Theory of percolation on hypergraphs

Ginestra Bianconi [1,2] and Sergey N. Dorogovtsev [3,4]

[1]*School of Mathematical Sciences, Queen Mary University of London, London E1 4NS, United Kingdom*
[2]*Alan Turing Institute, 96 Euston Road, London NW1 2DB, United Kingdom*
[3]*Departamento de Física da Universidade de Aveiro & I3N, 3810-193 Aveiro, Portugal*
[4]*Ioffe Physico-Technical Institute, 194021 St. Petersburg, Russia*

Hypergraphs capture the higher-order interactions in complex systems and always admit a factor graph representation, consisting of a bipartite network of nodes and hyperedges. As hypegraphs are ubiquitous, investigating hypergraph robustness is a problem of major research interest. In the literature the robustness of hypergraphs so far only has been treated adopting factor-graph percolation, which describes well higher-order interactions which remain functional even after the removal of one of more of their nodes. This approach, however, fall short to describe situations in which higher-order interactions fail when any one of their nodes is removed, this latter scenario applying, for instance, to supply chains, catalytic networks, protein-interaction networks, networks of chemical reactions, etc. Here we show that in these cases the correct process to investigate is hypergraph percolation, with is distinct from factor graph percolation. We build a message-passing theory of hypergraph percolation, and we investigate its critical behavior using generating function formalism supported by Monte Carlo simulations on random graph and real data. Notably, we show that the node percolation threshold on hypergraphs exceeds node percolation threshold on factor graphs. Furthermore we show that differently from what happens in ordinary graphs, on hypergraphs the node percolation threshold and hyperedge percolation threshold do not coincide, with the node percolation threshold exceeding the hyperedge percolation threshold. These results demonstrate that any fat-tailed cardinality distribution of hyperedges cannot lead to the hyper-resilience phenomenon in hypergraphs in contrast to their factor graphs, where the divergent second moment of a cardinality distribution guarantees zero percolation threshold.

I. INTRODUCTION

Hypergraphs and simplicial complexes form the important class of networks—so-called higher-order networks [1–5]—representing the systems of multinode interactions. Growing research interest is addressed to both modeling [3,6–8] higher-order network structure and investigating dynamical processes on top of them. Many processes and cooperative models on the higher-order networks significantly differ from those on ordinary networks in which each edge interconnects a pair of nodes [3,9,10]. These include opinion dynamics, game theory, synchronization, etc. Despite a few works having already addressed problems related to the robustness of hypergraphs [11–16] we still lack a theory for hyperedge and node percolation on hypergraphs. In this work we focus on the hypergraphs. The hyperedges of a hypergraph can be treated as a second type of nodes (factor nodes), and hence the hypergraphs are equivalent to the factor graphs which are bipartite graphs based on the original nodes and the factor nodes. This representation of hypergraphs provides one with a straightforward way of treating their structural properties [12,15] and

models and processes on them, since bipartite networks are well studied. For example, the giant connected component of a hypergraph coincides with the giant connected component of the corresponding factor graph. Similarly, the hyperedge percolation problem for a hypergraph (removal of hyperedges) conforms the result of the removal of the factor nodes from the factor graph. However, there exist problems, nonequivalent on the hypergraphs and on the factor graphs. The origin of this difference is the distinct effect of the removal of a node from a hypergraph and the removal of a node from the factor graph. Indeed, the deletion of a node in a hypergraph also removes all the adjacent hyperedges; no hyperedge can change its cardinality (unless some additional rule for transformation of hyperedges is implemented). In contrast to this, the deletion of a node in a factor graph doesn't lead to the removal of factor nodes; only the connections of the neighboring factor nodes to the removed node disappear.

There are two classes of higher-order interactions captured by two distinct types of hyperedges. In the first class, including networks of social interactions, the hyperedges break only if all but one of their nodes fail. Therefore these hyperedges can sustain the failure of one and even more of their nodes. For this class, existing theories and models treating hypergraphs as their factor graphs [12,14,17] work perfectly. However, there exists a second important class of hyperedges, which fail as soon as one of their node is damaged. In particular, this class includes supply chains and catalytic networks [18,19],

protein-interaction networks [20], and networks of chemical reactions [21]. For instance the removal of a raw material will impede the production of a product, the absence of a protein will impede the formation of a protein complex, and the absence of a reactant will impede a chemical reaction to occur. For these important hypergraphs, existing theories do not work. In the present paper we develop a percolation theory for such hypergraphs and show that the difference from percolation on factor graphs can be dramatic.

To demonstrate the difference between percolation on hypergraphs and on factor graphs, we explore the percolation problem for hypergraphs and compare it to percolation on factor graphs. Note that in a separate article published in this issue we address hypergraph k-core percolation [22].

Here our approach builds on the message-passing theory for percolation [23–33], which is a special case to message-passing algorithms widely used also in epidemic spreading, Ising models and combinatorial optimization [34–40], and the fundamental statistical mechanics theory of percolation as a paradigmatic example of critical phenomena [41–48]. We derive the message-passing equations for percolation on factor graphs and hypergraphs and apply them to random hypergraphs and random multiplex hypergraphs. In the first model, a random hypergraph is described by two given distributions, namely, a degree distribution for nodes and a cardinality distribution for hyperedges. In the second, more detailed model, a random hypergraph is described by a given joint degree distribution, where a degree is a list (i.e., a vector), whose entries are the numbers of hyperedges with each cardinality, adjacent to a node. For these two kinds of random, locally treelike hypergraphs we obtain the criterion of the presence of a percolation cluster (giant connected component) and the relative size of this cluster. We show that, in contrast to ordinary networks, the node and hyperedge percolation problems for hypergraphs strongly differ from each other. However we highlight that the percolation threshold for node percolation on hypergraph coincides with the percolation threshold of hyperedge percolation on the factor graphs provided one chooses the probability of removing hyperedges of a given cardinality in a suitable way.

The paper is organized as follows. In Sec. II we define the mapping between hypergraphs and factor graphs. In Secs. III and IV we review the key equations for message passing on factor graphs and the relevant analytical results on random factor graphs and random multiplex factor graphs. In Sec. V we derive the equations of the message-passing algorithm for percolation on hypergraphs. In Sec. VI we apply these equations to the model of random hypergraphs and random multiplex hypergraphs with given distributions of node degrees and hyperedge cardinalities and obtain the relative size of the giant connected component and the criterion of its existence. In Sec. VII we discuss our results and indicate other problems where results for hypergraphs and their factor graphs may be distinct.

II. HYPERGRAPHS AND THEIR MAPPING TO FACTOR GRAPHS

A hypergraph $H = (V, E_H)$ is formed by a set of nodes V and a set of hyperedges E_H where each hyperedge α of
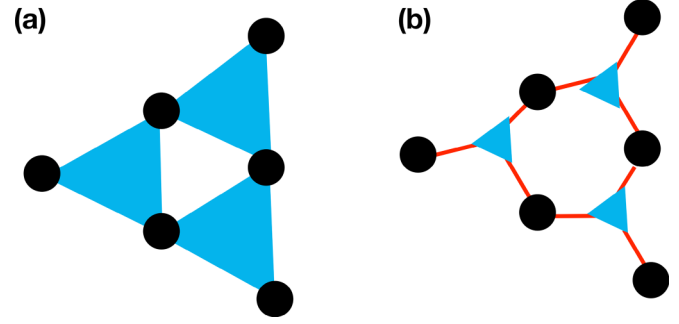

FIG. 1. Schematic representation of a hypergraph (a) and its corresponding factor graph (b). The factor graph is a bipartite network of connections between nodes (filled circles) representing the nodes of the hypergraph and factor nodes (filled triangles) representing the hyperedges of the hypergraph with a factor node connected to a node in the factor graph if and only if this node is incident to the corresponding hyperedge in the hypergraph.

cardinality m_α indicates a set of m_α nodes

$$\alpha = [v_1, v_2, \ldots v_{m_\alpha}]. \tag{1}$$

We indicate with $N = |V|$ the cardinality of the node set and with $M = |E_H|$ the cardinality of the hyperedge set.

A hypergraph $H = (V, E_H)$ can always be represented as a factor graph (see Fig. 1). The factor graph is a bipartite network $G = (V, U, E)$ formed by a set of nodes V, a set of factor nodes U, and a set of edges each one linking one node and one factor node. The factor graph $G = (V, U, E)$ represents the hypergraph $H = (V, E_H)$ when the set of nodes V of the factor graph coincides with the set of nodes of the hypergraph and when each factor node $\alpha \in U$ represents the hyperedge $\alpha \in E_H$. Therefore we have $N = |V|$ and $M = |U|$. In this setting the factor graph G uniquely represents the hypergraph H if each node $i \in V$ is linked to the factor node $\alpha \in U$ if and only if the node $i \in V$ belongs to the hyperedge $\alpha \in E_H$ in the corresponding hypergraph. It follows that in this mapping between hypergraphs and factor graphs, a hyperedge α of cardinality m maps into a factor node of degree m.

