Universality of the stochastic block model

Jean-Gabriel Young,[1,2,*] Guillaume St-Onge,[1,2] Patrick Desrosiers,[1,2,3] and Louis J. Dubé[1,2,†]

[1]*Département de physique, de génie physique et d'optique, Université Laval, Québec (Québec), Canada G1V 0A6*
[2]*Centre interdisciplinaire de modélisation mathématique de l'Université Laval, Québec (QC), Canada G1V 0A6*
[3]*Centre de recherche de CERVO, Québec (QC), Canada G1J 2G3*

Mesoscopic pattern extraction (MPE) is the problem of finding a partition of the nodes of a complex network that maximizes some objective function. Many well-known network inference problems fall in this category, including, for instance, community detection, core-periphery identification, and imperfect graph coloring. In this paper, we show that the most popular algorithms designed to solve MPE problems can in fact be understood as special cases of the maximum likelihood formulation of the stochastic block model (SBM) or one of its direct generalizations. These equivalence relations show that the SBM is nearly universal with respect to MPE problems.

I. INTRODUCTION

Whether it is called community detection, graphical inference, spectral embedding, unsupervised learning, bisection, or graph coloring, the idea of summarizing the structure of a complex system by grouping its elements in blocks is a popular one, discovered time and time again in different areas of science [1]. As such, there are now a plethora of algorithms and techniques—developed essentially in parallel—that provide good solutions to this ubiquitous problem [2]. In the past few years, a great deal of work has been done toward unifying and contrasting these approaches, building bridges across cultural divides [1,3]. This has been fruitful work thus far, for—sometimes surprising—equivalences between drastically different methods have turned up in the process, e.g., between modularity and the maximum likelihood formulation of the degree-corrected stochastic block models (SBM) [4–7], various spectral methods [8], normalized-cut [9], random-walks [10], and nonnegative matrix factorization [11]. These results invite the question: Is there a deeper reason for the correspondences, or are they simply mathematical coincidences?

The purpose of this paper is to show that equivalences arise because most of these *mesoscopic pattern extraction* (MPE) methods are actually the maximum likelihood formulation of the SBM in disguise (and a generalization of its degree-corrected version [12,13]). By MPE problems, we mean any problem where one is asked to find a partition of the network that maximizes some implicit or explicit score, encoded via an objective function.

Our results rest on the concepts of *equivalence* and *specialization* of the objective functions: Two objective functions are equivalent when they order any pair of partitions the same way (i.e., they implement the same notion of optimality), and specialization refers to the idea of limiting the expressiveness

of an objective function by fixing some of its parameters (see Sec. II). With these two operations, we delineate a hierarchy that crystallizes the idea of the SBM as a general MPE tool: Through specialization of its likelihood, it can be tailored to find patterns such as assortative and disassortative communities [5], bipartite structures [14], or core-periphery splits [15] (in Sec. III). Importantly, we show that these specialized likelihoods are *exactly* equivalent to the objective functions implemented by MPE algorithms such as modularity maximization, balanced cut, core-periphery search, etc. Our framework therefore offers principled methods to determine any arbitrary parameters that might arise in otherwise *ad hoc* modularities [5], but also suggests statistical techniques to carry out principled inference, in the spirit of Refs. [4,16] (see Sec. IV).

II. MESOSCOPIC STRUCTURES AND OPTIMIZATION

The mesoscopic pattern extraction (MPE) problem is usually stated as follows. We are given an extremely large complex network, generated by some random hidden process. Its overall organization is impossible to grasp, because its structure is much too detailed. Our goal with MPE is to reduce this complexity, by subsuming nodes in larger coherent units, using the structure of the network as our only input (and possibly additional metadata [17,18]). Sometimes, the hope is to reveal functional components and hints about assembly mechanisms, while at other times it is only a matter of making the dataset more manageable, or interpolating from what is known [2,3,19–24]. There is, however, a common theme: MPE algorithms take a complex network as their input, and produce as output a *partition* $\mathcal{B} = \{B_1, ..., B_q\}$ of the n nodes in q blocks $B_1, ..., B_q$, assigning precisely one block to each node. Despite these commonalities, the definition of what is a suitable partition will of course depend on the MPE problem at hand; there is thus a wealth of MPE *algorithms*, reflecting the wealth of MPE *problems* (see Fig. 1).

----
[*]jean-gabriel.young.1@ulaval.ca
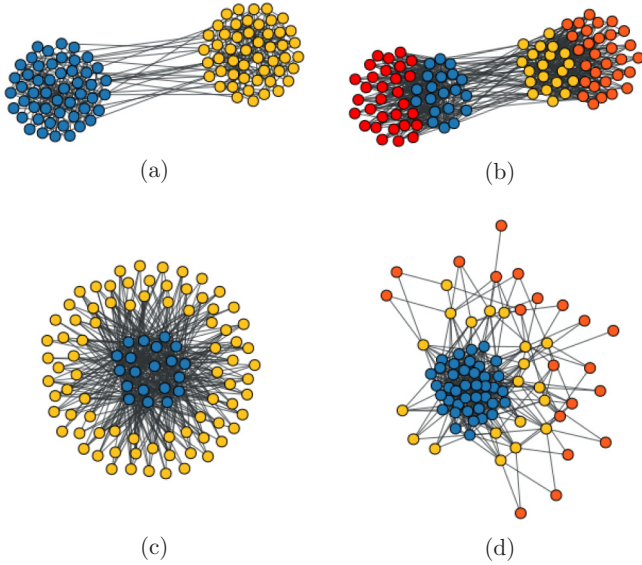[†]ljd@phy.ulaval.ca

FIG. 1. Four examples of mesoscopic pattern extraction problems on artificial networks. (a) Community detection [22,25], (b) community detection with further structure, (c) the identification of a simple core-periphery split [15,26], (d) identification of a core with a structured periphery. The targeted patterns are identified with colors.
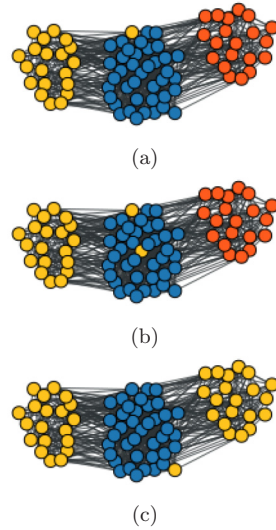

FIG. 2. Two-part algorithms in practice. We apply different combinations of maximizers and objective functions to a simple network with a clear block structure, generated using the SBM. Different MPE algorithms reveal different mesoscopic structures, but changing the objective function has the largest impact. (a, b) Same objective functions (modularity, see Sec. III B 3) and different maximizers [(a) spectral, (b) greedy]. (b, c) Same maximizer (greedy) with two different objective functions [(b) modularity, (c) core-periphery see Sec. III A 3]. The greedy maximizer is adapted from the Kernighan-Lin algorithm [30], in the spirit of Ref. [12], and the spectral maximizer relies on the embedding of the $q-1$ leading eigenvectors of the modularity matrix [25,31] in $\mathbb{R}^{q-1}$, followed by a clustering step, here implemented using a Gaussian mixture [32].

