

## Random graphs with specific degree distribution and giant component size

Laurent Hébert-Dufresne <sup>1,2,3</sup> Márton Pósfai <sup>4</sup> and Antoine Allard <sup>3,5,1</sup>

<sup>1</sup>Vermont Complex Systems Institute, *University of Vermont*, Burlington, Vermont 05405, USA

<sup>2</sup>Department of Computer Science, *University of Vermont*, Burlington, Vermont 05405, USA

<sup>3</sup>Département de Physique, de Génie Physique et d'Optique, *Université Laval*, Québec (Québec), Canada G1V 0A6

<sup>4</sup>Department of Network and Data Science, *Central European University*, 1100 Vienna, Austria

<sup>5</sup>Centre Interdisciplinaire en Modélisation Mathématique, *Université Laval*, Québec (Québec), Canada G1V 0A6



(Received 17 May 2024; accepted 21 May 2025; published 3 June 2025)

Random networks are a powerful tool in the analytical modeling of complex networks as they allow us to write approximate mathematical models for diverse properties and behaviors of complex systems. These models are often used to study stochastic processes like percolation, where the giant connected component breaks down as edges are removed, yet they fail to properly account for the size of that component, even in a deterministic setting where all edges exist. Here, we introduce a simple conceptual step to account for such connectivity constraints in existing models. We distinguish between network neighbors based on two types of connections that can lead, or not, to the giant component, which we refer to as critical and subcritical degrees. The giant component is the largest unique component of a network that scales with the network's size under our model. It is analogous to many properties of interest, such as the largest epidemic possible on a contact network or the connectivity of an infrastructure network. Accounting explicitly for this component also allows us to capture important structural features of the network in a system of only one or two equations. When applied to sparse connected networks, we show that our approach compares favorably with the predictions of state-of-the-art models, like message passing, which require a number of equations that are linear in system size. We discuss potential applications of this simple framework for studying infrastructure networks where connectivity constraints are critical to the function of the system.

DOI: [10.1103/PhysRevResearch.7.L022050](https://doi.org/10.1103/PhysRevResearch.7.L022050)

*Introduction: Models of networks.* Random network models are a useful tool that allows simple mathematical analysis of the structure and properties of real complex networks [1]. Such models are typically parameterized with local connection rules enforcing structural constraints while leaving the rest of their structure random [2,3]. In theory, these rules should encode constraints that capture important features of real networks while the randomness reduces bias and preserves mathematical tractability. One of the most popular versions of this approach is the *Configuration Model* (CM), specified using the degree sequence of the network while assuming that edges are connected at random [4]. Many generalizations have been introduced over the years, including degree-degree correlations between neighbors [5], motifs [6,7], or more macroscopic structures like  $k$ -core or onion decomposition [8–11].

One notable use of this approach is the study of site or bond percolation on random networks, which is used to model the robustness of real networks to random failure or targeted attacks [12] and understand how they support dynamics such as the spread of epidemics or cascading failures [13]. In these

contexts, models can approximate the percolation threshold of real networks quite well [10,14–16], but typically fail to capture the actual size of the largest connected component far above the threshold. Given that the size of this component is related to the robustness against damage or small changes in edge functionality, it is unfortunate that most models dramatically fail in the regime where there is no damage or changes to the structure.

Here, we propose shifting the perspective on the information encoded in the stubs (i.e., the half-edges matched to generate the network)—from purely local to global—and illustrate how this can yield simple mathematical models that are more realistic at the global scale. Focusing on the problem of preserving the size of the largest connected component, we introduce a new computationally informed model based on the idea of critical connections. In a nutshell, our goal is to specify a critical substructure of interest identified computationally beforehand—like the largest connected component—and model edges that contribute to this critical substructure separately from other edges of the network. The resulting model can implicitly account for important network features, such as degree correlations or unknown statistics, that drive the connectivity of the substructure. By changing the perspective on the information encoded in stubs, thereby moving some of the complexity of the problem to computational preprocessing, we hope to provide a path toward rich yet simple new models that further the analytical study of complex networks.

---

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

*Random networks with a constrained giant component.* The core idea of our approach builds upon the classic CM by considering two types of connections: tagging some connections as critically important and therefore following specific rules, and others as less important, subcritical connections. This tagging procedure can be arbitrarily complex, but must result in a joint degree distribution  $P(s, c)$  over the “subcritical degree” ( $s$ ) and “critical degree” ( $c$ ) of every node. In a physical sense, only the latter will have the potential for supercritical effects under bond percolation, while subcritical connections can only contribute second-order effects.

The critical constraint we explore involves identifying connections that point toward the giant component of a network. Doing so will not only account for the size of that component but also for some of its structure. To see this, consider two nodes of degree 100, both in the giant component, but one has 99 neighbors of degree 1, while the other has none and is part of multiple loops over the giant component. Both nodes are connected to the giant component, but the second more robustly so, and our approach accounts for this difference.

*Critical CM.* We assume an undirected network where every connection between nodes  $i$  and  $j$  can be thought of as two directed edges  $(i, j)$  and  $(j, i)$ . In this case, critical connections are tagged through the following procedure:

- (1) Go over every directed edge  $(i, j)$  in the network.
- (2) Temporarily mask all edges pointing toward  $i$ , including  $(j, i)$ , to avoid backtracking.
- (3) Ask whether the component downstream from  $(i, j)$ —that is, the number of nodes  $k$  for which at least one path excluding  $i$  exists from  $j$  to  $k$ —is larger than some threshold value (we use 1% of the total number of nodes).
- (4) If yes, increase the critical degree  $c_i$  of  $i$  by 1. If not, increase the subcritical degree  $s_i$  of  $i$  by 1.
- (5) Restore all directed edges pointing toward  $i$ .
- (6) After going over all pairs, return a distribution  $P(s, c)$ .

Based on the joint critical-subcritical degree distribution  $P(s, c)$ , one can then construct a random network that is coherent with the tagging procedure, such that critical degrees in the original network data remain critical in a randomized version of the data. To do so, we rely on the concept of *excess degree*, that is, the number of *other* stubs available when one reaches a node through a given stub. We then define this Critical CM as follows:

- (1) A subcritical stub can only connect with the stub of a node whose excess critical degree is 0.
- (2) A critical stub can only connect with the stub of a node whose excess critical degree is at least 1.

Stubs are otherwise connected uniformly at random. Note that the second rule has a condition based on the excess critical degree of possible neighbors—that is, the number of critical stubs that leave that node—which assures that there exists a giant component that contains all nodes with at least one critical stub, therefore preserving the size of the giant component. The resulting model is illustrated in Fig. 1.

*Size of the giant component.* We now consider the infinite random network ensemble defined by the aforementioned connection rules and the joint distribution  $P(s, c)$  (with everything else random), and compute the expected size of the giant component. We follow a standard approach based on probability-generating functions (PGFs) [17] which, while

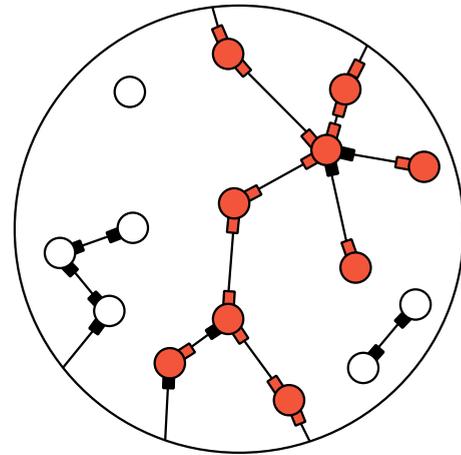


FIG. 1. Illustration of a Critical Configuration Model informed by the giant component of a network. Stubs are tagged as critical (red) and subcritical (black) based on whether they attach the node to the largest connected component. Nodes with a critical degree of at least 1 are then known to be in the giant component, but not all stubs in the giant component are critical, as some lead to dead ends.

very simple, give us an approximate solution to the problem. We highlight where the approximation comes into play and explain the slightly more involved formalism presented in the Supplemental Material [18].

When selecting a node at random in a random network drawn from the Critical CM, it will have a subcritical degree  $s$  and a critical degree  $c$  drawn from a joint distribution  $P(s, c)$ , which is generated by

$$G_0(x, y) = \sum_{s,c} P(s, c) x^s y^c. \quad (1)$$

When following any type of stubs, the type of stub through which we reach the neighboring node will be constrained by the excess degree of the reached node. Regardless, the joint degree pair of the reached node will be biased by either its subcritical degree  $s$  or critical degree  $c$  as per the *friendship paradox*, since a random edge is ten times more likely to reach a node of degree 10 than a node of degree 1. To that effect, we introduce the neighbor-generating function  $f(x, y)$ :

$$f(x, y) = \frac{\partial G_0(x, y)}{\partial x} + \frac{\partial G_0(x, y)}{\partial y}, \quad (2)$$

which is not normalized and therefore not a PGF.

More specifically, when following a subcritical stub, the connection rules defined previously state that we will reach a node proportionally to its number of subcritical stubs ( $s$ ), or of critical stubs if it has only one ( $c = 1$ ). The joint excess degree distribution of the reached node is therefore generated by  $G_1^s(x, y)$  and written as

$$G_1^s(x, y) = \frac{\sum_{s,c} P(s, c) (\delta_{c,1} x^s + s \delta_{c,0} x^{s-1})}{\sum_{s,c} P(s, c) (\delta_{c,1} + s \delta_{c,0})} = \frac{f(x, 0)}{f(1, 0)}, \quad (3)$$

where  $\delta_{a,b}$  is the Kronecker delta equal to 1 if  $a$  equals  $b$  and 0 otherwise. Note that  $G_1^s(x, y)$  does not depend on  $y$ , since the critical excess degree of the reached node is 0, as per the connection rules. Similarly, when following a critical stub,

the connection rules state that we reach nodes proportionally to their total degree as long as they have an excess critical degree of at least 1. The joint excess degree distribution of these reached nodes is therefore generated by

$$\begin{aligned} G_1^c(x, y) &= \frac{\sum_{s,c} P(s, c)(c\bar{\delta}_{c,1}x^s y^{c-1} + s\bar{\delta}_{c,0}x^{s-1}y^c)}{\sum_{s,c} P(s, c)(c\bar{\delta}_{c,1} + s\bar{\delta}_{c,0})} \\ &= \frac{f(x, y) - f(x, 0)}{f(1, 1) - f(1, 0)}, \end{aligned} \quad (4)$$

where  $\bar{\delta}_{a,b} \equiv 1 - \delta_{a,b}$ .

