On the universality of the stochastic block model

Jean-Gabriel Young\textsuperscript{a,b,1}, Guillaume St-Onge\textsuperscript{a,b}, Patrick Desrosiers\textsuperscript{a,b,c}, and Louis J. Dubé\textsuperscript{a,b}

\textsuperscript{a}Département de physique, de génie physique, et d’optique, Université Laval, Québec (QC), G1V 0A6, Canada
\textsuperscript{b}Centre interdisciplinaire de modélisation mathématique de l’Université Laval, Québec (QC), G1V 0A6, Canada
\textsuperscript{c}Centre de recherche de CERVO, Québec (QC), GIJ 2G3, Canada

\textsuperscript{1}Corresponding author: jean-gabriel.young.1@ulaval.ca

June 13, 2018

Abstract

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, imperfect graph colouring. 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, or one of its direct generalizations. These equivalence relations show that the SBM is nearly universal with respect to MPE problems.

1 Introduction

Whether it is called community detection, graphical inference, spectral embedding, unsupervised learning, bisection or graph colouring, 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 towards 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) [4–6], various spectral methods [7], normalized-cut [8], random-walks [9], and non-negative matrix factorization [10]. 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 mesoscopic pattern extraction (MPE) methods are actually the maximum likelihood formulation of the SBM in disguise (and a generalization of its degree-corrected version [11, 12]). 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. With these two operations, we delineate a hierarchy that crystalizes the idea of the SBM as a general MPE tool: Through specialization of its likelihood, it can be tailored to find specific patterns such as assortative and disassortative communities [5], bipartite structures [13], or core-periphery splits [14]. These specialized likelihoods, in turn, take the form of modularities, volumes,
cuts, all related through equivalence relationships. This 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, 15].

2 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 [16, 17]). Sometimes, the hope is to reveal functional components, while at other times
it is only a matter of making the dataset more manageable [2, 3, 18–22]. 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. The precise definition
of what is a suitable partition will depend on the type of MPE problem, and in turn, this will commend
different algorithms.

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 Sec. 3.1–3.2.
Our goal with the next four subsections is therefore to clarify these issues.

2.1 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 centred 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 analysed. 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.
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 this 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 this paper, we will focus on partial equivalence, essentially ignoring the maximizers. This is motivated by the observation that maximizers are, by necessity, efficient heuristics designed to find “good enough” optima in the rugged landscape of partitions [1, 25, 26] and that no free-lunch type theorems imply that different objective functions and different inputs are associated to different optimal maximizers [26]. From this point onward, therefore, by equivalence, we will refer to the equivalence of the objective functions used.

2.2 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 [20]. 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. [27], the fast unfolding method of Ref. [19], 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 / maximizer dichotomy. Consider as an example the classical label propagation algorithm of Ref. [28]. 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 neighbours). 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 $B_1$ is better than $B_2$ if the algorithm goes from $B_2$ to $B_1$ 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 a (complicated) objective function with these orderings, and therefore a two–part algorithm indistinguishable from the original.

2.3 General graphical objective function

Having established that a separation of algorithms in 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 to each pair of nodes. For the sake of generality, we will define these scores as real-valued functions, with essential dependencies on the partition $B = \{B_1, \ldots, B_q\}$ 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 $\lambda$ that contains any pairwise information not directly captured by $A$. Let us therefore write the score

1 MPE problems are quite generally in \textsc{np-hard} [23, 24].

2 Technically, partitions not connected by a majority update are not comparable with the above rule, which implies that we cannot write, even in principle, a well-defined objective function inducing a total ordering. The translation can however be completed by adding a trivial criterion that allows for the ordering of unrelated partitions: Define unrelated stable partitions and unrelated unstable partitions as equally good, and define any stable partition as better than any unstable partitions.
associated to the pair of nodes \((i, j) \in [n] \times [n]\) (we use the integers \([k] = \{1, \ldots, k\}\) to denote the nodes) as

\[
f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j),
\]

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

\[
H(A, \lambda, \sigma; f) = \sum_{i, j: 1 \leq i \leq j \leq n} f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j),
\]

yielding a global objective function based on pairwise scores. The choice of a sum is for mere convenience; equivalently, a product aggregate could have been implemented by taking \(f \mapsto \log f\) and \(H \mapsto e^H\).

