General and exact approach to percolation on random graphs

Antoine Allard,[*] Laurent Hébert-Dufresne,[†] Jean-Gabriel Young, and Louis J. Dubé
Département de physique, de génie physique, et d'optique, Université Laval, Québec, Québec, Canada G1V 0A6

We present a comprehensive and versatile theoretical framework to study site and bond percolation on clustered and correlated random graphs. Our contribution can be summarized in three main points. (i) We introduce a set of iterative equations that solve the exact distribution of the size and composition of components in finite-size quenched or random multitype graphs. (ii) We define a very general random graph ensemble that encompasses most of the models published to this day and also makes it possible to model structural properties not yet included in a theoretical framework. Site and bond percolation on this ensemble is solved exactly in the infinite-size limit using probability generating functions [i.e., the percolation threshold, the size, and the composition of the giant (extensive) and small components]. Several examples and applications are also provided. (iii) Our approach can be adapted to model interdependent graphs—whose most striking feature is the emergence of an extensive component via a discontinuous phase transition—in an equally general fashion. We show how a graph can successively undergo a continuous then a discontinuous phase transition, and preliminary results suggest that clustering increases the amplitude of the discontinuity at the transition.

I. INTRODUCTION

Percolation on graphs offers a simple theoretical framework to model and investigate the behavior of many complex systems, noteworthy examples being the growth and the robustness of their structure [1,2], their observability [3,4], as well as the effect of their structure on the propagation of emerging infectious agents [5,6]. On the analytical front, recent progress has been mainly achieved within the configuration model (CM) paradigm [7], which, in the limit of large graphs, allows an exact and simple analytical treatment with the use of probability generating functions (pgfs) [8,9]. The versatility of the pgf method has triggered the development of many variants of the CM, reproducing, to some extent, correlations and clustering found in real complex systems [10–31].

To move beyond what has been done thus far, we introduce a very general and comprehensive class of random graphs that increases significantly the nontrivial correlations and clustering patterns that can be handled analytically. Correlations and clustering are incorporated into the graphs through the use of types of vertices and types of stubs (i.e., half edge stemming from vertices). Hence, by explicitly controlling *who is connected to whom* and *through what kind of connection*, our approach reproduces any correlations as long as they can be mapped onto this multitype framework. For instance, the type of the vertices can correspond to their degree (the number of neighbors) [22,30], to their intrinsic properties such as age or ethnicity [11,23], or to their position in the k-core structure of the graph [16].

Furthermore, the use of types of stubs explicitly accounts for different categories of connections. On the one hand, these differences may be of a *conceptual* nature [32]. For instance, in *multilayer* or *multiplex* graphs the type of an edge refers to the layer of interaction to which it belongs (e.g., family

ties and acquaintances in social networks). On the other hand, the different types of stubs can describe different *topological* functions. Since some edges may be undirected or directed, different types of stubs can be used to identify *in-degrees*, *out-degrees*, or *undirected degrees* [9,20]. More importantly perhaps, stubs can be matched in groups of more than two vertices to form *motifs*, also called *hyperedges* [10,17], permitting the inclusion of clustering in a very general and natural fashion. These motifs can take a wide variety of forms: Simple triangles, cliques of several hundreds of vertices, or arbitrary graphs with directed and multiple edges [see Fig. 1(a)]. Additionally, these motifs can have a quenched (i.e., fixed) or a random structure (e.g., multitype Erdős-Rényi graphs).

We have developed a mathematical framework that solves the site *and* bond percolation (hereafter *hybrid* percolation) on this general class of random graphs. We build upon the well-known pgf-based formalism and obtain the analytical expression for the size of the extensive "giant" component, the percolation threshold, as well as the distribution of the size of the "small" components in the limit of large graph size. However, the pgf approach *de facto* assumes locally treelike graphs forbidding closed loops and therefore any clustering whatsoever. To circumvent this limitation, we present a set of iterative equations that exactly solves the size distribution of components in finite-size arbitrary, or quenched, graphs. These equations map the possible outcomes of hybrid percolation on any motif (i.e., the size distribution of the components) onto a distribution of branching trees and thereby reconcile the presence of motifs with the treelike requirement of the pgf approach.

The general nature of our model acts as a *theoretical laboratory* where the effect of a wide selection of structural features on the outcomes of hybrid percolation can be investigated on a common ground. To facilitate understanding and to provide support for our claims, several examples accompany the analysis and illustrate its practical implementation. Moreover, our model encompasses most variants of the CM published to date; we provide several examples supporting this claim as well.

---

[*]Now at Departament de Física Fonamental, Universitat de Barcelona, Carrer de Martí i Franquès 1, 08028 Barcelona, Spain.
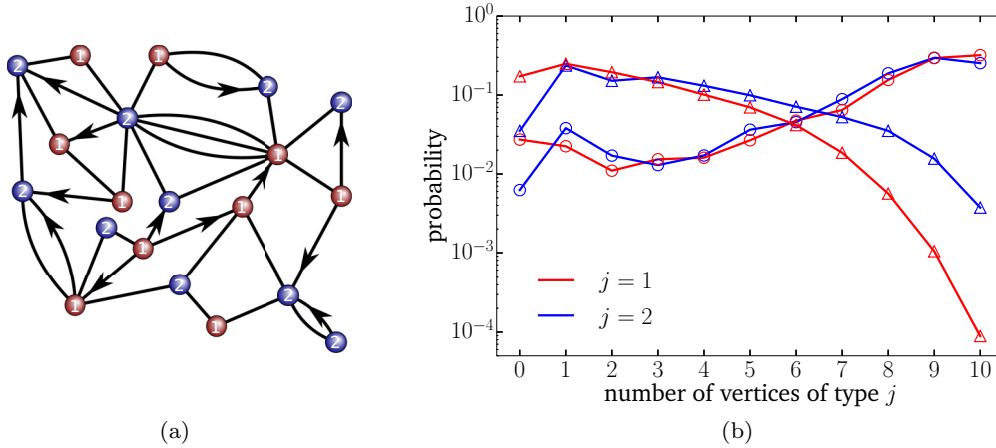[†]Now at Santa Fe Institute, Santa Fe, New Mexico 87501, USA.

FIG. 1. (Color online) (a) Example of an arbitrary graph that can be handled by our framework. Red and blue represent vertex types 1 and 2, respectively. There are ten vertices of each type. (b) Distribution of the number of vertices of type j in components reached from an initial vertex chosen at random for the graph shown in (a). Symbols are the results of 2.5×10^8 numerical simulations, and lines are the predictions of Eqs. (3a)–(3d). The distributions are discrete; lines have been added to guide the eye. Triangles ($\triangle$) correspond to pure site percolation, with $\{\tilde{r}'_s\} \equiv \{\tilde{r}'_1, \tilde{r}'_2\} = \{0.40, 0.70\}$ and $\{\tilde{p}'_{ij}\} \equiv \{\tilde{p}'_{11}, \tilde{p}'_{12}, \tilde{p}'_{21}, \tilde{p}'_{22}\} = \{1.00, 1.00, 1.00, 1.00\}$. Circles ($\circ$) correspond to hybrid percolation (site and bond) with $\{\tilde{r}'_s\} \equiv \{\tilde{r}'_1, \tilde{r}'_2\} = \{0.95, 0.90\}$ and $\{\tilde{p}'_{ij}\} \equiv \{\tilde{p}'_{11}, \tilde{p}'_{12}, \tilde{p}'_{21}, \tilde{p}'_{22}\} = \{1.00, 0.90, 0.85, 0.95\}$. These probabilities are given in terms of the original vertex types to lighten the presentation (thus the use of a prime). To use the mapping described in Sec. II B, a probability for each individual vertex and each individual edge must be defined. For instance, if vertex 8 is of type 1, we set $r_8 = \tilde{r}'_1$. Similarly, if vertex 5 is of type 2 and shares three undirected edges with vertex 8, we set $p_{58} = 1 - (1 - \tilde{p}'_{21})^3$ and $p_{85} = 1 - (1 - \tilde{p}'_{12})^3$.

Finally, we show how our approach can be adapted to model interdependent graphs—in which the extensive component emerges via a discontinuous transition instead of a continuous one [33,34]—through a suitable change in the definition of what constitutes an extensive component (i.e., the order parameter). This adaptation shows that a graph can successively undergo a continuous then a discontinuous phase transition [35] and provide a quantitative measure of the effect of clustering on the emergence of the extensive component.

The paper is organized as follows. In Sec. II, we introduce the set of iterative equations that exactly solves the size distribution of components in finite-size arbitrary graphs. In Sec. III, we formally define the general graph ensemble discussed above and obtain its exact structural properties under hybrid percolation (i.e., the size and composition of the components and the position of the percolation threshold). We then illustrate the workings of our formalism with several examples and special cases in Sec. IV. We finally show how our approach can be adapted to model interdependent graphs in Sec. V. Conclusions and final remarks are collected in Sec. VI.

II. PERCOLATION ON FINITE-SIZE ARBITRARY GRAPHS

To reconcile the treelike assumption of the pgf approach with the presence of motifs in graphs, the outcomes of percolation on these motifs—the distribution of the number of vertices that can be reached from a given vertex—must be obtained beforehand. These distributions can be computed by hand by enumerating each possible configuration where vertices and edges exist with given probabilities [17,21]. However, this procedure becomes rapidly unwieldy for motifs of more than a handful of vertices. Instead, we generalize the equations presented in Ref. [36] to obtain a set of iterative

equations that solve the outcome of hybrid percolation on small arbitrary graphs.

A. Multitype Erdős-Rényi graphs

Let us first consider multitype random graphs as a generalization of the $\mathcal{G}_{n,p}$ model (i.e., Erdős-Rényi random graphs) in which n vertices are linked by edges that exist individually and independently with a probability p [37]. We generalize this model by labeling vertices using types; the set of types is noted $\mathcal{N}$, and there are a total of $|\mathcal{N}|$ types of vertices. A directed edge from a vertex of type i towards a vertex of type j (noted $i \rightarrow j$) exists with a probability p_{ij} independently of other potential edges [38]. For the sake of conciseness, we refer to a graph composed of n_i vertices of type i (with $i = 1, \ldots, |\mathcal{N}|$) with the vector $\boldsymbol{n} \equiv (n_1, \ldots, n_{|\mathcal{N}|})^\mathsf{T}$. We use a similar notation for other quantities throughout this paper, unless specified otherwise.

Since edges may be directed, we define a component as the vertices that are reachable from a given initial vertex, including itself (i.e., the out-component rooted to this given vertex). This initial vertex is identified solely by its type since vertices of a given type are indistinguishable. We define $W_i(\boldsymbol{l}|\boldsymbol{n})$ as the probability that $\boldsymbol{l} \equiv (l_1, \ldots, l_{|\mathcal{N}|})^\mathsf{T}$ vertices can be reached from an initial vertex of type i in a graph containing $\boldsymbol{n}$ vertices. The calculation of $W_i(\boldsymbol{l}|\boldsymbol{n})$ begins with the initial condition $W_i(\boldsymbol{\delta_i}|\boldsymbol{\delta_i}) = 1$, where $\boldsymbol{\delta_i}$ is the vector of Kronecker δ's $(\delta_{i1}, \ldots, \delta_{i|\mathcal{N}|})^\mathsf{T}$ and corresponds to a single vertex of type i. This initial condition simply states that the probability of finding a component of one vertex of type i in a graph containing one vertex of type i is 1. Now suppose that there are other vertices in the graph and that it contains $\boldsymbol{n}$ vertices instead. The probability of finding a component of only one vertex of type i in this graph, $W_i(\boldsymbol{\delta_i}|\boldsymbol{n})$, is equal to the probability that there is a component containing $\boldsymbol{\delta_i}$ vertex,

$W_i(\boldsymbol{\delta}_i|\boldsymbol{\delta}_i)$ (which in this case is equal to one), multiplied by the probability that none of the potential edges from the vertex in the component (i.e., the vertex of type i) towards the other vertices of the graph of size $\boldsymbol{n}$ exist:

$$W_i(\boldsymbol{\delta}_i|\boldsymbol{n}) = W_i(\boldsymbol{\delta}_i|\boldsymbol{\delta}_i)\prod_{k\in\mathcal{N}}(1-p_{ik})^{n_k-\delta_{ik}}. \qquad (1)$$

Let us now consider a component made of two vertices of type i, which we note $2\boldsymbol{\delta}_i$. By definition of the multitype random graphs, we know that $W_i(2\boldsymbol{\delta}_i|2\boldsymbol{\delta}_i) = p_{ii}$ since the component exists only if there is a directed edge from the initial vertex to the other vertex of type i. Following the steps leading to Eq. (1), the probability of finding a component of size $2\boldsymbol{\delta}_i$ from an initial vertex of type i in a graph containing $\boldsymbol{n}$ vertices (we assume that $n_i \geqslant 2$) is

$$W_i(2\boldsymbol{\delta}_i|\boldsymbol{n}) = W_i(2\boldsymbol{\delta}_i|2\boldsymbol{\delta}_i)(n_i - 1)\prod_{k\in\mathcal{N}}(1-p_{ik})^{2(n_k-2\delta_{ik})}, \qquad (2)$$

where the extra factor $(n_i - 1)$ accounts for the number of ways to choose the second vertex among the $n_i - 1$ available vertices of type i in the graph, and $2(n_k - 2\delta_{ik})$ is the number of potential edges from the two vertices of type i towards the $n_k - 2\delta_{ik}$ vertices of type k. Repeating this exercise for larger components, we obtain a general form for a generic component of size $\boldsymbol{l}$,

$$W_i(\boldsymbol{l}|\boldsymbol{n}) = W_i(\boldsymbol{l}|\boldsymbol{l})\prod_{j\in\mathcal{N}}\binom{n_j-\delta_{ij}}{l_j-\delta_{ij}}\prod_{k\in\mathcal{N}}(1-p_{jk})^{l_j(n_k-l_k)}, \qquad (3a)$$

where $W_i(\boldsymbol{l}|\boldsymbol{l})$ is the probability that $\boldsymbol{l}$ vertices form a component considering an initial vertex of type i. In this last equation, the binomial coefficients count the number of ways the other $\boldsymbol{l} - \boldsymbol{\delta}_i$ vertices in the component can be chosen from the $\boldsymbol{n} - \boldsymbol{\delta}_i$ vertices available in the graph, and the other terms correspond to the probability that no other vertices can be reached from the $\boldsymbol{l}$ vertices in the component.