These equations are, unfortunately, only an approximation. The excess degree of the node reached by a random edge depends not only on the type of stub we follow, but also on the type of the node to which that stub belongs. Indeed, while the model did not specify it directly, there are three types of nodes: nodes with zero critical stubs, nodes with a single critical stub, and nodes with more than one critical stub. We provide an exact treatment to account for their different structural roles in the Supplemental Material [18], where we show that the simpler description presented here provides an excellent approximation, nevertheless.

Once we know the distribution of nodes at the end of random stubs, we can then solve for the giant component of the network under a bond percolation process where every edge exists independently with probability  $p$ . We denote  $u_s$  and  $u_c$  the probabilities that a subcritical or critical stub (respectively) does not lead to the giant component. The first,  $u_s$ , is fixed to 1 by construction, whereas the second is the solution of the following self-consistent equation:

$$u_c = 1 - p + pG_1^c(1, u_c). \quad (5)$$

In other words, the probability that a critical stub does not lead to the giant component must be equal to the probability that the remaining stubs of the reached node do not lead to the giant component either. This occurs if the critical stub belongs to an edge that has been removed (probability  $1 - p$ ), or, when the edge has not been removed, if none of the *other* critical stubs of the reached node lead to the giant component either (probability  $pG_1^c(1, u_c)$ ). References [10,13,21] provide more details on this self-consistent argument.

Having solved Eq. (5), we may now compute the size of the giant component as the fraction of nodes for which at least one critical stub leads to it. Recalling that an individual critical stub does not lead to the giant component with probability  $u_c$ , we find:

$$S = \sum_{s,c} P(s, c)[1 - u_c^c] = 1 - G_0(1, u_c). \quad (6)$$

Notice that  $u_c = 0$  is the only nontrivial solution to Eq. (5) when  $p = 1$ , meaning that the nodes with at least one critical stub will be connected to the giant component with probability 1 in the limit of infinite networks in the absence of damage, as expected. In other words, our model exactly preserves the size of the largest connected component perfectly when all edges exist, and approximates its size under a percolation process ( $p < 1$ ).

*Connections with other models.* Most, if not all, random network models can be seen as mathematical frameworks used to compress complex network data based on some important

features or constraints. Our approach differs from common models in two ways.

First, instead of calibrating the model based on observed local properties from network data (e.g., degree distribution or degree-degree correlations) and validating the model by predicting the robustness of the corresponding network, the critical CM combines the calibration and validation steps. It uses information about how every node is embedded within the components found in the network data, thereby calibrating the model based on global information (i.e., the size of the largest connected component), and then attempts to extrapolate to the perturbed version of the same network.

Second, calibrating the critical CM requires more computational preprocessing than the CM. It does not merely count edges around every node as in the CM—an operation whose complexity scales as the number of edges  $E$ —but tags them all based on some criterion for “critical” connections, which can be costly to evaluate. A naive implementation of the procedure described would scale as  $E$  for every edge, and therefore as  $E^2$  for the full preprocessing of a network of  $E$  edges.

There is therefore an important conceptual jump from the CM to the critical CM, since we now distinguish types of connections *and* have two different types of stubs informed by *global* connectivity patterns. This is similar to the difference between the CM and the popular Stochastic Block Model, which is meant to capture mesoscopic block or community structure in networks [25], which does not directly translate to a good description of dynamics like percolation [10]. For the critical CM, because we do not need a complicated inference procedure to tag stubs and because the role of one type of stub is fixed by construction (recall that  $u_s = 1$ ), the resulting model is mathematically as complex as the CM, as we only need to solve for a single polynomial self-consistent quantity.

By contrast, state-of-the-art approaches rely on a similar conceptual jump but greatly increase the complexity of the resulting mathematics. The Message Passing Approach (MPA) is based on the idea that all stubs and all edges are distinguishable, and therefore tags the stub from node  $i$  leading to node  $j$  as a unique edge type  $i \rightarrow j$  [22,26]. It then follows a similar calculation: Assuming an infinite number of nodes of any type (i.e., ignoring feedback through loops) whose degree sequence in types of stubs is explicitly given by the adjacency matrix of the true network, we can write a system of  $2E$ -coupled self-consistent equations describing percolation on a network with  $E$  edges.

Also, depending on the joint distribution  $P(s, c)$ , the connection rules listed in Sec. II may enforce some level of degree correlation between neighbors. For instance, if only low-degree nodes have  $c = 0$ , then the critical CM will generate networks with degree correlations differing from that of the classic CM. In a sense, one could also view the critical CM as a crude version of the correlated configuration model [5] or of the Layered Correlated Configuration Model [10]. However, our preliminary investigation did not reveal any systematic relationship between the critical CM and these two models (see Supplemental Material [18]).

The critical CM therefore lies somewhere in between the CM and MPA. It can be seen as a fine-grained version of the CM where stubs are distinguished based on the structure of

the nonbacktracking component to which they lead. Alternatively, the critical CM can be seen as a coarse-graining of the MPA where we compress the  $2E$  types of unique connections into only two, again based on the nonbacktracking component to which they lead. By comparing the critical CM to the CM and MPA, we hope to justify the development of more creative random network models. In other words, we can, in theory, efficiently compress mathematical models using network analysis as a preprocessing tool.

**Results.** We first compare the results of the critical CM to other models and simulations based on the giant component of a common corporate ownership dataset, shown in Fig. 2. This network was chosen because the CM completely fails to match simulations of a percolation process, in large part because the giant component contains a gigantic hub in its periphery. The CM completely overestimates the robustness of the giant component by underestimating the epidemic threshold, while conversely underestimating the final relative size of this giant component at  $p = 1$ . By comparing to simulations on a connected subset of the CM [20], we can show that by capturing the complete size of the giant component, one also captures the higher-than-expected percolation threshold due to the prevalence of negative degree correlations required for this connectivity (hubs have to connect multiple low-degree nodes). Interestingly, the critical CM does a much better job at capturing the unique structure of this network by accounting not only for the size of the giant component but also for implicit correlations between the degree of nodes and their role in the cohesion of the giant component (the so-called *critical degree*). This highlights how our analytical model differs from a uniform connected subset of the CM, but it is an effective model that captures how the degree sequence of the network interacts with its giant component through the correlations between the degree of a node and its critical degree. In fact, the critical CM mimics results from the MPA very closely while decreasing the number of equations by a factor equal to twice the number of edges (1 versus 9304 self-consistent equations).

We further compare the Critical CM to the classic CM and to simulations for four other connected networks in Fig. 3. These networks were chosen to highlight how the previous results hold in technological networks (e.g., the structure of the Internet) where the connectivity of the empirical networks might be surprising to the CM based on their degree distributions alone. However, in denser networks such as protein-protein interaction networks and social networks, the critical CM offers less of a gain and falls closer to the classic CM. The usefulness of our approach therefore relies on the structure of interest, in this case, the giant component, being surprising to classic random networks.

We can build on this intuition and look at the behavior of the critical CM on regular trees where all nodes have the same degree, up to a certain number of layers  $L$  where all nodes have degree one to close the tree. These structures are interesting for three reasons. First, our PGFs assume a treelike structure [most notably in Eq. (5), where we assume all stubs leaving a node are independent] but do not perform well on trees because of their highly constrained structure [10]. Second, the fact that trees are connected as a single component is surprising under the lens of most random net-

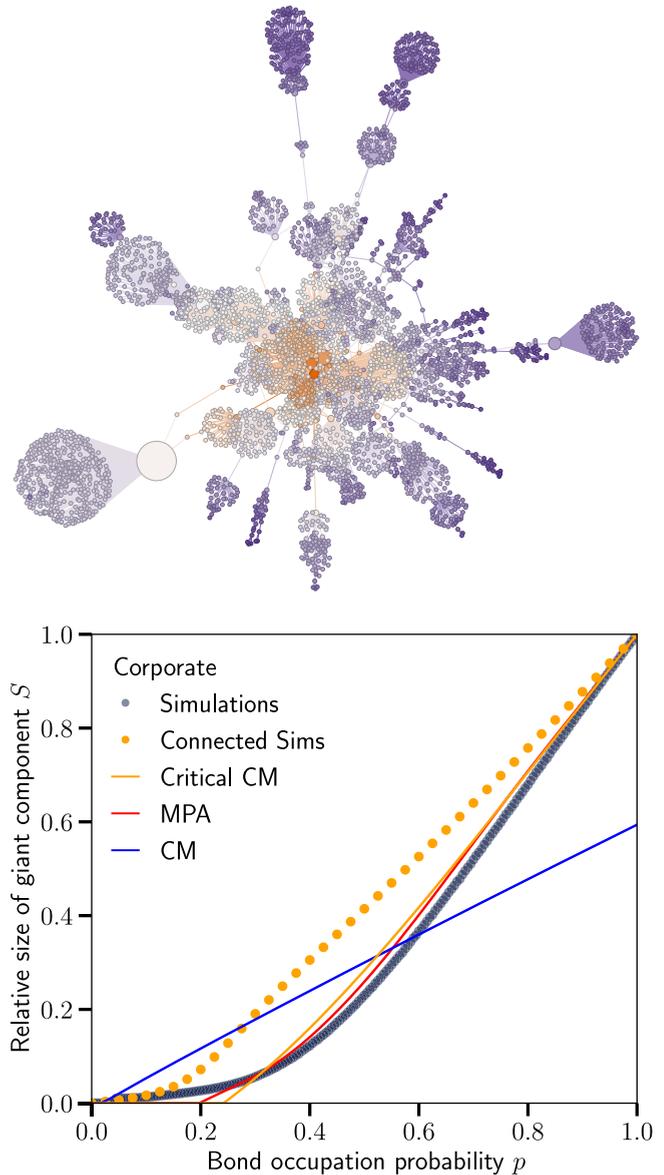


FIG. 2. (Top) The giant component of a corporate ownership network [19]. Node size corresponds to degree (linearly from 1 to 552) and node color corresponds to closeness centrality (inverse of the average distance to other nodes). This is an interesting network structure in part because of its extremely skewed degree distribution, and because its most connected node has a much higher degree than its second most connected (552 vs. 178), but the former is located in the periphery of the network while the latter is located in its core. (Bottom) We simulate bond percolation on this network (Simulations) and compare the results against three analytical models: the Message Passing Approach (MPA; 9304 self-consistent equations), the Configuration Model (CM; 1 self-consistent equation), and our Critical Configuration Model (Critical CM; 1 self-consistent equation). We also compare the results against simulations on a connected subset of the CM (Connected Sims [20]) to show that the critical CM does more than capture the size of the connected component at  $p = 1$ .