### 2.4 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.

#### 2.4.1 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

\[
H(A, \lambda, \sigma; f) < H(A, \lambda, \sigma'; f) \implies g \circ H(A, \lambda, \sigma; f) < g \circ H(A, \lambda, \sigma'; f),
\]

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. 3, 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

\[
f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) = f_1(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) + f_2(a_{ij}, \lambda_{ij}),
\]

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

\[
H(A, \lambda, \sigma; f) = \sum_{i \leq j} f_1(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) + \sum_{i \leq j} f_2(a_{ij}, \lambda_{ij})
\]

\[
\sim \sum_{i \leq j} f_1(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j)
\]

\[
= H'(A, \lambda, \sigma; f)
\]

where \(\sim\) denotes equivalence, and \(i \leq j\) is a shorthand for the more precise statement \(i, j : 1 \leq i \leq j \leq n\). The equivalence holds because the additive terms are independent from \(\sigma\) and therefore do not affect the ordering. Thus, equivalence will often follow from a simple linear transformation of the form \(g \circ H = H - \sum_{i \leq j} f_2(a_{ij}, \lambda_{ij})\).
2.4.2 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. 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 \( H'(A, \lambda, \sigma; f_s) \) is a specialization of the objective function \( H(A, \lambda, \sigma; f) \) 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

\[
f_s(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) = f_s(a_{ij}, \lambda_{ij}, \sigma'_i, \sigma'_j) \quad \text{and} \quad f(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) \neq f(a_{ij}, \lambda_{ij}, \sigma'_i, \sigma'_j),
\]

where \( \sigma \neq \sigma' \), 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.

3 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 3.1 and 3.2 are devoted to showing how this comes about. We begin with the methods that do not account for any side-information \( \lambda_{ij} \), in Section 3.1. We show that they can be understood as specialization of the maximum likelihood formulation of the classical SBM [29]. We then move on to general MPE methods, in Sec. 3.2, 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. [11]. This part of the hierarchy sits above the methods of Sec. 3.1, since the generalized SBM contains the classical SBM as a special case. We summarize the relations between the various methods in Fig. 1, and illustrate the methods in Fig. 2.

3.1 Partial hierarchy (no side-information)

3.1.1 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 \( B \) of its nodes. The SBM becomes a MPE algorithm once this likelihood is used to infer the hidden partition \( B \) of \( G \). Although there are many ways of harnessing the likelihood to extract the mesoscopic patterns encoded by \( B \), we will only focus on likelihood maximization, because it directly fits within the two–part model of MPE algorithms defined in Sec. 2.1; 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 \( \sigma \), the classical SBM prescribes that the number of edges between nodes \((i, j)\) should be drawn from a Poisson distribution of
mean $\omega_{\sigma_1, \sigma_2}$. All edges are assumed to be independent, such that the likelihood of the complete graph is given by
\[
P(G|B, \omega) = \prod_{i \leq j} \frac{(\omega_{\sigma_i, \sigma_j})^{a_{ij}} e^{-\omega_{\sigma_i, \sigma_j}}}{a_{ij}!}.
\] (7)

It is parametrized by the $q \times q$ matrix $\omega$ and the partition $B$ (or equivalently by the block assignments $\sigma$). The standard inference procedure calls for the estimation of both, $\omega$ and $B$, usually through alternated learning of the two sets of parameters (via the expectation–maximization algorithm [30]). However, we will focus on the estimation of $B$ alone, treating the parameters $\omega$ as “control buttons.” The freedom to impose parameters $\omega$ on the network will ultimately allow us to draw relations with other MPE algorithms.

To extract $B^*(G)$—the “true” partition of the nodes—from the network, we maximize the likelihood of the SBM with respect to the partition (see also Sec. 4). Since the logarithm is a strictly increasing function of its argument, we may equivalently maximize the log-likelihood \(^6\)
\[
\log P(G|B, \omega) = \sum_{i \leq j} [a_{ij} \log \omega_{\sigma_i, \sigma_j} - \omega_{\sigma_i, \sigma_j} - \log a_{ij}!].
\] (8)

It becomes evident upon inspection of Eq. (8) that it is, in fact, a graphical objective function of the general form appearing in Eq. (2), associated to the pairwise score function
\[
\tilde{f}_{SBM}(a_{ij}, \sigma_i, \sigma_j) \sim a_{ij} \log \omega_{\sigma_i, \sigma_j} - \omega_{\sigma_i, \sigma_j}.
\] (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.