Percolation processes characterize the size of the giant connected component when nodes or factor nodes or hyperedges are randomly removed.

In this work our goal is to highlight the similarity and differences between percolation on hypergraphs and on their corresponding factor graphs. These two problems coincide when the hyperedges of the hypergraph and hence the corresponding factor nodes of the factor graph are randomly removed but differ when nodes are randomly removed.

III. MESSAGE PASSING FOR FACTOR GRAPH PERCOLATION

Factor graph percolation monitors the size of the giant connected component, i.e., the fraction of nodes and/or the fraction of factor nodes in the giant connected component of the factor graph, when either nodes or factor nodes are randomly removed. Let us define the message-passing algorithm which implements percolation on the factor graph on locally treelike factor graphs. In the locally treelike bipartite networks, finite cycles virtually vanish as network sizes approach

infinity, and almost all cycles in these networks are infinite [43]. In other words, the probability that a node of any kind belongs to a finite cycle vanishes in these infinite networks. We emphasize that this notion doesn't differ from that for ordinary networks and hypergraphs. Importantly, the factor graph of a hypergraph is locally treelike if and only if the hypergraph is locally treelike. The message-passing algorithms are exact for all locally treelike networks (including ordinary one-partite networks, multipartite networks, and hypergraphs), and they are approximate for networks having a significant fraction of finite cycles, which is typical for real-world networks. The message-passing algorithm is formulated on the factor graph $G = (V, U, E)$ and consists in updating recursively messages sent by nodes to factor nodes and messages sent by factor nodes to nodes. These messages are then used to predict the probability that each node and each factor node belong to the giant component.

We distinguish between two types of message-passing algorithms that implement factor graph percolation depending on the type of information that is available. In particular, the first algorithm assumes that we exactly know the initial damage configuration, namely, the set of the removed nodes and the removed factor nodes (importantly, this array does not include the additional damage induced by these removals), while the second algorithm assumes that the initial damage configuration is unknown and we have access only to the probabilities that nodes and factor nodes are left intact by the initial damage.

To this end, in order to define the message-passing algorithm implementing factor graph percolation we first assume that we know whether each node i is initially damaged, $x_i = 0$, or it is intact, $x_i = 1$, and whether each factor node α is initially damaged, $y_\alpha = 0$, or it is intact, $y_\alpha = 1$.

Let us indicate with $N(i)$ the set of factor nodes α that are the neighbors of node i and with $N(\alpha)$ the set of nodes that are the neighbors of factor node α.

The message-passing algorithm consists in updating the messages $\hat{w}_{i \to \alpha}$ and $\hat{v}_{\alpha \to i}$ going from node i to factor node α and from factor node α to node i, respectively. The message $\hat{w}_{i \to \alpha}$ is equal to one, i.e., $\hat{w}_{i \to \alpha} = 1$ if

(i) Node i is not initially damaged, i.e., $x_i = 1$

(ii) Node i receives at least one positive message $\hat{v}_{\beta \to i} = 1$ from at least one of its neighbor factor nodes β different from α, indicating that β is connected to the giant connected component of the factor graph.

In all the other cases $\hat{w}_{i \to \alpha} = 0$.

The message $\hat{v}_{\alpha \to i}$ is set to one, i.e., $\hat{v}_{\alpha \to i} = 1$ if

(i) The factor node α is not initially damaged, i.e., $y_\alpha = 1$

(ii) At least one of the neighbor nodes j of the factor node α that is different from node i is connected to the giant component. This event occurs if the factor node α receives at least one positive message $\hat{w}_{j \to \alpha} = 1$ from one of its neighbor nodes j different from i.

In all the other cases $\hat{v}_{\alpha \to i} = 0$.

Consequently these messages are updated according to the following rules:

$$\hat{w}_{i \to \alpha} = x_i \left[1 - \prod_{\beta \in N(i) \backslash \alpha} (1 - \hat{v}_{\beta \to i}) \right],$$

$$\hat{v}_{\alpha \to i} = y_\alpha \left[1 - \prod_{j \in N(\alpha) \backslash i} (1 - \hat{w}_{j \to \alpha}) \right]. \tag{2}$$

The indicator function $\hat{r}_i$ that node i is in the giant connected component is 1 ($\hat{r}_i = 1$) if and only if

(i) Node i is not initially damaged, i.e., $x_i = 1$

(ii) Node i receives at least one positive message $\hat{v}_{\alpha \to i} = 1$ from at least one of its neighbor factor nodes.

In all the other cases $\hat{r}_i = 0$.

The indicator function $\hat{s}_\alpha$ that the factor node α is in the giant component is 1 ($\hat{s}_\alpha = 1$) if and only if

(i) The factor node is not initially damaged, i.e., $y_\alpha = 1$

(ii) At least one of the neighbor nodes i of the factor node α is connected to the giant component. This event occurs if the factor node α receives at least one positive message $\hat{w}_{i \to \alpha} = 1$ from one of its neighbor nodes.

In all the other cases $\hat{s}_\alpha = 0$.

We have therefore that $\hat{r}_i$ and $\hat{s}_\alpha$ are defined in terms of the messages as

$$\hat{r}_i = x_i \left[1 - \prod_{\alpha \in N(i)} (1 - \hat{v}_{\alpha \to i}) \right],$$

$$\hat{s}_\alpha = y_\alpha \left[1 - \prod_{i \in N(\alpha)} (1 - \hat{w}_{i \to \alpha}) \right]. \tag{3}$$

In a number of situations, however, although we might have access to the real hypergraph structure, we might not have full knowledge of the initial damage of the nodes. In particular, we might know only that nodes and factor nodes of degree m are initially intact (i.e., not damaged) independently at random with probability p_N and $p_H^{[m]}$, respectively, and that the probability of the initial damage configuration $\{x_i\} = \{x_1, x_2, \ldots, x_N\}, \{y_\alpha\} = \{u_{\alpha_1}, x_{\alpha_2}, \ldots, x_M\}$ is

$$P(\{x_i\}, \{y_\alpha\}) = \prod_{i=1}^{N} p_N^{x_i} (1 - p_N)^{1-x_i}$$

$$\times \prod_{\alpha=1}^{M} \left(p_H^{[m_\alpha]} \right)^{y_\alpha} \left(1 - p_H^{[m_\alpha]} \right)^{1-y_\alpha}. \tag{4}$$

Consequently in this scenario the message-passing algorithm needs to predict the fraction of nodes and factor in the giant component using only the probabilities p_N and $p_H^{[m]}$.

In this case we should consider an alternative message-passing algorithm in which the messages sent from nodes to factor nodes are indicated by $w_{i \to \alpha}$ and the messages sent from factor nodes to nodes are indicated with $v_{\alpha \to i}$. The message $w_{i \to \alpha}$ is obtained by averaging $\hat{w}_{i \to \alpha}$ over the probability distribution $P(\{x_i\}, \{y_\alpha\})$ and similarly $v_{\alpha \to i}$ is obtained by averaging $\hat{v}_{\alpha \to i}$ over the probability distribution $P(\{x_i\}, \{y_\alpha\})$. In this way we obtain the following message-passing algorithm:

$$w_{i \to \alpha} = p_N \left[1 - \prod_{\beta \in N(i) \backslash \alpha} (1 - v_{\beta \to i}) \right],$$

$$v_{\alpha \to i} = p_H^{[m_\alpha]} \left[1 - \prod_{j \in N(\alpha) \backslash i} (1 - w_{j \to \alpha}) \right]. \tag{5}$$

