with an ER null model. This is mainly due to the use of the KL divergence to quantify the difference between the empirical partition and the null model. The larger the network and the smaller the communities, the better KL methods perform with respect to modularity. Indeed, whereas modularity suffers from the resolution limit, this problem (nearly) does not affect surprise. On the other hand, surprise tends to find substructure in larger communities, arising from random fluctuations, whereas this problems appears less prominent for modularity. In short, modularity tends to work well in the limit of community sizes $n_c \rightarrow \infty$, keeping the number of communities r fixed. Surprise, on the other hand, works well when $r \rightarrow \infty$, keeping the community sizes n_c fixed. Stated differently, modularity tends to underestimate the number of communities, whereas surprise tends to overestimate the number of communities. The question of which method works well in both limits deserves further study.

The slight differences between surprise and significance stem from two things either the one or the other measure ignores. Significance relies on the fraction of edges that are present within a community. It thus implicitly considers missing edges within communities, because this fraction is relative to the total number of possible edges within that community, which surprise does not. Surprise, on the other hand, considers the fraction of total edges that fall within communities. It thus implicitly considers edges that fall between communities, whereas significance does not. Indeed, it should be possible to address these shortcomings by also explicitly examining missing links (for surprise) or links between communities (for significance).

Another shortcoming is that surprise does not depend on the actual distribution of the internal edges among communities. One way to address this issue is to consider edges for all communities separately, by using a multivariate hypergeometric distribution. In that case, we would be interested in the probability to observe m_{cd} edges between communities

c and d as

$$\Pr(X_{cd} = m_{cd}) = \frac{\prod_{cd} \binom{n_c n_d}{m_{cd}}}{\binom{M}{m}}. \quad (30)$$

Again deriving an asymptotic expression, we arrive at

$$\mathcal{S}(\mathcal{V}) = m \sum_{cd} q_{cd} \ln \frac{q_{cd}}{\langle q_{cd} \rangle} = m D(\mathbf{q} \| \langle \mathbf{q} \rangle), \quad (31)$$

where $q_{cd} = \frac{m_{cd}}{m}$ is the fraction of edges between communities c and d and $\langle q_{cd} \rangle$ the expected value.

Interestingly, the extension of surprise in Eq. (30) is identical to a stochastic block model (using an ER null model) [47,48]. However, Karrer and Newman found that this method did not work satisfyingly [47]. This might be because the measure does not focus on communities specifically, but rather on all types of block structures. Hence, there is no reason why a community structure should maximize this likelihood, rather than any other type of block structure. One possible way to address this is to compare our partition to the ideal type we are looking for, rather than maximizing the difference to a random null model. This would be an interesting avenue to consider in future research.

ACKNOWLEDGMENTS

V.A.T. acknowledges funding by the Royal Netherlands Academy of Arts and Sciences (KNAW) through its eHumanities project. J.-C.D. acknowledges support from a grant ‘‘Actions de recherche concertées–big data’’ of the ‘‘Communauté Française de Belgique’’ and from the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The authors would like to thank Vincent Blondel and Arnaud Browet for helpful comments and discussions.

APPENDIX A: ASYMPTOTIC SURPRISE

As stated in the main text, q denotes the fraction of internal edges, so that we can write $m_{\text{int}} = qm$. Since $m = p \binom{n}{2} = pM$, we thus have $m_{\text{int}} = qpM$. Similarly, we can write $M_{\text{int}} = \langle q \rangle M$. Hence, we obtain

$$m = pM, \quad (A1)$$

$$m_{\text{int}} = qpM, \quad (A2)$$

$$M_{\text{int}} = \langle q \rangle M. \quad (A3)$$

Notice that all quantities now depend on M . We only take into account the dominant term, so to obtain

$$\mathcal{S}(\mathcal{V}) \approx - \ln \frac{\binom{\langle q \rangle M}{pqM} \binom{(1-\langle q \rangle)M}{p(1-q)M}}{\binom{M}{pM}}, \quad (A4)$$