### 3.1.2 General modular graph model

One such (explicit) specialization is the general modular graph model (GMGM) [31, 32]. 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 $\omega$ of the GMGM are much more structured than that of the SBM.

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 $(B_a, B_b)$ is of type $a$, then we set $x_{rs} = 1$. Contrariwise, we set $x_{rs} = 0$ if the pair $(B_a, B_b)$ is of type $b$. Pairs of blocks of type $a$ are then all associated to a connectivity $\omega_{rs} = \omega_a$, while pairs of type $b$ are associated to a connectivity $\omega_{rs} = \omega_b$, where we take $\omega_b < \omega_a$ without loss of generality\(^6\). Every connection matrices of the GMGM can therefore be written as
\[
\omega = \omega_a \mathbf{1} \mathbf{1}^T + (\omega_a - \omega_b)X
\] (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 [31, 32]. In particular, the two identities
\[
\omega_{rs} = \omega_a + x_{rs}(\omega_a - \omega_b),
\] (11a)
\[
\log \omega_{rs} = \log \omega_a + x_{rs}(\log \omega_a - \log \omega_b),
\] (11b)

\(^5\)A first (trivial) example of the concept of equivalence of Sec. 2.4.

\(^6\)A parametrization where $\omega_a > \omega_b$ can be represented by an equivalent parametrization $(X', \omega')$, defined as $X' = \mathbf{1} \mathbf{1}^T - X$ 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 $X = \mathbf{1} \mathbf{1}^T$, $\omega'_a = \omega_a$ and $\omega'_b = \delta$, for any $\delta < \omega_a$.  

6
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 to the score function

\[ f_{\text{GMGM}}(a_{ij}, \sigma_i, \sigma_j) = a_{ij} \left[ \log \omega_b + x_{\sigma_i, \sigma_j} (\log \omega_a - \log \omega_b) \right] - [\omega_b + x_{\sigma_i, \sigma_j} (\omega_a - \omega_b)] \\
\sim x_{\sigma_i, \sigma_j} [a_{ij}] + \gamma \]

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 to 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 to a block pair of type \(b\), then it only has an indirect impact, by omission.

### 3.1.3 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. Functions of the form (13) are plagued by many trivial optima, since it is often possible to maximize \(H\) 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 \(h(B)\) that penalizes the objective function \(H\) for partitions that contain large blocks aligned with \(X\). Specifically, we use a quadratic constraint on the block sizes [15, 33]

\[ h(B) = 2\gamma \sum_{r,s} x_{rs} n_r n_s, \quad \gamma < 0, \]

where \(n_r\) is the size of block \(B_r\), and where \(|\gamma|\) controls the overall strength of the constraint \(h\). Because constraint (14) can be rewritten as

\[ 2\gamma \sum_{r,s} x_{rs} n_r n_s = 2\gamma \sum_{r,s} x_{rs} \left( \sum_{i=1}^{n} I[\sigma_i = r] \right) \left( \sum_{j=1}^{n} I[\sigma_j = s] \right) = \gamma \sum_{i \leq j} x_{\sigma_i, \sigma_j}, \]

where \(I[x]\) is the indicator function (equal to 1 only if the statement \(x\) is true), the constrained version of Eq. (13) is equivalent to

\[ H(G|B) = \tilde{H}(G|B) + h(B) = \sum_{i \leq j} x_{\sigma_i, \sigma_j} [a_{ij} + \gamma]. \]
This balanced objective function is obviously associated to a pairwise score function equivalent to that of the GMGM [c.f. 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 $\gamma$ can be seen as a function of the parameters $(\omega_a, \omega_b)$ of the corresponding GMGM: The greater the difference between $\omega_a$ and $\omega_b$, the stronger the balance constraint.