The only missing information in Eq. (3a) is the probability $W_i(\boldsymbol{l}|\boldsymbol{l})$. As seen in the two simple examples above, it is possible to compute the probability $W_i(\boldsymbol{l}|\boldsymbol{l})$ by hand, but this calculation becomes rapidly tedious as the size of the component increases. Fortunately, we can use Eq. (3a) to circumvent this difficulty. For example, from $W_i(\boldsymbol{\delta}_i|\boldsymbol{\delta}_i) = 1$, Eq. (1) yields $W_i(\boldsymbol{\delta}_i|2\boldsymbol{\delta}_i) = 1 - p_{ii}$. Since the probabilities must sum to 1 for a given graph size, we conclude that $W_i(2\boldsymbol{\delta}_i|2\boldsymbol{\delta}_i) = 1 - W_i(\boldsymbol{\delta}_i|2\boldsymbol{\delta}_i) = p_{ii}$. Hence, it is possible to build upon the probabilities computed for smaller graph size to obtain the missing probability $W_i(\boldsymbol{l}|\boldsymbol{l})$ by simply asking for normalization. In general terms,

$$W_i(\boldsymbol{l}|\boldsymbol{l}) = 1 - \sum_{\boldsymbol{m}<\boldsymbol{l}}W_i(\boldsymbol{m}|\boldsymbol{l}), \qquad (3b)$$

where the probabilities $W_i(\boldsymbol{m}|\boldsymbol{l})$ are obtained with Eq. (3a) and where the sum covers every possible instance of $\boldsymbol{m}$ such that $m_j \leqslant l_j$ for all j but excludes the case in which all elements of the two vectors are equal (i.e., $m_j = l_j$ for every j). In short, Eqs. (3a) and (3b) are mutually dependent: The left-hand side of one feeds the right-hand side of the other. Thus, from a graph consisting of a single vertex (the initial condition), Eqs. (3a) and (3b) extend the graph to the desired size $\boldsymbol{n}$ and keep track

of the component size distribution along the way to build the final distribution $\{W_i(\boldsymbol{l}|\boldsymbol{n})\}$.

A mass of information is produced during the iteration of Eqs. (3a) and (3b): The probability of finding every possible components $\boldsymbol{l}$ in each intermediate graph whose size is smaller than $\boldsymbol{n}$. When interested in bond percolation solely (as in Ref. [36]), the only probabilities of interest are the ones related to the graph of maximum size $\boldsymbol{n}$. This ultimately leaves most of the calculated probabilities unused. However, if interested in hybrid percolation, that is, when edges *and* vertices exist with given probabilities, all the calculated probabilities can be put to contribution.

The probability for a graph of original size $\boldsymbol{n}$ to be left with $\boldsymbol{b}$ vertices after each of its vertices has been independently *kept* with probabilities $\{r_j\}_{j\in\mathcal{N}}$ (i.e., a vertex of type j is kept with probability r_j) is

$$B_i(\boldsymbol{b}|\boldsymbol{n}) \equiv \prod_{j\in\mathcal{N}}\binom{n_j-\delta_{ij}}{b_j-\delta_{ij}}r_j^{b_j-\delta_{ij}}(1-r_j)^{n_j-b_j}, \qquad (3c)$$

where we assume that the initial vertex of type i exists. Hence, from a starting vertex of type i, the probability to find a component of size $\boldsymbol{l}$ in a graph of original size $\boldsymbol{n}$ when vertices and edges exist with given probabilities, $Q_i(\boldsymbol{l}|\boldsymbol{n})$, is

$$Q_i(\boldsymbol{l}|\boldsymbol{n}) = \sum_{\boldsymbol{b}=\boldsymbol{l}}^{\boldsymbol{n}}W_i(\boldsymbol{l}|\boldsymbol{b})B_i(\boldsymbol{b}|\boldsymbol{n}), \qquad (3d)$$

where the sum covers every possible instances of $\boldsymbol{b}$ such that $l_j \leqslant b_j \leqslant n_j$ for every $j \in \mathcal{N}$. Thus, by slightly increasing the computational effort, it is possible to incorporate site percolation into the systematic method introduced in Ref. [36] for bond percolation.

B. Arbitrary graphs

Our framework can also be used to predict the outcomes of hybrid percolation on small arbitrary graphs. By small arbitrary graphs, we mean graphs of finite size with a fixed structure, in which edges may be directed and/or multiple, and whose vertices may belong to types. We use the *adjacency matrix* $\mathbf{A}$, whose element A_{ij} is the number of edges leaving vertex i towards vertex j, to specify the structure. Figure 1(a) depicts an example of such an arbitrary graph. In such graphs, percolation corresponds to the random removal of edges and vertices according to some given probabilities which may depend on the type of the vertices involved. Predicting the outcome of percolation then consists of predicting the probability that a component of size $\boldsymbol{l}$ can be reached from a given initial vertex in a graph of size $\boldsymbol{n}$.

For Eqs. (3a)–(3d) to be applicable, we need to map arbitrary graphs onto multitype random graphs. This mapping is achieved by assigning to each vertex its own type ($|\mathcal{N}|$ equals the number of vertices) and by setting the probabilities $\{p_{ij}\}$ to mimic the structure of the original arbitrary graph. To account for the fact that more than one edge may exist between two vertices in the original graph, we set $p_{ij} = 1 - (1 - \tilde{p}_{ij})^{A_{ij}}$, where $\tilde{p}_{ij}$ is the probability that an individual edge from vertex i to vertex j remains after the random removal of edges in the arbitrary graph. Note that $\tilde{p}_{ij}$ may depend on the original types of vertices i and j in the graph [e.g., there are two types

of vertices in Fig. 1(a)]. The same applies for the existence probabilities of vertices (i.e., $\{r_i\}$ must be equal to $\{\tilde{r}_i\}$). An example is given in the caption of Fig. 1. Using this mapping, Eqs. (3a)–(3d) offer a systematic procedure to compute the outcomes of hybrid percolation on small arbitrary graphs. Figure 1(b) compares the predictions of Eqs. (3a)–(3d) with the results of numerical simulations for the arbitrary graph shown in Fig. 1(a). As expected, a perfect agreement is observed.

III. PERCOLATION ON CORRELATED AND CLUSTERED INFINITE RANDOM GRAPHS

We now turn our attention to the generalized version of the CM briefly described in the Introduction. We provide a formal definition of the model and analytically solve percolation for this general ensemble of random graphs.

A. A *stub matching* scheme

The CM defines an ensemble of graphs that are random in all respects except for the *degree* of their vertices (the number of neighbors), which is prescribed by a given distribution $\{P(k)\}_{k \in \mathbb{N}}$. More precisely, to generate graphs of this ensemble, we start with N vertices and assign a degree to each one by drawing an integer from $\{P(k)\}_{k \in \mathbb{N}}$. We then build a list of stubs (half edges) in which a vertex whose degree is k appears k times. We shuffle the list and pair stubs according to this randomized list to create edges. Up to corrections of order $O(N^{-1})$, this procedure uniformly samples the ensemble of graphs with a given distribution whose second moment is bounded [7]. Moreover, as any closed loop occurs with a probability proportional to N^{-1}, this procedure generates graphs that are locally treelike in the limit $N \to \infty$.

We generalize this scheme to account for types of vertices and types of stubs. In our model, each of the N vertices belongs to a type and we note $\mathcal{N}$ the set of vertex types, as in the last section. We also note w_i the fraction of vertices whose type is i. As in the CM, vertices are assigned a number of stubs, but now these stubs are identified with types as well. We say that a vertex has k_α stubs of type α, and we note $\mathcal{E}$ the set of stub types. Unless specified otherwise, greek and latin letters refer to types of edges and vertices, respectively. The number of stubs of each type belonging to a vertex of type i is prescribed by the joint degree distribution $\{P_i(k_1, \ldots, k_{|\mathcal{E}|})\}_{k_1, \ldots, k_{|\mathcal{E}|} \in \mathbb{N}} \equiv \{P_i(\boldsymbol{k})\}_{\boldsymbol{k} \in \mathbb{N}^{|\mathcal{E}|}}$. Hence, when generating graphs from this ensemble, each of the N vertices is assigned a type according to $\{w_i\}_{i \in \mathcal{N}}$ and then assigned a number of stubs of each type according to the corresponding joint degree distribution.

To generate graphs from this sequence of vertices, we build a list of stubs for each pair (α, i), where $\alpha \in \mathcal{E}$ and $i \in \mathcal{N}$. For example, a vertex of type i that has k_α stubs of type α and k_β stubs of type β appears k_α times in the list (α, i) and k_β times in the list (β, i). Stubs are then randomly matched according to a set of rules—noted $\mathcal{R}$—to generate graphs. The information encoded in these rules is twofold. On the one hand, they prescribe from *which* lists should stubs be picked during the matching step. Mathematically, this is encoded in the distribution $\{R(\boldsymbol{n})\}_{\boldsymbol{n} \in \mathbb{N}^{|\mathcal{E}| \times |\mathcal{N}|}}$, where $\boldsymbol{n}$ is a matrix whose elements, $n_{\alpha i}$ (for every $\alpha \in \mathcal{E}$ and $i \in \mathcal{N}$), give the number of

stubs from each list involved in the edge (or *hyperedge*, if more than two stubs are involved). The probability that a hyperedge contains $\boldsymbol{n}$ stubs is then $R(\boldsymbol{n})$.

On the other hand, the rules $\mathcal{R}$ prescribe *how* the vertices are connected to one another within the hyperedge. For example, stubs from the list (α, i) and (α, j) could be paired to create undirected edges between layers i and j of *multilayer* graphs. Similarly, stubs from the lists (β, i) and (γ, i) could be paired to create directed edges between vertices of a same type (the two types of stubs corresponding, respectively, to the *in*-degree and *out*-degree). Moreover, three stubs from a same list could be matched to create triangles, or m stubs of type ε stemming from different types of vertices could be matched to form a multitype Erdős-Rényi motif, where edges exist with probability p (see Sec. II A). In fact, the hyperedges can take any imaginable form and composition as long as they can be mapped onto the multitype random graphs defined in Sec. II. Note that only one stub is required to be part of an hyperedge, even if this hyperedge contributes more than one to the degree of vertices. For instance, if stubs of type Δ correspond to triangles, a vertex with $k_\Delta = 2$ will belong to two triangles. An illustration of the stub matching scheme is given in Fig. 2.