In this framework the probability r_i that node i is in the giant component and the probability s_α that factor node α is in the giant component are given by

$$r_i = p_N \left[1 - \prod_{\alpha \in N(i)} (1 - v_{\alpha \to i}) \right],$$

$$s_\alpha = p_H^{[m]} \left[1 - \prod_{i \in N(\alpha)} (1 - w_{i \to \alpha}) \right], \qquad (6)$$

where these probabilities can be obtained by averaging $\hat{r}_i$ and $\hat{s}_\alpha$ over the probability distribution $P(\{x_i\}, \{y_\alpha\})$. Node percolation on a factor graph implies characterizing the fraction of nodes R and the fraction S of factor nodes in the giant component,

$$R = \frac{1}{N} \sum_{i=1}^{N} r_i,$$

$$S = \frac{1}{M} \sum_{\alpha=1}^{M} s_\alpha, \qquad (7)$$

as a function of p_N and $p_H^{[m]}$. In particular, putting $p_H^{[m]} = 1$ for all values of m one can characterize node percolation on the factor graph as a function of p_N, while by putting $p_N = 1$ and $p_H^{[m]} = p_H$ for all m we can characterize factor node percolation on the factor graph. Both transitions are continuous and occur at a percolation threshold that can be determined by linearizing the message-passing algorithm. Indeed for $w_{i \to \alpha} \ll 1$ and $v_{\alpha \to i} \ll 1$, Eqs. (5) can be linearized to obtain

$$w_{i \to \alpha} = p_N \sum_{\beta \in N(i) \backslash \alpha} v_{\beta \to i},$$

$$v_{\alpha \to i} = p_H^{[m_\alpha]} \sum_{j \in N(\alpha) \backslash i} w_{j \to \alpha}. \qquad (8)$$

This linearized equation admits a nonzero solution if and only if

$$\Lambda(p_N, \mathbf{p}_H) > 1, \qquad (9)$$

where $\Lambda(p_N, \mathbf{p}_H)$ with $\mathbf{p}_H = (p_H^{[2]}, p_H^{[3]}, \ldots)$ is the maximum eigenvalue of the factor graph nonbacktracking matrix $\mathcal{A}$ of block structure

$$\mathcal{A} = \begin{pmatrix} 0 & \mathcal{B}^{NH} \\ \mathcal{B}^{HN} & 0 \end{pmatrix} \qquad (10)$$

with the nonbacktracking matrices $\mathcal{B}^{NH}$ and $\mathcal{B}^{HN}$ having elements given by

$$\mathcal{B}^{NH}_{(i \to \alpha);(\beta \to j)} = p_N \delta_{j,i} (1 - \delta_{\beta,\alpha}),$$

$$\mathcal{B}^{HN}_{(\alpha \to i);(j \to \beta)} = p_H^{[m_\alpha]} \delta_{\alpha,\beta} (1 - \delta_{i,j}), \qquad (11)$$

where $\delta_{x,y}$ indicates the Kronecker delta. Therefore the transition occurs for

$$\Lambda(p_N, \mathbf{p}_H) = 1. \qquad (12)$$

For a uniform damage, i.e., $p_H^{[m_\alpha]} = p_H$ independent of m_α, two percolation thresholds for an arbitrary locally treelike

factor graph coincide, $p_{cH}^{(\text{factor graph})} = p_{cN}^{(\text{factor graph})}$, which can be obtained from the block matrix $\mathcal{A}$, Eqs. (10) and (11).

IV. ANALYTICAL SOLUTION OF FACTOR GRAPH PERCOLATION

A. Percolation on a random factor graph

The message-passing algorithms from Sec. III enable us to obtain the giant connected component for an arbitrary factor graph as long as its structure is close to a locally treelike one. However, in a number of cases we do not have access to the full topology of the factor graph, and we know only its structural statistical properties. In this case we can get analytical predictions for factor graph percolation as long as we assume that the factor graph can be modeled by a random bipartite network. In particular we assume that the factor graph is a random sparse bipartite network where the nodes have a degree distribution $P(q)$ and the factor nodes have a degree distribution $Q(m)$, where $P(q)$ and $Q(m)$ are arbitrary, provided the total number of links incident to the nodes is equal to the total number of links of the factor nodes, $N\langle q \rangle = M\langle m \rangle$. The factor graph G is uniquely determined by the $N \times M$ incidence matrix $\mathbf{a}$ of elements $a_{i\alpha}$, where $a_{i\alpha} = 1$ if node i is connected to factor node α and otherwise $a_{i\alpha} = 0$. We assume that the factor graph G is drawn from the distribution

$$P(G) = \prod_{i,\alpha} p_{i\alpha}^{a_{i\alpha}} (1 - p_{i\alpha})^{1 - a_{i,\alpha}}, \qquad (13)$$

with $p_{i\alpha}$ indicating the probability that in an uncorrelated factor graph node i is connected to factor node α,

$$p_{i\alpha} = \frac{q_i m_\alpha}{\langle m \rangle M}. \qquad (14)$$

Here the degree sequences $\{q_i\}$ and $\{m_\alpha\}$ of the nodes and of the factors nodes are drawn from the distribution $P(q)$ and $Q(m)$. In the limit $N \to \infty$ and $M \to \infty$ with $M/N = O(1)$, standard percolation describes a critical phenomenon that can be studied with statistical mechanics approaches.

In particular, factor graph percolation can be captured by two nonlinear equations: (1) for the probability W, that if we follow a random edge of a factor node, then we reach a node in the giant component, and (2) for the probability V, that by following a random edge of a node we reach a factor node in the giant component. The probability W can be obtained by averaging the messages $w_{\alpha \to i}$ over $P(G)$, and, similarly, the probability V can be obtained by averaging the messages $v_{i \to \alpha}$ over the probability $P(G)$. In this way, starting from the message-passing algorithm Eqs. (5), it is straightforward to derive the following equations for W and V:

$$W = p_N \sum_m \frac{q P(q)}{\langle q \rangle} [1 - (1 - V)^{q-1}],$$

$$V = \sum_m p_H^{[m]} \frac{m Q(m)}{\langle m \rangle} [1 - (1 - W)^{m-1}]. \qquad (15)$$

The fractions R and S of, respectively, the nodes and factor nodes in the giant connected component are expressed in

terms of the probabilities W and V,

$$R = p_N \sum_q P(q)[1 - (1 - V)^q],$$

$$S = \sum_m p_H^{[m]} Q(m)[1 - (1 - W)^m], \quad (16)$$

where these equations are obtained by averaging Eqs. (6) and (7) over $P(G)$. Assuming that both the degree distributions $P(q)$ and $Q(m)$ have finite second moments, one can linearize the right-hand sides of Eqs. (15), which leads to the following criterion of the existence of the giant connected component in this network:

$$p_N \frac{\langle q(q-1) \rangle}{\langle q \rangle} \frac{\langle p_H^{[m]} m(m-1) \rangle}{\langle m \rangle} > 1. \quad (17)$$

In particular if $p_H^{[m]} = p_H$ do not depend on m, then we obtain

$$p_N p_H \frac{\langle q(q-1) \rangle}{\langle q \rangle} \frac{\langle m(m-1) \rangle}{\langle m \rangle} > 1. \quad (18)$$

One can check that the phase transition occurring when the left-hand side of Eq. (17) equals 1 is continuous. Setting $p_H = 1$ in Eq. (18) we obtain the node percolation threshold $p_{cN}^{\text{(factor graph)}}$, and setting $p_N = 1$ we obtain the factor node percolation threshold $p_{cH}^{\text{(factor graph)}}$. These two thresholds coincide similarly to ordinary uncorrelated networks. Hence we have $p_{cN}^{\text{(factor graph)}} = p_{cH}^{\text{(factor graph)}} = p^\star$ with $p^\star$ satisfying

$$p^\star \frac{\langle q(q-1) \rangle}{\langle q \rangle} \frac{\langle m(m-1) \rangle}{\langle m \rangle} = 1. \quad (19)$$

This equation and Eq. (17) can be compared with the Molloy-Reed criterion $p\langle q(q-1) \rangle / \langle q \rangle > 1$ for ordinary networks [42], which is valid for both the node and edge percolation problems when $\langle q^2 \rangle$ is finite.