work models, given their high number of nodes of degree one (which explains why configuration models do not perform well on trees). Third, the MPA is exact on trees, where it will see the finite size of the system and exactly predict the

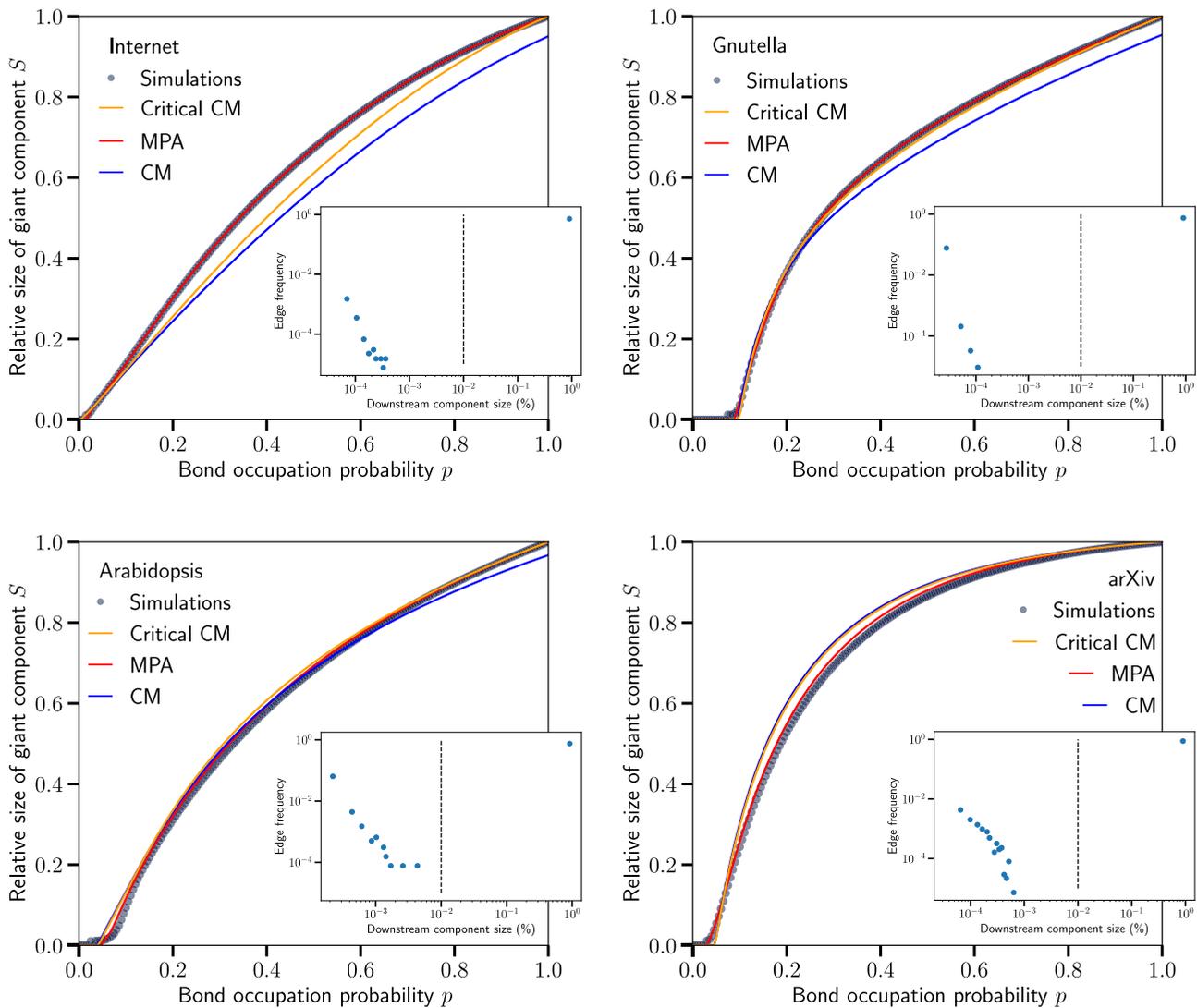


FIG. 3. Bond percolation on the Critical Configuration Model (Critical CM) and Configuration Model (CM) compared with simulations on the giant connected component of: (top left) the structure of the Internet at the level of autonomous systems [22], (top right) the peer-to-peer Gnutella network [23], (bottom left) the Arabidopsis protein-protein interaction network [24], and (bottom right) a coauthorship network from the arXiv preprint archive [17]. The insets show the (binned) distribution of downstream component sizes per directed edge. The distribution is highly bimodal, with edges either leading to a very large macroscopic fraction of the network or to a very small subset of the network. The critical CM is therefore highly robust to the choice of the threshold for criticality, as long as that threshold lies somewhere between 1% and 90% of the entire system.

distribution of small component sizes while predicting the lack of supercritical giant component [26].

The results of the critical CM on regular trees are shown in Fig. 4. As trees grow, only nodes in the first or second layer around the root will have more than one critical stub; all other nodes will have a single critical stub pointing toward the root of the tree. The critical CM therefore creates a very small core of central nodes around which we have directed chains of nodes, all with a single critical stub pointing toward the core. These chains make up the vast majority of nodes, and are always subcritical for  $p < 1$  while only the small core can be supercritical. As the relative size of the core vanishes, the critical CM therefore converges to the same prediction as the MPA, but for a conceptually different reason. While MPA sees the finite size of the system, the critical CM instead sees

chains of critical stubs that are subcritical as soon as one edge is removed. Both predict the absence of a phase transition below  $p = 1$ .

*Conclusion.* Random network models are used for multiple reasons across network science and beyond. While many problems and dynamical systems are impossible to solve on a fixed network structure, they are often solvable on infinite random network ensembles. When these models work well, we build intuition about which structural constraints matter the most for the problems and dynamics of interest. This, in turn, also allows us to control for important features and therefore provide important null models for network analyses. Indeed, we often need to ask how surprising a network feature is given some other metrics. This is why modularity or even degree assortativity control for degree distribution, using

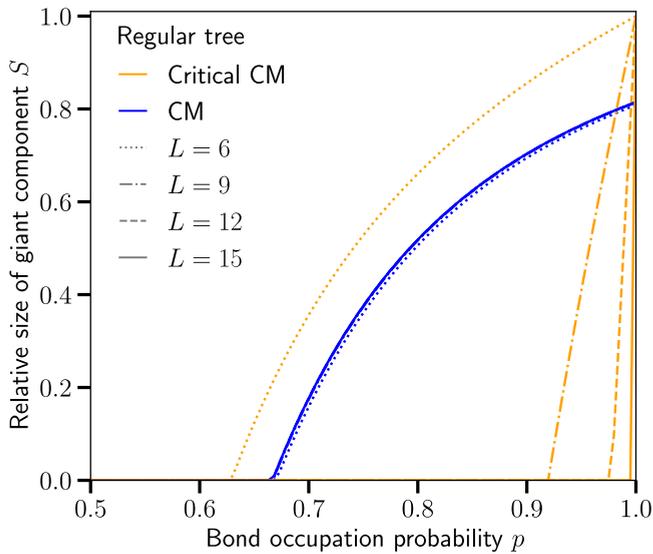


FIG. 4. Comparison of the behavior of the CM and critical CM parametrized by data coming from perfect trees of coordination number 3 and different depths  $L$ . As the trees grow, the critical CM converges to directed chains where all nodes have a single critical stub pointing toward the core of the tree. Those chains preserve the treelike structure of the original network and converge to a system of equations that predicts the absence of the giant component for any  $T < 1$ . This is the same prediction as the MPA, which effectively sees the finite size of the system and therefore does not predict any giant (infinite) connected component [26].

the CM as an underlying null model. Through this pipeline, improvements in random network models improve much of network science.

Given that network scientists often care about global properties of network structures, such as their connectivity and robustness, we need random network models that can control for these features. Unfortunately, most models are only built around simple local connection rules, leaving a significant gap in the network science toolbox. While we certainly do not claim that the critical CM introduced here should become a standard null model for network analyses and mathematical descriptions, it highlights the potential for rethinking the information encoded in the stubs to design models that reproduce ever more realistic network structures. As illustrated here, such a change in perspective may involve a more complicated parametrization requiring computational preprocessing, while nevertheless having a mathematical description that remains parsimonious [27].

Future variations of our approach could consider a parametrization of random networks based on statistics of their minimum spanning tree or dynamical features relevant to network control. Along these lines, we have explored similar models constrained by  $k$ -core decomposition in Refs. [8–10]. There remain many more models to explore. This line of research can be summarized as two simple questions: What features do we care about in a given network structure, and how can we best describe networks using these features? And when the answer to the first question is the global connectivity of a network, variations of the critical CM presented here should prove useful.

*Acknowledgments.* This work was supported by the Vermont Complex Systems Institute (LHD), ERC Grant No. 810115-DYNASNET (MP), the Natural Sciences and Engineering Research Council of Canada (AA), and the Sentinelle Nord program of Université Laval funded by the Canada First Research Excellence Fund (AA).