To establish parallels between algorithms of diverse natures, we must first clearly answer: What is the essence of an MPE algorithm? And what do we mean, when we say that two algorithms are *equivalent*? The answers to these questions are not trivial, and crucial to the interpretation of the results of Secs. III A–III B. Our goal with the next four subsections is therefore to clarify these issues.

A. Anatomy of a black box

There are essentially two possible ways to formulate our answers, depending on how we think of MPE algorithms.

First, we may take the empirical point of view and declare that the essence of algorithms is their *action*, independent of their inner workings. According to this point of view, equivalence is functional and context dependent: If two algorithms give the same result on a series of networks $G_1, G_2, ...G_k$, then the algorithms are equivalent with respect to these k networks. This allows us to treat algorithms as black boxes: Network in, partition out. It is certainly an appealing approach, because it may be used to compare algorithms of widely different natures—say a genetic algorithm with an evolved objective function and a label propagation method. Functional equivalence, however, has the drawback that it depends on the context, which makes it hard to draw definitive conclusions about algorithms. Furthermore, it may identify somewhat artificial parallels, because it is insensitive to the *origin* of the equivalences.

A second point of view is centered on the *definitions* of MPE algorithms rather than their action therefore appears necessary. Due to the diversity of existing MPE algorithms, this point of view will only be useful if we are able to first express MPE algorithms in some canonical form that can be readily analyzed. One possibility is a two–part model

expressed as the coupling of (i) an objective function that induces a total ordering of the partitions, and (ii) a maximizer that can find a—potentially local—optimum of the objective function (see Fig. 2). This two-part model captures the two important mechanisms that any MPE algorithm must possess. On the one hand, the objective function captures the notion of quality of the partition and, consequently, tells the algorithm when to stop, and what partition to prefer whenever it has a choice. On the other hand, the maximizer provides a mean of moving in the solution space, and of pinpointing the best partitions, as per the above criterion. These mechanisms might be interwoven or hidden—we will touch on the subject shortly—but the separation holds quite generally.

With the two-part model in place, equivalence takes on a crisp and clear meaning. Two algorithms are either partially equivalent—same objective **or** same maximizer—or completely equivalent—same objective **and** same maximizer. In the present paper, we will focus on partial equivalence, essentially ignoring the maximizers. This choice is motivated by the observations that (a) maximizers are, by necessity [27], only efficient heuristics designed to find "good enough" optima in the rugged landscape of partitions [1,28,29] (b) the no free lunch theorem implies that different objective functions and different inputs are associated with different optimal maximizers [29].

Hereafter, by *equivalence*, we will therefore refer to the equivalence of the objective functions used.

B. A glance under the hood

In the simplest—and quite common—case, the separation in two parts is explicit. For example, modularity–based methods famously attempt to maximize the modularity function over the set of all partitions of a network [22]. If there are many modularity optimization *algorithms*, it is because there are many different mechanisms that can propose and refine partitions to find the optima of the modularity, e.g., the iterative spectral method of Ref. [31], the fast unfolding method of Ref. [21], or the message-passing algorithm of Ref. [4]. The two–part algorithmic model is an exact description of these methods because they are framed in the language of objective functions.

Importantly, the two-part algorithmic model also holds in many cases where the emphasis is shifted away from an explicit objective function and maximizer dichotomy. Consider as an example the classical label propagation algorithm of Ref. [33]. This algorithm moves through the partition space by first assigning temporary labels (blocks) to nodes, and then repeatedly updating the labels with a majority rule (a node takes the label worn by the majority of its neighbors). Optimality is thus not defined for arbitrary pairs of partitions; it is instead expressed as a dynamical, initial condition dependent concept. But a description in two parts can still be given, provided that we do some translation work: The label propagation mechanism can be thought of as a maximizer, which naturally leads to *partition flow* as a notion of optimality. A partition $\mathcal{B}_1$ is better than $\mathcal{B}_2$ if the algorithm goes from $\mathcal{B}_2$ to $\mathcal{B}_1$ when it updates labels based on the majority rule. With this definition, the best partitions are those that are stable against majority updates, and they are found via the propagation of labels. One can construct an objective function with these orderings, and therefore a two–part algorithm indistinguishable from the original [34–36].

C. General graphical objective function

Having established that a separation of algorithms into an objective function and a maximizing mechanism is often possible, let us turn to the functions themselves.

The outcome of pairwise interactions determines the structure of a complex network. A *general* objective function devised to uncover the mesoscopic patterns of a network therefore ought to include all these interactions in its calculation, at the very least. If it does no more than that, then the function can be called *graphical*, in the sense that no high-order terms are considered (i.e., there are no direct dependency on triplet of nodes, etc.). From this point onward, we will focus on graphical objective functions alone; the remainder of this paper is a testament to the generality of such a "limited" approach.

The definition of graphical objective function begins with the definition of its basic elements: Scores associated with each pair of nodes. For the sake of generality, we will define these scores as real-valued functions, with essential dependencies on the partition $\mathcal{B} = \{B_1, \ldots, B_q\}$ under consideration, on the structure of the network as encoded by the $n \times n$ adjacency matrix A, and on an additional $n \times n$ side-information matrix λ that contains any pairwise information not directly captured by A. Let us therefore write the score associated with the pair of nodes $(i, j) \in [n] \times [n]$ (we use the integers $[k] = \{1, ..., k\}$ to denote the nodes) as

$$f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j), \qquad (1)$$

where $\sigma_i \in [q]$ is the index of the block of node i, i.e., $\sigma_i = r$ if and only if $i \in B_r$. We then express the aggregate of these local scores as

$$H(A, \lambda, \sigma; f) = \sum_{i,j: 1 \leqslant i \leqslant j \leqslant n} f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j), \qquad (2)$$

yielding a global objective function based on pairwise scores. This form highlights the close parallel that exists between graphical objective functions and Edward-Anderson Hamiltonians [28], explicitly harnessed in a number of specific cases in Refs. [37–41], for example. The choice of a sum is, otherwise, for mere convenience; a product aggregate could have been equivalently implemented by taking $f \mapsto \log f$ and $H \mapsto e^H$.

D. Equivalence and hierarchy under specialization

The last piece of the theoretical framework is a clear notion of connections among functions. We use two concepts to establish these connections: Equivalence and specialization.