B. Generalized degree distribution

The random bipartite network is not the only possible ensemble in which factor graph percolation can be solved analytically. Indeed, it is possible for any nonuniform hypergraph (and corresponding factor graph) to construct a multiplex hypergraph [12,49] (see Fig. 2) encoding for its structure, where each layer is formed by a random hypergraph having hyperdges of fixed cardinality. These leads to a correlated hypergraph (and corresponding factor graph) structure in which each node i is associated a vector of degrees $\mathbf{q}(i) = \{q_2(i), q_3(i) \ldots q_m(i) \ldots\}$ where $q_m(i)$ indicates the number of hyperedges of cardinality m to which node i belongs and each hyperedge α of cardinality $m_\alpha = m$ describes one higher-order interaction in the layer m. The corresponding factor graph ensemble is a multiplex ensemble of factor graphs $\vec{G} = (G^{[2]}, G^{[3]}, \ldots, G^{[m]} \ldots)$, such that in each network $G^{[m]}$ in layer m the generic node i has degree $q_m(i)$ and all the factor nodes have constant degree m. Let us indicate with the matrix $\mathbf{a}^{(m)}$ the incidence matrix of the factor graph in the m layer of the multiplex factor graph. The probability of this multiplex factor graph $\vec{G}$ is given by

$$P(\vec{G}) = \prod_{m \geqslant 2} \prod_{\alpha \in \mathcal{K}(m)} \prod_i \left[p_{i\alpha}^{(m)} \right]^{a_{i\alpha}^{(m)}} \left(1 - p_{i\alpha}^{(m)} \right)^{1 - a_{i\alpha}^{(m)}}, \quad (20)$$
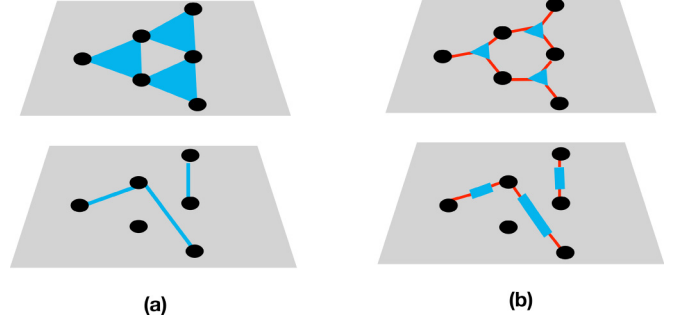

(a)   **(b)**

FIG. 2. A multiplex hypergraph (a) [12,49] is a nonuniform (i.e., containing hyperedges with different cardinalities) hypergraph represented by a multiplex network in which each layer m accounts for only the hyperedge of cardinality m in the hypergraph. In (a) the multiplex hypergraph has two layers $m = 2$ and $m = 3$. A multiplex hypergraph can be mapped to a multiplex factor graph (b) in which each layer is formed by the factor graph corresponding to the hypergraph in corresponding layer of the multiplex hypergraph. When each layer of the hypergraph is drawn from a random hypergraph ensemble, the multiplex hypergraph in general will be correlated. Similarly the corresponding multiplex factor graph will also be correlated.

where $\mathcal{K}(m)$ indicates the set of hyperedges of cardinality m and where the probability $p_{i\alpha}^{(m)}$ of observing a link between node i and factor node α in layer m is given by

$$p_{i\alpha}^{(m)} = \frac{q_m(i)}{\langle q_m \rangle N}, \quad (21)$$

where $\langle q_m \rangle N = m M_m$. In this ensemble, factor graph percolation is captured by the probability W_m that by following a random edge of a factor node in layer m we reach a node in the giant component and by the probability V_m that by following a random edge of a node in layer m we reach a factor node in the giant component. These probabilities obey the equations

$$W_m = p_N \sum_{\mathbf{q}} \frac{q_m P(\mathbf{q})}{\langle q_m \rangle} \left[1 - \prod_{m'=2}^{M} (1 - V_{m'})^{q_{m'} - \delta_{m,m'}} \right],$$

$$V_m = p_H^{[m]} [1 - (1 - W_m)^{m-1}], \quad (22)$$

which can be obtained directly by averaging the message-passing algorithm (5) over the probability $P(\vec{G})$ given by Eq. (20). Here $q_m P(\mathbf{q}) / \langle q_m \rangle$ is the degree distribution of an end node of an edge in layer m. The probabilities R and S that a node and a factor node are in the giant component of the multiplex factor graph are given by

$$R = p_N \sum_{\mathbf{q}} P(\mathbf{q}) \left[1 - \prod_{m'=2}^{M} (1 - V_{m'})^{q_{m'}} \right],$$

$$S = \sum_m p_H^{[m]} Q(m)[1 - (1 - W_m)^m]. \quad (23)$$

These equations describe a continuous second-order phase transition whose critical point is determined by the condition

$$\Lambda = 1, \quad (24)$$

where Λ is the maximum eigenvalue of the matrix $\mathbf{G}$ of elements

$$G_{m,m'} = p_N p_H^{[m]}(m-1) \frac{\langle q_m(q_m - \delta_{m',m}) \rangle}{\langle q_m \rangle}. \qquad (25)$$

This network has a giant connected component when $\Lambda > 1$.

V. MESSAGE-PASSING ALGORITHM FOR HYPERGRAPH PERCOLATION

While factor node percolation on a factor graph fully accounts for hyperedge percolation on the corresponding hypergraph, node percolation on a factor graph and on an hypergraph are distinct. In node percolation on a factor graph, a factor node (hyperedge) is still able to connect its nodes also if one or more of its nodes are initially damaged, provided that at least one of its nodes is connected to the giant component. However, in node percolation on hypergraphs, a hyperedge is able to connect a node to the giant component only if none of its other nodes are initially damaged. In other words, the initial damage of a single node of an hyperedge deactivates the entire hyperedge.

Here and in the following we formulate the message-passing algorithms that fully account for this difference. Interestingly, the message passing for hypergraph percolation is still conveniently defined on the factor graph representing the hypergraph, although it fully accounts for the differences between factor graph and hypergraph percolation. We assume that we have full information about the hypergraph structure completely encoded in the corresponding factor graph wiring.

The message-passing algorithm for hypergraph percolation can be implemented for any real-world hypergraph topology as long as the factor graph corresponding to the hypergraph is locally treelike.

As we did for the message-passing algorithms for factor graph percolation, we will first formulate the message-passing algorithm under the assumption that we will have full knowledge of the initial damage. Subsequently, we will formulate the message-passing algorithm that can be applied to the case in which we have only access to the probability that nodes and hyperedges are randomly removed.

To start with, let us formulate the message-passing algorithm able to predict hypergraph percolation when we have full information about the entity of the initial damage. To this end, let us assume that we know whether each node i is initially damaged $x_i = 0$ or intact $x_i = 1$ and whether each hyperedge α is initially damaged $y_\alpha = 0$ or intact $y_\alpha = 1$. Note that the variable y_α does not take into account the damage induced by the removal of nodes. The message-passing algorithm implementing hypergraph percolation consists in updating the messages $\hat{\omega}_{i \to \alpha}$ and $\hat{v}_{\alpha \to i}$ going from node i to factor node α and from factor node α to node i. The message $\hat{\omega}_{i \to \alpha}$ is equal to 1, i.e., $\hat{\omega}_{i \to \alpha} = 1$, if

(i) Node i is not initially damaged, i.e., $x_i = 1$

(ii) Node i receives at least one positive message $\hat{v}_{\beta \to i} = 1$ from at least one of its neighbor factor nodes β different from α, indicating that it is connected to the giant component of the hypergraph.

In all the other cases $\hat{\omega}_{i \to \alpha} = 0$.

The message $\hat{v}_{\alpha \to i}$ is equal to 1, i.e., $\hat{v}_{\alpha \to i} = 1$ if

(i) The hyperedge is not initially damaged, i.e., $y_\alpha = 1$

(ii) Each of the nodes j that belong to hyperedge α and differ from the node i is intact, i.e., $\prod_{j \in N(\alpha) \setminus i} x_j = 1$

At least one of the nodes j that belong to hyperedge α and differ from the node i is connected to the giant component. This event occurs if the factor node α receives at least one positive message $\hat{\omega}_{j \to \alpha} = 1$ from one of its neighbor nodes different from i.

In all the other cases $\hat{v}_{\alpha \to i} = 0$.