- [1] A. C. C. Coolen, A. Annibale, and E. S. Roberts, *Generating Random Networks and Graphs* (Oxford University Press, Oxford, 2017).
- [2] J. Park and M. E. J. Newman, Statistical mechanics of networks, *Phys. Rev. E* **70**, 066117 (2004).
- [3] P. van der Hoorn, G. Lippner, and D. Krioukov, Sparse maximum-entropy random graphs with a given power-law degree distribution, *J. Stat. Phys.* **173**, 806 (2018).
- [4] B. K. Fosdick, D. B. Larremore, J. Nishimura, and J. Ugander, Configuring random graph models with fixed degree sequences, *SIAM Rev.* **60**, 315 (2018).
- [5] A. Vázquez and Y. Moreno, Resilience to damage of graphs with degree correlations, *Phys. Rev. E* **67**, 015101(R) (2003).
- [6] B. Karrer and M. E. J. Newman, Random graphs containing arbitrary distributions of subgraphs, *Phys. Rev. E* **82**, 066118 (2010).
- [7] 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: Math. Theor.* **45**, 405005 (2012).
- [8] 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).
- [9] L. Hébert-Dufresne, J. A. Grochow, and A. Allard, Multi-scale structure and topological anomaly detection via a new network statistic: The onion decomposition, *Sci. Rep.* **6**, 31708 (2016).
- [10] A. Allard and L. Hébert-Dufresne, Percolation and the effective structure of complex networks, *Phys. Rev. X* **9**, 011023 (2019).
- [11] F. Thibault, L. Hébert-Dufresne, and A. Allard, On the uniform sampling of the configuration model with centrality constraints, [arXiv:2409.20493](https://arxiv.org/abs/2409.20493).
- [12] R. Albert, H. Jeong, and A.-L. Barabási, Error and attack tolerance of complex networks, *Nature (London)* **406**, 378 (2000).
- [13] M. E. J. Newman, Spread of epidemic disease on networks, *Phys. Rev. E* **66**, 016128 (2002).
- [14] S. Melnik, A. Hackett, M. A. Porter, P. J. Mucha, and J. P. Gleeson, The unreasonable effectiveness of tree-based theory for networks with clustering, *Phys. Rev. E* **83**, 036112 (2011).
- [15] F. Radicchi, Predicting percolation thresholds in networks, *Phys. Rev. E* **91**, 010801(R) (2015).
- [16] F. Radicchi and C. Castellano, Beyond the locally treelike approximation for percolation on real networks, *Phys. Rev. E* **93**, 030302(R) (2016).
- [17] M. E. J. Newman, The structure of scientific collaboration networks, *Proc. Natl. Acad. Sci. USA* **98**, 404 (2001).

- [18] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevResearch.7.L022050> for additional analysis of the model.
- [19] K. Norlen, G. Lucas, M. Gebbie, and J. Chuang, EVA: Extraction, visualization and analysis of the telecommunications and media ownership network, in *Proceedings of International Telecommunications Society 14th Biennial Conference (ITS2002)* (Seoul, Korea, 2002), pp. 27–129.
- [20] J. H. Ring, J.-G. Young, and L. Hébert-Dufresne, Connected graphs with a given degree sequence: Efficient sampling, correlations, community detection and robustness, in *Proceedings of NetSci-X 2020: Sixth International Winter School and Conference on Network Science, Tokyo, Japan* (Springer, Cham, 2020), pp. 33–47.
- [21] A. Allard, L. Hébert-Dufresne, J.-G. Young, and L. J. Dubé, General and exact approach to percolation on random graphs, *Phys. Rev. E* **92**, 062807 (2015).
- [22] B. Karrer, M. E. J. Newman, and L. Zdeborová, Percolation on sparse networks, *Phys. Rev. Lett.* **113**, 208702 (2014).
- [23] R. Matei, A. Iamnitchi, and P. Foster, Mapping the Gnutella network, *IEEE Internet Comput.* **6**, 50 (2002).
- [24] M. Dreze, A.-R. Carvunis, B. Charloteaux, M. Galli, S. J. Pevzner, M. Tasan, Y.-Y. Ahn, P. Balumuri, A.-L. Barabasi, V. Bautista, P. Braun, D. Byrdsong, H. Chen, J. D. Chesnut, M. E. Cusick, J. L. Dangl, C. de los Reyes, A. Dricot, M. Duarte, J. R. Ecker *et al.*, Evidence for network evolution in an arabidopsis interactome map, *Science* **333**, 601 (2011).
- [25] T. P. Peixoto, Nonparametric Bayesian inference of the microcanonical stochastic block model, *Phys. Rev. E* **95**, 012317 (2017).
- [26] A. Allard and L. Hébert-Dufresne, On the accuracy of message-passing approaches to percolation in complex networks, [arXiv:1906.10377](https://arxiv.org/abs/1906.10377).
- [27] L. Hébert-Dufresne, J.-G. Young, A. Daniels, A. Kirkley, and A. Allard, Network compression with configuration models and the minimum description length, *Phys. Rev. E* **110**, 034305 (2024).

# Random graphs with specific degree distribution and giant component size

## Supplemental material

Laurent Hébert-Dufresne, Márton Pósfai and Antoine Allard

May 19, 2025

**Abstract.** The first part of this supplemental material presents the results of our preliminary investigation into correlations included in the Critical Configuration Model (CritCM). The second part of the document investigates the validity of the approximation behind the derivation of Eqs. (3)–(4) in the main text using the multitype formalism from Ref. [1].

## Contents

<b>1</b>	<b>Correlations in the Critical Configuration Model</b>	<b>2</b>
<b>2</b>	<b>Exact description</b>	<b>5</b>
<b>3</b>	<b>References</b>	<b>13</b>

## 1 Correlations in the Critical Configuration Model

As mentioned in the main text, it is clear that some level of degree correlation can be enforced by the joint distribution  $P(s, c)$  and the connection rules listed in Sec. II.A. As a preliminary investigation, we compare the predictions for the size of the giant component under random edge removal obtained using following five models:

- ▷ the Configuration Model (CM) [2] which fixes the degree distribution;
- ▷ the Correlated Configuration Model (CCM) [3] which fixes the degree distribution as well as the degree-degree correlations;
- ▷ the Critical Configuration Model (main text) which fixes the degree distribution as well as the size of the giant component when edges have not been removed (i.e. no percolation);
- ▷ the Layered Correlated Configuration Model (LCCM) [4] which fixes the degree distribution, the  $k$ -core and onion decompositions [5], as well as degree-layer-degree-layer correlations;
- ▷ the Message Passing Approach (MPA) [6] which fixes the correlations imposed by the adjacency matrix but neglects the correlations between the state of the neighbors of any given vertex [7].

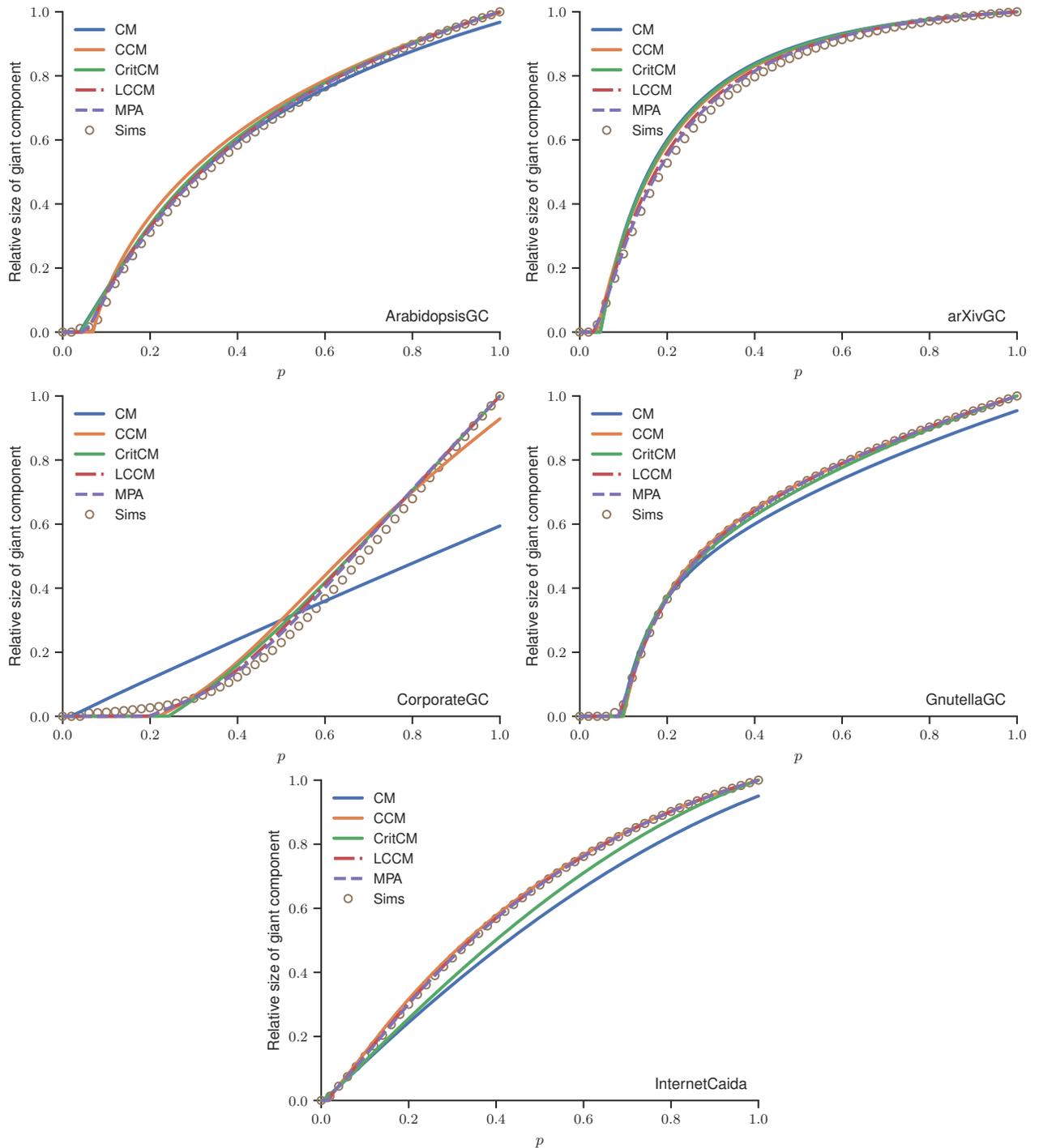
The results of a preliminary investigation on the matter are presented in Supp. Tables 1 and 2 and in Supp. Figs. 1 and 2. Unfortunately, no clear relationship with the density or the assortativity can be identified when it comes to predict the role of density or correlation on the accuracy of the Critical Configuration Model. Further investigation is needed, which goes beyond the scope of this work.