which corresponds to the probability of observing exactly m_{int} internal links. The binomial coefficient $\binom{M}{pM}$ is independent of the partition, so we ignore it. We use Stirling’s approximation

of the binomial coefficient, which reads

$$\binom{n}{k} \approx \left(\frac{n}{k} \right)^k. \quad (A5)$$

Hence, for the dominant term, we obtain

$$\mathcal{S}(\mathcal{V}) = - \ln \left(\frac{\langle q \rangle M}{pqM} \right)^{pqM} \left(\frac{(1-\langle q \rangle)M}{p(1-q)M} \right)^{p(1-q)M} \quad (A6)$$

$$= - \ln p^{-pN} \left[\left(\frac{\langle q \rangle}{q} \right)^q \left(\frac{1-\langle q \rangle}{1-q} \right)^{1-q} \right]^{pM}. \quad (A7)$$

The term p^{-pN} is independent of the partition and we ignore it, which yields

$$\mathcal{S}(\mathcal{V}) = -pM \left[q \ln \frac{\langle q \rangle}{q} + (1-q) \ln \frac{1-\langle q \rangle}{1-q} \right]. \quad (A8)$$

Using $pM = m$, we can rewrite this to

$$\mathcal{S}(\mathcal{V}) = m D(q \| \langle q \rangle), \quad (A9)$$

where $D(x \| y)$ is the KL divergence [31],

$$D(x \| y) = x \ln \frac{x}{y} + (1-x) \ln \frac{1-x}{1-y}, \quad (A10)$$

which can be interpreted as the distance between the two probability distributions q and $\langle q \rangle$.

APPENDIX B: SIGNIFICANCE

We can calculate the approximate difference of moving an edge from one community to another. Assume we move an edge from community r to community s . The change in the density will be approximately $p_r - \frac{1}{n_r^2}$ and $p_s + \frac{1}{n_s^2}$, respectively. The corresponding difference in significance will be approximately

$$\begin{aligned} \mathcal{Z}(\mathcal{V}') - \mathcal{Z}(\mathcal{V}) &= n_s^2 \left[D\left(p_s + \frac{1}{n_s^2} \| p\right) - D(p_s \| p) \right] \\ &\quad + n_r^2 \left[D\left(p_r - \frac{1}{n_r^2} \| p\right) - D(p_r \| p) \right] \end{aligned} \quad (B1)$$

$$\approx \frac{\partial}{\partial p_s} D(p_s \| p) - \frac{\partial}{\partial p_r} D(p_r \| p) \quad (B2)$$

$$= \ln \frac{p_s}{1-p_s} \frac{1-p_r}{p_r} = \Delta \mathcal{Z}. \quad (B3)$$

This quantity is particularly straightforward (the logarithmic odds ratio), and if $p_r > p_s$ the difference will be negative, and if $p_r < p_s$ this quantity will be positive. Moving edges from a denser community to a less dense community decreases the significance. In other words, making two densities more equal decreases the significance. Repeating these steps, we should expect to find the lowest significance when the communities are of equal density.

Alternatively, by convexity of the KL divergence, we obtain for significance that

$$\mathcal{Z}(\mathcal{V}) \geq \left[\sum_c \binom{n_c}{2} \right] D \left[\sum_c \frac{\binom{n_c}{2}}{\sum_c \binom{n_c}{2}} p_c \| p \right]. \quad (B4)$$

Realizing that $m_c = p_c \binom{n_c}{2}$, we see that

$$\sum_c \frac{\binom{n_c}{2}}{\sum_c \binom{n_c}{2}} p_c = \frac{m_{\text{int}}}{M_{\text{int}}} = p \frac{q}{\langle q \rangle}. \quad (\text{B5})$$

Notice that this can be interpreted as an average internal density $\langle p_c \rangle$ as stated in the main text. Using this we arrive at

$$\mathcal{Z}(\mathcal{V}) \geq M_{\text{int}} D\left(p \frac{q}{\langle q \rangle} \parallel p\right). \quad (\text{B6})$$

Hence, the significance of a partition with different community densities p_c is generally larger than a partition where all communities have the same average density $p_c = \frac{m_{\text{int}}}{M_{\text{int}}}$. Notice that $p \frac{q}{\langle q \rangle}$ should be bounded by 1 so that $q > \langle q \rangle > p \langle q \rangle$ in general.