For this graph ensemble to be consistent, the distributions $\{P_i(\boldsymbol{k})\}_{k \in \mathbb{N}^{|\mathcal{E}|}}$ and $\{R(\boldsymbol{n})\}_{\boldsymbol{n} \in \mathbb{N}^{|\mathcal{E}| \times |\mathcal{N}|}}$ must obey certain constraints in the limit $N \to \infty$. Namely,

$$\frac{w_i \langle k_\alpha \rangle_{P_i}}{w_j \langle k_\nu \rangle_{P_j}} = \frac{\langle n_{\alpha i} \rangle_R}{\langle n_{\nu j} \rangle_R} \tag{4}$$

for each $i, j \in \mathcal{N}$ and $\alpha, \nu \in \mathcal{E}$, where $\langle x \rangle_Y$ represents the average of x according to the distribution $Y(x)$. These constraints simply require that the ratio of the average number of elements in each list (left) equals the relative proportion in which pairs appear in hyperedges (right).

As for the CM, this *stub matching* scheme uniformly samples—up to corrections of order $O(N^{-1})$—a maximally random ensemble of graphs defined by the distributions $\{w_i\}_{i \in \mathcal{N}}$ and $\{P_i(\boldsymbol{k})\}_{i \in \mathcal{N}; k \in \mathbb{N}^{|\mathcal{E}|}}$ and by the rules $\mathcal{R}$. Since stubs are matched randomly, the graphs of that ensemble have an underlying treelike structure in the limit $N \to \infty$, except within clustered hyperedges.

B. Probability generating functions

To solve percolation on this general ensemble of random graphs, we adapt the well-known pgf approach [9,10] to account for vertex and stub types. As mentioned above, this approach assumes that the structure of the graphs is locally treelike, an assumption that is not valid whenever a hyperedge contains a loop (e.g., a triangle). However, by solving the component size distribution on each hyperedge beforehand, it is possible to consider that the hyperedge has an *effective* treelike structure: The probability that there is an *effective* edge from vertex A to vertex B is simply the probability that vertex B can be reached from vertex A either directly or through the other vertices in the hyperedge. Figures 3(a) and 3(b) illustrate the idea behind the effective treelike structure. This slight change of perspective allows the use of the pgf approach even though the treelike structure assumption is not valid in the original graph ensemble.

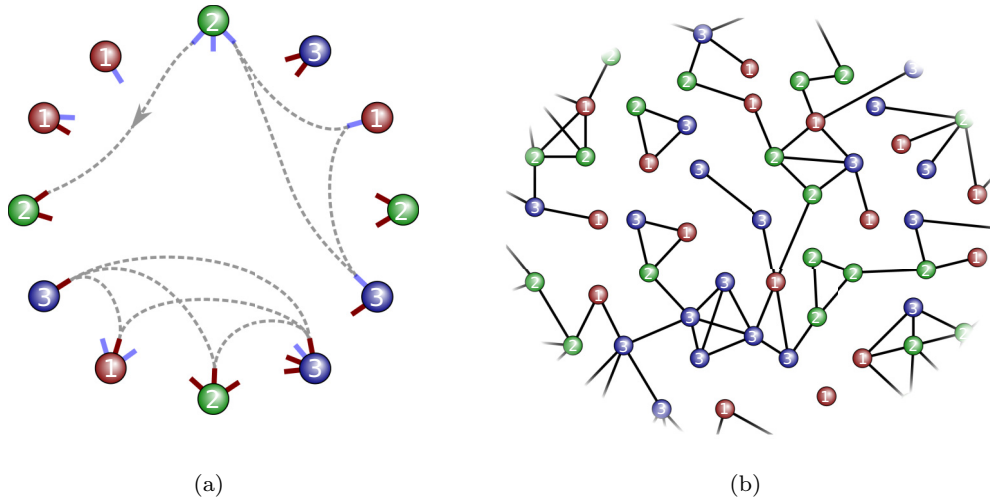(a)   (b)

FIG. 2. (Color online) Illustration of the stub matching scheme. (a) Vertices are attributed a type according to the distribution $\{w_i\}$, and are given a number of stubs of each type according to the distribution $\{P_i(\boldsymbol{k})\}$. There are $N = 12$ vertices, $|\mathcal{N}| = 3$ types of vertices (four vertices of type 1 in red, four vertices of type 2 in green, and four vertices of type 3 in blue), and $|\mathcal{E}| = 2$ types of stubs (light blue and dark red). Stubs are then randomly matched according to a set of rules, $\mathcal{R}$, to create hyperedges. For example, a light blue stub and a dark red stub that both stem from vertices of type 2 can be matched to create a directed edge, or three light blue stubs stemming from three vertices of types 1, 2, and 3 can be matched to create a triangle. More complex hyperedges are possible and can be handled by our mathematical approach. (b) Example of a graph obtained with the stub matching scheme which can reproduce a great variety of nontrivial correlations and clustering patterns. In the infinite-size limit ($N \to \infty$), the resulting graphs have an underlying treelike structure: There are no closed paths other than within clustered hyperedges.

The effective treelike structure of hyperedges is unveiled with Eqs. (3), where a vertex is now identified by the pair (α, i) instead of by its vertex type solely. In other words, we keep track of the type of the vertices but also the type of the stubs through which they are involved in the hyperedge. As a result, bold variables like $\boldsymbol{n}$ and $\boldsymbol{l}$ now contain $|\mathcal{E}| \times |\mathcal{N}|$ elements instead of the $|\mathcal{N}|$ elements as in Sec. II. The quantity $Q_{\alpha i}(\boldsymbol{l}|\boldsymbol{n}; \mathcal{R})$ therefore corresponds to the probability that a pair (α, i) leads to $\boldsymbol{l}$ pairs—i.e., $l_{\nu j}$ pairs (ν, j), for each $\nu \in \mathcal{E}$ and $j \in \mathcal{N}$—in a hyperedge containing $\boldsymbol{n}$ pairs. A dependency on the rules $\mathcal{R}$ has been added in $Q_{\alpha i}(\boldsymbol{l}|\boldsymbol{n}; \mathcal{R})$ to explicitly mark that the inner structure of the hyperedges (e.g., quenched or random nature, probabilities of existence of vertices or edges) is prescribed by these rules [39].

The pgf that generates the distribution of the number of pairs that can be reached from an initial pair (α, i) in an hyperedge containing $\boldsymbol{n}$ pairs is

$$\sum_{\boldsymbol{l}=\boldsymbol{\delta}_\alpha \circ \boldsymbol{\delta}_i}^{\boldsymbol{n}} Q_{\alpha i}(\boldsymbol{l}|\boldsymbol{n}; \mathcal{R}) \prod_{\substack{\nu \in \mathcal{E} \\ j \in \mathcal{N}}} x_{\nu j}^{l_{\nu j} - \delta_{\alpha\nu}\delta_{ij}}, \qquad (5)$$

where the sum covers every possible instances of $\boldsymbol{l}$ such that $\delta_{\mu\alpha}\delta_{mi} \leqslant l_{\mu m} \leqslant n_{\mu m}$ for every $m \in \mathcal{N}$ and $\mu \in \mathcal{E}$ ("$\circ$" denotes the entrywise product). These two δ's of Kronecker account for the fact that there is at least one pair (α, i) in the hyperedge. Similarly, the two others δ's of Kronecker $\delta_{\alpha\nu}\delta_{ij}$ appearing in Eq. (5) remove the initial pair—by definition included in $\boldsymbol{l}$—from the count of reachable pairs. Because we are ultimately interested in the number of pairs that can be reached from a given initial pair regardless of the specifics of the hyperedge, we must remove the dependency of Eqs. (5)

on the composition $\boldsymbol{n}$. To do so, we average this pgf over the probabilities that the initial pair (α, i) belongs to a hyperedge whose composition is $\boldsymbol{n}$. Since stubs are matched randomly, a vertex identified by the pair (α, i) is ten times more likely to belong to an hyperedge containing ten pairs (α, i) than to belong to an hyperedge that contains only one pair (α, i). Consequently, the probabilities $R(\boldsymbol{n})$ must be weighted by the number of pairs (α, i) that each composition contains, i.e., averaged over $n_{\alpha i} R(\boldsymbol{n})/\langle n_{\alpha i}\rangle_R$, where the normalizing factor, $\langle n_{\alpha i}\rangle_R$, is the average value of $n_{\alpha i}$ with respect to the distribution $R(\boldsymbol{n})$. Doing so yields the pgf generating the distribution of the number of pairs of each types that can be reached from a pair (α, i),

$$\theta_{\alpha i}(\boldsymbol{x}) = \sum_{\boldsymbol{n}} \frac{n_{\alpha i} R(\boldsymbol{n})}{\langle n_{\alpha i}\rangle_R} \sum_{\boldsymbol{l}=\boldsymbol{\delta}_\alpha \circ \boldsymbol{\delta}_i}^{\boldsymbol{n}} Q_{\alpha i}(\boldsymbol{l}|\boldsymbol{n}; \mathcal{R}) \prod_{\substack{\nu \in \mathcal{E} \\ j \in \mathcal{N}}} x_{\nu j}^{l_{\nu j} - \delta_{\alpha\nu}\delta_{ij}},$$

$$(6)$$

where the sum over $\boldsymbol{n}$ covers all hyperedge compositions such that $R(\boldsymbol{n}) \neq 0$. Computed for each initial pair (α, i), $\theta_{\alpha i}(\boldsymbol{x})$ provides the projection of the outcomes of percolation on the hyperedges onto an effective branching tree and therefore permits the use of the pgf approach.

To solve percolation on the graphs defined in the previous section, we first need to compute the distribution of the composition of the neighborhood of vertices. The neighborhood of a vertex is the set of *reachable* vertices with which it shares a hyperedge. In other words, vertex B is a neighbor of vertex A if there exists an effective edge from vertex A to vertex B. The pgf $\theta_{\alpha i}(\boldsymbol{x})$ generates the distribution of neighbors that a vertex of type i has through one of its stubs of type α. In the

(a) Before projection   (b) After projection   (c) Representation of $f_{\mu i}(\boldsymbol{x})$
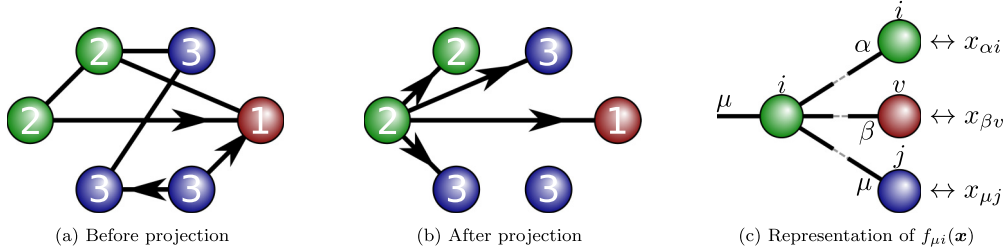
FIG. 3. (Color online) (a), (b) Effective treelike structure of an hyperedge from the point of view of a vertex of type 2 before and after projection, respectively. There exists an effective edge between the initial vertex of type 2 and any vertices that are directly or indirectly reachable from it. The probability for an effective edge to exist corresponds to the probability that a direct or indirect path exists. (c) Schematic representation of the pgf $f_{\mu i}(\boldsymbol{x})$. Knowing that a vertex of type i has been reached from one of its stubs of type μ [i.e., a pair (μ, i)], this pgf generates the distribution of the number of vertices of each type in its neighborhood, as well as the type of stubs from which they have been reached. The types of the vertices and of the stubs are identified with the subscripts of the variables $\boldsymbol{x} = \{x_{vi}\}$.

limit $N \to \infty$, the treelike structure of the graphs ensures that the neighboring vertices reachable through two different stubs do not overlap. Hence, the composition of the neighborhood of a vertex of type i that has k_α and k_β stubs of type α and β is generated by $[\theta_{\alpha i}(\boldsymbol{x})]^{k_\alpha} \cdot [\theta_{\beta i}(\boldsymbol{x})]^{k_\beta}$, which corresponds to the convolution of the distributions. Since the number of stubs belonging to vertices of type i is distributed according to $\{P_i(\boldsymbol{k})\}_{\boldsymbol{k} \in \mathbb{N}^{|\mathcal{E}|}}$, we obtain that the distribution of the composition of the neighborhood of vertices of type i is generated by

$$g_i(\boldsymbol{x}) = \sum_{\boldsymbol{k}} P_i(\boldsymbol{k}) \prod_{\alpha \in \mathcal{E}} [\theta_{\alpha i}(\boldsymbol{x})]^{k_\alpha}, \qquad (7)$$

where the sum covers all cases where $P_i(\boldsymbol{k}) \neq 0$. As in $\theta_{\alpha i}(\boldsymbol{x})$, this pgf keeps track of the type of the stubs from which the neighboring vertices have been reached through the subscripts of the variables $\boldsymbol{x} = \{x_{vj}\}_{v \in \mathcal{E}; j \in \mathcal{N}}$. In other words, $g_i(\boldsymbol{x})$ generates the number of pairs that are in the neighborhood of a vertex of type i. This pgf is analogous to the function $G_0(x)$ generating the degree distribution in the CM [7,9].