Supp. Table 1: Network datasets used for this investigation, as well as a few global properties. Note that the suffix GC is added to denote network datasets for which we only considered the largest connected component.

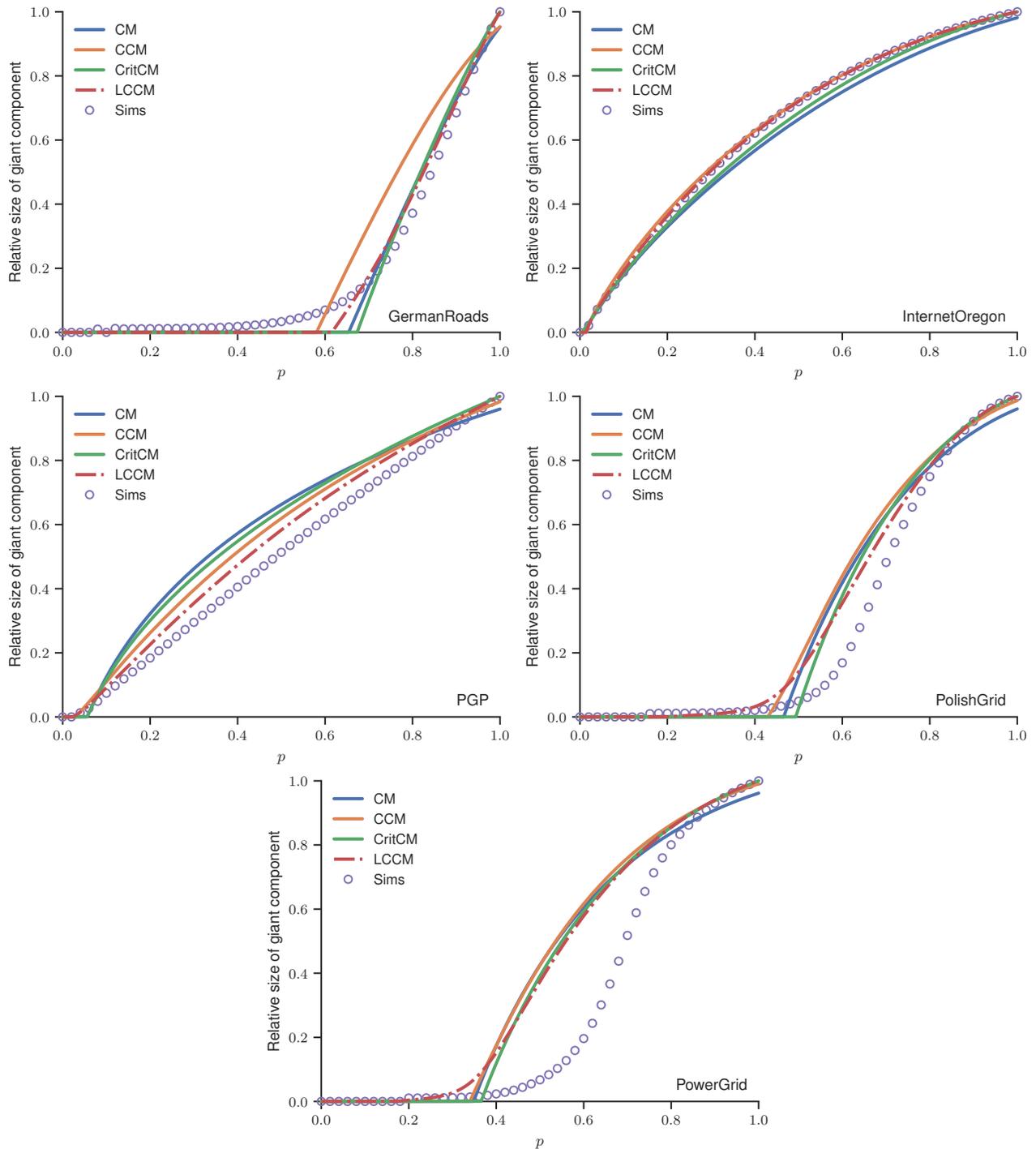
Dataset name	number of vertices	density ( $\times 10^{-3}$ )	assortativity	Ref.
ArabidopsisGC	4519	1.050	-0.197	[8]
arXivGC	28502	0.304	0.138	[9]
CorporateGC	4475	0.465	-0.185	[10]
GnutellaGC	36646	0.132	-0.104	[11]
InternetCaida	26475	0.152	-0.195	[6]
GermanRoadsGC	1168	1.824	0.025	[12]
InternetOregon	11461	0.498	-0.164	[13]
PGP	10680	0.426	0.238	[14]
PolishGrid	3374	0.715	0.050	[15]
PowerGrid	4941	0.540	0.003	[16]

Supp. Table 2: Average absolute difference between the prediction of the size of the giant component between all pairs of models considered in the study. The name of each model has been shortened as CM→C, CCM→CC, CritCM→Cr, LCCM→LCC and MPA→M.

Dataset name	C-CC	C-Cr	C-LCC	CC-Cr	CC-LCC	Cr-LCC	C-M	CC-M	Cr-M	LCC-M
ArabidopsisGC	0.0265	0.0153	0.0133	0.0114	0.0148	0.0055	0.0141	0.0175	0.0082	0.0030
arXivGC	0.0062	0.0029	0.0151	0.0037	0.0095	0.0124	0.0185	0.0130	0.0158	0.0037
CorporateGC	0.1273	0.1332	0.1336	0.0149	0.0180	0.0050	0.1340	0.0214	0.0083	0.0035
GnutellaGC	0.0369	0.0270	0.0347	0.0099	0.0034	0.0084	0.0346	0.0036	0.0083	0.0003
InternetCaida	0.0774	0.0331	0.0720	0.0442	0.0054	0.0389	0.0712	0.0061	0.0381	0.0008
GermanRoadsGC	0.0479	0.0085	0.0084	0.0502	0.0455	0.0127	—	—	—	—
InternetOregon	0.0428	0.0159	0.0374	0.0269	0.0054	0.0215	—	—	—	—
PGP	0.0319	0.0165	0.0550	0.0225	0.0242	0.0440	—	—	—	—
PolishGrid	0.0170	0.0189	0.0254	0.0225	0.0265	0.0214	—	—	—	—
PowerGrid	0.0110	0.0170	0.0217	0.0145	0.0185	0.0110	—	—	—	—



Supp. Fig. 1: Relative size of the giant connected component as predicted by the Configuration Model (CM) [2], the Correlated Configuration Model (CCM) [3], the Critical Configuration Model (CritCM; see main text) and the Message Passing Approach (MPA) [6] as a function of the edge occupation probability  $p$ . The relative size of the largest connected component as obtained from the numerical simulation of bond percolation of the original network datasets is also shown (Sims). See Table 1 for details about the datasets.



Supp. Fig. 2: Relative size of the giant connected component as predicted by the Configuration Model (CM) [2], the Correlated Configuration Model (CCM) [3] and the Critical Configuration Model (CritCM; see main text) as a function of the edge occupation probability  $p$ . The relative size of the largest connected component as obtained from the numerical simulation of bond percolation of the original network datasets is also shown (Sims). See Table 1 for details about the datasets.

## 2 Exact description

This section presents an exact description of the formalism introduced in the main text through the multitype formalism of Ref. [1], and discusses the conditions for which the two approaches coincide.

### Joint degree distribution

The joint degree distribution  $P(s, c)$  specifies the fraction of vertices with  $s$  *subcritical stubs* and  $c$  *critical stubs*. It is the main mathematical object that must be provided to the model.

### Types of vertices

The vertices are divided into three types based on their number of critical stubs. Type 0 vertices do not have critical stubs and are *de facto* excluded from the giant component identified at the preprocessing stage of the Critical Configuration Model. They correspond to a fraction

$$w_0 = \sum_s \sum_c P(s, c) \delta_{c0} = \sum_s P(s, 0) \quad (1a)$$

of all vertices. Type 1 vertices have only 1 critical stub and correspond to a fraction

$$w_1 = \sum_s \sum_c P(s, c) \delta_{c1} = \sum_s P(s, 1) \quad (1b)$$

of all vertices. Type 2 vertices have at least 2 critical stubs and correspond to a fraction

$$w_2 = \sum_s \sum_c P(s, c) (1 - \delta_{c0}) (1 - \delta_{c1}) = \sum_s \sum_{c>1} P(s, c) \quad (1c)$$

of all vertices. Clearly,  $w_0 + w_1 + w_2 = 1$  since the joint degree distribution  $P(s, c)$  is normalized.

### Types of connections

There are 4 ways vertices can be connected to each other according to the rules described in the main text:

- ▷ (0s–0s): vertices of type 0 are connected through their subcritical stubs;
- ▷ (1c–1s): vertices of type 1 are connected through one critical stub and one subcritical stub;
- ▷ (1c–2s): vertices of types 1 and 2 are connected through one critical stub (type 1) and one subcritical stub (type 2);
- ▷ (2c–2c): vertices of type 2 are connected through their critical stubs.

It is convenient to define the following first-order auxiliary quantities

$$M_{0s} = \sum_s \sum_c sP(s, c) \delta_{c0} = \sum_s sP(s, 0) \quad (2a)$$

$$M_{1s} = \sum_s \sum_c sP(s, c) \delta_{c1} = \sum_s sP(s, 1) \quad (2b)$$

$$M_{1c} = \sum_s \sum_c cP(s, c) \delta_{c1} = \sum_s P(s, 1) = w_1 \quad (2c)$$

$$M_{2s} = \sum_s \sum_c sP(s, c) (1 - \delta_{c0}) (1 - \delta_{c1}) = \sum_s \sum_{c>1} sP(s, c) \quad (2d)$$

$$M_{2c} = \sum_s \sum_c cP(s, c) (1 - \delta_{c0}) (1 - \delta_{c1}) = \sum_s \sum_{c>1} cP(s, c) , \quad (2e)$$

which are proportional to the fraction of all stubs that are of a given type and emanating from a given type of vertices<sup>1</sup>. The average number of each type of stub per vertex (i.e., the *typed degree*) is obtained by dividing by the fraction of vertices of the corresponding type<sup>2</sup>.