This points to a bound such that $\mathcal{Z}(\mathcal{V}) \geq \mathcal{S}(\mathcal{V})$ when $\langle q \rangle < p$ in the following way. Define $\lambda = \frac{\langle q \rangle}{p}$ so that $\lambda < 1$ if $\langle q \rangle < p$. Again applying convexity, we obtain

$$\mathcal{Z}(\mathcal{V}) \geq M_{\text{int}} D\left(\frac{pq}{\langle q \rangle} \parallel p\right) \quad (\text{B7})$$

$$= \frac{M_{\text{int}}}{\lambda} \left[\lambda D\left(\frac{pq}{\langle q \rangle} \parallel p\right) + (1 - \lambda) D(0 \parallel 0) \right] \quad (\text{B8})$$

$$\geq \frac{M_{\text{int}}}{\lambda} D\left(\lambda \frac{pq}{\langle q \rangle} \parallel \lambda p\right) \quad (\text{B9})$$

$$= m_{\text{int}} D(q \parallel \langle q \rangle) = \mathcal{S}(\mathcal{V}). \quad (\text{B10})$$

If there are fewer communities (i.e., if $p > \langle q \rangle$), the relationship is not entirely clear, but there are cases for which surprise may be larger than significance. For example, if we assume an equisized equidense partition with r communities, then $q = \frac{p_c \langle q \rangle}{p}$ and $\langle q \rangle = \frac{1}{r}$, and the difference can be written as

$$\begin{aligned} \mathcal{S}(\mathcal{V}) - \mathcal{Z}(\mathcal{V}) &= m(1 - q) \ln \frac{1 - q}{1 - \langle q \rangle} \\ &\quad - M_{\text{int}}(1 - p_c) \ln \frac{1 - p_c}{1 - p}. \end{aligned} \quad (\text{B11})$$

Indeed, if $\langle q \rangle > p$, then $\mathcal{S}(\mathcal{V}) > \mathcal{Z}(\mathcal{V})$ for equisized equidense partitions. Keep in mind, though, that an equisized equally dense partition will have a lower significance in general, so that this does not hold for $\langle q \rangle > p$ in general.