This equivalence has far reaching consequences, because many MPE methods are based on specializations of the above. A few well-known examples are: Balanced minimum cut, with $X = I$ where $I$ is an identity matrix [7]; approximative graph colouring with $X = 11^T - I$ [30, 34]; non-overlapping core-peripheries (CP) under size constraints [35, 36] with, e.g.,

$$X_{CP1} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad X_{CP2} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \quad \text{or} \quad X_{MultiCP} = \begin{pmatrix} 1 & 1 & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 1 \\ 0 & 0 & \ldots & 1 & 0 \end{pmatrix}.$$ 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 $\gamma$ and $q$, there will be $2^q(q^2) + q$ 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, colouring, hierarchies, etc.

### 3.2 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 $\lambda$ absent from the pairwise scores of Eq. (9). The purpose of the present section is to expand on the classification of Sec. 3.1 to accommodate these functions.

#### 3.2.1 Stochastic block model with side–information

In the spirit of Ref. [11], we define a generalization of the Poisson SBM, whose likelihood is given by

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

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

$$\log P(G | B, \omega, \Lambda) = \sum_{i \leq j} [a_{ij} \log \omega_{\sigma_i, \sigma_j} \lambda_{ij} - \omega_{\sigma_i, \sigma_j} \lambda_{ij} - \log a_{ij}!] .$$ (17)

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

$$f_{\text{SISBM}}(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) \sim a_{ij} \log \omega_{\sigma_i, \sigma_j} - \omega_{\sigma_i, \sigma_j} \lambda_{ij} .$$ (18)
Figure 1: **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 is obviously transitive. 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 coloured in blue (general classification) and red (binary classification).
This likelihood is not useful in itself, because there are too many parameters for the amount of information encoded in $A$. However, considering Eq. (18) not as a proper MPE methods, but rather as the starting point of a general objective function hierarchy, it becomes a useful classification tool.

### 3.2.2 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. 3.1.2, the idea is again to classify all pairs of blocks according to their density category (via $X$), and to re-use the identities appearing in Eq. (11) to rewrite the log-likelihood. This likelihood is still over-parametrized through $\Lambda$, but much simpler than that of the general SISBM, since the connection matrices $\omega$ are now restricted to the form (10). It is easy to show that the pairwise score function is now

$$f_{\text{SIGMGM}}(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) = a_{ij} \left[ \log \omega_b + x_{rs}(\log \omega_a - \log \omega_b) \right] - \left[ \omega_b + x_{\sigma_i\sigma_j}(\omega_a - \omega_b) \right] \lambda_{ij}$$

$$\sim x_{\sigma_i\sigma_j} \left[ a_{ij} (\log \omega_a - \log \omega_b) - \lambda_{ij} (\omega_a - \omega_b) \right]$$

$$\sim x_{\sigma_i\sigma_j} \left[ a_{ij} + \gamma \lambda_{ij} \right]$$

(19)

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

### 3.2.3 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 [37]. 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 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.

Modularity is a graphical objective function, since it can be written as a sum over pairs of nodes [27, 38]. Writing the expected number of edges between the nodes $\{i, j\}$ as $\lambda_{ij}$ under the null model of choice, the modularity of a partition reads

$$H_{\text{Mod}}(B, \Lambda, G) \propto \sum_{i \leq j} [a_{ij} - \lambda_{ij}] I[\sigma_i = \sigma_j] .$$

(20)

Modularity-type objective functions pervade the community detection literature [2, 20]; they rely on various null models, including but not limited to:

- **Configuration model** [37] \( \lambda_{ij} = \frac{k_i k_j}{2m} \) (21a)
- **Configuration model with resolution** [38] \( \lambda_{ij} = \gamma \frac{k_i k_j}{2m} \) (21b)
- **Erdős-Rényi (ER)** \( \lambda_{ij} = \rho \) (21c)
- **Gravity Model** [39] \( \lambda_{ij} = k_i k_j \phi(r_{ij}) \) (21d)

where $\rho$ is the density of the network [Eq. (21c)], and where we have included (21d) an example of exotic null model, derived for spatially embedded network, with $r_{ij}$ being the Euclidean distance between nodes $(i, j)$ and $\phi$ some reference connection propensity in space. Importantly, the pairwise score function associated to the modularity is always given by

$$f_{\text{Mod}}(a_{ij}, \lambda_{ij}, \sigma_i, \sigma_j) = (a_{ij} - \lambda_{ij}) I[\sigma_i = \sigma_j] \sim (a_{ij} + \gamma \lambda_{ij}) I[\sigma_i = \sigma_j] ,$$

(22)
independent from the choice of null model (where $\gamma < 0$ and $\tilde{\lambda}_{ij}$ is a rescaled connection probability under the null model).