It is also convenient to introduce the second-order auxiliary quantities

$$M_{0s0s} = \sum_s \sum_c s(s-1)P(s,c)\delta_{c0} = \sum_s s(s-1)P(s,0) \quad (3a)$$

$$M_{2s2c} = \sum_s \sum_c scP(s,c)(1-\delta_{c0})(1-\delta_{c1}) = \sum_s \sum_{c>1} scP(s,c) \quad (3b)$$

$$M_{2c2c} = \sum_s \sum_c c(c-1)P(s,c)(1-\delta_{c0})(1-\delta_{c1}) = \sum_s \sum_{c>1} c(c-1)P(s,c). \quad (3c)$$

### Generating functions

We define the following three generating functions associated with the joint degree distribution of each type of vertices:

$$G_0(\mathbf{x}) = \frac{1}{w_0} \sum_s P(s,0)[1-T+Tx_{0s}]^s \quad (4a)$$

$$G_1(\mathbf{x}) = \frac{1}{w_1} \sum_s P(s,1)[1-T+Tx_{1c}]^s \left[ 1-T + \frac{T(M_{1s}x_{1s} + M_{2s}x_{2s})}{M_{1s} + M_{2s}} \right] \quad (4b)$$

$$G_2(\mathbf{x}) = \frac{1}{w_2} \sum_s \sum_{c>1} P(s,c)[1-T+Tx_{1c}]^s [1-T+Tx_{2c}]^c, \quad (4c)$$

where edges have been kept (removed) with probability  $T$  ( $1-T$ ). We also introduced  $\mathbf{x} = (x_{0s}, x_{1s}, x_{1c}, x_{2s}, x_{2c})$ , where the subscripts keep track of the type of the neighbors and of the type of stubs through which it is reached. The underlying assumption behind the last bracket in Eq. (4b) is that stubs are chosen uniformly when creating edges<sup>3</sup>.

We also introduce the following generating functions associated with the joint excess degree distribution:

$$F_{0s}(\mathbf{x}) = \frac{1}{M_{0s}} \sum_s sP(s,0)[1-T+Tx_{0s}]^{s-1} \quad (5a)$$

$$F_{1s}(\mathbf{x}) = \frac{1}{M_{1s}} \sum_s sP(s,1)[1-T+Tx_{1c}]^{s-1} \left[ 1-T + \frac{T(M_{1s}x_{1s} + M_{2s}x_{2s})}{M_{1s} + M_{2s}} \right] \quad (5b)$$

$$F_{1c}(\mathbf{x}) = \frac{1}{M_{1c}} \sum_s P(s,1)[1-T+Tx_{1c}]^s \quad (5c)$$

$$F_{2s}(\mathbf{x}) = \frac{1}{M_{2s}} \sum_s \sum_{c>1} sP(s,c)[1-T+Tx_{1c}]^{s-1} [1-T+Tx_{2c}]^c \quad (5d)$$

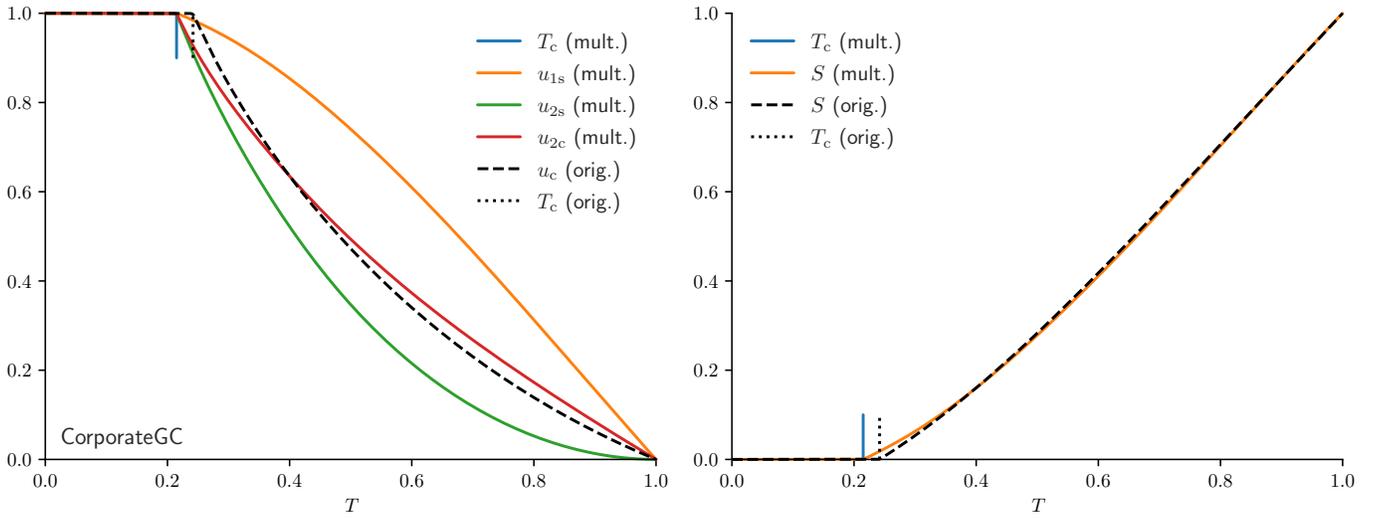
$$F_{2c}(\mathbf{x}) = \frac{1}{M_{2c}} \sum_s \sum_{c>1} cP(s,c)[1-T+Tx_{1c}]^s [1-T+Tx_{2c}]^{c-1}, \quad (5e)$$

where the subscript indicates the type of the vertex and the type of the stubs via which it has been reached.

<sup>1</sup>For example,  $M_{2s}$  is proportional to the fraction of all stubs that are subcritical and emanating from vertices of type 2.

<sup>2</sup>For example, the average number of subcritical stubs emanating from vertices of type 2 is  $M_{2s}/w_2$ .

<sup>3</sup>This is a basic assumption in most if not all variants of the configuration model.



Supp. Fig. 3: The giant component of a corporate ownership network [10] for which  $w_1 = 0.93$ ,  $\alpha_1 = 5.46$  and  $\alpha_2 = 0.89$ .

### Size of the extensive component

We define  $\mathbf{u} = (u_{0s}, u_{1s}, u_{1c}, u_{2s}, u_{2c})$  as the probabilities that a vertex reached via one specific type of stub does not lead to the extensive component<sup>4</sup>. These probabilities are the solution of the following set of self-consistent equations [1]

$$u_{0s} = F_{0s}(\mathbf{u}) = \frac{1}{M_{0s}} \sum_s sP(s, 0) [1 - T + Tu_{0s}]^{s-1} \quad (6a)$$

$$u_{1s} = F_{1s}(\mathbf{u}) = \frac{1}{M_{1s}} \sum_s sP(s, 1) [1 - T + Tu_{1c}]^{s-1} \left[ 1 - T + \frac{T(M_{1s}u_{1s} + M_{2s}u_{2s})}{M_{1s} + M_{2s}} \right] \quad (6b)$$

$$u_{1c} = F_{1c}(\mathbf{u}) = \frac{1}{M_{1c}} \sum_s P(s, 1) [1 - T + Tu_{1c}]^s \quad (6c)$$

$$u_{2s} = F_{2s}(\mathbf{u}) = \frac{1}{M_{2s}} \sum_s \sum_{c>1} sP(s, c) [1 - T + Tu_{1c}]^{s-1} [1 - T + Tu_{2c}]^c \quad (6d)$$

$$u_{2c} = F_{2c}(\mathbf{u}) = \frac{1}{M_{2c}} \sum_s \sum_{c>1} cP(s, c) [1 - T + Tu_{1c}]^s [1 - T + Tu_{2c}]^{c-1}. \quad (6e)$$

We compute  $S_i$  which corresponds to the fraction of vertices that are of type  $i$  and that are in the extensive component:

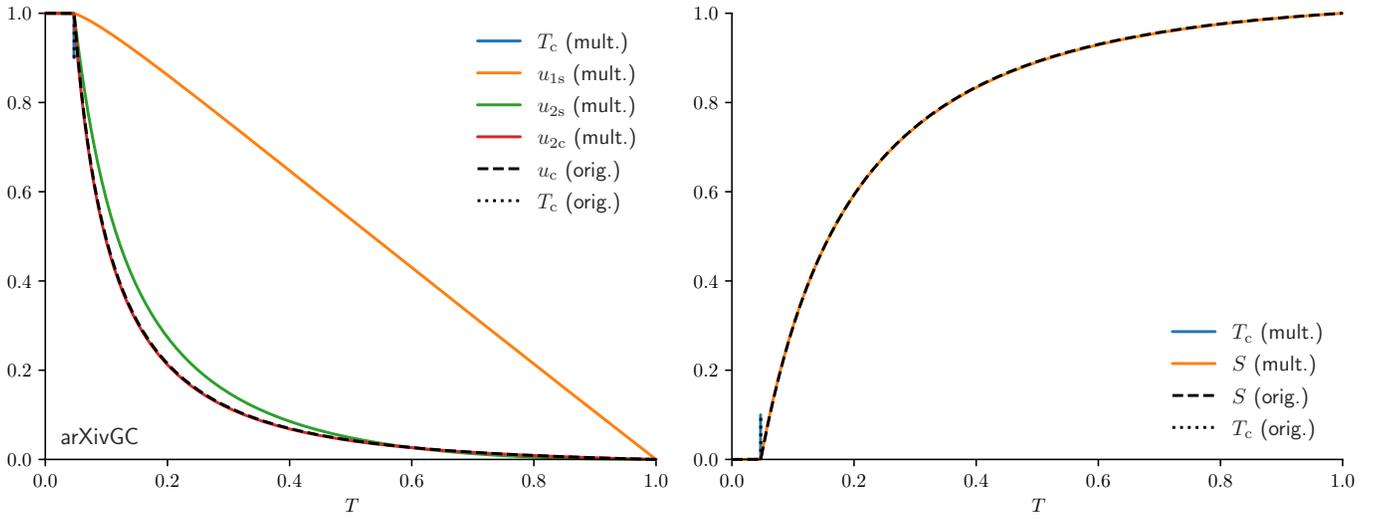
$$S_0 = w_0 [1 - G_0(\mathbf{u})] = w_0 - \sum_s P(s, 0) [1 - T + Tu_{0s}]^s \quad (7a)$$

$$S_1 = w_1 [1 - G_1(\mathbf{u})] = w_1 - \sum_s P(s, 1) [1 - T + Tu_{1c}]^s \left[ 1 - T + \frac{T(M_{1s}u_{1s} + M_{2s}u_{2s})}{M_{1s} + M_{2s}} \right] \quad (7b)$$

$$S_2 = w_2 [1 - G_2(\mathbf{u})] = w_2 - \sum_s \sum_{c>1} P(s, c) [1 - T + Tu_{1c}]^s [1 - T + Tu_{2c}]^c. \quad (7c)$$

The relative size of the extensive component is then  $S = S_0 + S_1 + S_2$ .