-
- [1] R. Albert and A.-L. Barabási, *Rev. Mod. Phys.* **74**, 47 (2002).
[2] S. Dorogovtsev, *Lectures on Complex Networks* (Oxford University Press, Oxford, UK, 2010).
[3] M. Newman, *Networks: An Introduction* (Oxford University Press, USA, 2010).
[4] A.-L. Barabási and R. Albert, *Science* **286**, 509 (1999).
[5] D. J. Watts and S. H. Strogatz, *Nature (London)* **393**, 440 (1998).
[6] S. Fortunato, *Phys. Rep.* **486**, 75 (2010).
[7] M. Girvan and M. E. J. Newman, *Proc. Natl. Acad. Sci. USA* **99**, 7821 (2002).
[8] P. M. Gleiser and L. Danon, *Adv. Complex Syst.* **06**, 565 (2003).
[9] G. Palla and I. Dere, *Nature (London)* **435**, 814 (2005).
[10] R. Guimerá, S. Mossa, A. Turtleschi, and L. A. N. Amaral, *Proc. Natl. Acad. Sci. USA* **102**, 7794 (2005).
[11] J. M. Olesen, J. Bascompte, Y. L. Dupont, and P. Jordano, *Proc. Natl. Acad. Sci. USA* **104**, 19891 (2007).
[12] Y. Lupu and V. Traag, *J. Conflict Resolut.* **57**, 1011 (2013).
[13] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, *J. Stat. Mech. Theory Exp.* (2005) P09008.
[14] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, *PLoS ONE* **6**, e18961 (2011).
[15] R. Aldecoa and I. Marín, *Sci. Rep.* **3**, 1060 (2013).
[16] M. E. J. Newman and M. Girvan, *Phys. Rev. E* **69**, 026113 (2004).
[17] J. Reichardt and S. Bornholdt, *Phys. Rev. E* **74**, 016110 (2006).
[18] P. Ronhovde and Z. Nussinov, *Phys. Rev. E* **81**, 046114 (2010).
[19] A. Lancichinetti and S. Fortunato, *Phys. Rev. E* **80**, 056117 (2009).
[20] B. H. Good, Y.-A. de Montjoye, and A. Clauset, *Phys. Rev. E* **81**, 046106 (2010).
[21] R. Aldecoa and I. Marín, *Sci. Rep.* **3**, 2216 (2013).
[22] V. A. Traag, G. Krings, and P. Van Dooren, *Sci. Rep.* **3**, 2930 (2013).
[23] S. Fortunato and M. Barthélemy, *Proc. Natl. Acad. Sci. USA* **104**, 36 (2007).
[24] J. M. Kumpula, J. Saramäki, K. Kaski, and J. Kertész, *Eur. Phys. J. B* **56**, 41 (2007).
[25] A. Lancichinetti and S. Fortunato, *Phys. Rev. E* **84**, 066122 (2011).
[26] J. P. Bagrow, *Phys. Rev. E* **85**, 066118 (2012).
[27] J. Xiang and K. Hu, *Phys. A: Stat. Mech. Appl.* **391**, 4995 (2012).
[28] A. Kehagias, and L. Pitsoulis, *Eur. Phys. J. B* **86**, 330 (2013).
[29] V. A. Traag, P. Van Dooren, and Y. Nesterov, *Phys. Rev. E* **84**, 016114 (2011).
[30] R. Aldecoa and I. Marín, *PLoS One* **6**, e24195 (2011).
[31] S. Kullback and R. A. Leibler, *Ann. Math. Stat.* **22**, 79 (1951).
[32] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborova, *Phys. Rev. Lett.* **107**, 065701 (2011).
[33] V. Arnau, S. Mars, and I. Marín, *Bioinformatics* **21**, 364 (2005).
[34] R. Aldecoa and I. Marín, *Bioinformatics* **30**, 1041 (2014).
[35] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, *J. Stat. Mech. Theory Exp.* (2008) P10008.
[36] <https://github.com/vtraag/louvain-igraph>
[37] <https://pypi.python.org/pypi/louvain/>
[38] M. A. Porter, J.-P. Onnela, and P. J. Mucha, *Notices of the American Mathematical Society* **56**, 1082 (2009); **56**, 1164 (2009);
[39] M. Molloy and B. Reed, *Random Struct. Algorithms* **6**, 161 (1995).
[40] G. Krings and V. D. Blondel, [arXiv:1103.5569](https://arxiv.org/abs/1103.5569) (2011).
[41] V. A. Traag, *Algorithms and Dynamical Models for Communities and Reputation in Social Networks*, Springer Theses (Springer, Heidelberg, 2014).

- [42] M. T. Schaub, J.-C. Delvenne, S. N. Yaliraki, and M. Barahona, *PloS One* **7**, e32210 (2012).
- [43] F. Radicchi, *Europhys. Lett.* **106**, 38001 (2014).
- [44] R. R. Nadakuditi and M. E. J. Newman, *Phys. Rev. Lett.* **108**, 188701 (2012).
- [45] F. Radicchi, *Phys. Rev. E* **88**, 010801 (2013).
- [46] A. Lancichinetti, S. Fortunato, and F. Radicchi, *Phys. Rev. E* **78**, 046110 (2008).
- [47] B. Karrer and M. E. J. Newman, *Phys. Rev. E* **83**, 016107 (2011).
- [48] P. J. Bickel and A. Chen, *Proc. Natl. Acad. Sci. USA* **106**, 21068 (2009).