1. Equivalence

We say that two objective functions are equivalent if they induce the same total ordering of partitions, regardless of their inputs. This definition captures the correct notion of equivalence, because it is clear that two equivalent objective functions—by this standard—will yield two MPE indistinguishable algorithms when they are paired with the same maximizer. As it stands, however, this notion of equivalence is not easy to handle mathematically. We therefore resort to a second, stronger, criterion that leads to a more direct comparison procedure. It is obvious that if

$$H(A, \lambda, \sigma; f) < H(A, \lambda, \sigma'; f)$$
$$\Downarrow$$
$$g \circ H(A, \lambda, \sigma; f) < g \circ H(A, \lambda, \sigma'; f), \qquad (3)$$

for some strictly increasing function g, then H and $g \circ H$ are equivalent according to the first definition. While this second version is more restrictive, it reduces the comparison of objective functions to the identification of the transformation g—an often straightforward process.

As we will see in Sec. III, in practice, an even stronger criterion that limits g to a particular subset of all *linear* transformation will often suffice to establish many equivalence relations. Namely, whenever a pairwise score functions $f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j)$ can be split as

$$f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) = f_1(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) + f_2(a_{ij}, \lambda_{ij}), \quad (4)$$

where f_2 does not depend on the partition, we will be able to rewrite the global objective function as

$$H(A, \lambda, \sigma; f) = \sum_{i \leqslant j} f_1(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) + \sum_{i \leqslant j} f_2(a_{ij}, \lambda_{ij})$$
$$\sim \sum_{i \leqslant j} f_1(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j)$$
$$= H'(A, \lambda, \sigma; f), \qquad (5)$$

where "∼" denotes equivalence, and "$i \leqslant j$" is a shorthand for the more precise statement "$i, j : 1 \leqslant i \leqslant j \leqslant n$." The equivalence holds because the additive terms are independent from $\boldsymbol{\sigma}$ and therefore do not affect the ordering. Thus, equivalence will often follow from a simple linear transformation of the form $g \circ H = H - \sum_{i \leqslant j} f_2(a_{ij}, \lambda_{ij})$.

2. Specialization

With specialization, we aim to capture the idea that an objective function can be less expressive than its *parent* function, i.e., that it is possible to fix some parameters of a function (the parent) to obtain a "simpler" version of the function [42]. It is more straightforward to define specialization at the level of pairwise score, and so we will say informally that a pairwise score function f_S is a specialization of f if f_S is constructed by fixing some of the free parameters of f, in a way that alters the ranking of partitions, for some inputs. Furthermore, we will say that the objective function $H'(\boldsymbol{A}, \boldsymbol{\lambda}, \boldsymbol{\sigma}; f_S)$ is a specialization of the objective function $H(\boldsymbol{A}, \boldsymbol{\lambda}, \boldsymbol{\sigma}; f)$ when f_S is a specialization of f.

In the context of MPE, if f_S is derived from f and there exists at least one pair of nodes (i, j) such that

$$f_S(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) = f_S(a_{ij}, \lambda_{ij}, \sigma_i', \sigma_j')$$

and

$$f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) \neq f(a_{ij}, \lambda_{ij}, \sigma_i', \sigma_j'), \quad (6)$$

where $\boldsymbol{\sigma} \neq \boldsymbol{\sigma}'$, then f_S is a specialization of f (and similarly for the resulting H' and H).

Specialization is, in a sense, a one-way operation, because it involves reducing the complexity of a function. In Eq. (6), f could act as f_S but not the other way around, because f_S is derived from f by specialization. Thus, specialization induces a hierarchy, with the most general functions at the top, and the most specialized ones at the bottom. This is the hierarchy that we propose to delineate in the next sections.

III. OBJECTIVE FUNCTION HIERARCHY UNDER SPECIALIZATION

Recall that our claim is essentially the following: The objective functions of many mesoscopic pattern extraction algorithms are, in fact, special cases of the maximum likelihood formulation of the SBM. Sections III A and III B are devoted to showing how this comes about. We begin with the methods that *do not* account for any side information λ_{ij}, in Sec. III A. We show that they can be understood as specialization of the maximum likelihood formulation of the classical SBM [43]. We then move on to general MPE methods, in Sec. III B, by adding a side-information dependency to the score functions. Again, we show that these methods can be seen as specializations of a generalized SBM, close in spirit to the degree-corrected SBM of Ref. [12]. This part of the hierarchy sits above the methods of Sec. III A, since the generalized SBM contains the classical SBM as a special case. We summarize the relations between the various methods in Fig. 3, and we show in Fig. 4 that they can be used to extract various patterns from a same real network.

A. Partial hierarchy (no side information)

1. Stochastic block model

Our starting point is the stochastic block model (SBM). It is not an MPE algorithm *per se*, but rather a random network model, amenable to statistical inference. It prescribes a likelihood for the network G, parametrized by a latent partition $\mathcal{B}$ of its nodes. The SBM becomes a MPE algorithm once this likelihood is used to infer the hidden partition $\mathcal{B}$ of G. Although there are many ways of harnessing the likelihood to extract the mesoscopic patterns encoded by $\mathcal{B}$, we will only focus on *likelihood maximization*, because it directly fits within the two–part model of MPE algorithms defined in Sec. II A; the likelihood is the objective function and the maximizer does not matter.

Given a network and a partition of the nodes in blocks associated with the vector $\boldsymbol{\sigma}$, the classical SBM [44] prescribes that the number of edges between nodes (i, j) should be drawn from a Poisson distribution of mean $\omega_{\sigma_i \sigma_j}$. All edges are assumed to be independent, such that the likelihood of the complete graph is given by

$$\mathbb{P}(G|\mathcal{B}, \boldsymbol{\omega}) = \prod_{i \leqslant j} \frac{(\omega_{\sigma_i \sigma_j})^{a_{ij}}}{a_{ij}!} e^{-\omega_{\sigma_i \sigma_j}}. \quad (7)$$

It is parametrized by the $q \times q$ matrix $\boldsymbol{\omega}$ and the partition $\mathcal{B}$ (or equivalently by the block assignments $\boldsymbol{\sigma}$). The standard inference procedure calls for the estimation of both, $\boldsymbol{\omega}$ and $\mathcal{B}$, usually through alternated learning of the two sets of parameters (via the expectation–maximization algorithm [39]). However, we will focus on the estimation of $\mathcal{B}$ alone, treating the parameters $\boldsymbol{\omega}$ as "control buttons." The freedom to *impose* parameters $\boldsymbol{\omega}$ on the network will ultimately allow us to draw relations with other MPE algorithms.

To extract $\mathcal{B}^*(G)$—the "true" partition of the nodes—from the network, we maximize the likelihood of the SBM with respect to the partition (see also Sec. IV). Since the logarithm is a strictly increasing function of its argument, we may equivalently maximize the log-likelihood

$$\log \mathbb{P}(G|\mathcal{B}, \boldsymbol{\omega}) = \sum_{i \leqslant j} [a_{ij} \log \omega_{\sigma_i \sigma_j} - \omega_{\sigma_i \sigma_j} - \log a_{ij}!]. \quad (8)$$

This is a first (trivial) example of the concept of equivalence of Sec. II D. It becomes evident upon inspection of Eq. (8) that the log-likelihood is, in fact, a graphical objective function of the general form appearing in Eq. (2), associated with the pairwise score function

$$f_{\text{SBM}}(a_{ij}, \sigma_i, \sigma_j) \sim a_{ij} \log \omega_{\sigma_i \sigma_j} - \omega_{\sigma_i \sigma_j}. \quad (9)$$

Thus, any objective function that can be written as a special case of Eq. (9) will be a specialization of the maximum likelihood formulation of the SBM.