<sup>4</sup>For example,  $u_{1s}$  is the probability that a vertex of type 1 reached via one of its subcritical stub does not lead to the extensive component.



Supp. Fig. 4: Co-authorship network from the arXiv preprint archive [9] for which  $w_1 = 0.08$ ,  $\alpha_1 = 1.02$  and  $\alpha_2 = 0.99$ .

### Extensive component formed by vertices of type 0

Equation (6a) is uncoupled from the other ones, and by applying the methods from Ref. [17], we find that some vertices of type 0 will form an extensive component if

$$\frac{TM_{0s0s}}{M_{0s}} > 1. \quad (8)$$

The relative size of this extensive component is obtained using Eq. (7a).

### Can reaching a vertex of type 1 via its only critical stub lead to an extensive component?

According to the definition of the critical configuration model, reaching a vertex of type 1 via its only critical stub should not lead to an extensive component. This is due to the fact that all remaining stubs of that vertex are subcritical, by definition<sup>5</sup>. As a consequence, we expect  $u_{1c} = 1$  to be the only solution of Eq. (6c) for any  $P(c, m)$  coherent with the connection rules of the Critical Configuration Model. Again, applying the methods from Ref. [17], we find that this will be true whenever

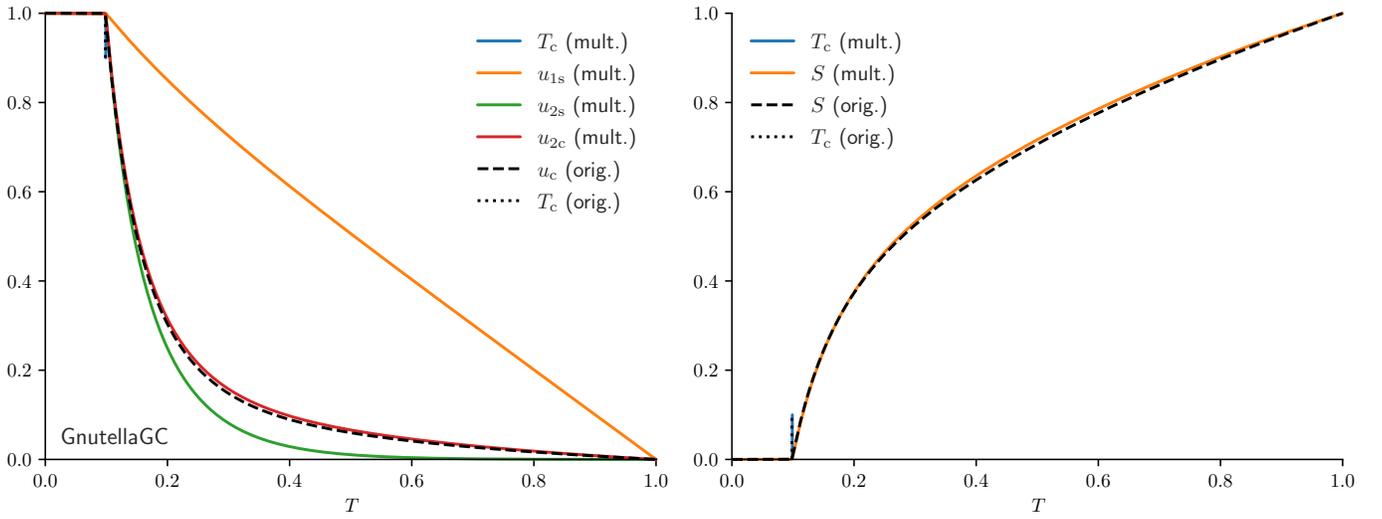
$$\frac{TM_{1s}}{M_{1c}} < 1. \quad (9)$$

This inequality will always be verified for any value of  $T$  if the vertices of type 1 reached via their critical stub always lead to a finite tree of type-1 vertices, which is what we expect from the connection rules of the Critical Configuration Model. If true, we know that  $M_{1s} < w_1$  since trees have one less edge than they have vertices and that all edges in these trees involve one subcritical stub. It then follows that

$$1 > \frac{M_{1s}}{w_1} = \frac{M_{1s}}{M_{1c}} > \frac{TM_{1s}}{M_{1c}}, \quad (10)$$

thereby proving that  $u_{1c} = 1$  for any well behaved  $P(s, c)$ .

<sup>5</sup>This is the reason why we set  $u_s = 1$  in the main text.



Supp. Fig. 5: Peer-to-peer Gnutella network [11] for which  $w_1 = 0.45$ ,  $\alpha_1 = 1.11$  and  $\alpha_2 = 1.01$ .

### Comparison with the approach in the main text

We now compare the multitype approach introduced above with the one presented in the main text.

### Self-consistent equations

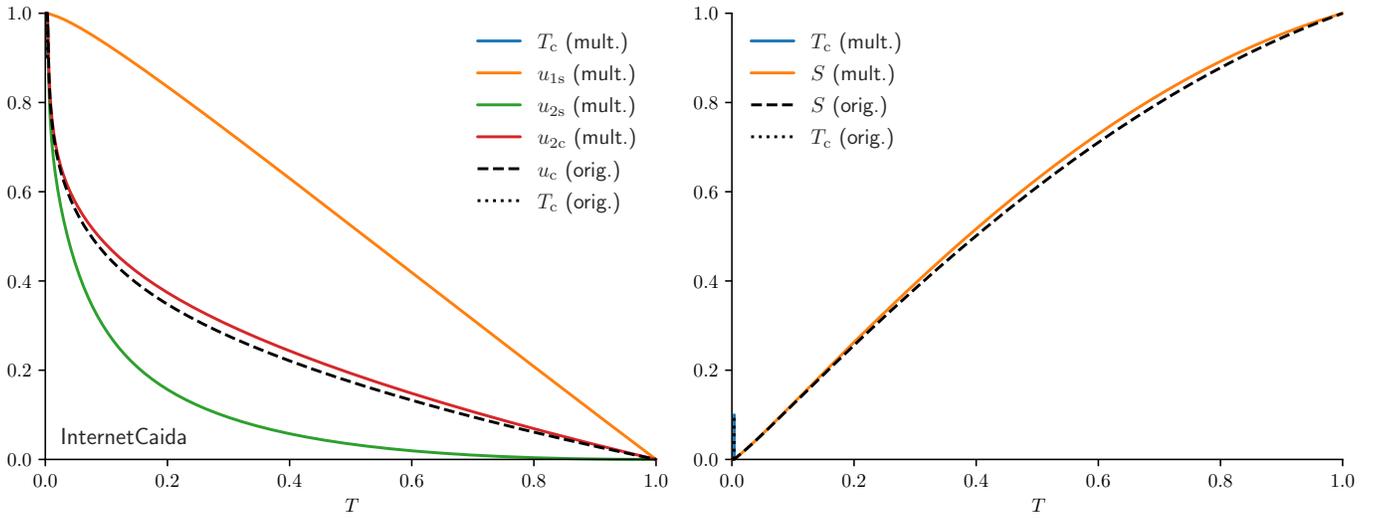
The self-consistent equation needed to be solved in the main text can be rewritten as

$$\begin{aligned}
 u_c &= 1 - T + T \frac{\sum_{s,c} P(s,c) \left( c(1 - \delta_{c,0})(1 - \delta_{c,1})u_c^{c-1} + s(1 - \delta_{c,0})u_c^c \right)}{\sum_{s,c} P(s,c) \left( c(1 - \delta_{c,0})(1 - \delta_{c,1}) + s(1 - \delta_{c,0}) \right)} \\
 &= 1 - T + T \frac{\left( \sum_s sP(s,1)u_c \right) + \left( \sum_s \sum_{c>1} sP(s,c)u_c^c \right) + \left( \sum_s \sum_{c>1} cP(s,c)u_c^{c-1} \right)}{\left( \sum_s sP(s,1) \right) + \left( \sum_s \sum_{c>1} sP(s,c) \right) + \left( \sum_s \sum_{c>1} cP(s,c) \right)} \\
 &= 1 - T + T \frac{\left( M_{1s}u_c \right) + \left( \sum_s \sum_{c>1} sP(s,c)u_c^c \right) + \left( \sum_s \sum_{c>1} cP(s,c)u_c^{c-1} \right)}{M_{1s} + M_{2s} + M_{2c}}, \tag{11}
 \end{aligned}$$

where  $u_c$  is the probability that leaving any vertex via a critical stub will not lead to the extensive component.

