



Skolkovo Institute of Science and Technology

Skolkovo Institute of Science and Technology

PERCOLATION ON COMPLEX NETWORKS AND ITS APPLICATIONS

Doctoral Thesis

by

SAEED OSAT

DOCTORAL PROGRAM IN
COMPUTATIONAL AND DATA SCIENCE AND ENGINEERING

Supervisor
Assistant Professor Vladimir Palyulin

Moscow - 2021

© Saeed Osat 2021

I hereby declare that the work presented in this thesis was carried out by myself at Skolkovo Institute of Science and Technology, Moscow, except where due acknowledgement is made, and has not been submitted for any other degree.

Candidate (Saeed Osat)

Supervisor (Assistant Professor Vladimir Palyulin)

ACKNOWLEDGMENT

I have been lucky to receive many supports and encouragements during my Ph.D. studies at Skoltech. First of all, I thank my supervisor Prof. Palyulin. He kindly accepted me in his group while I was at the late stages of my studies. His friendly attitude is a blessing and brought me a calm and stress-less graduation.

Writing a short format thesis was a big challenge for me since I wanted to keep the material condensed and at the same time accessible and self-contained. I could not achieve this without the help from Prof. Palyulin. He patiently went with me through the text in each iteration of the editing part and gave me a detailed feedback. I believe this also helped me to improve my scientific writing and I thank him for that.

Current thesis and the resulting publications could not be done without the help of my colleagues and collaborators. I would like to thank Prof. Radicchi and Dr. Ali Fafeeh for helping me to gain a perspective in network science. I am grateful to other colleagues and collaborators including Prof. Biamonte, Prof. Papadopoulos, Prof. Gleeson, Prof. Dorogovtsev, Prof. Azimi-Tafreshi and Prof. Dylov.

Apart from science, I would like to appreciate the support from Skoltech and CDISE department, administrative crew in education office and Skoltech's dorm team.

And last but not least, I would like to thank my parents and my family for their unconditional love and support during all stages of my life.

PERCOLATION ON COMPLEX NETWORKS AND ITS APPLICATIONS

Abstract

Real complex systems can be easily and effectively represented by networks. Networks are ubiquitous in our daily life, including, social networks, transportation networks, technological networks, etc. Networks or graphs are mathematical objects in which nodes represent the constituent units of the system and links represent interaction between these units. Recently a more general framework of multiplex networks has been introduced in order to study networks with more than one layer. In an L-layer multiplex network each node is present simultaneously in all layers, however, each layer corresponds to a different type or flavor of interactions between nodes. Beyond the structure of the network (either it is monoplex or multiplex) one is interested to unravel the interplay between structure of a network and the dynamic taking place on top of it.

Percolation and some of its variants as candidate processes taking place on top of the networks are studied in this thesis. More explicitly the thesis includes the following. **(a)** Bond percolation on monoplex networks. This study models neuronal systems and proves that systems can attain power-law indications of criticality without being at critical point. **(b)** Generalized core percolation of monoplex networks. This study helps us to improve our understanding of structure of networks by offering new layout based on core and k -core percolation. **(c)** Observability phase transition on multiplex networks. Observability phase transition of synthetic and real multiplex networks are studied. We show that while the latter has second order phase transition, the former has first order type. **(d)** Community structure and hyperbolic embedding of networks. We show that there is a one-to-one correspondence between hyperbolic embedding and community structure. Utilizing this analogy we unravel the reason behind the robustness of real-world multiplex networks. **(e)** Optimal percolation on multiplex networks. We define the model and design algorithms to approximately solve this problem. We differentiate this problem from its monoplex counterpart and show that forgetting multiplex nature of the network can overestimate robustness of the network. **(f)** k -core structure of real multiplex networks. We show that real multiplex networks have rich k -core structure. Moreover, we investigate the role of geometric correlations in this regard.

Publications

1. Optimal percolation on multiplex networks

S. Osat, A. Faqeeh and F. Radicchi

Nature Communications, 8, 1540, (2017)

2. Observability transition in multiplex networks

S. Osat and F. Radicchi

Physica A, 503, 745-761, (2018)

3. Characterizing the analogy between hyperbolic embedding

and community structure of complex networks

A. Faqeeh, **S. Osat** and F. Radicchi

Physical Review Letters, 121, 098301, (2018)

4. Generalization of core percolation on complex networks

N. Azimi, **S. Osat** and S. N. Dorogovtsev

Physical Review E, 99, 022312, (2019)