A comparison with Eq. (19) reveals that the above score function—and therefore that of any modularity-type function—is in fact a specialization of the GMGM with side-information, recovered by setting $X = I$, $\gamma = -1$, and by using the null model of the modularity as the side-information matrix $\Lambda$. In other words, every modularity function is equivalent to some variant of the SISBM, where the null model appears as a scaling factor of the likelihood and where $X$ is simply the identity.

With this in mind, it is worth pointing out that using a flat null model (i.e., $\tilde{\lambda}_{ij} = 1$) in Eq. (22) leads to a pairwise score function analogous to the one appearing in Eq. (15) (for $X = I$). Because the latter are derived from edge counts coupled with quadratic balance constraints, it follows that flat null models act exactly like quadratic balance constraints. This explains the regularization properties of the ER null model investigated in Ref. [40, 41] (among others).

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

4 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 $\omega$. 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. 1), and to understand all methods as increasingly simplified SBM. 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 [42].

Apart from a better understanding of MPE methods, in the light of the hierarchy of Fig. 1, 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 all the machinery developed to tackle the hard problem of estimating $B$ in the general SBM can be re-used to solve more specific MPE problems that are also hard [see, e.g., Refs. [30, 43]]. This application is direct: Simply fix the matrix $\omega$ (and $\Lambda$ if there is side-information) with some target mesoscopic pattern in mind, and run an inference procedure of choice to uncover $B$. This is precisely how we have obtained the results of Fig. 2. Changing our point of view reveals different facets of the mesoscopic organization of a same network.

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. $\gamma$ in Eqs. (19) and (12)—can be interpreted as functions of the connectivity matrix $\omega$, thereby providing a statistically principled estimation procedure—expectation–maximization [30]—for otherwise arbitrary parameters.

Third, understanding equivalence relations allows us to 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 $\omega$. This is not necessarily 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]. 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 [4, 44–46].
Figure 2: 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 US 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 coloured according to the identified partition, (centre) a cartoon of the matrix $\omega$ imposed for the equivalent SBM [darker shades of blue represent larger values of $\omega_{rs}$], and (bottom) the adjacency matrix with the limit of blocks indicated as coloured lines and edges as white dots. The optima of the objective functions are found via simulated annealing and greedy search [32]. (a) Natural partition of the network in $q = 2$ blocks, as found with the classical Bernoulli SBM via expectation–maximization (EM) on $B$ and $\omega$. In this case alone, $\omega$ is learned and not imposed. (b) Balanced cut obtained with $\gamma \approx -25$ and the GMGM. The two blocks have size $n^T = [650, 572]$. 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).
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 [47, 48]—can avoid being lured by what appears to be a strong consensus of many methods, whereas the agreement comes about because they simply share the same notion of optimality.

Finally, the equivalences lead to a number of theoretical shortcuts. One, the consistency results pertaining to the SBM [49] apply directly to these MPE algorithms, by specialization. This further justifies the study of the consistency of the SBM, since it has consequences for virtually every MPE algorithm studied thus far. Two, formal NP–hardness results can be extended to many methods using trivial reductions. For example, since it is known that modularity maximization is in NP–hard [23], 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.

Acknowledgments

We thank Laurent Hébert-Dufresne for useful comments. This work was funded by the Fonds de recherche du Québec-Nature et technologies (JGY, PD), the Conseil de recherches en sciences naturelles et en génie du Canada (GSO, LJD), and the Sentinel North program, financed by the Canada First Research Excellence Fund (JGY, GSO, PD, LJD).