In the multitype approach, the probability that leaving any vertex via a critical stub will not lead to the extensive component is

$$\hat{u}_c = 1 - T + T \frac{M_{1s}u_{1s} + M_{2s}u_{2s} + M_{2c}u_{2c}}{M_{1s} + M_{2s} + M_{2c}}, \tag{12}$$



Supp. Fig. 6: Structure of the Internet at the level of autonomous systems [6] for which  $w_1 = 0.38$ ,  $\alpha_1 = 1.12$  and  $\alpha_2 = 1.03$ .

which can be rewritten using Eqs. (6) as

$$\begin{aligned} \hat{u}_c = 1 - T + T & \frac{M_{1s} \left[ 1 - T + T \frac{M_{1s}u_{1s} + M_{2s}u_{2s}}{M_{1s} + M_{2s}} \right]}{M_{1s} + M_{2s} + M_{2c}} \\ & + T \frac{\sum_s \sum_{c>1} sP(s, c) \left[ 1 - T + T \frac{M_{2c}u_{2c}}{M_{2c}} \right]^c}{M_{1s} + M_{2s} + M_{2c}} \\ & + T \frac{\sum_s \sum_{c>1} cP(s, c) \left[ 1 - T + T \frac{M_{2c}u_{2c}}{M_{2c}} \right]^{c-1}}{M_{1s} + M_{2s} + M_{2c}} . \end{aligned} \quad (13)$$

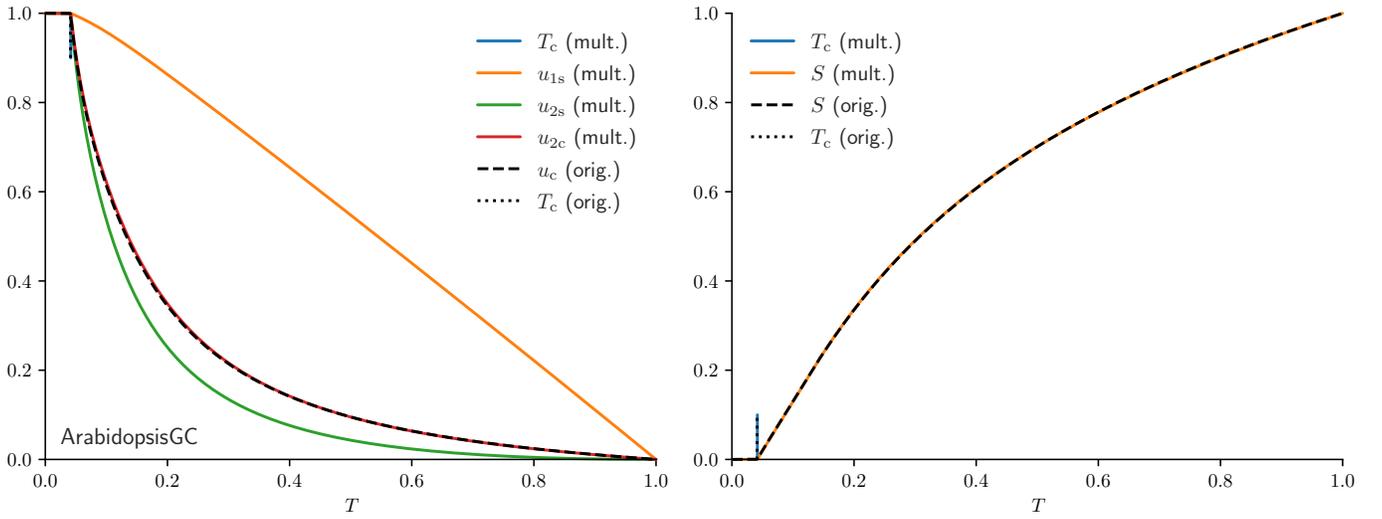
Since the terms in the three brackets of Eq. (13) are not equal to  $\hat{u}_c$ , we conclude that it does not coincide with Eq. (11). This comparison highlights the major difference between the two approaches: by identifying vertices with types, the multitype approach keeps a memory about how a vertex has been reached and is therefore able to specify the ways through which it can be left. The approach presented in the main text does not make such a distinction, and *any critical stub will be equally likely to not lead to the extensive component regardless of how the vertex has been reached*.

The two approaches will coincide if there is no vertex of type 1 (i.e.  $w_1 = 0$ ), implying that  $P(s, c) = 0$  for all  $s > 0$  and  $c > 0$  and therefore that the first two terms of both Eqs. (11) and (13) vanish as well as  $M_{1s} = M_{2s} = 0$ .

They will also yield similar results if

$$\alpha_1 \equiv \frac{M_{1s} + M_{2s} + M_{2c}}{M_{2c}} \simeq 1 . \quad (14)$$

This is confirmed in Supp. Figs. 3–6.



Supp. Fig. 7: The Arabidopsis brain interactome [8] for which  $w_1 = 0.40$ ,  $\alpha_1 = 1.10$  and  $\alpha_2 = 1.01$ .

**Percolation threshold**

A linear stability analysis of the solution  $u_c = 1$  of Eq. (11) yields the following condition for the existence of an extensive component

$$T \frac{M_{1s} + M_{2s2c} + M_{2c2c}}{M_{1s} + M_{2s} + M_{2c}} > 1 . \tag{15}$$

For the multitype approach, linear stability analysis of the solution  $(u_{1s}, u_{2s}, u_{2c}) = (1, 1, 1)$  yields the following Jacobian matrix

$$\mathbf{J} = \begin{bmatrix} \frac{TM_{1s}}{M_{1s} + M_{2s}} & \frac{TM_{2s}}{M_{1s} + M_{2s}} & 0 \\ 0 & 0 & \frac{TM_{2s2c}}{M_{2s}} \\ 0 & 0 & \frac{TM_{2c2c}}{M_{2c}} \end{bmatrix} \tag{16}$$

whose largest eigenvalue will be greater than one, thereby implying the existence of an extensive component, when

$$T \frac{M_{2c2c}}{M_{2c}} > 1 . \tag{17}$$

From Eq. (17), we see that only vertices of type 2 (connected through their critical stubs) contribute to the emergence of an extensive component in the multitype approach, whereas all vertices with at least 1 critical stubs (i.e. types 1 and 2) play a part in the approach of the main text, as shown in Eq. (15).

Comparing Eqs. (15) and (17) also confirms our previous conclusion in that the two approaches will coincide if there is no vertex of type 1 (i.e.  $w_1 = 0$ ), which implies that  $M_{1s} = M_{2s} = M_{2s2c} = 0$ .

They will also yield similar results if

$$\alpha_2 \equiv \frac{M_{2c}}{M_{2c2c}} \frac{M_{1s} + M_{2s2c} + M_{2c2c}}{M_{1s} + M_{2s} + M_{2c}} \simeq 1 . \quad (18)$$

This is confirmed in Supp. Figs. 3–6.

### 3 References

- [1] A. Allard, L. Hébert-Dufresne, J.-G. Young, and L. J. Dubé, General and exact approach to percolation on random graphs, *Phys. Rev. E* **92**, 062807 (2015).
- [2] M. E. J. Newman, Spread of epidemic disease on networks, *Phys. Rev. E* **66**, 016128 (2002).
- [3] A. Vázquez and Y. Moreno, Resilience to damage of graphs with degree correlations, *Phys. Rev. E* **67**, 015101 (2003).
- [4] A. Allard and L. Hébert-Dufresne, Percolation and the Effective Structure of Complex Networks, *Phys. Rev. X* **9**, 011023 (2019).
- [5] L. Hébert-Dufresne, J. A. Grochow, and A. Allard, Multi-scale structure and topological anomaly detection via a new network statistic: The onion decomposition, *Sci. Rep.* **6**, 31708 (2016).
- [6] B. Karrer, M. E. J. Newman, and L. Zdeborová, Percolation on Sparse Networks, *Phys. Rev. Lett.* **113**, 208702 (2014).
- [7] A. Allard and L. Hébert-Dufresne, *On the accuracy of message-passing approaches to percolation in complex networks*, Preprint arXiv:1906.10377 (2019).
- [8] M. Dreze, A.-R. Carvunis, B. Charletoeux, M. Galli, S. J. Pevzner, M. Tasan, Y.-Y. Ahn, P. Balumuri, A.-L. Barabasi, V. Bautista, P. Braun, D. Byrdsong, H. Chen, J. D. Chesnut, M. E. Cusick, J. L. Dangl, C. de los Reyes, A. Dricot, M. Duarte, J. R. Ecker, C. Fan, L. Gai, F. Gebreab, G. Ghoshal, P. Gilles, B. J. Gutierrez, T. Hao, D. E. Hill, C. J. Kim, R. C. Kim, C. Lurin, A. MacWilliams, U. Matrubutham, T. Milenkovic, J. Mirchandani, D. Monachello, J. Moore, M. S. Mukhtar, E. Olivares, S. Patnaik, M. M. Poulin, N. Przulj, R. Quan, S. Rabello, G. Ramaswamy, P. Reichert, E. A. Rietman, T. Rolland, V. Romero, F. P. Roth, B. Santhanam, R. J. Schmitz, P. Shinn, W. Spooner, J. Stein, G. M. Swamilingiah, S. Tam, J. Vandenhoute, M. Vidal, S. Waaijers, D. Ware, E. M. Weiner, S. Wu, and J. Yazaki, Evidence for Network Evolution in an Arabidopsis Interactome Map, *Science* **333**, 601 (2011).
- [9] M. E. J. Newman, The structure of scientific collaboration networks, *Proc. Natl. Acad. Sci. U.S.A.* **98**, 404 (2001).
- [10] K. Norlen, G. Lucas, M. Gebbie, and J. Chuang, EVA: Extraction, Visualization and Analysis of the Telecommunications and Media Ownership Network, in *Proceedings of International Telecommunications Society 14th Biennial Conference (ITS2002)* (2002) pp. 27—129.
- [11] R. Matei, A. Iamnitchi, and P. Foster, Mapping the Gnutella network, *IEEE Internet Comput.* **6**, 50 (2002).
- [12] M. Kaiser and C. C. Hilgetag, Spatial growth of real-world networks, *Phys. Rev. E* **69**, 036103 (2004).
- [13] J. Leskovec, J. Kleinberg, and C. Faloutsos, Graphs over time: Densification laws, shrinking diameters and possible explanations, in *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD '05)* (2005) p. 177.
- [14] M. Boguñá, R. Pastor-Satorras, A. Díaz-Guilera, and A. Arenas, Models of social networks based on social distance attachment, *Phys. Rev. E* **70**, 056122 (2004).
- [15] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education, *IEEE Trans. Power Syst.* **26**, 12 (2011).
- [16] D. J. Watts and S. H. Strogatz, Collective dynamics of small-world networks, *Nature* **393**, 440 (1998).
- [17] 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).