2. General modular graph model

One such (explicit) specialization is the general modular graph model (GMGM) [45,46]. Like its general counterpart, the GMGM is a generative model for networks that supposes a latent partition of the nodes in blocks. The crucial difference is that the connection matrices $\boldsymbol{\omega}$ of the GMGM are much more structured than that of the SBM.
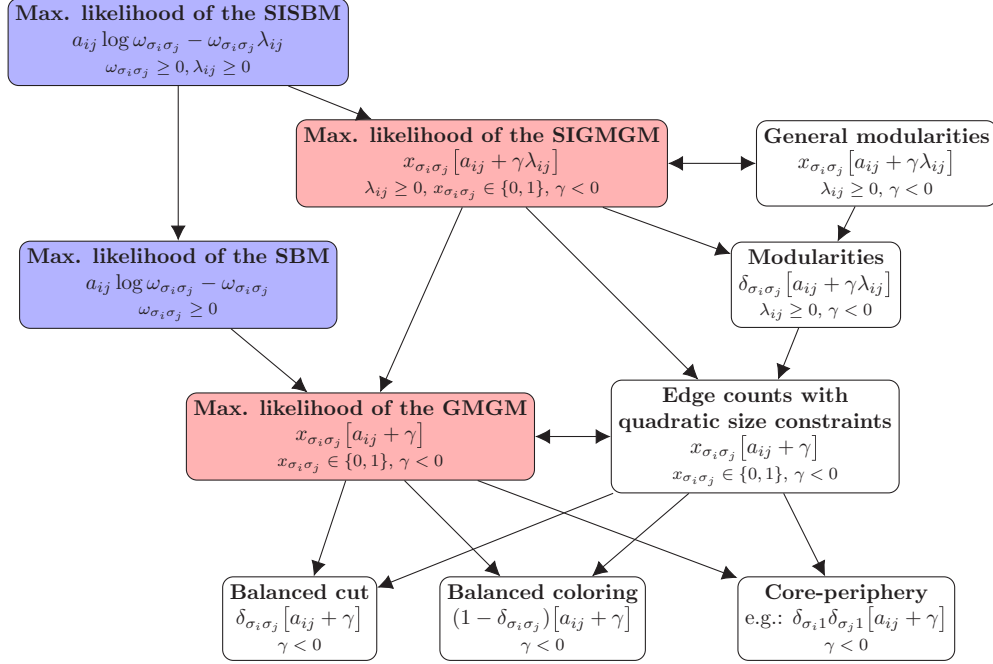
FIG. 3. Partial hierarchy of objective functions. The pairwise score function of MPE methods are shown with the range of parameters below. Arrows denote specialization; doubled–sided arrows denote equivalence. Only the most direct arrows are drawn for the sake of clarity; specialization and equivalence are transitive operations. The abbreviations are: stochastic block model (SBM), with side information (SISBM); and general modular graph modular model (GMGM), with side information (SIGMGM). Functions derived from the perspective of statistical inference are colored in blue (general classification) and red (binary classification).

Pairs of blocks are assigned one of two types, say a and b, and this information is encoded in a $q \times q$ binary (and symmetric) matrix X. If a pair of blocks (B_r, B_s) is of type a, then we set $x_{rs} = 1$. Contrariwise, we set $x_{rs} = 0$ if the pair (B_r, B_s) is of type b. Pairs of blocks of type a are then *all* associated with a connectivity $\omega_{rs} = \omega_a$, while pairs of type b are associated with a connectivity $\omega_{rs} = \omega_b$, where we take $\omega_b < \omega_a$ without loss of generality [47]. Every connection matrix of the GMGM can therefore be written as

$$\boldsymbol{\omega} = \omega_b \mathbf{1}\mathbf{1}^\intercal + (\omega_a - \omega_b)X, \tag{10}$$

where $\mathbf{1}$ is column vector of ones.

The principal motivation for using the simplified matrices of Eq. (10) is that the mathematical treatment of the model becomes simpler at the expense of a moderately reduced flexibility [45,46]. In particular, the two identities (used in similar derivations in Refs. [5,48])

$$\omega_{rs} = \omega_b + x_{rs}(\omega_a - \omega_b), \tag{11a}$$

$$\log \omega_{rs} = \log \omega_b + x_{rs}(\log \omega_a - \log \omega_b), \tag{11b}$$

lead to a likelihood and a log-likelihood analogous to—but much simpler than—the ones appearing in Eqs. (7) and (8). They are associated with the score function

$$
\begin{aligned}
f_{\text{GMGM}}&(a_{ij}, \sigma_i, \sigma_j) \\
&= a_{ij}[\log \omega_b + x_{\sigma_i \sigma_j}(\log \omega_a - \log \omega_b)] \\
&\quad - [\omega_b + x_{\sigma_i \sigma_j}(\omega_a - \omega_b)] \\
&\sim x_{\sigma_i \sigma_j}[a_{ij}(\log \omega_a - \log \omega_b) - (\omega_a - \omega_b)] \\
&\sim x_{\sigma_i \sigma_j}[a_{ij} + \gamma], \tag{12}
\end{aligned}
$$

where $\gamma = (\omega_b - \omega_a)/(\log \omega_a - \log \omega_b) \in (-\infty, 0]$, a drastic simplification when contrasted with Eq. (9). In essence, the GMGM only cares about the type of a block pair. If a pair of nodes (i, j) is associated with a block pair of type a, then the global objective function is increased by a factor of $a_{ij} + \gamma$ (greater when $a_{ij} = 1$ than when $a_{ij} = 0$). If a pair of nodes (i, j) is associated with a block pair of type b, then it only has an indirect impact, by omission.

3. Combinatorial objective functions

The GMGM specialization of the SBM is interesting not only for its mathematical simplicity but also because its pairwise score function can be obtained from a completely different perspective. As we have seen, the essence of the GMGM is its binary classification of block pairs; it turns out that there are countless examples of MPE objective functions that rely on a similar dichotomy (see, for instance, Ref. [2] for a recent review). Their design is essentially the following. Some subsets or intersections of nodes are first identified as special. The MPE objective function is then designed as to maximize the number of edges *within* or *to* these subsets. Finally, because there are often trivial maxima (e.g., place all nodes in the special subset), some constraints are added to avoid trivial optima.