References

[1] C. Moore, “The computer science and physics of community detection: landscapes, phase transitions, and hardness,” arXiv:1702.00467, 2017.
[2] S. Fortunato and D. Hric, “Community detection in networks: A user guide,” Phys. Rep., vol. 659, pp. 1–44, 2016.
[3] M. T. Schaub, J.-C. Delvenne, M. Rosvall, and R. Lambiotte, “The many facets of community detection in complex networks,” Appl. Netw. Sci., vol. 2, no. 1, p. 4, 2017.
[4] P. Zhang and C. Moore, “Scalable detection of statistically significant communities and hierarchies, using message passing for modularity,” Proc. Natl. Acad. Sci. U.S.A., vol. 111, no. 51, pp. 18144–18149, 2014.
[5] M. E. J. Newman, “Equivalence between modularity optimization and maximum likelihood methods for community detection,” Phys. Rev. E, vol. 94, p. 052315, Nov 2016.
[6] A. Roxana Pamfil, S. D. Howison, R. Lambiotte, and M. A. Porter, “Relating modularity maximization and stochastic block models in multilayer networks,” arXiv:1804.01964, 2018.
[7] M. E. J. Newman, “Spectral methods for community detection and graph partitioning,” Phys. Rev. E, vol. 88, no. 4, p. 042822, 2013.
[8] T. Kawamoto and Y. Kabashima, “Detectability of the spectral method for sparse graph partitioning,” Europhys. Lett., vol. 112, no. 4, p. 40007, 2015.
[9] N. Masuda, M. A. Porter, and R. Lambiotte, “Random walks and diffusion on networks,” Phys. Rep., vol. 716, 2017.
[10] Z. Chang, H.-M. Cheng, C. Yan, X. Yin, and Z.-Y. Zhang, “On approximate equivalence of modularity, d and non-negative matrix factorization,” arXiv:1801.03618, 2018.
[11] B. Karrer and M. E. J. Newman, “Stochastic blockmodels and community structure in networks,” Phys. Rev. E, vol. 83, no. 1, p. 016107, 2011.
[12] B. Ball, B. Karrer, and M. E. J. Newman, “Efficient and principled method for detecting communities in networks,” *Phys. Rev. E*, vol. 84, no. 3, p. 036103, 2011.

[13] D. B. Larremore, A. Clauset, and A. Z. Jacobs, “Efficiently inferring community structure in bipartite networks,” *Phys. Rev. E*, vol. 90, no. 1, p. 012805, 2014.

[14] S. P. Borgatti and M. G. Everett, “Models of core/periphery structures,” *Soc. Networks*, vol. 21, no. 4, pp. 375–395, 2000.

[15] P. Šuc and L. Zdeborová, “Belief propagation for graph partitioning,” *J. Phys. A*, vol. 43, no. 28, p. 285003, 2010.

[16] D. Hric, T. P. Peixoto, and S. Fortunato, “Network structure, metadata, and the prediction of missing nodes and annotations,” *Phys. Rev. X*, vol. 6, no. 3, p. 031038, 2016.

[17] M. E. J. Newman and A. Clauset, “Structure and inference in annotated networks,” *Nat. Comm.*, vol. 7, 2016.

[18] A. Arenas, J. Duch, A. Fernández, and S. Gómez, “Size reduction of complex networks preserving modularity,” *New J. Phys.*, vol. 9, no. 6, p. 1082–1097, 2009.

[19] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” *J. Stat. Mech. Theor. Exp.*, vol. 2008, no. 10, p. P10008, 2008.

[20] S. Fortunato, “Community detection in graphs,” *Phys. Rep.*, vol. 486, no. 3, pp. 75–174, 2010.

[21] M. A. Porter, J.-P. Onnela, and P. J. Mucha, “Communities in networks,” *Notices of the AMS*, vol. 56, no. 9, pp. 1082–1097, 2009.

[22] S. Shai, N. Stanley, C. Granell, D. Taylor, and P. J. Mucha, “Case studies in network community detection,” *arXiv:1705.02305*, 2017.

[23] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner, “Maximizing modularity is hard,” *arXiv:0608255*, 2006.

[24] P. Crescenzi and V. Kann, *A compendium of NP optimization problems*. Universit di Roma, 1995.

[25] D. L. Stein and C. M. Newman, *Spin Glasses and Complexity*. Princeton University Press, 2013.

[26] L. Peel, D. B. Larremore, and A. Clauset, “The ground truth about metadata and community detection in networks,” *Sci. Adv.*, vol. 3, no. 5, p. e1602548, 2017.