Consequently these messages are updated according to the following rule:

$$\hat{\omega}_{i \to \alpha} = x_i \left[1 - \prod_{\beta \in N(i) \setminus \alpha} (1 - \hat{v}_{\beta \to i}) \right],$$

$$\hat{v}_{\alpha \to i} = y_\alpha \left(\prod_{j \in N(\alpha) \setminus i} x_i \right) \left[1 - \prod_{j \in N(\alpha) \setminus i} (1 - \hat{\omega}_{j \to \alpha}) \right]. \quad (26)$$

These equations can be further simplified by introducing a message $\hat{w}_{i \to \alpha}$ that indicates the message sent by a node i to a neighbor factor node α, under the assumption that $x_i = 1$,

$$\hat{w}_{i \to \alpha} = \left[1 - \prod_{\beta \in N(i) \setminus \alpha} (1 - \hat{v}_{\beta \to i}) \right], \qquad (27)$$

which is related to the previously defined message $\hat{\omega}_{i \to \alpha}$ by

$$\hat{\omega}_{i \to \alpha} = x_i \hat{w}_{i \to \alpha}. \qquad (28)$$

The message $\hat{v}_{\alpha \to i}$ can now be expressed directly in terms of $\hat{w}_{i \to \alpha}$ as

$$\hat{v}_{\alpha \to i} = y_\alpha \left(\prod_{j \in N(\alpha) \setminus i} x_i \right) \left[1 - \prod_{j \in N(\alpha) \setminus i} (1 - x_j \hat{w}_{j \to \alpha}) \right]$$

$$= y_\alpha \left(\prod_{j \in N(\alpha) \setminus i} x_i \right) \left[1 - \prod_{j \in N(\alpha) \setminus i} (1 - \hat{w}_{j \to \alpha}) \right], \quad (29)$$

where in this expression we have used the fact that $\hat{v}_{\alpha \to i}$ is nonzero only if $x_j = 1$ for every $j \in N(\alpha) \setminus i$, hence we can safely substitute $\hat{\omega}_{i \to \alpha}$ with $\hat{w}_{i \to \alpha}$. The indicator function $\hat{r}_i$ that node i is in the giant component is 1 ($\hat{r}_i = 1$) if and only if

(i) Node i is not initially damaged, i.e., $x_i = 1$

(ii) Node i receives at least one positive message ($\hat{v}_{\alpha \to i} = 1$) from at least one of its neighbor factor nodes.

In all the other cases $\hat{r}_i = 0$. The indicator function $\hat{s}_\alpha$ that the factor node α is in the giant component equals 1 ($\hat{s}_\alpha = 1$) if and only if

(i) The hyperedge α is not initially damaged

(ii) Each of the nodes j that belong to the hyperedge α is intact, i.e., $\prod_{j \in N(\alpha)} x_j = 1$

(iii) At least one of the nodes j that belong to the hyperedge α is connected to the giant component. This event occurs if the factor node α receives at least one positive message $\hat{w}_{i \to \alpha} = 1$ from one of its neighbor nodes.

We obtain therefore that the indicator functions $\hat{s}_\alpha$ and $\hat{r}_i$ are given by

$$\hat{s}_\alpha = y_\alpha \left(\prod_{j \in N(\alpha) \setminus i} x_i \right) \left[1 - \prod_{i \in N(\alpha)} (1 - \hat{w}_{i \to \alpha}) \right], \quad (30)$$

$$\hat{r}_i = x_i \left[1 - \prod_{\alpha \in N(i)} (1 - \hat{v}_{\alpha \to i}) \right]. \quad (31)$$

In a number of situations, although we might have access to the real hypergraph structure, we might not have full knowledge of the initial damage of the nodes. In particular, we might know only that nodes and hyperedges of cardinality m are damaged independently at random with probability p_N and $p_H^{[m]}$, respectively, which provides the probability distribution $P(\{x_i\}, \{y_\alpha\})$ given by Eq. (4).

In this scenario we should consider an alternative message-passing algorithm in which the messages sent from nodes to factor nodes are indicated by $w_{i \to \alpha}$ and the messages sent from factor nodes to nodes are indicated with $v_{\alpha \to i}$, where $w_{i \to \alpha}$ is obtained by averaging $\hat{w}_{i \to \alpha}$ over the probability distribution $P(\{x_i\}, \{y_\alpha\})$ and, similarly, $v_{\alpha \to i}$ is obtained by averaging $\hat{v}_{\alpha \to i}$ over the probability distribution $P(\{x_i\}, \{y_\alpha\})$. In this way we obtain the following message-passing algorithm:

$$w_{i \to \alpha} = \left[1 - \prod_{\beta \in N(i) \setminus \alpha} (1 - v_{\beta \to i}) \right], \quad (32)$$

$$v_{\alpha \to i} = p_H^{[m]} p_N^{m_\alpha - 1} \left[1 - \prod_{j \in N(\alpha) \setminus i} (1 - w_{j \to \alpha}) \right], \quad (33)$$

where m_α indicates the cardinality of hyperedge α (degree of the corresponding factor node in the bipartite network). In this framework, the probability r_i that node i is in the giant component and the probability s_α that factor node α (hyperedge) is in the giant component are given by

$$r_i = p_N \left[1 - \prod_{\alpha \in N(i)} (1 - v_{\alpha \to i}) \right], \quad (34)$$

$$s_\alpha = p_H^{[m]} p_N^{m_\alpha} \left[1 - \prod_{i \in N(\alpha)} (1 - w_{i \to \alpha}) \right], \quad (35)$$

where these probabilities can be obtained by averaging $\hat{r}_i$ and $\hat{s}_\alpha$ over the probability distribution $P(\{x_i\}, \{y_\alpha\})$. By comparing the message-passing equations for hypergraph percolation and for factor node percolation, we conclude that the hypergraph percolation algorithm reduces to factor node percolation for $p_N = 1$; however, the two algorithms differ for $p_N \neq 1$.

Both hyperedge and node percolation are continuous and occur at a percolation threshold that can be determined by linearising the message-passing algorithm. Indeed, for $w_{i \to \alpha} \ll 1$ and $v_{\alpha \to i} \ll 1$, Eqs. (33) and (32) can be linearized to obtain

$$w_{i \to \alpha} = \sum_{\beta \in N(i) \setminus \alpha} v_{\beta \to i},$$

$$v_{\alpha \to i} = p_H^{[m_\alpha]} p_N^{m_\alpha - 1} \sum_{j \in N(\alpha) \setminus i} w_{j \to \alpha}. \quad (36)$$

This linearized equation admits a nonzero solution if and only if

$$\Lambda(p_N, \mathbf{p}_H) > 1, \quad (37)$$

where $\Lambda(p_N, \mathbf{p}_H)$ with $\mathbf{p}_H = (p_H^{[2]}, p_H^{[3]}, \ldots)$ is the maximum eigenvalue of the factor graph nonbacktracking matrix $\mathcal{A}$ of block structure

$$\mathcal{A} = \begin{pmatrix} 0 & \mathcal{B}^{NH} \\ \mathcal{B}^{HN} & 0 \end{pmatrix} \quad (38)$$

with the nonbacktracking matrices $\mathcal{B}^{NH}$ and $\mathcal{B}^{HN}$ having elements given by

$$\mathcal{B}^{NH}_{(i \to \alpha);(\beta \to j)} = \delta_{j,i}(1 - \delta_{\beta,\alpha}),$$

$$\mathcal{B}^{HN}_{(\alpha \to i);(j \to \beta)} = p_H^{[m_\alpha]} p_N^{m_\alpha - 1} \delta_{\alpha,\beta}(1 - \delta_{i,j}), \quad (39)$$

where $\delta_{x,y}$ indicates the Kronecker delta. Therefore the condition for the transition is

$$\Lambda(p_N, \mathbf{p}_H) = 1. \quad (40)$$

Note that that the hyperedge percolation threshold for a hypergraph coincides with the factor node percolation threshold for the corresponding factor graph. We observe that node percolation on a factor graph and on a hypergraph is distinct, and it can be shown that the percolation threshold on a hypergraph is always strictly larger than for node percolation on its factor graph provided the hypergraph is not only formed by hyperedges of cardinality $m = 2$, i.e., provided the hypergraph is not a graph. Indeed, the maximum eigenvalue of the nonnegative matrix $\mathcal{A}$ obeys the Collatz-Wielandt formula

$$\Lambda = \max_{\mathbf{x} > 0} \min_{\gamma} \frac{(\mathcal{A}\mathbf{x})_\gamma}{x_\gamma}. \quad (41)$$

Now let us compare the entries of the matrices $\mathcal{A}$ of hypergraphs and factor graphs in the case of $p_H^{[m]} = 1$, i.e., for the node percolation problems, Eqs. (39) and (11). Both matrices have non-negative elements. Furthermore for an arbitrary p_N, each entry of the matrix $\mathcal{A}$ of a hypergraph cannot be larger than the corresponding entry of the matrix $\mathcal{A}$ of the factor graph. Consequently, for each choice of the vector $\mathbf{x} > 0$ and each value of $\gamma = (i \to \alpha)$ or $\gamma = (\alpha \to i)$ the ratio $(\mathcal{A}\mathbf{x})_\gamma / x_\gamma$ for the hypergraph node percolation problem is smaller than or equal to the ratio for the factor node percolation problem. This implies that for a given value of p_N the largest eigenvalue of the nonbacktracking matrix for factor graph node percolation is larger than or equal to the largest eigenvalue of the nonbracktracking matrix for hypergraph node percolation. Thus this argument confirms that the node percolation threshold on a hypergraph is either equal to or exceeds the threshold for node percolation on the factor graph. A closer look at the eigenvalue problem will reveal that the equality holds only if the hypergraph reduces to a network.