The general mathematical construction closely parallels that of the GMGM. First, we designate special pairs of blocks, and encode the result in a binary matrix X. We then assume, without loss of generality, that the number of edges within these blocks should be *maximized* by the target partition $\mathcal{B}^*$.
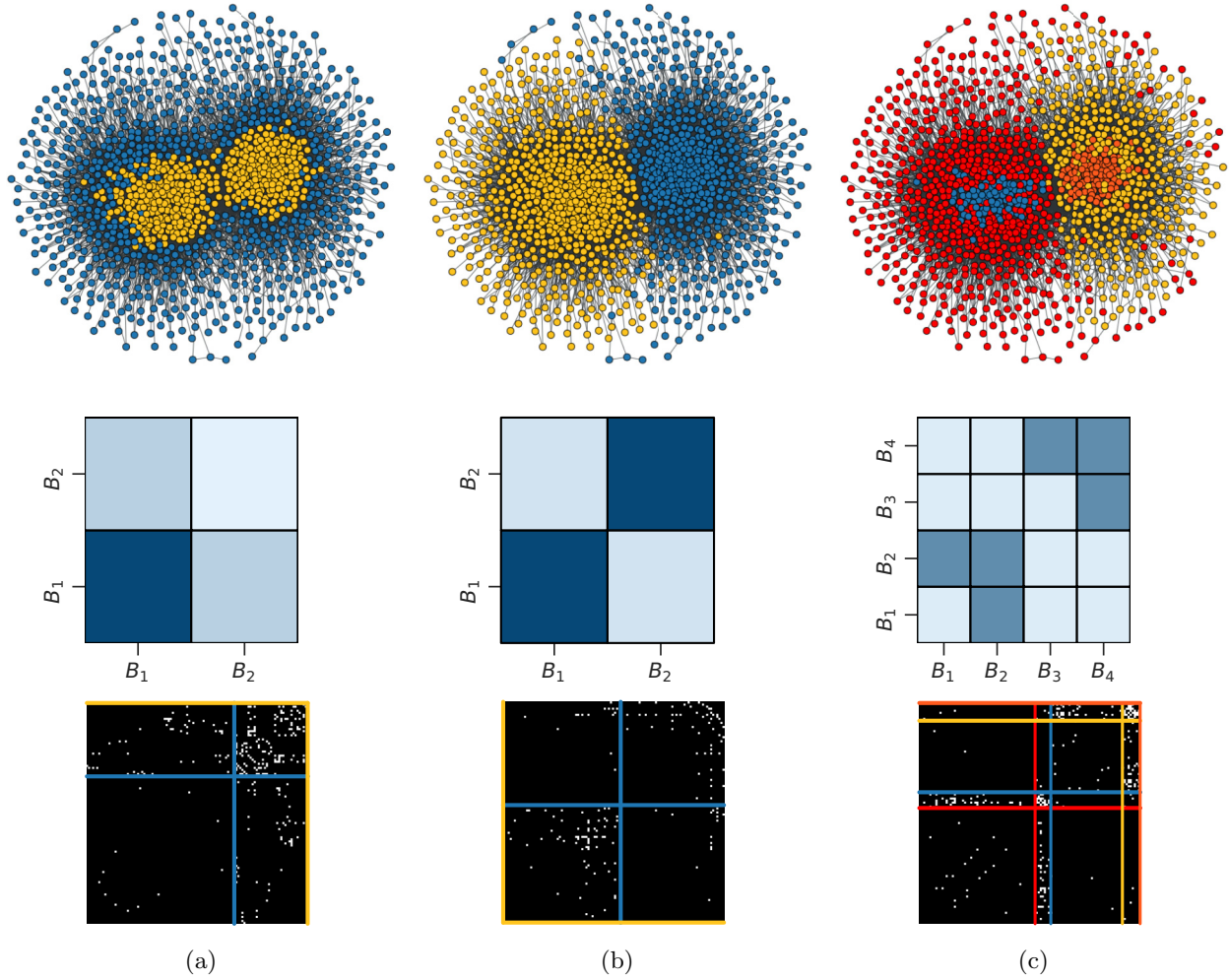
FIG. 4. Mesoscopic patterns learned from and imposed on a real complex network. All results are obtained on the `polblog` dataset, a directed network of hyperlinks between weblogs on U.S. politics, recorded shortly after the 2004 presidential election. There are a total of 1 222 nodes (weblogs) and 16 714 edges. We use an undirected, self-loop free version of the network. All subfigures show (top) the network with nodes colored according to the identified partition, (center) a cartoon of the matrix $\boldsymbol{\omega}$ *imposed* for the equivalent SBM [darker shades of blue represent larger values of ω_{rs}], and (bottom) the adjacency matrix with the limit of blocks indicated as colored lines and edges as white dots. The optima of the objective functions are found via simulated annealing and greedy search [46]. (a) Natural partition of the network in $q = 2$ blocks, as found with the classical Bernoulli SBM via expectation–maximization (EM) on $\mathcal{B}$ and $\boldsymbol{\omega}$. In this case alone, $\boldsymbol{\omega}$ is learned and not imposed. (b) Balanced cut obtained with $\gamma \approx -25$ and the GMGM. The two blocks have size $\boldsymbol{n}^{\mathsf{T}} = [650, 572]$. A similar partition is identified by the modularity. (c) Double core-periphery found with $\gamma \approx -9$. The cores are of sizes 98 and 87 while their respective peripheries contain 396 and 641 nodes. There are only 3 703 edges between nodes of the peripheries (out of a maximum of 283 330 possible edges).

This leads to the graphical objective function

$$\widetilde{H}(G|\mathcal{B}) = \sum_{i \leqslant j} a_{ij} x_{\sigma_i \sigma_j}. \qquad (13)$$

Functions of the form of Eq. (13) are plagued by many trivial optima, since it is often possible to maximize $\widetilde{H}$ by placing all nodes in one or a few blocks. For instance, if $x_{rr} = 1$ for at least one r, then Eq. (13) is maximized by putting all nodes in block B_r—it is obvious that no mesoscopic information is contained in the resulting partition. In general, if Eq. (13) rewards placing many edges between some pair of blocks (B_r, B_s) via $x_{rs} = 1$, then it is possible to find good solutions simply by putting a lot of *nodes* in these blocks: The more nodes, the more edges, and therefore the better score. We

discourage these uninformative solutions by introducing an additive *balance constraints* $h(\mathcal{B})$ that penalizes the objective function $\widetilde{H}$ for partitions that contain large blocks aligned with X. Specifically, we use a quadratic constraint on the block sizes [16,48]

$$h(\mathcal{B}) = 2\gamma \sum_{r,s} x_{rs} n_r n_s, \qquad \gamma < 0, \qquad (14)$$

where n_r is the size of block B_r, and where $|\gamma|$ controls the overall strength of the constraint h. Because the constraint

appearing in Eq. (14) can be rewritten as

$$2\gamma \sum_{r,s} x_{rs} n_r n_s = 2\gamma \sum_{r,s} x_{rs} \left(\sum_{i=1}^{n} \delta_{\sigma_i r} \right) \left(\sum_{j=1}^{n} \delta_{\sigma_j s} \right)$$
$$= \gamma \sum_{i \leqslant j} x_{\sigma_i \sigma_j},$$

where δ_{ab} is the Kronecker δ (equal to 1 if $a = b$ and to zero otherwise), the constrained version of Eq. (13) is equivalent to

$$H(G|\mathcal{B}) = \widetilde{H}(G|\mathcal{B}) + h(\mathcal{B}) = \sum_{i \leqslant j} x_{\sigma_i \sigma_j}[a_{ij} + \gamma]. \quad (15)$$