[27] M. E. J. Newman, “Modularity and community structure in networks,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 103, no. 23, pp. 8577–8582, 2006.

[28] U. N. Raghavan, R. Albert, and S. Kumara, “Near linear time algorithm to detect community structures in large-scale networks,” *Phys. Rev. E*, vol. 76, no. 3, p. 036106, 2007.

[29] P. W. Holland, K. B. Laskey, and S. Leinhardt, “Stochastic blockmodels: First steps,” *Soc. Networks*, vol. 5, no. 2, pp. 109–137, 1983.

[30] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, “Inference and phase transitions in the detection of modules in sparse networks,” *Phys. Rev. Lett.*, vol. 107, no. 6, p. 065701, 2011.

[31] T. Kawamoto and Y. Kabashima, “Detectability thresholds of general modular graphs,” *Phys. Rev. E*, vol. 95, no. 1, p. 012304, 2017.

[32] J.-G. Young, P. Desrosiers, L. Hébert-Dufresne, E. Laurence, and L. J. Dubé, “Finite-size analysis of the detectability limit of the stochastic block model,” *Phys. Rev. E*, vol. 95, no. 6, p. 062304, 2017.

[33] M. E. J. Newman, “Community detection and graph partitioning,” *Europhys. Lett.*, vol. 103, no. 2, p. 28003, 2013.

[34] F. Krzakala and L. Zdeborová, “Hiding quiet solutions in random constraint satisfaction problems,” *Phys. Rev. Lett.*, vol. 102, no. 23, p. 238701, 2009.

[35] M. P. Rombach, M. A. Porter, J. H. Fowler, and P. J. Mucha, “Core-periphery structure in networks,” *SIAM J. Appl. Math.*, vol. 74, no. 1, p. 167, 2014.
[36] S. Kojaku and N. Masuda, “Finding multiple core-periphery pairs in networks,” Phys. Rev. E, vol. 96, no. 5, p. 052313, 2017.
[37] M. E. J. Newman and M. Girvan, “Finding and evaluating community structure in networks,” Phys. Rev. E, vol. 69, p. 026113, 2004.
[38] J. Reichardt and S. Bornholdt, “Statistical mechanics of community detection,” Phys. Rev. E, vol. 74, no. 1, p. 016110, 2006.
[39] P. Expert, T. S. Evans, V. D. Blondel, and R. Lambiotte, “Uncovering space-independent communities in spatial networks,” Proc. Natl Acad. of Sci., vol. 108, no. 19, pp. 7663–7668, 2011.
[40] V. A. Traag, P. Van Dooren, and Y. Nesterov, “Narrow scope for resolution-limit-free community detection,” Phys. Rev. E, vol. 84, no. 1, p. 016114, 2011.
[41] P. Ronhovde and Z. Nussinov, “Local resolution-limit-free potts model for community detection,” Phys. Rev. E, vol. 81, no. 4, p. 046114, 2010.
[42] S. C. Olhede and P. J. Wolfe, “Network histograms and universality of blockmodel approximation,” Proc. Natl Acad. of Sci., vol. 111, no. 41, p. 14722, 2014.
[43] T. P. Peixoto, “Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models,” Phys. Rev. E, vol. 89, no. 1, p. 012804, 2014.
[44] T. Kawamoto, “Algorithmic infeasibility of community detection in higher-order networks,” arXiv:1710.08816, 2017.
[45] T. Kawamoto, “Algorithmic detectability threshold of the stochastic block model,” Phys. Rev. E, vol. 97, no. 3, p. 032301, 2018.
[46] T. P. Peixoto, “Bayesian stochastic blockmodeling,” arXiv:1705.10225, 2017.
[47] A. Ghasemian, H. Hosseinmardi, and A. Clauset, “Evaluating overfit and underfit in models of network community structure,” arXiv:1802.10582, 2018.
[48] T. Kawamoto and Y. Kabashima, “Comparative analysis on the selection of number of clusters in community detection,” Phys. Rev. E, vol. 97, no. 2, p. 022315, 2018.
[49] E. Abbe, “Community detection and the stochastic block model: recent developments,” arXiv:1703.10146, 2017.