This mathematical results also can be derived by the following physical argument. Let us compare the node and hyperedge percolation thresholds of a hypergraph. Inspecting the matrix elements in Eq. (39), we see that the node percolation threshold p_{cN} coincides with the hyperedge percolation threshold if the probability that hyperedge α of cardinality m_α is retained with probability $p_H^{[m_\alpha]} = p_{cN}^{m_\alpha - 1} \leqslant p_{cN}$. If we remove hyperedges uniformly (independently of
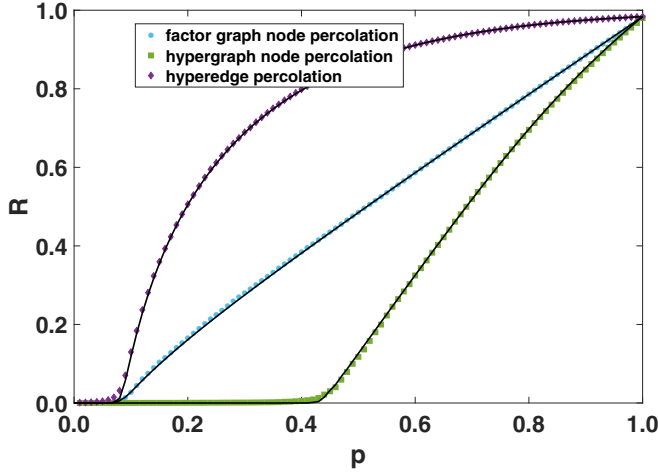
FIG. 3. Factor graph node percolation (circles), hypergraph node percolation (squares), and hyperedge percolation (diamonds) for a factor graph and the corresponding hypergraph with $N = 10^4$ nodes and $M = 10^4$ factor nodes (hyperedges) having the Poisson degree distribution $P(q)$ with $\langle q \rangle = 4$ for nodes and $Q(m) = \delta(m, 4)$ for factor nodes. The symbols indicate results of the Monte Carlo simulations of the percolation processes, and the solid lines indicate results obtained with the corresponding message-passing algorithms. Hyperedge percolation on a hypergraph coincides with factor node percolation on the corresponding factor graph. The percolation threshold for factor graph percolation (and hyperedge percolation) is $p_c = 1/12 = 0.0833\ldots$ [Eq. (18)], while for hypergraph node percolation, it is $p_c = 12^{-1/3} = 0.437\ldots$ [Eq. (48)].

their cardinalities), then the resulting hyperedge percolation threshold $p_{cH}^{[m]}$ must be lower than the maximum probability $p_H^{[m_\alpha]} = p_{cN}^{m_\alpha - 1}$ for the nonuniform removal of edges, i.e., $p_{cH} < p_{cN}^{m_{\min} - 1} \leqslant p_{cN}$. Thus this argument also confirms that node percolation threshold for a treelike hypergraph exceeds its hyperedge percolation threshold.

Now we recall that for a locally treelike graph, the node and factor node percolation thresholds coincide, $p_{cN}^{\text{(factor graph)}} = p_{cH}^{\text{(factor graph)}}$. Consequently we have as long as the hypergraph does not reduce to a network,

$$p_{cN} > p_{cH} = p_{cH}^{\text{(factor graph)}} = p_{cN}^{\text{(factor graph)}}, \qquad (42)$$

and hence indeed the node percolation threshold of a locally treelike hypergraph exceeds the factor node percolation threshold for the corresponding factor graph.

This effect is also evident from our numerical comparison of factor graph percolation and hypergraph percolation on a random hypergraph of 10^4 nodes and 10^4 hyperedges, having a Poisson degree distribution with $\langle q \rangle = 4$ and all hyperedges of the same cardinality $m = 4$ (Fig. 3) and on the real hypergraph of U.S. Senate committees [50,51] (see Fig. 4). This is a hypergraph where nodes are members of the U.S. House of Representatives and hyperedges correspond to committee memberships. The hypergraph has 1290 nodes with average degree 9.2 and 341 hyperedges with the average size of hyperedges $\langle m \rangle = 34.8$ and the maximum size of the hyperedges $m_{\max} = 82$. These two figures compare the results of numerical simulations and of the message-passing algorithm for each of these networks having, notably, rapidly decaying degree
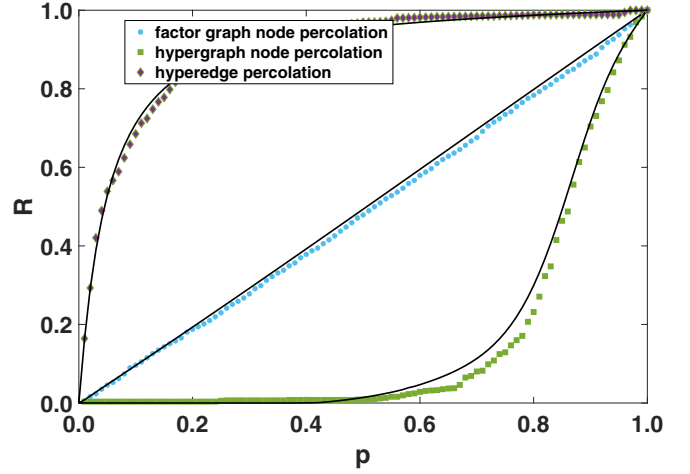

FIG. 4. Factor graph node percolation and hypergraph node percolation and hyperedge percolation for a factor graph and the corresponding hypergraph on the U.S. Senate Committee hypergraph [50,51]. The symbols indicate results of the Monte Carlo simulations for factor graph node percolation (circles), hypergraph node percolation (squares), and hyperedge percolation (diamonds), while the solid lines indicate results obtained with the corresponding message-passing algorithms. The data set has $N = 1290$ nodes and $M = 341$ hyperedges; the average degree of nodes is $\langle q \rangle = 9.2$, the average cardinality of the hyperedges is $\langle m \rangle = 34.8$, and their maximum cardinality is $m_{\max} = 82$.

and cardinality distributions. The message-passing algorithm perfectly describes the indicated difference on the random hypergraph whose corresponding factor graph is treelike, and this algorithm provides still a good prediction of the percolation process also in the case of the real hypergraph, which deviates from the pure locally treelike approximation. In both cases we find confirmation that the percolation threshold for hypergraph percolation can be significantly higher than the percolation on the corresponding factor graph. Even for the synthetic random hypergraph with $\langle q \rangle = 4$ and $m = 4$, the difference between these thresholds is surprisingly big, namely, 0.44, vs 0.08, and for the real hypergraph of U.S. Senate committees, whose factor graph has a very low percolation threshold; this difference is dramatic (see Fig. 4).

Interestingly, it turns out that by setting $p_H^{[m]} = \pi^{m-1}$ in the factor graph percolation, one finds that the percolation threshold π_c and the node percolation threshold of a hypergraph, p_{cN}, coincide, i.e., $\pi_c = p_{cN}$. That is, the percolation threshold of node percolation on hypergraph coincides with the percolation threshold for factor node percolation on the corresponding factor graph when factor nodes of degree m are left intact with probability π^{m-1}. Note, however, that the this mapping does not extent to the fraction of nodes in the giant connected component.

Let us indicate with R the fraction of nodes in the giant component of node percolation on the hypergraph (assuming that $p_N = p \in [0, 1]$ and $p_H^{[m]} = 1$ for every m), with $R^{\text{(factor graph)}}$ the fraction of nodes in the giant connected component for factor graph percolation with $p_H^{[m]} = p^{m-1}$ and $p_N = 1$, and let us indicate with S_m and $S_m^{\text{(factor graph)}}$ the fractions of, respectively, hyperedges of cardinality m and factor

nodes of degree m within the giant connected components for these problems. Then we have

$$R = pR^{(\text{factor graph})},$$
$$S_m = p^m S_m^{(\text{factor graph})}. \tag{43}$$

VI. ANALYTICAL SOLUTION OF HYPERGRAPH PERCOLATION

A. Percolation for the configuration model of uncorrelated hypergraphs

We can apply the same configuration model as in Sec. IV to uncorrelated hypergraphs. In this case a random hypergraph is described by a given node degree distribution $P(q)$ and a given hyperedge cardinality distribution $Q(m)$. Similarly to Sec. IV, we introduce the averages $V = \langle v_{\alpha \to i} \rangle$ and $W = \langle w_{i \to \alpha} \rangle$ of the messages considered in Sec. V over the distribution $P(H)$ for a random hypergraph, given by the same Eq. (13) as for a random factor graph. Averaging Eqs. (33) and (32) over the distribution $P(H)$, we arrive at the self-consistency equations for W and V,

$$W = \sum_q \frac{qP(q)}{\langle q \rangle}[1 - (1-V)^{q-1}], \tag{44}$$

$$V = \sum_{m=2}^M p_H^{[m]} p_N^{m-1} \frac{mQ(m)}{\langle m \rangle}[1 - (1-W)^{m-1}]. \tag{45}$$