This balanced objective function is obviously associated with a pairwise score function equivalent to that of the GMGM [cf. Eq. (12)]. Therefore, all objective functions formulated as an *edge count maximization* coupled with an additive *quadratic balance constraint* are equivalent to the GMGM. Furthermore, the strength of the balance constraint γ can be seen as a function of the parameters (ω_a, ω_b) of the corresponding GMGM: The greater the difference between ω_a and ω_b, the stronger the balance constraint.

The equivalence of the GMGM with combinatorial objective function has far reaching consequences, because many MPE methods are based on variation on these functions. A few well-known examples are: Balanced minimum cut, with $X = I$ where I is an identity matrix [8]; approximative graph coloring with $X = 11^{\mathsf{T}} - I$ [39,49]; nonoverlapping core-peripheries (CP) under size constraints [26,50] with, e.g.,

$$X_{\mathrm{CP1}} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \qquad X_{\mathrm{CP2}} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix},$$

$$X_{\mathrm{MultiCP}} = \begin{pmatrix} 1 & 1 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix}.$$

If anything, these simple examples show that the GMGM and Eq. (15) can be used as an "objective function factory" of sort: For any choice of γ and q, there will be $2^{\binom{q}{2}+q}$ different binary symmetric matrices X, and as many MPE objective functions. Those that are named and well studied are but a tiny fraction of the full spectrum of possibilities; most will uncover exotic patterns that are mixtures of core-peripheries, cuts, coloring, hierarchies, etc.

B. Complete hierarchy

While the SBM and its GMGM are general enough to specialize to many well-known MPE methods, there are also numerous objective functions that cannot be written as in Eqs. (9) and (12)—e.g., modularity functions—because they rely also on some *side-information* matrix λ absent from the pairwise scores of Eq. (9). The purpose of the present section is to expand on the classification of Sec. III A to accommodate these functions.

1. Stochastic block model with side information

In the spirit of Ref. [12], we define a generalization of the Poisson SBM, whose likelihood is given by

$$\mathbb{P}(G|\mathcal{B}, \boldsymbol{\omega}, \boldsymbol{\Lambda}) = \prod_{i \leqslant j} \frac{(\omega_{\sigma_i \sigma_j} \lambda_{ij})^{a_{ij}}}{a_{ij}!} e^{-\omega_{\sigma_i \sigma_j} \lambda_{ij}}. \quad (16)$$

This (over-parametrized) version of the SBM combines mesoscopic information (via $\boldsymbol{\omega}$) with side information at the level of edges (via $\boldsymbol{\Lambda}$). It directly specializes to many well–known likelihoods, including the classical Poisson SBM (with $\boldsymbol{\Lambda} = 11^{\mathsf{T}}$), or the degree-corrected SBM of Ref. [12] (with $\boldsymbol{\Lambda} = kk^{\mathsf{T}}/2m$ where k is the vector of degrees).

As with its classical counterpart, one can find the most likely partition of the nodes of G by maximizing the logarithm of the likelihood (16):

$$\log \mathbb{P} = \sum_{i \leqslant j} [a_{ij} \log \omega_{\sigma_i \sigma_j} \lambda_{ij} - \omega_{\sigma_i \sigma_j} \lambda_{ij} - \log a_{ij}!].$$

Therefore, the maximum likelihood formulation of the SBM with side information (hereafter SISBM) is associated with the pairwise score function

$$f_{\mathrm{SISBM}}(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) \sim a_{ij} \log \omega_{\sigma_i \sigma_j} - \omega_{\sigma_i \sigma_j} \lambda_{ij}. \quad (17)$$

The likelihood is not useful in itself, because there are too many parameters for the amount of information encoded in A. However, considering Eq. (17) not as a proper MPE method, but rather as the starting point of a general objective function hierarchy, it becomes a useful classification tool.

2. General modular graph model with side–information

As with the classical SBM, it is possible to define a GMGM specialization of the SISBM. Following Sec. III A 2, the idea is again to classify all pairs of blocks according to their density category (via X), and to reuse the identities appearing in Eq. (11) to rewrite the log-likelihood. The resulting likelihood is still over-parametrized because of $\boldsymbol{\Lambda}$, but much simpler than that of the general SISBM, since the connection matrices $\boldsymbol{\omega}$ are now restricted to the form of Eq. (10). It is easy to show that the pairwise score function is now

$$\begin{aligned} f_{\mathrm{SIGMGM}}&(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) \\ &= a_{ij}[\log \omega_b + x_{rs}(\log \omega_a - \log \omega_b)] \\ &\quad - [\omega_b + x_{\sigma_i \sigma_j}(\omega_a - \omega_b)]\lambda_{ij} \\ &\sim x_{\sigma_i \sigma_j}[a_{ij}(\log \omega_a - \log \omega_b) - \lambda_{ij}(\omega_a - \omega_b)] \\ &\sim x_{\sigma_i \sigma_j}[a_{ij} + \gamma \lambda_{ij}], \end{aligned} \quad (18)$$

where $\gamma < 0$ is the same parameter as the one appearing in Eq. (12).

3. Modularity functions

One of the reasons why the GMGM specialization is useful is, again, that it can be derived from first principles in a completely different manner, this time from the point of view of the *modularity* [25]. In a nutshell, modularity is defined as the difference between the number of internal edges of a partition (edges that connect two nodes in the same block), and the *expected* number of internal edges for this partition, if

TABLE I. Examples of null models. In all cases, $\zeta > 0$ is a free parameter, $\rho \in [0, 1]$ is the density of the network. The gravity model is included as an example of an exotic null model; it is derived for spatially embedded network, with r_{ij} being the Euclidean distance between nodes i and j and ϕ some reference connection propensity in space.

Model	λ_{ij}	Ref.
Configuration model (CM)	$k_i k_j/(2m)$	[25]
CM with resolution	$\zeta k_i k_j/(2m)$	[38]
Erdős-Rényi	ρ	[51]
Constant Potts model	ζ	[51]
Gravity model	$k_i k_j \phi(r_{ij})$	[52]

the network were to be drawn from some null model. The idea behind modularity is to maximize the number of edges within blocks, while accounting for the edges that would have been there in the first place, just by pure chance (assuming some model for the network, see Table I).