The complete solution to the percolation problem requires the distribution of possible neighborhood compositions for vertices reached through one of their stubs. As discussed for $\theta_{\alpha i}(\boldsymbol{x})$, the probability for a stub of type μ to be attached to a vertex with a total of $\boldsymbol{k}$ stubs (i.e., k_α stubs of type α for every $\alpha \in \mathcal{E}$) is weighted by the number of stubs of type μ that this vertex has. Hence, given that a vertex of type i has been reached through one of its stubs of type μ [i.e., a pair (μ, i)], the composition of the neighborhood accessible from its *other* stubs is generated by

$$f_{\mu i}(\boldsymbol{x}) = \sum_{\boldsymbol{k}} \frac{k_\mu P_i(\boldsymbol{k})}{\langle k_\mu \rangle_{P_i}} \prod_{\alpha \in \mathcal{E}} [\theta_{\alpha i}(\boldsymbol{x})]^{k_\alpha - \delta_{\alpha \mu}}, \qquad (8)$$

where $\delta_{\alpha \mu}$ has been added to exclude from the count the stub of type μ from which the vertex has been reached and where $\langle k_\mu \rangle_{P_i}$ is the average number of stubs of type μ that vertices of type i have. The distributions generated by $f_{\mu i}(\boldsymbol{x})$ are analogous to the *excess* degree distribution generated by $G_1(x)$ in the CM [7,9]. Figure 3(c) illustrates the information encoded in the pgfs $f_{\mu i}(\boldsymbol{x})$.

C. Extensive "giant" component

Having defined the pgfs $g_i(\boldsymbol{x})$ and $f_{\mu i}(\boldsymbol{x})$, the behavior of the extensive "giant" component can be predicted in the limit $N \to \infty$ using simple self-consistency arguments. We define $a_{\mu i}$ as the probability that a vertex of type i reached via one of its stubs of type μ does *not* lead to the giant component. Self-consistency then requires that if this pair does not lead to the giant component, then neither should the pairs that are reachable from it. Since the distribution of the number of pairs reachable from a given pair (μ, i) is generated by Eq. (8), this self-consistency requirement can be rewritten as

$$a_{\mu i} = f_{\mu i}(\boldsymbol{a}) \qquad (9)$$

for every $\mu \in \mathcal{E}$ and $i \in \mathcal{N}$. Because the coefficients of $f_{\mu i}(\boldsymbol{x})$ are normalized (they form a probability distribution), the point $\boldsymbol{a} = \boldsymbol{1}$ (every $a_{\mu i}$ equals 1) is always a solution of Eqs. (9). However, as the density of edges and/or vertices increases with increasing $\{r_j\}_{j \in \mathcal{N}}$ and/or $\{p_{jk}\}_{j,k \in \mathcal{N}}$, another solution where at least one element of $\boldsymbol{a}$ is smaller than 1 appears. This new solution marks the emergence of an extensive component.

Because their coefficients are all positives, the pgfs $f_{\mu i}(\boldsymbol{x})$ are all convex and monotonic increasing in $[0,1]^{|\mathcal{E}| \times |\mathcal{N}|}$. Hence, when $\boldsymbol{a} = \boldsymbol{1}$ is the only solution of Eqs. (9) in $[0,1]^{|\mathcal{E}| \times |\mathcal{N}|}$, it is the stable fixed point of (with $n \in \mathbb{N}$),

$$\boldsymbol{a}^{(n+1)} = \boldsymbol{f}(\boldsymbol{a}^{(n)}), \qquad (10)$$

for any initial condition $\boldsymbol{a}^{(0)}$ in $[0,1]^{|\mathcal{E}| \times |\mathcal{N}|}$, and where the map $\boldsymbol{f}(\boldsymbol{x})$ consists of every $f_{\mu i}(\boldsymbol{x})$. This fixed point becomes unstable through a transcritical bifurcation as soon as another solution in $[0,1]^{|\mathcal{E}| \times |\mathcal{N}|}$ appears. The shape of $\boldsymbol{f}(\boldsymbol{x})$ in $[0,1]^{|\mathcal{E}| \times |\mathcal{N}|}$ and the fact that $\boldsymbol{f}(\boldsymbol{1}) = \boldsymbol{1}$ implies that this other solution is unique in the interval of interest, that it is a stable fixed point of Eq. (10), and that the transition is continuous. Analyzing the stability of $\boldsymbol{f}(\boldsymbol{x})$ around the fixed point $\boldsymbol{a} = \boldsymbol{1}$ leads to the criterion for the emergence of the giant component

$$\det(\mathbf{J} - \mathbf{I}) = 0, \qquad (11)$$

where $\mathbf{J}$ is the Jacobian matrix of $\boldsymbol{f}(\boldsymbol{x})$ around $\boldsymbol{x} = \boldsymbol{1}$, and $\mathbf{I}$ is the identity matrix. Put differently, an extensive component exists whenever the largest eigenvalue of $\mathbf{J}$, $\lambda_{\max}(\mathbf{J})$, is greater than one.

Having solved Eqs. (9), the probability that a vertex of type i leads to the giant component through at least one of

its neighbors is given by $\mathcal{P}_i = 1 - g_i(\boldsymbol{a})$. Consequently, the probability that a randomly chosen vertex does lead to the giant component is

$$\mathcal{P} = \sum_{i \in \mathcal{N}} \frac{r_i w_i \mathcal{P}_i}{\sum_{j \in \mathcal{N}} r_j w_j} = 1 - \sum_{i \in \mathcal{N}} \frac{r_i w_i g_i(\boldsymbol{a})}{\sum_{j \in \mathcal{N}} r_j w_j}, \qquad (12)$$

where r_i is the probability that a vertex of type i exists.

As shown in Sec. II, hyperedges may include directed edges, or edges that are more likely to exist in one direction than the other [i.e., $p_{ij} \neq p_{ji}$ in Eq. (3a)]. This implies that while vertex B is in the neighborhood of vertex A, vertex A may not be in the neighborhood of vertex B. From such local asymmetries, a global asymmetry arises between the probability that a vertex leads to the giant component, $\mathcal{P}$, and the relative size $\mathcal{S}$ of the giant component. In such case, the extensive component has a "bow-tie" structure [11,40], meaning that the vertices involved in the extensive component belong to one of the three nonoverlapping sets $\mathcal{I}^{\text{in}}$, $\mathcal{I}^{\text{both}}$, and $\mathcal{I}^{\text{out}}$. The set $\mathcal{I}^{\text{in}}$ includes vertices that lead to the giant component but that cannot be reached from it; these vertices are somehow "hidden" behind directed edges. The set $\mathcal{I}^{\text{out}}$ contains vertices that cannot lead to the giant component but that can be reached from it; they are positioned downstream of directed edges. The set $\mathcal{I}^{\text{both}}$ contains vertices that lead to the giant component and that can be reached from it. From this, we conclude $\mathcal{P} = |\mathcal{I}^{\text{in}} \bigcup \mathcal{I}^{\text{both}}|/N$ and $\mathcal{S} = |\mathcal{I}^{\text{both}} \bigcup \mathcal{I}^{\text{out}}|/N$.

This perspective offers a direct and intuitive way to calculate $\mathcal{S}$: It is the probability that a vertex does not lead to the extensive component when the direction of every edges is reversed. This edge reversal is fully encoded in $\bar{Q}_{\alpha i}(\boldsymbol{l}|\boldsymbol{n}; \mathcal{R})$ computed with Eqs. (3) with incoming directed edges swapped into outgoing ones (and vice versa) and with edges that were more likely to exist in a given direction now more likely to exist in the opposite direction (i.e., p_{ij} becomes p_{ji}). From these probabilities, we define the pgfs $\bar{\theta}_{\alpha i}(\boldsymbol{x})$, $\bar{g}_i(\boldsymbol{x})$, and $\bar{f}_{\mu i}(\boldsymbol{x})$, which are analogous to the ones previously defined [$P_i(\boldsymbol{k})$ and $R(\boldsymbol{n})$ remain unchanged]. Defining $\bar{a}_{\mu i}$ as the probability that a vertex of type i reached by one of its stubs of type μ does not lead to the giant component in the *reversed* graph ensemble, self-consistency now requires

$$\bar{a}_{\mu i} = \bar{f}_{\mu i}(\bar{\boldsymbol{a}}) \qquad (13)$$

for every $\mu \in \mathcal{E}$ and $i \in \mathcal{N}$. As for Eqs. (9), the solution of this set of equations corresponds to the fixed point of the corresponding map and can therefore be obtained by successive iterations of any initial condition in $[0,1]^{|\mathcal{E}| \times |\mathcal{N}|}$. The elements of the Jacobian matrix of both Eqs. (9) and (13) are the average number of pairs, say (α, j), that are in the neighborhood of a pair, say (μ, i), in their respective graph ensemble. Since Eqs. (9) and (13) correspond to different perspectives of the same graph ensemble, the transcritical bifurcation occurs simultaneously in both systems [41].

Having obtained $\bar{\boldsymbol{a}}$ from Eqs. (13), the probability for a vertex of type i to be part of the giant component is $\mathcal{S}_i = 1 - \bar{g}_i(\bar{\boldsymbol{a}})$, and the relative size of the giant component is

$$\mathcal{S} = \sum_{i \in \mathcal{N}} \frac{r_i w_i \mathcal{S}_i}{\sum_{j \in \mathcal{N}} r_j w_j} = 1 - \sum_{i \in \mathcal{N}} \frac{r_i w_i \bar{g}_i(\bar{\boldsymbol{a}})}{\sum_{j \in \mathcal{N}} r_j w_j}. \qquad (14)$$

Clearly, when all hyperedges are symmetric (i.e., $p_{ij} = p_{ji}$ for every $i, j \in \mathcal{N}$), there is no global asymmetry in the graph ensemble, and $\mathcal{P} = \mathcal{S}$. Also, whenever Eqs. (5)–(14) are used in the context of site percolation—where vertices exist or are activated with a given set of probabilities—the value of $\mathcal{P}$ and $\mathcal{S}$ is relative to the number of vertices that exist. In other words, $\mathcal{P}$ is the probability that an existing vertex leads to an extensive component, and $\mathcal{S}$ is the probability that an existing vertex is part of it.

D. Small components

Substituting x_{vj} by z_j for every $j \in \mathcal{N}$ and $v \in \mathcal{E}$ in Eq. (7) yields a pgf that generates the number of vertices of each type that are *directly accessible* from a vertex of type i (i.e., vertices that are in its neighborhood). In other words, the information concerning the types of stubs is lost. Using self-consistency arguments similar to the one used in the previous section, it is possible to obtain a pgf that generates the distribution of the number of vertices of each type that will be *eventually reached* from a vertex of type i; the *reach* of this new pgf is no longer limited to the immediate neighborhood. In fact, this new pgf makes it possible to investigate the composition and the sizes of the components that contain a finite number of vertices. Let this new pgf be denoted $K(\boldsymbol{z})$.

To compute $K(\boldsymbol{z})$, we first consider the pgf $A_{\alpha i}(\boldsymbol{x})$ that generates the distribution of the number of all pairs of each type that will *eventually* be reached (i.e., not limited to the first neighbors) from a vertex of type i given that this vertex has been reached from one of its stubs of type α. In other words, this function generates the distribution of the number of the vertices that are eventually reached from a pair (α, i). Note that $A_{\alpha i}(\boldsymbol{x})$ is a function of $\boldsymbol{x}$ so that it keeps track of the type of the stubs from which each vertex has been reached. Besides yielding a treelike structure, the *stub matching* scheme used to generate graphs implies that the pgfs $\{f_{\mu i}(\boldsymbol{x})\}$ are invariant under translations on the graphs in the limit $N \to \infty$. In other words, while navigating on a graph from this ensemble, the number and the type of the vertices downstream from any given vertex does not depend on the types of the vertices (or the types of the stubs) previously encountered; navigating on graphs from this ensemble is a stationary Markov process (i.e., it only depends on the *current position* on the graph). Consequently, a vertex of type i reached from one of its stubs of type α and a pair (α, i) present in its neighborhood should both lead to a finite tree whose size and composition are identically distributed; this distribution is generated by $A_{\alpha i}(\boldsymbol{x})$. Considering every combination (α, i), this self-consistency requirement can be mathematically formulated as

$$A_{\alpha i}(\boldsymbol{x}) = x_{\alpha i} f_{\alpha i}[\boldsymbol{A}(\boldsymbol{x})], \qquad (15)$$

where the extra $x_{\alpha i}$ accounts for the vertex of type i that has been reached through one of its stubs of type α. Analogously to the set of probabilities $\{a_{\alpha i}\}$, the pgfs $\{A_{\alpha i}(\boldsymbol{x})\}$ are the fixed point of (with $n \in \mathbb{N}$)

$$\boldsymbol{A}^{(n+1)}(\boldsymbol{x}) = \boldsymbol{x} \circ \boldsymbol{f}[\boldsymbol{A}^{(n)}(\boldsymbol{x})], \qquad (16)$$

where "$\circ$" denotes the entrywise product and $\boldsymbol{f}(\boldsymbol{x})$ is the same map as in Eq. (10). It is, in fact, straightforward to show that the extra $\boldsymbol{x}$ guarantees that the distributions generated by $\boldsymbol{A}(\boldsymbol{x})$

can be obtained for components of n vertices or less in $n + 1$ iterations of Eq. (16) from the initial condition $\boldsymbol{A}^{(0)}(\boldsymbol{x}) = \boldsymbol{1}$ [i.e., $A_{\alpha i}^{(0)}(\boldsymbol{x}) = 1$ for every $i \in \mathcal{N}$ and $\alpha \in \mathcal{E}$].