Averaging Eqs. (34) and (35) over $P(H)$, we obtain the expressions for the relative size $R = \langle r_i \rangle$ of the giant connected component in the hypergraph and for the probability $S = \langle s_\alpha \rangle$ that a hyperedge connects two nodes in the giant connected component,

$$R = p_N \sum_q P(q)[1 - (1-V)^q], \tag{46}$$

$$S = \sum_m p_H^{[m]} p_N^m Q(m)[1 - (1-W)^m]. \tag{47}$$

Assuming that $p_H^{[m]} = p_H$ is independent of m and that $\langle q^2 \rangle$ is finite and linearizing Eqs. (45) and (44), we arrive at the following criterion of the presence of the giant connected component in this problem:

$$p_N p_H \frac{\langle q(q-1) \rangle}{\langle q \rangle} \sum_m \frac{m(m-1)Q(m)}{\langle m \rangle} p_N^{m-2} > 1. \tag{48}$$

Introducing the generating functions $G_P(x) \equiv \sum_q Q(q)x^q$ of the degree distribution $P(q)$ and $G_{1P}(x) \equiv G_P'(x)/G_P(1)$, and $G_Q(x) \equiv \sum_m Q(m)x^m$ of the cardinality distribution $Q(m)$ and $G_{1Q}(x) \equiv G_Q'(x)/G_Q(1)$, we can rewrite this criterion in the compact form:

$$p_N p_H G_{1P}'(1) G_{1Q}'(p_N) > 1. \tag{49}$$

It is worthwhile to compare the criteria, Eqs. (17) and (48). The term p_N^{m-2} effectively introduces a cutoff in the cardinality distribution and makes the sum in Eq. (48) finite even if the first moment of this distribution diverges. Since $\sum_m m(m-1)Q(m)p_N^{m-2}/\langle m \rangle < \langle m(m-1) \rangle/\langle m \rangle$ if $p_N < 1$ and $Q(m) \neq \delta_{m,2}$, the node percolation threshold for this model of an uncorrelated hypergraph exceeds the hyperedge percolation

threshold in contrast to ordinary uncorrelated networks, as we have already shown for general locally treelike hypergraphs.

Substituting the degree and cardinality distributions $P(Q)$ and $Q(m)$ of the synthetic random hypergraph used for simulations in Fig. 3 into Eqs. (18) and (48) [or Eq. (49)], we obtain the exact values of the percolation thresholds for hyperedge percolation (and factor graph percolation) and hypergraph node percolation, $1/12 = 0.0833...$ and $12^{-1/3} = 0.437...$, respectively. The percolation thresholds observed in the simulations agree with these values.

Finally, let us consider Eqs. (45)–(49) in the special case of $Q(m) = \delta_{m,2}$, corresponding to an ordinary network, where each edge has cardinality 2. In this case, Eqs. (48) and (49) take the form of the standard Molloy-Reed criterion [42] (node percolation problem). Equation (45) gives $V = pW$, and the expression for the relative size of the giant connected component R, Eq. (46), also turns out to be standard for uncorrelated networks. Thus in this case our equations properly reduce to the known results for node percolation on uncorrelated networks.

B. Percolation on multiplex random hypergraphs

On a random multiplex hypergraph we can capture hypergraph percolation by introducing the probability W_m that a node belonging to a random hyperedge in layer m is occurs to be the root of an infinite tree when this hyperedge is deleted and the probability V_m that a random hyperedge of a node in layer m leads to the giant component. These probabilities are determined by the self-consistency equations:

$$W_m = \sum_{\mathbf{q}} \frac{q_m P(\mathbf{q})}{\langle q_m \rangle}\left[1 - \prod_{m'=2}^M (1-V_{m'})^{q_{m'}-\delta_{m,m'}}\right],$$
$$V_m = p_H^{[m]} p_N^{m-1}[1 - (1-W_m)^{m-1}], \tag{50}$$

which are Eqs. (32) and (33) averaged over the distribution $P(\vec{G})$ for a random multiplex hypergraph, described by the same Eq. (20) as for the corresponding random multiplex factor graph. The probabilities R and S that, respectively, a node and an hyperedge are in the giant component of the multiplex hypergraph are given by

$$R = p_N \sum_{\mathbf{q}} P(\mathbf{q})\left[1 - \prod_{m'=2}^M (1-V_{m'})^{q_{m'}}\right],$$
$$S = \sum_m Q(m) p_H^{[m]} p_N^m [1 - (1-W_m)^m], \tag{51}$$

where for this model of a random hypergraph,

$$Q(m) = \frac{\langle m \rangle}{m} \frac{\langle q_m \rangle}{\langle q \rangle}. \tag{52}$$

These equations lead to a continuous second-order phase transition for

$$\Lambda(p_N, \mathbf{p}_H) = 1, \tag{53}$$

where $\Lambda(p_N, \mathbf{p}_H)$ is the maximum eigenvalue of the matrix $\mathbf{G}$ of elements

$$G_{m,m'} = p_H^{[m]} p_N^{m-1}(m-1)\frac{\langle q_m(q_m - \delta_{m',m}) \rangle}{\langle q_m \rangle}. \tag{54}$$

The criterion of the existence of the giant connected component is this problem is $\Lambda(p_N, \mathbf{p}_H) > 1$. Inspecting the matrix elements in Eq. 54, we see that the node percolation threshold p_{cN} in this random hypergraph coincides with the hyperedge percolation threshold if each hyperedge of cardinality m is retained with probability $p_H^{[m]} = p_{cN}^{m-1} \leqslant p_{cN}$. If we remove hyperedges uniformly (independently of their cardinalities), then the resulting hyperedge percolation threshold p_{cH} must be lower than the maximum probability $p_H^{[m]} = p_{cN}^{m-1}$ for the nonuniform removal of edges, i.e., $p_{cH} < p_{cN}^{m_{\min}-1} \leqslant p_{cN}$. Thus the node percolation threshold for this model of an uncorrelated hypergraph exceeds the hyperedge percolation threshold, as we have already shown for general locally treelike hypergraphs.

VII. DISCUSSION AND CONCLUSION

We have formulated message-passing algorithms to investigate percolation on hypergraphs, we have studied the critical behavior of this process on arbitrary topologies and on two versions of the configuration model of random hypergraphs, and we have observed a large difference between the node and hyperedge percolation problems. The difference is particularly big when a hypergraph contains hyperedges with large cardinalities. We have shown that the node percolation threshold for a locally treelike hypergraph exceeds the hyperedge percolation threshold. This qualitative difference between node and hyperedge percolation on hypergraphs is in marked contrast to ordinary networks, where these two types of percolation do not differ much from each other, and

the node and hyperedge percolation thresholds coincide for a locally treelike network. Moreover, if the second moment of the degree distribution of nodes in a hypergraph is finite, then its node percolation threshold is finite for any cardinality distribution of hyperedges, even with the first moment diverging. That is, any fat-tailed cardinality distribution cannot lead to the hyper-resilience phenomenon in hypergraphs in contrast to their factor graphs, where the divergent second moment of a cardinality distribution guarantees zero percolation threshold and the hyper-resilience phenomenon.

The node percolation problem is the basic problem in which results for bipartite networks cannot be directly applied to their hypergraph counterparts. We have shown that the node percolation threshold of a locally treelike hypergraph exceeds this threshold for the corresponding factor graph. We have observed this effect also in a sparse real-world hypergraph whose structure is not treelike. One can indicate a number of other problems and processes, where such a direct mapping is impossible. These problems involve the removal of nodes. This occurs, in particular, during various pruning processes, including the k-core pruning process, cascading failures in multilayer interdependent networks, etc., and during disease spreading in various epidemic models.

Importantly, our equations of the message-passing algorithm do not assume the absence of correlations in hypergraphs. They allow a numerical treatment of the problem. The final formulas have been obtained for hypergraphs having no degree-degree correlations between different nodes. We suggest that an analytical treatment of correlated hypergraphs is also possible in the framework of our approach.

[1] A. R. Benson, D. F. Gleich, and J. Leskovec, Higher-order organization of complex networks, Science **353**, 163 (2016).