Modularity is a graphical objective function, since it can be written as a sum over pairs of nodes [31,38]. Writing the expected number of edges between the nodes i and j as λ_{ij} under the null model of choice, the modularity of a partition reads

$$H_{\text{Mod}}(\mathcal{B}, \boldsymbol{\lambda}, G) \propto \sum_{i \leqslant j} [a_{ij} - \lambda_{ij}] \delta_{\sigma_i \sigma_j}. \quad (19)$$

Importantly, the pairwise score function associated with the modularity is always given by

$$f_{\text{Mod}}(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) = (a_{ij} - \lambda_{ij}) \delta_{\sigma_i \sigma_j}$$
$$\sim (a_{ij} + \gamma \widetilde{\lambda}_{ij}) \delta_{\sigma_i \sigma_j}, \quad (20)$$

independent from the choice of null model (where $\gamma < 0$ and $\widetilde{\lambda}_{ij}$ is a rescaled connection probability under the null model).

A comparison with Eq. (18) reveals that the above score function—and therefore *any* modularity-type function—is in fact a specialization of the GMGM with side information, recovered by setting $\boldsymbol{X} = \boldsymbol{I}$, $\gamma = -1$, and by using the null model of the modularity as the side-information matrix $\boldsymbol{\Lambda}$. In other words, every modularity function is equivalent to some variant of the SIGMGM, where the null model is multiplied with the connection matrix $\boldsymbol{\omega}$ where $\boldsymbol{X}$ is simply the identity.

It is worth pointing out that using a flat null model (i.e., $\widetilde{\lambda}_{ij} = 1$) in Eq. (20) amounts to opting for a GMGM without side information, with $\boldsymbol{X} = \boldsymbol{I}$. Because the latter is associated with a pairwise score functions that is equivalent to edge counts coupled with quadratic balance constraints, it follows that *flat null models* act exactly like quadratic balance constraints. This correspondence explains the regularization properties of the ER null model investigated in Refs. [40,51] (among others).

We note in closing that while the connection between the modularity and the GMGM is presented here for $\boldsymbol{X} = \boldsymbol{I}$, it is of course possible to define "modularities" associated with different matrices $\boldsymbol{X}$, in the spirit of the side-information free equivalence. These modularities will be able to uncover any mixture of mesoscopic patterns reflected in $\boldsymbol{X}$.

IV. DISCUSSION

In this paper, we have shown that the maximum likelihood formulation of the SBM is perfectly equivalent to a number of standard mesoscopic pattern extraction (MPE) methods, upon appropriate specialization of its density matrix $\boldsymbol{\omega}$. Specifically, we have found that different classes of density matrices are associated with various classes of MPE algorithms, such as minimum cuts, modularities, core-periphery algorithm, and combinations thereof. This has allowed us to delineate a hierarchy of MPE methods (Fig. 3) and to understand all methods as increasingly simplified SBMs. In doing so, we have shown that the SBM is universal with respect to mesoscopic pattern extraction with graphical functions—a conclusion that is complementary to the recent observation that the SBM is a universal network approximator [53].

Apart from a better understanding of MPE methods, in the light of the hierarchy of Fig. 3, there are a number of practical consequences to the fact that many of the MPE methods of network science are, after all, the SBM in disguise. Let us mention a few in closing.

First and foremost, these equivalences imply that the efficient maximizers (see, e.g., Ref. [54]) developed to tackle the hard problem of estimating $\mathcal{B}$ for the general SBM can be reused to solve more specific MPE problems that are also hard. This application of the equivalences is direct: To optimize an MPE objective function, simply fix the matrix $\boldsymbol{\omega}$ (and $\boldsymbol{\Lambda}$ if there is side information) with some target mesoscopic pattern in mind, and run an SBM likelihood maximization procedure to uncover $\mathcal{B}$ (we have used this method to obtain the results of Fig. 4).

Second, as is also pointed out in Ref. [5] (for the special case of modularity), arbitrary MPE methods that are specializations of the SBM now stand on sounder statistical foundations, once their connection with the SBM is recognized. This is due to the fact that their free parameters—e.g., γ in Eqs. (12) and (18)—can be interpreted as functions of the connectivity matrix $\boldsymbol{\omega}$, thereby providing a statistically principled estimation procedure—expectation-maximization [39]—for otherwise arbitrary parameters.

Third, we can conclude that a number of hidden assumptions are built into popular MPE methods. In particular, they amount to fitting simplified SBMs by maximum likelihood, often with misspecified density matrices $\boldsymbol{\omega}$. Doing so is not a problem *per se*, because the goal of MPE is not always to find the most natural or most statistically robust decomposition of a network [3]; it might instead be to reveal different facets of the mesoscopic organization of a same network (see Fig. 4). However, one should bear in mind that using these MPE methods amounts to fitting an ill-defined model, with all the problems that this may bring about, such as missing the best description of a network or preventing inference algorithms from converging at all [4,55,56].

Fourth, a knowledge of equivalences can help us better interpret the empirical outcomes of mesoscopic pattern extraction. Two algorithms may behave similarly on a set of networks not due to the robustness of the patterns therein but because they share an equivalent notion of optimality. Hence, empirical studies that rely on many MPE algorithms— say, comparative analyses [57,58]—can avoid being lured by

what appears to be a strong consensus of many methods that actually implement the same notion of optimality.

Finally, the equivalences lead to a number of theoretical shortcuts. One, the consistency results derived for the SBM [59] apply directly to all MPE algorithms in the hierarchy, by specialization. The consistency of the SBM in most scaling regimes (and the existence of a *detectability limit* [39]) therefore extends to virtually every MPE algorithm studied thus far. Two, formal NP-hardness results can be extended to many MPE methods, using trivial reductions. For example, since it is known that modularity maximization is in NP-hard [60], the equivalence of modularity with the likelihood maximization of the GMGM specialization of the degree-corrected SBM [5] directly implies the NP-hardness

of the latter, and therefore of the SBM. Three, the universality of the SBM suggests that there is an extension of the no free lunch Theorem of Ref. [29] to a more generalized notion of MPE problems—not just community detection.

[1] C. Moore, Bulletin of the EATCS, issue 121, 25 (2017).

[2] S. Fortunato and D. Hric, Phys. Rep. **659**, 1 (2016).

[3] M. T. Schaub, J.-C. Delvenne, M. Rosvall, and R. Lambiotte, Appl. Netw. Sci. **2**, 4 (2017).

[4] P. Zhang and C. Moore, Proc. Natl. Acad. Sci. USA **111**, 18144 (2014).

[5] M. E. J. Newman, Phys. Rev. E **94**, 052315 (2016).

[6] A. Roxana Pamfil, S. D. Howison, R. Lambiotte, and M. A. Porter, arXiv:1804.01964 (2018).

[7] N. Veldt, D. F. Gleich, and A. Wirth, *WWW '18 Proceedings of the 2018 World Wide Web Conference* (ACM, New York, 2018), pp. 439–448.