Having obtained $\boldsymbol{A}(\boldsymbol{x})$ up to a sufficient size of components, n, the number of vertices of each type that can be reached in a finite component from a randomly chosen vertex of type i is generated by $K_i(\boldsymbol{z}) \equiv z_i g_i[\boldsymbol{A}(\boldsymbol{z})]$. The pgf generating the number of vertices of each type that are accessible in a small component from a randomly chosen (existing) vertex is

$$K(\boldsymbol{z}) = \sum_{i \in \mathcal{N}} \frac{r_i w_i K_i(\boldsymbol{z})}{\sum_{j \in \mathcal{N}} r_j w_j} = \sum_{i \in \mathcal{N}} \frac{r_i w_i z_i g_i[\boldsymbol{A}(\boldsymbol{z})]}{\sum_{j \in \mathcal{N}} r_j w_j}. \quad (17)$$

It is worth mentioning that the distributions generated by $K(\boldsymbol{z})$ and $\{A_{\alpha i}(\boldsymbol{z})\}$ are not normalized in the presence of an extensive component as there is a nonzero probability that a pair (α, i) leads to the giant component. In fact, comparing Eqs. (10) and (16) leads to the conclusion that $A_{\alpha i}(\boldsymbol{1}) = a_{\alpha i}$ and that $K(\boldsymbol{1}) = 1 - \mathcal{P}$.

IV. SPECIAL CASES AND APPLICATIONS

To demonstrate the versatility and the flexibility of the formalism, we present a series of representative examples. This will also clarify the conceptual and numerical steps necessary to implement such a general approach.

A. Semidirected random graphs

Semidirected random graphs are composed of indistinguishable vertices connected via undirected and directed edges. They were used in Ref. [20] to study the impact of nonreciprocal connections in contact networks on the propagation of an emerging infectious disease. These nonreciprocal connections accounted for the susceptibility of health-care workers to get infected from infectious individuals seeking treatments in hospitals. Semidirected is also a good first example for it has the well-known undirected graphs and directed graphs as special cases.

Every vertex in these graphs belongs to the same type ($|\mathcal{N}| = 1$, type 1, $w_1 = 1$), and there are $|\mathcal{E}| = 3$ types of stubs: Stubs of type A are paired together to form undirected edges, and stubs of types B (outgoing) and C (incoming) are paired to form a directed edge. The joint degree distribution $P_1(\boldsymbol{k}) = P_1(k_A, k_B, k_C)$ corresponds to the distribution of *undirected* degree, *out*-degree and *in*-degree. In this scenario, the conditions given by Eq. (4) imply that there must be as much incoming stubs as there are outgoing stubs, $\langle k_B \rangle_{P_1} = \langle k_C \rangle_{P_1}$, and they fix the values of $R(\boldsymbol{n}) = R(n_{A1}, n_{B1}, n_{C1})$ in terms of the average degrees, $R(2,0,0) = 1 - R(0,1,1) = \langle k_A \rangle_{P_1} / (\langle k_A \rangle_{P_1} + 2\langle k_B \rangle_{P_1})$. Assuming that edges exist with probability p_{11} and vertices exist with probability r_1, we find from Eqs. (3) and (6)

$$\theta_{A1}(\boldsymbol{x}) = (1 - r_1 p_{11}) + r_1 p_{11} x_{A1}, \quad (18a)$$

$$\theta_{B1}(\boldsymbol{x}) = (1 - r_1 p_{11}) + r_1 p_{11} x_{C1}, \quad (18b)$$

$$\theta_{C1}(\boldsymbol{x}) = 1, \quad (18c)$$

from which we define the pgfs $g_1(\boldsymbol{x})$, $f_{A1}(\boldsymbol{x})$, and $f_{C1}(\boldsymbol{x})$ from Eqs. (7) and (8). Note that $f_{B1}(\boldsymbol{x})$ does not exist as vertices cannot be reached by an outgoing stub. Similarly,

when reversing the direction of edges (directed edges now run from C stubs to B stubs), we obtain

$$\bar{\theta}_{A1}(\boldsymbol{x}) = (1 - r_1 p_{11}) + r_1 p_{11} x_{A1}, \quad (19a)$$

$$\bar{\theta}_{B1}(\boldsymbol{x}) = 1, \quad (19b)$$

$$\bar{\theta}_{C1}(\boldsymbol{x}) = (1 - r_1 p_{11}) + r_1 p_{11} x_{B1}, \quad (19c)$$

which yield the pgfs $\bar{g}_1(\boldsymbol{x})$, $\bar{f}_{A1}(\boldsymbol{x})$, and $\bar{f}_{B1}(\boldsymbol{x})$ [$\bar{f}_{C1}(\boldsymbol{x})$ is nondefined]. Using Eqs. (18) and (19) in Eqs. (7)–(17) with $r_1 = 1$ yields the results obtained in Ref. [20] and the ones obtained for purely directed [9] or purely undirected random graphs [42] in the appropriate limits.

B. Correlated random graphs

Other interesting special cases of our model are correlated random graphs: Graphs where vertices are more likely to be connected with vertices having specific intrinsic properties (e.g., degree, centrality, ethnicity, age group, gender). In such cases, there are $|\mathcal{N}|$ types of vertices, one for each intrinsic property, and there are as many types of stubs: Each type of stub corresponds to the type of the vertex that is at the other end of the edge. To simplify the notation, types of stubs will be identified by the type of the vertex toward which they point (i.e., $\mathcal{E} = \mathcal{N}$). Hence, the joint degree distribution $P_i(\boldsymbol{k})$ prescribes the number of vertices of each type that vertices of type i are connected to. The conditions (4) ask that there are as many stubs stemming from vertices of type i toward vertices of type j as in the reverse direction, $w_i \langle k_j \rangle_{P_i} = w_j \langle k_i \rangle_{P_j}$. These constraints also prescribe the distribution

$$R(\boldsymbol{n}) = \frac{1}{R'} \sum_{i,j \in \mathcal{N}} (1 - \delta_{0,n_{ji}}) w_i \langle k_j \rangle_{P_i}, \quad (20)$$

where $R' = \sum_{i',j' \in \mathcal{N}} w_{i'} \langle k_{j'} \rangle_{P_{j'}}$ is simply the normalization factor. Assuming that vertices of type i exist with probability r_i and that edges going from a vertex of type i to a vertex of type j exist with probability p_{ij} (i.e., edges may be more likely to exist in one direction than in the other), we get from Eqs. (3) and (6)

$$\theta_{ji}(\boldsymbol{x}) = (1 - r_j p_{ij}) + r_j p_{ij} x_{ij}, \quad (21a)$$

$$\bar{\theta}_{ji}(\boldsymbol{x}) = (1 - r_j p_{ji}) + r_j p_{ji} x_{ij}, \quad (21b)$$

for $i, j \in \mathcal{N}$. Using Eqs. (21) in Eqs. (7)–(17) and setting every $r_i = 1$ yields the results obtained in Ref. [11] for *multitype* graphs, which are themselves a generalization of several other formalisms [9,23,42]. We have also used this approach in Ref. [3] to study the observability of random graphs and in Ref. [16] to define an ensemble of graphs with an arbitrary k-core structure.

C. Degree-correlated random graphs

An important category of correlations is the one based on the degree of vertices [22,30]. These correlations are encoded in the conditional probability $P(d'|d)$ corresponding to the probability that the neighbor of a vertex with a degree d has a degree equal to d'. This can be reproduced with our formalism by considering that every vertex with the same degree are of the same type (i.e., a vertex of type i has i neighbors, and,

consequently, $\{w_i\}$ corresponds to the degree distribution) and by using the type-specific joint degree distribution

$$P_i(\mathbf{k}) = \frac{i!}{\prod_{j' \in \mathcal{N}} k_{j'}!} \prod_{j \in \mathcal{N}} [P(j|i)]^{k_j}. \qquad (22)$$

From Eq. (7), we obtain

$$g_i(\mathbf{x}) = \left[\sum_{j \in \mathcal{N}} P(j|i)\theta_{ji}(\mathbf{x}) \right]^i, \qquad (23)$$

where $\theta_{ji}(\mathbf{x})$ is given by Eq. (21a), and Eq. (8) yields

$$f_{li}(\mathbf{x}) = \left[\sum_{j \in \mathcal{N}} P(j|i)\theta_{ji}(\mathbf{x}) \right]^{i-1}, \qquad (24)$$

which is independent of the type of the vertex/stub, namely l, from which the vertex has been reached. This is a direct consequence of the multinomial distribution in Eq. (22) and shows that our approach, through the joint distribution $P_i(\mathbf{k})$, can include more detailed correlations in the degree of the neighbors of vertices.

It may be useful at this point to illustrate the precise connection with previous works. Consider the quantity $u_i \equiv \sum_{j \in \mathcal{N}} P(j|i)[(1 - r_j p_{ji}) + r_j p_{ji}\bar{a}_{ij}]$, which under successive application of Eqs. (13), (21b), and (24) becomes the self-consistent expression

$$u_i = \sum_{j \in \mathcal{N}} P(j|i)\big[(1 - r_j p_{ji}) + r_j p_{ji} u_j^{j-1}\big] \qquad (25)$$

for every $i \in \mathcal{N}$. Setting every $r_j p_{ji} = 1 - f$, with $0 \leqslant f \leqslant 1$, in this last equation yields Eqs. (5) and (13) of Ref. [30], while Eq. (8) of Ref. [22] is obtained by setting $r_j p_{ji} = 1$. Similarly, replacing $P(j|i)$ with $jw_j / \sum_{l \in \mathcal{N}} lw_l$ and assuming every $u_i = u$ in Eq. (25) yields the results of Ref. [8]. More precisely, setting every $p_{ji} = 1$ makes it possible to retrieve their Eq. (15), and their Eq. (8) is obtained by setting every $r_j p_{ji} = q_s q_b$, with $0 \leqslant q_s, q_b \leqslant 1$. Expressions for the size of the extensive component derived in Refs. [8,22,30] can be obtained from our formalism similarly.

D. Clustered random graphs

We now show how many variants of the CM containing clustered hyperedges (i.e., hyperedges that contain loops) are special cases of the approach presented in this paper.

Since the clustering property is related to the number of triangles found in graphs—hence capturing the idea that *the friend of my friend is also my friend*—it is natural to introduce clustering in graphs through the use of triangles (i.e., three vertices all connected together) [21,25,27]. The simplest clustered graph ensemble then has $|\mathcal{N}| = 1$ types of vertices (type 1, $w_1 = 1$) and $|\mathcal{E}| = 2$ types of stubs: Two stubs of type A are paired to form undirected edges and three stubs of type B are matched to create triangles. Note that only one stub of type B is required to belong to a triangle even though its contribution amounts to two to the degree of the vertex; stubs can be seen as a membership to a hyperedge. The constraints given by Eq. (4) *de facto* set the values of $R(\mathbf{n}) = R(n_{A1}, n_{B1})$ since $R(2,0) = 1 - R(0,3) = 3\langle k_A \rangle_{P_1} /$

$(3\langle k_A \rangle_{P_1} + 2\langle k_B \rangle_{P_1})$. Assuming that vertices and edges exist with probabilities r_1 and p_{11}, we obtain from Eqs. (3) and (6)

$$\theta_{A1}(\mathbf{x}) = (1 - r_1 p_{11}) + r_1 p_{11} x_{A1}, \qquad (26a)$$

$$\theta_{B1}(\mathbf{x}) = (1 - r_1 p_{11})^2 + 2r_1 p_{11}[1 - r_1 p_{11}(2 - p_{11})]x_{B1}$$
$$+ r_1^2 p_{11}^2 [3 - 2p_{11}]x_{B1}^2. \qquad (26b)$$

Using these two functions in Eqs. (7)—(17) leads directly to the results obtained in Refs. [21,25,28]. Similarly, the results of Ref. [13] can be obtained with three types of vertices, $\mathcal{N} = \{1,2,3\}$, and one type of stubs, $\mathcal{E} = \{A\}$, where all hyperedges are triangles containing one vertex of each type [$R(1,1,1) = 1$ and $\theta_{Ai}(\mathbf{x}) = x_{A1} x_{A2} x_{A3} / x_{Ai}$].