[2] F. Battiston, G. Cencetti, I. Iacopini, V. Latora, M. Lucas, A. Patania, J.-G. Young, and G. Petri, Networks beyond pairwise interactions: Structure and dynamics, Phys. Rep. **874**, 1 (2020).

[3] G. Bianconi, *Higher-Order Networks* (Cambridge University Press, Cambridge, 2021).

[4] S. Boccaletti, P. De Lellis, C. I. del Genio, K. Alfaro-Bittner, R. Criado, S. Jalan, and M. Romance, The structure and dynamics of networks with higher order interactions, Phys. Rep. **1018**, 1 (2023).

[5] L. Torres, A. S. Blevins, D. Bassett, and T. Eliassi-Rad, The why, how, and when of representations for complex systems, SIAM Rev. **63**, 435 (2021).

[6] G. Bianconi, Statistical physics of exchangeable sparse simple networks, multiplex networks, and simplicial complexes, Phys. Rev. E **105**, 034310 (2022).

[7] M. Barthelemy, Class of models for random hypergraphs, Phys. Rev. E **106**, 064310 (2022).

[8] P. L. Krapivsky, Random recursive hypergraphs, J. Phys. A: Math. Theor. **56**, 195001 (2023).

[9] F. Battiston, E. Amico, A. Barrat, G. Bianconi, G. Ferraz de Arruda, B. Franceschiello, I. Iacopini, S. Kéfi, V. Latora, Y. Moreno *et al.*, The physics of higher-order interactions in complex systems, Nat. Phys. **17**, 1093 (2021).

[10] S. Majhi, M. Perc, and D. Ghosh, Dynamics on higher-order networks: A review, J. R. Soc. Interface **19**, 20220043 (2022).

[11] B. C. Coutinho, A.-K. Wu, H.-J. Zhou, and Y.-Y. Liu, Covering problems and core percolations on hypergraphs, Phys. Rev. Lett. **124**, 248301 (2020).

[12] H. Sun and G. Bianconi, Higher-order percolation processes on multiplex hypergraphs, Phys. Rev. E **104**, 034306 (2021).

[13] H. Sun, F. Radicchi, J. Kurths, and G. Bianconi, The dynamic nature of percolation on networks with triadic interactions, Nat. Commun. **14**, 1308 (2023).

[14] J. Lee, K.-I. Goh, D.-S. Lee, and B. Kahng, (k, q)-core decomposition of hypergraphs, Chaos Solit. Fractals. **173**, 113645 (2023).

[15] H. Peng, C. Qian, D. Zhao, M. Zhong, X. Ling, and W. Wang, Disintegrate hypergraph networks by attacking hyperedge, J. King Saud Univ.–Comput. Info. Sci. **34**, 4679 (2022).

[16] H. Peng, C. Qian, D. Zhao, M. Zhong, J. Han, R. Li, and W. Wang, Message passing approach to analyze the robustness of hypergraph, arXiv:2302.14594.

[17] M. Mancastroppa, I. Iacopini, G. Petri, and A. Barrat, Hypercores promote localization and efficient seeding in higher-order processes, Nat. Commun. **14**, 6223 (2023).

[18] S. Thurner, P. Klimek, and R. Hanel, Schumpeterian economic dynamics as a quantifiable model of evolution, New J. Phys. **12**, 075029 (2010).

[19] R. Hanel, S. A. Kauffman, and S. Thurner, Phase transition in random catalytic networks, Phys. Rev. E **72**, 036117 (2005).

[20] F. Klimm, C. M. Deane, and G. Reinert, Hypergraphs for predicting essential genes using multiprotein complex data, J. Complex Networks **9**, cnaa028 (2021).

[21] J. Jost and R. Mulas, Hypergraph Laplace operators for chemical reaction networks, Adv. Math. **351**, 870 (2019).

[22] G. Bianconi and S. N. Dorogovtsev, following paper, Nature of hypergraph k-core percolation problems, Phys. Rev. E **109**, 014307 (2024).

[23] G. Bianconi, *Multilayer Networks: Structure and Function* (Oxford University Press, Oxford, 2018).

[24] M. E. J. Newman, Message passing methods on complex networks, Proc. R. Soc. A **479**, 20220774 (2023).

[25] A. K. Hartmann and M. Weigt, *Phase Transitions in Combinatorial Optimization Problems: Basics, Algorithms and Statistical Mechanics* (John Wiley & Sons, Weinheim, 2006).

[26] B. Karrer and M. E. J. Newman, Message passing approach for general epidemic models, Phys. Rev. E **82**, 016101 (2010).

[27] K. Zhao and G. Bianconi, Percolation on interacting, antagonistic networks, J. Stat. Mech.: Theory Exp. (2013) P05005.

[28] D. Cellai, S. N. Dorogovtsev, and G. Bianconi, Message passing theory for percolation models on multiplex networks with link overlap, Phys. Rev. E **94**, 032301 (2016).

[29] F. Radicchi and G. Bianconi, Redundant interdependencies boost the robustness of multiplex networks, Phys. Rev. X **7**, 011013 (2017).

[30] S. Watanabe and Y. Kabashima, Cavity-based robustness analysis of interdependent networks: Influences of intranetwork and internetwork degree-degree correlations, Phys. Rev. E **89**, 012808 (2014).

[31] G. T. Cantwell, A. Kirkley, and F. Radicchi, Heterogeneous message passing for heterogeneous networks, Phys. Rev. E **108**, 034310 (2023).

[32] G. Bianconi, Rare events and discontinuous percolation transitions, Phys. Rev. E **97**, 022314 (2018).

[33] G. T. Cantwell and M. E. J. Newman, Message passing on networks with loops, Proc. Natl. Acad. Sci. USA **116**, 23398 (2019).

[34] M. Mezard and A. Montanari, *Information, Physics, and Computation* (Oxford University Press, Oxford, 2009).

[35] S. Yoon, A. V. Goltsev, S. N. Dorogovtsev, and J. F. F. Mendes, Belief-propagation algorithm and the Ising model on networks with arbitrary distributions of motifs, Phys. Rev. E **84**, 041144 (2011).

[36] H. Sun, D. Saad, and A. Y. Lokhov, Competition, collaboration, and optimization in multiple interacting spreading processes, Phys. Rev. X **11**, 011048 (2021).

[37] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, Controllability of complex networks, Nature (London) **473**, 167 (2011).

[38] G. Menichetti, L. Dall'Asta, and G. Bianconi, Network controllability is determined by the density of low in-degree and out-degree nodes, Phys. Rev. Lett. **113**, 078701 (2014).

[39] G. Bianconi, H. Sun, G. Rapisardi, and A. Arenas, Message-passing approach to epidemic tracing and mitigation with apps, Phys. Rev. Res. **3**, L012014 (2021).

[40] M. Weigt and H. Zhou, Message passing for vertex covers, Phys. Rev. E **74**, 046110 (2006).

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

[42] M. Molloy and B. Reed, A critical point for random graphs with a given degree sequence, Random Struct. Algor. **6**, 161 (1995).

[43] 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).

[44] M. E. J. Newman, Random graphs with clustering, Phys. Rev. Lett. **103**, 058701 (2009).

[45] Y. S. Cho, J. S. Kim, J. Park, B. Kahng, and D. Kim, Percolation transitions in scale-free networks under the Achlioptas process, Phys. Rev. Lett. **103**, 135702 (2009).

[46] M. Li, R.-R. Liu, L. Lü, M.-B. Hu, S. Xu, and Y.-C. Zhang, Percolation on complex networks: Theory and application, Phys. Rep. **907**, 1 (2021).

[47] D. Lee, B. Kahng, Y. S. Cho, K.-I. Goh, and D.-S. Lee, Recent advances of percolation theory in complex networks, J. Korean Phys. Soc. **73**, 152 (2018).

[48] S. N. Dorogovtsev and J. F. F. Mendes, *The Nature of Complex Networks* (Oxford University Press, Oxford, 2022).

[49] G. Ferraz de Arruda, M. Tizzani, and Y. Moreno, Phase transitions and stability of dynamical processes on hypergraphs, Commun. Phys. **4**, 24 (2021).

[50] P. S. Chodrow, N. Veldt, and A. R. Benson, Generative hypergraph clustering: From blockmodels to modularity, Sci. Adv. **7**, eabh1303 (2021).

[51] C. Stewart III and J. Woon, Congressional committee assignments, 103rd to 114th Congresses, 1993–2017: House, Tech. Rep. (MIT, Cambridge, MA, 2008).