[8] M. E. J. Newman, Phys. Rev. E **88**, 042822 (2013).

[9] T. Kawamoto and Y. Kabashima, Europhys. Lett. **112**, 40007 (2015).

[10] N. Masuda, M. A. Porter, and R. Lambiotte, Phys. Rep. **716**, 1 (2017).

[11] Z. Chang, H.-M. Cheng, C. Yan, X. Yin, and Z.-Y. Zhang, arXiv:1801.03618 (2018).

[12] B. Karrer and M. E. J. Newman, Phys. Rev. E **83**, 016107 (2011).

[13] B. Ball, B. Karrer, and M. E. J. Newman, Phys. Rev. E **84**, 036103 (2011).

[14] D. B. Larremore, A. Clauset, and A. Z. Jacobs, Phys. Rev. E **90**, 012805 (2014).

[15] S. P. Borgatti and M. G. Everett, Soc. Networks **21**, 375 (2000).

[16] P. Šulc and L. Zdeborová, J. Phys. A **43**, 285003 (2010).

[17] D. Hric, T. P. Peixoto, and S. Fortunato, Phys. Rev. X **6**, 031038 (2016).

[18] M. E. J. Newman and A. Clauset, Nat. Commun. **7**, 11863 (2016).

[19] A. Arenas, J. Duch, A. Fernández, and S. Gómez, New J. Phys. **9**, 176 (2007).

[20] A. Clauset, C. Moore, and M. E. J. Newman, Nature **453**, 98 (2008).

[21] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, J. Stat. Mech.: Theory Exp. (2008) P10008.

[22] S. Fortunato, Phys. Rep. **486**, 75 (2010).

[23] M. A. Porter, J.-P. Onnela, and P. J. Mucha, Not. AMS **56**, 1082 (2009).

[24] S. Shai, N. Stanley, C. Granell, D. Taylor, and P. J. Mucha, arXiv:1705.02305 (2017).

[25] M. E. J. Newman and M. Girvan, Phys. Rev. E **69**, 026113 (2004).

[26] M. P. Rombach, M. A. Porter, J. H. Fowler, and P. J. Mucha, SIAM J. Appl. Math. **74**, 167 (2014).

[27] MPE problems are quite generally in NP-HARD [60,61].

[28] D. L. Stein and C. M. Newman, *Spin Glasses and Complexity* (Princeton University Press, Princeton, NJ, 2013).

[29] L. Peel, D. B. Larremore, and A. Clauset, Sci. Adv. **3**, e1602548 (2017).

[30] B. W. Kernighan and S. Lin, Bell Syst. Tech. J. **49**, 291 (1970).

[31] M. E. J. Newman, Proc. Natl. Acad. Sci. USA **103**, 8577 (2006).

[32] S. White and P. Smyth, in *Proceedings of the 2005 SIAM International Conference on Data Mining* (SIAM, Philadelphia, PA, 2005), pp. 274–285.

[33] U. N. Raghavan, R. Albert, and S. Kumara, Phys. Rev. E **76**, 036106 (2007).

[34] G. Tibély and J. Kertész, Physica A (Amsterdam) **387**, 4982 (2008).

[35] M. J. Barber and J. W. Clark, Phys. Rev. E **80**, 026129 (2009).

[36] G. Ver Steeg, C. Moore, A. Galstyan, and A. Allahverdyan, Europhys. Lett. **106**, 48004 (2014).

[37] J. Reichardt and M. Leone, Phys. Rev. Lett. **101**, 078701 (2008).

[38] J. Reichardt and S. Bornholdt, Phys. Rev. E **74**, 016110 (2006).

[39] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, Phys. Rev. Lett. **107**, 065701 (2011).

[40] P. Ronhovde and Z. Nussinov, Phys. Rev. E **81**, 046114 (2010).

[41] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Science **220**, 671 (1983).

[42] Formally, f is in fact a placeholder for a function space with some parametrization set $\boldsymbol{\pi}$; f only represents a unique function upon choosing some $p \in \boldsymbol{\pi}$. Specializing f corresponds to defining a function space f_S associated with a parameter set $\boldsymbol{\pi}_S \subseteq \boldsymbol{\pi}$.

[43] P. W. Holland, K. B. Laskey, and S. Leinhardt, Soc. Networks **5**, 109 (1983).

[44] We base our derivation on the Poisson SBM, nearly equivalent to the somewhat more standard Bernoulli SBM. Our choice is

justified by the fact that the full hierarchy of objective functions follows more naturally from the Poisson SBM.

[45] T. Kawamoto and Y. Kabashima, Phys. Rev. E **95**, 012304 (2017).

[46] J.-G. Young, P. Desrosiers, L. Hébert-Dufresne, E. Laurence, and L. J. Dubé, Phys. Rev. E **95**, 062304 (2017).

[47] A parametrization where $\omega_b > \omega_a$ can be represented by an equivalent parametrization $(X', \boldsymbol{\omega}')$, defined as $X' = \mathbf{11}^T - X$ with $\omega_a' = \omega_b$, and $\omega_b' = \omega_a$. The case $\omega_a = \omega_b$ is somewhat pathological, but it can be handled nonetheless, with $X = \mathbf{11}^T$, $\omega_a' = \omega_a$ and $\omega_b' = \delta$, for any $\delta < \omega_a$.

[48] M. E. J. Newman, Europhys. Lett. **103**, 28003 (2013).

[49] F. Krzakala and L. Zdeborová, Phys. Rev. Lett. **102**, 238701 (2009).

[50] S. Kojaku and N. Masuda, Phys. Rev. E **96**, 052313 (2017).

[51] V. A. Traag, P. Van Dooren, and Y. Nesterov, Phys. Rev. E **84**, 016114 (2011).

[52] P. Expert, T. S. Evans, V. D. Blondel, and R. Lambiotte, Proc. Natl. Acad. Sci. USA **108**, 7663 (2011).

[53] S. C. Olhede and P. J. Wolfe, Proc. Natl. Acad. Sci. USA **111**, 14722 (2014).

[54] T. P. Peixoto, Phys. Rev. E **89**, 012804 (2014).

[55] T. Kawamoto, arXiv:1710.08816 (2017).

[56] T. Kawamoto, Phys. Rev. E **97**, 032301 (2018).

[57] A. Ghasemian, H. Hosseinmardi, and A. Clauset, arXiv:1802.10582 (2018).

[58] T. Kawamoto and Y. Kabashima, Phys. Rev. E **97**, 022315 (2018).

[59] E. Abbe, J. Mach. Learn. Res. **18**, 1 (2018).

[60] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner, arXiv:physics/0608255 (2006).

[61] P. Crescenzi and V. Kann, *International Workshop on Randomization and Approximation Techniques in Computer Science, RANDOM*, Lecture Notes in Computer Science, Vol. 1269 (Springer, Berlin, 1997), pp. 111–118.