Besides triangles, clustering—or any digression from a perfect treelike structure—has been introduced in random graphs through the inclusion of various categories of hyperedges that involve more than three vertices. For instance, in Ref. [14,15,24] clustering is incorporated through fully connected hyperedges, or *cliques*, where vertices or edges exist with given probabilities (i.e., Erdős-Rényi graphs). In all cases, there is only one type of vertex. We retrieve the model of Ref. [24] by using one type of stubs; $P_1(k_A)$ prescribes the number of cliques to which vertices belong, and $R(n_{A1})$ prescribes the size of cliques (respectively the distributions r_m and s_n in Ref. [24]). In the model considered in Refs. [14,15], vertices belong to only one clique, but can have many single edges. Since the number of single edges and the size of the clique can be correlated in the original model, there is one type of stub for each clique size and an additional type for single edges; cliques of size m are formed by matching m stubs of the type assigned to cliques of size m. Hence, the structure of the graphs is fully prescribed by $P_1(\mathbf{k})$ whose argument indicates the number of single edges and the size of the clique. The constraints (4) then yield

$$R(\mathbf{n}) = \frac{1}{R''} \sum_{\beta \in \mathcal{E}} \left(1 - \delta_{0,n_{\beta 1}}\right) \frac{\langle k_\beta \rangle_{P_1}}{n_{\beta 1}}, \qquad (27)$$

where $R'' = \sum_{\beta \in \mathcal{E}} \langle k_\beta \rangle_{P_1} / n_{\beta 1}$ is the normalization constant. Using these distributions and quantities, our model reproduces the ones presented in Refs. [14,15]. Also, we have used a version of our model that is similar to the one introduced in Ref. [15] to uncover a transition in the effectiveness of immunization strategies [5].

Finally, two of the most versatile models published to date are also special cases of our model. Reference [10] is a previous version of the model presented in this paper. The two main differences are that the previous version did not handle site percolation, and that only stubs of same type could be matched to create hyperedges (e.g., forbidding directed edge between vertices of a same type). The model introduced in Ref. [17] can be retrieved from our model with one type of vertex ($|\mathcal{N}| = 1$) and with one type of stubs for each *role* that a vertex can play in hyperedges. However, this approach lacks the systematic method offered by Eqs. (3) to solve percolation on each hyperedge beforehand, thereby limiting the number of hyperedges that can effectively be handled analytically.

E. Weak and strong clustering regimes

We now use our model to test a conjecture regarding the effect of clustering (e.g., triangles) on bond percolation. References [27,43] proposed that clustering has opposite effects on the bond percolation threshold depending of the density of triangles in a graph. This density is measured through the degree dependent clustering coefficient $\bar{c}(k)$: The probability that two neighbors of a vertex of degree k are also neighbors (i.e., they complete the triangle). The conjecture states that the *weak* clustering regime $\bar{c}(k) < (1-k)^{-1}$ leads to a higher percolation threshold than in an equivalent unclustered graph. Contrariwise, *strong* clustering, defined as $\bar{c}(k) > (1-k)^{-1}$, leads to a lower percolation threshold than in an equivalent unclustered graph.

Let us consider the following graph ensemble in which there are two types of vertices $\mathcal{N} = \{1,2\}$ and three types of edges $\mathcal{E} = \{A,B,C\}$. Every vertex of type 1 has one stub of type A and one stub of type B, while each vertex of type 2 has one stub of type B and one stub of type C. In other words, we set $P_i(\boldsymbol{k}) = P_i(k_A, k_B, k_C)$ as $P_1(1,1,0) = P_2(0,1,1) = 1.0$. Hyperedges are formed by matching either four stubs of type A, four stubs of type B (two stemming from vertices of type 1 and two from vertices of type 2), or eight stubs of type C; vertices are all connected to one another in every hyperedge. The constraints given by Eq. (4) imply that $w_1 = w_2$ and that $R(\boldsymbol{n}) = R(n_{A1}, n_{B1}, n_{C1}, n_{A2}, n_{B2}, n_{C2})$ follows the relation $2R(4,0,0,0,0,0) = R(0,2,0,0,2,0) = 4R(0,0,0,0,0,8) = 4/7$. Vertices of type 1 all have a degree equal to 6, and vertices of type 2 all have a degree equal to 10. Consequently, we see that $\bar{c}(6) = \frac{6}{15} > \frac{1}{5}$ and $\bar{c}(10) = \frac{24}{45} > \frac{1}{9}$, which implies that this graph ensemble qualifies for the strong regime.

To isolate the effect of clustering on bond percolation, we compare the results obtained for the graph ensemble described above with the ones obtained with an equivalent unclustered version [21]. This equivalent graph ensemble possesses identical correlations, but hyperedges are broken into individual independent edges instead (e.g., each vertex in an hyperedge containing n vertices now have $n-1$ independent edges). The behavior of its giant component is obtained as in Sec. IV B with $P_1(4,2,0) = P_2(0,2,8) = 1$.

Figure 4 compares the behavior of the giant component in both ensembles when edges exist with probability p_{11}. We conclude that although the clustered graph ensemble qualifies for the strong regime, the behavior observed is the one of the weak regime: Higher percolation threshold than for the equivalent unclustered graph. This behavior can be understood in terms of branching factors. The unclustered graphs have a treelike structure and therefore maximize the number of *new* vertices encountered while navigating the graph: Every edge leads to a new vertex. The redundancy caused by clustering means that not all edges lead to a new vertex in the clustered graphs, which reduces the average number of vertices that can be reached from any given vertex. Hence, a larger number of edges must be present for a giant component to appear (e.g., larger threshold) since many edges will be *wasted* by leading to vertices previously reached.

This counterexample suggests that the criterion on $\bar{c}(k)$ could be a necessary condition for a strong clustering regime but that it is not a sufficient one. The explanation in terms
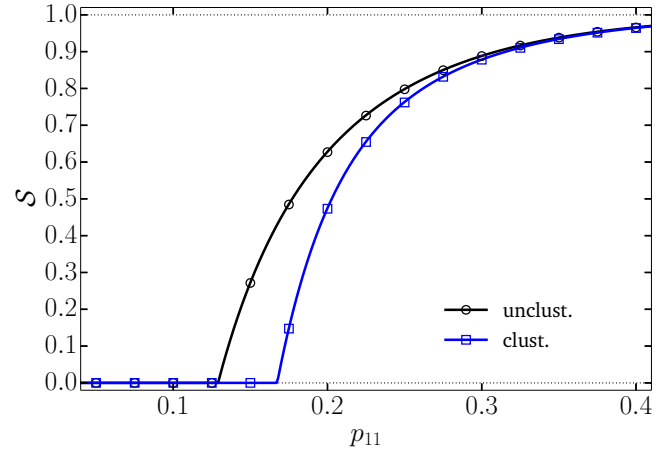

FIG. 4. (Color online) Comparison of the emergence of the giant component in a clustered graph ensemble that qualifies for the strong clustering regime with its unclustered counterpart. We see that the latter has a lower percolation threshold. Details on the graphs used are given in Sec. IV E.

of branching factors alongside the results in Refs. [21,44,45] point toward the conclusion that the effect of clustering on random graphs with an *underlying* treelike structure is best described by the weak clustering regime. Indeed, Ref. [46] has recently shown that strong clustering may induce a double continuous phase transition which conciliates the conjectured antagonistic effects of weak and strong clustering and whose effects are in line with our conclusion.

F. Bijection between site and bond percolation thresholds

From Eqs. (8), we see that the elements of the Jacobian matrix $\mathbf{J}$ used to determine the point at which the giant component appears have the general form

$$\frac{\partial f_{\mu i}(\mathbf{1})}{\partial x_{\nu j}} = \sum_{\alpha \in \mathcal{E}} \frac{\langle k_\mu (k_\alpha - \delta_{\mu\alpha}) \rangle_{P_i}}{\langle k_\mu \rangle_{P_i}} \frac{\partial \theta_{\alpha i}(\mathbf{1})}{\partial x_{\nu j}} \qquad (28)$$

for every $i,j \in \mathcal{N}$ and $\mu, \nu \in \mathcal{E}$. These terms are, in fact, branching factors: Each element is the average number of pairs (ν, j) that are present in the neighborhood of a pair (μ, i). More precisely, the first term corresponds to the average number of stubs of type α that a vertex of type i has if it has been reached from one of its stubs of type μ (this stub is excluded from the count if $\alpha = \mu$). The second term is the average number of pairs (ν, j) that can be reached in hyperedges accessed via a stub of type α of a vertex of type i. The value of these latter terms depends on the structure of hyperedges (i.e., rules $\mathcal{R}$) and on the probabilities for vertices and edges to exist (i.e., $\{r_j\}_{j \in \mathcal{N}}$ and $\{p_{jk}\}_{j,k \in \mathcal{N}}$).

Let us assume that all hyperedges have the same structure; vertices of different types may be involved in a nontrivial manner as long as all hyperedges have the same shape (e.g., they all are triangles). We also suppose that vertices and edges exist with probabilities that are independent of their type, that is $r_i = r$ and $p_{ij} = p$ for all $i,j \in \mathcal{N}$. In such case, every nonzero element of the Jacobian matrix, $\frac{\partial \theta_{i\alpha}(\mathbf{1})}{\partial x_{\nu j}}$, is a polynomial in r and p, $h(r,p)$, and is independent of i, j, α, and ν. Consequently,

the dependency in r and p can be factored out of the Jacobian matrix

$$\mathbf{J} = h(r,p)\mathbf{J}'. \qquad (29)$$

Since the giant component appears when $\lambda_{\max}(\mathbf{J}) = h(r,p)\lambda_{\max}(\mathbf{J}') = 1$, the points (r',p') at which the phase transition occurs all belong to the critical surface

$$h(r',p') = \frac{1}{\lambda_{\max}(\mathbf{J}')}. \qquad (30)$$

Whenever the Jacobian matrix can be written like in Eq. (29), any given point (r_1,p_1) at which a graph ensemble is known to percolate can be related to any other critical point (r_2,p_2) through $h(r_1,p_1) = h(r_2,p_2)$. For instance, this relation leads to a direct bijection between the thresholds of pure site percolation $(r_c,1)$ and pure bond percolation $(1,p_c)$ through $h(r_c,1) = h(1,p_c)$. Additionally, $h(r,p) = rp$ for unclustered correlated random graphs, and the fact that r and p only appear as rp in Eqs. (21) implies that site and bond percolation are equivalent for this random graph ensemble.

V. INTERDEPENDENT RANDOM GRAPHS

In this last section, we briefly show how our approach can be adapted to model interdependent graphs through a redefinition of Eqs. (9)–(14) and use the resulting formalism to investigate the emergence of an extensive component on interdependent clustered random graphs.

To lighten the description, we consider the case of two interdependent graphs, graph A and graph B, without loss of generality (guidelines for a straightforward generalization to an arbitrary number of graphs are given in Ref. [47]). We assume that every edge in each graph is undirected such that there is no global asymmetry: The probability that a randomly chosen vertex leads to the extensive component is equal to the relative size of the extensive component (i.e., $\mathcal{P} = \mathcal{S}$). Furthermore, we consider that the pgfs $g_i(\boldsymbol{x})$ and $f_{\mu i}(\boldsymbol{x})$ and all other related quantities defined in the previous sections (i.e., $\{w_i\}$, $\mathcal{R}$, $\mathcal{N}$, $\{r_i\}$, ...) are known for both graphs and are identified with the superscript A or B. Both graphs contain the same number of vertices which tends to infinity: $N^A = N^B = N \to \infty$.

The change in the nature of the transition (i.e., from continuous to discontinuous) originates from the existence of *dependencies* between vertices of the two graphs. Again, to lighten the description, we consider the case in which each vertex has either one *twin vertex* on which it depends or none. To specify the dependencies between vertices, we define q_{iv}^{AB} as the probability that a vertex of type i in graph A has a twin vertex of type v in graph B. Note that allowing graphs to be partially dependent—not all vertices have a twin vertex—implies that

$$\sum_{v \in \mathcal{N}^B} q_{iv}^{AB} \equiv 1 - \bar{q}_i^{AB} \leqslant 1, \qquad (31)$$

for each $i \in \mathcal{N}^A$, and where the sum is over the types of the vertices in graph B. Therefore, a fraction $\bar{q}_i^{AB}$ of the vertices of type i in graph A do not have a twin vertex in graph B. A similar set of probabilities, $\{q_{ju}^{BA}\}_{j,u}$ with $j \in \mathcal{N}^B$ and $u \in \mathcal{N}^A$, is defined to specify the dependencies of vertices in graph B.

Moreover, we add the constraint that the dependency between two vertices must be reciprocal unless a vertex's twin has no dependency whatsoever. In other words, if vertex n^A in graph A depends on vertex n^B in graph B, then either vertex n^B depends on vertex n^A as well or it depends on no vertex at all. In the latter case, vertex n^A is the only vertex in graph A that can depend on vertex n^B. This constrains the two probability sets, $\{q_{iv}^{AB}\}_{i,v}$ and $\{q_{ju}^{BA}\}_{j,u}$, as there must be enough "independent" vertices in graph A (graph B) to account for the vertices in graph B (graph A) whose dependency is not reciprocal. Mathematically, these conditions can be written as

$$\sum_{i \in \mathcal{N}^A} \max\{N^A w_i^A q_{iv}^{AB} - N^B w_v^B q_{vi}^{BA}, 0\} \leqslant N^B w_v^B \bar{q}_v^{BA} \qquad (32)$$

for each $u \in \mathcal{N}^A$ and $v \in \mathcal{N}^B$. A similar expression in which the superscripts A and B are swapped must also hold. In the expression above, $N^A w_i^A q_{iv}^{AB}$ corresponds to the number of vertices of type i in graph A that depend on a vertex of type v in graph B, and $N^B w_v^B \bar{q}_v^{BA}$ is the number of vertices of type v in graph B that have no dependency.

A discontinuous phase transition is associated with the emergence of an extensive *functional* component: An extensive component composed solely of vertices with no dependency or whose twin vertex is part of the extensive functional component in its respective graph. To compute the size of the extensive functional component, we define $a_{\mu i}^A$ as the probability that a vertex of type i reached from one of its stubs of type μ in graph A *does not* lead to the functional extensive component in graph A. Similar probabilities are defined for the other types of vertices and stubs in graph A and in graph B. Following the locally treelike structure argument of Sec. III C, we now derive a set of self-consistent equations similar to Eqs. (9) for these probabilities.

Let us consider the case of a vertex of type i in graph A reached via one of its stubs of type μ. By definition, this vertex belongs to the extensive functional component in graph A with probability $1 - a_{\mu i}^A$. Only two scenarios can lead to this situation. The first one consists of the vertex being part of the extensive functional component in graph A and having no twin vertex. This happens with probability $[1 - f_{\mu i}^A(\boldsymbol{a}^A)]\bar{q}_i^{AB}$. The second scenario consists of the vertex being part of the extensive functional component in graph A and having a twin vertex that is part of the extensive functional component in its own graph. This scenario happens with probability

$$\left[1 - f_{\mu i}^A(\boldsymbol{a}^A)\right] \sum_{v \in \mathcal{N}^B} q_{iv}^{AB} r_v^B \left[1 - g_v^B(\boldsymbol{a}^B)\right], \qquad (33)$$

where q_{iv}^{AB} is the probability that the twin vertex is of type v, r_v^B is the probability that it exists (i.e., it has not been removed), and $1 - g_v^B(\boldsymbol{a}^B)$ is the probability that it belongs to the extensive functional component in graph B. Summing these two scenarios yields

$$a_{\mu i}^A = 1 - \left[1 - f_{\mu i}^A(\boldsymbol{a}^A)\right]\left\{\bar{q}_i^{AB} + \sum_{v \in \mathcal{N}^B} q_{iv}^{AB} r_v^B \left[1 - g_v^B(\boldsymbol{a}^B)\right]\right\}, \qquad (34)$$

which must hold for every $i \in \mathcal{N}^A$ and $\mu \in \mathcal{E}^A$, as well as for graph B (i.e., simply swap the superscripts A and B in the

last expression). Having solved Eqs. (34) for the probabilities $\boldsymbol{a}^A \equiv \{a_{\mu i}^A\}$ and $\boldsymbol{a}^B \equiv \{a_{\nu j}^B\}$, the probability that a randomly chosen vertex of type i in graph A belongs to the functional component is

$$\mathcal{S}_i^A = \left[1 - g_i^A(\boldsymbol{a}^A)\right]\left\{\bar{q}_i^{AB} + \sum_{v \in \mathcal{N}^B} q_{iv}^{AB} r_v^B \left[1 - g_v^B(\boldsymbol{a}^B)\right]\right\},$$
(35)

which is similar to the calculation of $\mathcal{S}_i$ in Sec. III C, but in this case the probability that a vertex of type i belongs to the functional extensive component, $1 - g_i^A(\boldsymbol{a}^A)$, is weighted by the probability that its twin vertex, if any, belongs to the extensive functional component as well. Averaging over the fraction of existing vertices of each type (e.g., a fraction $r_i^A w_i^A$ of vertices in graph A corresponds to vertices of type i that have not been removed), we finally obtain the size of the extensive functional component in graph A,

$$\mathcal{S}^A = \sum_{i \in \mathcal{N}^A} \frac{r_i^A w_i^A}{\sum_{j \in \mathcal{N}^A} r_j^A w_j^A} \mathcal{S}_i^A.$$
(36)

Similar equations for vertices in graph B are obtained by swapping the superscripts A and B in the last two expressions. As for the quantities $\mathcal{P}$ and $\mathcal{S}$ defined previously, the fraction $\mathcal{S}^A$ (and $\mathcal{S}^B$) is relative to the number of existing vertices (i.e., vertices that have not been removed). Note also that Eqs. (9) are retrieved from Eqs. (34) if there are no dependencies. However, contrariwise to Eqs. (9) and Eqs. (13), the right-hand side of Eqs. (34) does not necessarily correspond to monotonously increasing functions (some coefficients in the polynomials are negative). This implies that, although the point $\boldsymbol{a}^A \oplus \boldsymbol{a}^B = \boldsymbol{1}$ is still a solution, another solution in the hypercube $[0,1]^{|\mathcal{N}^A| \times |\mathcal{E}^A| + |\mathcal{N}^B| \times |\mathcal{E}^B|}$ corresponding to the presence of an extensive functional component may not appear continuously from $\boldsymbol{a}^A \oplus \boldsymbol{a}^B = \boldsymbol{1}$ through a transcritical bifurcation as in Sec. III C. Hence, the values of $\mathcal{S}^A$ and $\mathcal{S}^B$ *jump* abruptly from zero to a finite value in [0,1] which corresponds to a discontinuous phase transition.

To illustrate this behavior, we investigate the emergence of extensive components on interdependent clustered random graphs. To do so, we consider the edge-triangle clustered graph ensemble presented in Sec. IV D with the joint degree distribution $P_1(0,3) = P_1(2,1) = 2P_1(2,0) = 4/10$ and $p_{11} = 1.0$. Notice that this joint degree distribution forces assortative mixing since high and low degree vertices tend to be segregated. The size of the extensive component in this isolated random graph ensemble is given as a function of the vertex existence probability r_1 in Fig. 5 (black curve labeled $\mathcal{S}_{(c,i)}$).

To illustrate the impact of interdependence on the phase transition, we consider the case of two identical partially dependent edge-triangle clustered graphs with $q_{11}^{AB} = 0.6$ and $q_{11}^{BA} = 1.0$. In other words, only 60% of the vertices in graph A depend on a vertex in graph B, whereas every vertex in graph B has a twin vertex. The green curves labeled $\mathcal{S}_{(c,d)}^A$ and $\mathcal{S}_{(c,d)}^B$ in Fig. 5 show the size of the extensive functional component in graph A and graph B as a function of r_1. We see that the extensive functional components indeed emerge through a discontinuous phase transition, unlike the
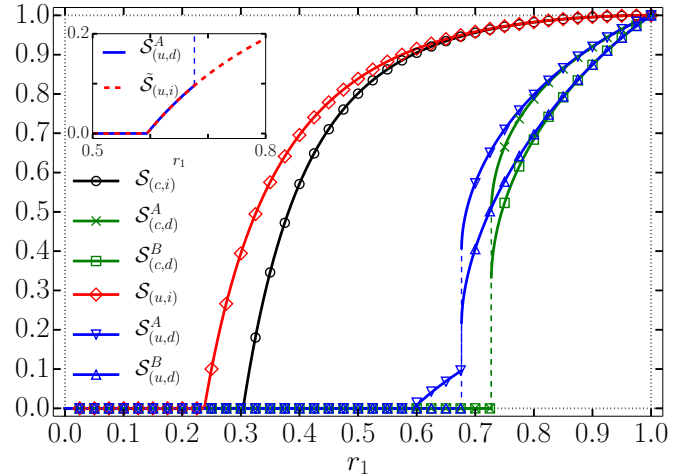

FIG. 5. (Color online) Comparison of the size of the extensive (functional) components as a function of the vertex existence probability r_1 in four related graph ensembles. The details of each graph ensemble are given in the main text of Sec. V. The curves $\mathcal{S}_{(c,i)}$ and $\mathcal{S}_{(u,i)}$ were obtained with Eq. (14) and the curves $\mathcal{S}_{(c,d)}^A$, $\mathcal{S}_{(c,d)}^B$, $\mathcal{S}_{(u,d)}^A$, and $\mathcal{S}_{(u,d)}^B$ were obtained with Eq. (36). Symbols show the results of numerical simulations on these graph ensembles with $N = 10^6$ vertices. (Inset) Comparison between the curve $\mathcal{S}_{(u,d)}^A$ and the curve $\mathcal{S}_{(u,i)}$ rescaled according to $\tilde{\mathcal{S}}_{(u,i)}[r_1] = \bar{q}_1^{AB} \mathcal{S}_{(u,i)}[r_1 \bar{q}_1^{AB}]$ (the dependence to r_1 in shown in brackets).

extensive component in the isolated clustered graphs that emerges continuously. We also see that the extensive functional component in graph A is always bigger than the one on graph B since vertices with no dependency are more likely to be in the extensive functional component than vertices with a dependency. Indeed, a vertex in graph A that has no dependency belongs to the extensive functional component with probability $[1 - g_1^A(\boldsymbol{a}^A)]$, which is clearly greater than $[1 - g_1^A(\boldsymbol{a}^A)]r_1^B[1 - g_1^B(\boldsymbol{a}^B)]$ for a vertex with a dependency since $0 \leqslant r_1^B[1 - g_1^B(\boldsymbol{a}^B)] \leqslant 1$ (the second equality holds only when there is no extensive component and $r_1^B = 1$). Figure 5 also shows that the size of the extensive functional component in the interdependent graphs is bounded by the size of the giant component in the corresponding isolated graphs. Again, this is expected since being part of the extensive component is a *sine qua non* condition for being part of the extensive functional component.

As in Sec. IV E, we consider the unclustered version of the edge-triangle random graph ensemble—in which triangles are broken into two independent single edges—to isolate the impact of clustering. In order to preserve the correlations present in the clustered dependent graph ensemble, two types of stubs are used to distinguish the original single edges from the single edges due to the broken triangles. As expected, the extensive component in this isolated unclustered graph ensemble appears at a lower value of r_1 (red curve labeled $\mathcal{S}_{(u,i)}$ in Fig. 5) than for the isolated clustered random graph ensemble (see Sec. IV E). The same conclusion holds for the extensive functional component in the unclustered version of the two partially dependent graphs described in the last paragraph (blue curves labeled $\mathcal{S}_{(u,d)}^A$ and $\mathcal{S}_{(u,d)}^B$ in Fig. 5). Comparing the size of the extensive functional component in

the clustered and unclustered versions of these interdependent graphs also suggests that clustering increases the jump size at the transition.

One interesting observation from Fig. 5 is that graph A in the interdependent unclustered graph ensemble (curve labeled $\mathcal{S}^A_{(u,d)}$) successively undergoes two phase transitions: A continuous then a discontinuous one. While the discontinuous transition is caused by the interdependency with graph B, the continuous one is due to the fact that the 40% of vertices in graph A that have no dependency are able to form an extensive component before the discontinuous phase transition occurs. From Fig. 5, we see that the continuous phase transition happens at $r_1 \simeq 0.23$ in the isolated unclustered graph ensemble, while the phase transitions occur at $r_1 \simeq 0.58$ (continuous) and at $r_1 \simeq 0.68$ (discontinuous) in the interdependent unclustered graph ensemble. Below $r_1 \simeq 0.68$, the vertices in graph A that depend on a vertex in graph B can be effectively considered as removed since there is no extensive functional component in graph B (i.e., they cannot be part of an extensive component). Hence, we expect graph A to behave as its isolated graph ensemble counterpart in which an effective fraction $1 - r_1 \bar{q}_1^{AB}$ of its vertices has been removed. This is indeed confirmed in the inset of Fig. 5. In fact, successive phase transitions should occur whenever independent vertices are able to form an extensive component *before* the extensive functional component emerges. In other words, we observe a double phase transition whenever the rescaled value at which the continuous phase transition happens in the isolated graph ensemble is below the value at which the discontinuous phase transition occurs in the interdependent graph ensemble.

VI. CONCLUSION

Building upon our previous works [5,10,11,16,36], we have presented a unifying conceptual framework that offers a comprehensive mathematical description of a wide variety of structural properties found in graphs extracted from real complex systems (e.g., correlations, segregation, clustering of various forms). The generality of the formalism resides on a multitype perspective for a precise prescription on how vertices are connected to one another and on a set of iterative equations for the solution of the distribution of the size of components in small arbitrary graphs. Interestingly, these iterative equations are by themselves a valuable addition to graph theoretical methodology. In fact, besides being a cornerstone of our formalism, allowing a mapping of hyperedges onto an effective

treelike structure, they also have potential applications in the theoretical description of fragmentation processes and of percolation on lattices [48–51] (see Ref. [36] for further details).

Our approach leads to the definition of a very general random graph ensemble for which site and/or bond percolation can be solved exactly using pgfs in the infinite-size limit (e.g., size of the giant component, percolation threshold, distribution of the size of small components). We have shown that this random graph ensemble encompasses most random graph models published until now and can incorporate structural properties not yet included in a theoretical framework. This versatility makes it a perfect theoretical laboratory to investigate the role of specific local structural properties on the global connectivity of the graphs. We have illustrated this point by implementing our method to provide a counterexample to a conjecture [27] on the effect of clustering on the percolation threshold.

Our formalism is also naturally equipped for the modeling of interdependent graphs whose most striking feature is the emergence of the extensive component via a discontinuous phase transition. We have provided a specific implementation for this application that demonstrates how a graph can successively undergo a continuous then a discontinuous phase transition, and how clustering increases the amplitude of discontinuity at the transition.

By offering one of the most comprehensive mathematical description of percolation on random graphs, we believe that the present work contributes to a better understanding of the interplay between local structural properties and the global connectivity of graphs. Moreover, our approach can easily accommodate other types of dynamics for which the pgf technique has already proven to be useful [3,52,53]. We are hopeful that several extensions (different dynamics and/or percolation models, see for instance Refs. [54,55]) will shed further light on the role of structure in the behavior of complex systems. We put forward that some of the tools to perform these studies are now available.

[1] R. Cohen and S. Havlin, *Complex Networks: Structure, Robustness and Function* (Cambridge University Press, Cambridge, UK, 2010).

[2] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, Critical phenomena in complex networks, Rev. Mod. Phys. **80**, 1275 (2008).

[3] A. Allard, L. Hébert-Dufresne, J.-G. Young, and L. J. Dubé, Coexistence of phases and the observability of random graphs, Phys. Rev. E **89**, 022801 (2014).

[4] Y. Yang, J. Wang, and A. E. Motter, Network Observability Transitions, Phys. Rev. Lett. **109**, 258701 (2012).

[5] L. Hébert-Dufresne, A. Allard, J.-G. Young, and L. J. Dubé, Global efficiency of local immunization on complex networks, Sci. Rep. **3**, 2171 (2013).

[6] L. A. Meyers, Contact network epidemiology: Bond percolation applied to infectious disease prediction and control, Bull. Am. Math. Soc. **44**, 63 (2007).

[7] M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Oxford, UK, 2010).

[8] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Network Robustness and Fragility: Percolation on Random Graphs, Phys. Rev. Lett. **85**, 5468 (2000).

[9] M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Random graphs with arbitrary degree distributions and their applications, Phys. Rev. E **64**, 026118 (2001).

[10] A. Allard, L. Hébert-Dufresne, P.-A. Noël, V. Marceau, and L. J. Dubé, Bond percolation on a class of correlated and clustered random graphs, J. Phys. A **45**, 405005 (2012).

[11] A. Allard, P.-A. Noël, L. J. Dubé, and B. Pourbohloul, Heterogeneous bond percolation on multitype networks with an application to epidemic dynamics, Phys. Rev. E **79**, 036113 (2009).

[12] Y. Berchenko, Y. Artzy-Randrup, M. Teicher, and L. Stone, Emergence and Size of the Giant Component in Clustered Random Graphs with a Given Degree Distribution, Phys. Rev. Lett. **102**, 138701 (2009).

[13] G. Ghoshal, V. Zlatić, G. Caldarelli, and M. E. J. Newman, Random hypergraphs and their applications, Phys. Rev. E **79**, 066118 (2009).

[14] J. P. Gleeson, Bond percolation on a class of clustered random networks, Phys. Rev. E **80**, 036107 (2009).

[15] A. Hackett, J. P. Gleeson, and S. Melnik, Site percolation in clustered random networks, Int. J. Comp. Syst. Sci. **1**, 25 (2011).

[16] L. Hébert-Dufresne, A. Allard, J.-G. Young, and L. J. Dubé, Percolation on random networks with arbitrary k-core structure, Phys. Rev. E **88**, 062820 (2013).

[17] B. Karrer and M. E. J. Newman, Random graphs containing arbitrary distributions of subgraphs, Phys. Rev. E **82**, 066118 (2010).

[18] E. Kenah and J. M. Robins, Second look at the spread of epidemics on networks, Phys. Rev. E **76**, 036113 (2007).

[19] E. A. Leicht and R. M. D'Souza, Percolation on interacting networks, arXiv:0907.0894.

[20] L. A. Meyers, M. E. J. Newman, and B. Pourbohloul, Predicting epidemics on directed contact networks, J. Theor. Biol. **240**, 400 (2006).

[21] J. C. Miller, Percolation and epidemics in random clustered networks, Phys. Rev. E **80**, 020901(R) (2009).

[22] M. E. J. Newman, Assortative Mixing in Networks, Phys. Rev. Lett. **89**, 208701 (2002).

[23] M. E. J. Newman, Mixing patterns in networks, Phys. Rev. E **67**, 026126 (2003).

[24] M. E. J. Newman, Properties of highly clustered networks, Phys. Rev. E **68**, 026121 (2003).

[25] M. E. J. Newman, Random Graphs with Clustering, Phys. Rev. Lett. **103**, 058701 (2009).

[26] P.-A. Noël, B. Davoudi, R. C. Brunham, L. J. Dubé, and B. Pourbohloul, Time evolution of epidemic disease on finite and infinite networks, Phys. Rev. E **79**, 026101 (2009).

[27] M. Á. Serrano and M. Boguñá, Percolation and Epidemic Thresholds in Clustered Networks, Phys. Rev. Lett. **97**, 088701 (2006).

[28] X. Shi, L. A. Adamic, and M. J. Strauss, Networks of strong ties, Phys. A (Amsterdam, Neth.) **378**, 33 (2007).

[29] A. Vázquez, Spreading dynamics on heterogeneous populations: Multitype network approach, Phys. Rev. E **74**, 066114 (2006).

[30] A. Vázquez and Y. Moreno, Resilience to damage of graphs with degree correlations, Phys. Rev. E **67**, 015101(R) (2003).

[31] V. Zlatić, D. Garlaschelli, and G. Caldarelli, Networks with arbitrary edge multiplicities, Europhys. Lett. **97**, 28005 (2012).

[32] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, Multilayer networks, J. Comp. Netw. **2**, 203 (2014).

[33] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin, Catastrophic cascade of failures in interdependent networks, Nature (London) **464**, 1025 (2010).

[34] R. Parshani, S. V. Buldyrev, and S. Havlin, Interdependent Networks: Reducing the Coupling Strength Leads to a Change from a First to Second Order Percolation Transition, Phys. Rev. Lett. **105**, 048701 (2010).

[35] It has recently come to our attention that similar phenomena have been studied in G. Bianconi and S. N. Dorogovtsev, Multiple percolation transitions in a configuration model of a network of networks, Phys. Rev. E **89**, 062814 (2014); C. Wu, S. Ji, R. Zhang, L. Chen, J. Chen, X. Li, and Y. Hu, Multiple hybrid phase transition: Bootstrap percolation on complex networks with communities, Europhys. Lett. **107**, 48001 (2014).

[36] A. Allard, L. Hébert-Dufresne, P.-A. Noël, V. Marceau, and L. J. Dubé, Exact solution of bond percolation on small arbitrary graphs, Europhys. Lett. **98**, 16001 (2012).

[37] E. N. Gilbert, Random graphs, Ann. Math. Stat. **30**, 1141 (1959).

[38] An undirected edge between a type i vertex and a type j vertex (noted $i \leftrightarrow j$) therefore occurs with a probability $p_{ij} p_{ji}$. The symmetric case where $p_{ij} = p_{ji}$ for all i and j is statistically equivalent to the situation where all edges are undirected.

[39] It is not forbidden for two hyperedges to have the same composition $\boldsymbol{n}$ while having different nature or different rules of connection. In such case, the probabilities $Q_{\alpha i}(\boldsymbol{l}|\boldsymbol{n}; \mathcal{R})$ should represent the appropriate weighted average of the probabilities computed with Eqs. (3) for each hyperedge. It is, however, strongly encouraged to use different types of stubs—hence different compositions—for each different categories of hyperedges to keep the analysis as clear as possible.

[40] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener, Graph structure in the Web, Comput. Netw. **33**, 309 (2000).

[41] J. C. Miller, Bounding the size and probability of epidemics on networks, J. Appl. Prob. **45**, 498 (2008).

[42] M. E. J. Newman, Spread of epidemic disease on networks, Phys. Rev. E **66**, 016128 (2002).

[43] M. Á. Serrano and M. Boguñá, Clustering in complex networks. II. Percolation properties, Phys. Rev. E **74**, 056115 (2006).

[44] J. P. Gleeson, S. Melnik, and A. Hackett, How clustering affects the bond percolation threshold in complex networks, Phys. Rev. E **81**, 066114 (2010).

[45] I. Z. Kiss and D. M. Green, Comment on "Properties of highly clustered network, Phys. Rev. E **78**, 048101 (2008).

[46] P. Colomer-de-Simón and M. Boguñá, Double Percolation Phase Transition in Clustered Complex Networks, Phys. Rev. X **4**, 041020 (2014).

[47] S.-W. Son, G. Bizhani, C. Christensen, P. Grassberger, and M. Paczuski, Percolation theory on interdependent networks based on epidemic spreading, Europhys. Lett. **97**, 16006 (2012).

[48] P. Désesquelles, Exact solution of finite size mean field percolation and application to nuclear fragmentation, Phys. Lett. B **698**, 284 (2011).

[49] P. Désesquelles, D.-T. Nga, and A. Cholet, Exact solution of random graph fragmentation and physical, chemical and biological applications, J. Phys.: Conf. Ser. **410**, 012058 (2013).

[50] P. J. Reynolds, H. E. Stanley, and W. Klein, Large-cell Monte Carlo renormalization group for percolation, Phys. Rev. B **21**, 1223 (1980).

[51] C. R. Scullard and R. M. Ziff, Critical Surfaces for General Bond Percolation Problems, Phys. Rev. Lett. **100**, 185701 (2008).

[52] A. Hackett, S. Melnik, and J. P. Gleeson, Cascades on a class of clustered random networks, Phys. Rev. E **83**, 056107 (2011).

[53] D. J. Watts, A simple model of global cascades on random networks, Proc. Natl. Acad. Sci. USA **99**, 5766 (2002).

[54] M. Ritchie, L. Berthouze, and I. Z. Kiss, Beyond clustering: Mean-field dynamics on networks with arbitrary subgraph composition, arXiv:1405.6234.

[55] J. C. Miller, A. C. Slim, and E. M. Volz, Edge-based compartmental modelling for infectious disease spread, J. R. Soc. Interface **9**, 890 (2012).