5. Emergence of power laws in noncritical neuronal systems

A. Faqeeh, **S. Osat**, F. Radicchi and J. Gleeson

Physical Review E, 100, 010401(R), (2019)

6. k -core structure of real multiplex networks

S. Osat, F. Radicchi and F. Papadopoulos

Physical Review Research, 2, 023176, (2020)

7. Deep Learning Super-Diffusion in Multiplex Networks

V.M Leli, **S. Osat**, T. Tlyachev, D. Dylov, J. Biamonte,

Journal of Physics: Complexity, 2, 035011, (2021)

Dedication

To my parents
For their endless love and support

TABLE OF CONTENTS

	Page
ABSTRACT	iv
PUBLICATIONS	v
LIST OF TABLES	ix
LIST OF FIGURES	x
CHAPTER	
1 Introduction	1
1.1 Networks and Percolation Models	3
1.1.1 Basic definitions	3
1.1.2 Monoplex Networks: Random v.s. Real	5
1.1.3 Configuration Model	6
1.1.4 Hyperbolic Networks	7
1.1.5 Multiplex networks	8
1.1.6 Why Multiplexity?	10
1.1.7 Degree Correlation	11
1.1.8 Edge Overlap	12
1.1.9 Community Correlation	12
1.1.10 Geometric Correlation	13
1.1.11 Phase Transition	15
1.1.12 Generating Function	18
1.1.13 Percolation on monoplex networks	19
1.1.14 Percolation on multiplex networks	22
1.1.15 Implications of multiplexity	25
1.2 Results	26

1.2.1	Bond percolation and neuronal avalanches	26
1.2.2	Core percolation and k -core decomposition	29
1.2.3	Depth-one percolation or observability phase transition	32
1.2.4	Community-correlated multiplex	36
1.2.5	Optimal percolation	40
1.2.6	k -core of real multiplex networks	46
2	Emergence of power laws in noncritical neuronal systems	52
3	Generalization of core percolation on complex networks	84
4	Observability transition in multiplex networks	94
5	Characterizing the analogy between hyperbolic embedding and community structure of complex networks	113
6	Optimal percolation on multiplex networks	127
7	k-core structure of real multiplex networks	151
8	Conclusions	189
REFERENCES		191

LIST OF TABLES

1.1 Basic Definitions	4
---------------------------------	---

LIST OF FIGURES

1.1	Multiplex networks	9
1.2	Site and bond percolation on monoplex networks	21
1.3	Site percolation on multiplex networks	24
1.4	Probability and cumulative distributions of avalanche sizes.	27
1.5	Greedy leaf removal.	30
1.6	G _k -core percolation	31
1.7	Observability phase transition of random multiplex networks	33
1.8	Observability phase transition of real multiplex networks	34
1.9	Hyperbolic embedding community structure of networks.	37
1.10	Robustness of community correlated multiplex networks.	39
1.11	Optimal percolation on a random multiplex network.	43
1.12	Effect of edge overlap on the size of structural sets.	45
1.13	<i>k</i> -core of multiplex network of arXiv.	48
1.14	Randomization of multiplex network of arXiv	49
1.15	Randomization of multiplex network of the Internet.	51

Chapter One

Introduction

Many of the real and man-made complex systems around us are represented efficiently by graphs or networks [1, 2]. Transportation networks, social networks, technological networks and brain networks are just a few examples of real complex networks. A network or a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is a collection of nodes (vertices) \mathcal{V} and links (edges) \mathcal{E} . Nodes represent the constituent units of a system and links capture the interaction pattern between these units. Network science was developed in order to study their emergent and collective behavior [3–5]. Understanding the structure of networks is the first step in analysis of a complex system [2]. However, in most of the real networks, the structure is supplemented by a dynamic which is taking place on top of it [6]. For instance, epidemic spreading on a social network [7, 8], cascades and black outs occurring on an interdependent networks of the Internet and power grids [9], percolation phenomenon and its variants on monoplex and multiplex networks [9–14] and so forth. The approach of the network science in dealing with a complex system can be summarized in three main steps:

- Analysis of the structure of the complex network.
- Unravelling the dynamics of and on the complex network.
- Investigation of the controllability of the complex network.

Initially one needs to characterize the structure of a complex system before delving into its

dynamic and controllability [2]. Network science facilitates structural analysis of a complex system by considering the graph associated with that system. Then, characterizing the structure of a complex system reduces to the topological description of the equivalent underlying graph. Structural analysis of a graph is achieved by measuring some well-defined mathematical measures, metrics and etc. Topological descriptors of a graph include quantities such as: degree distribution, clustering (abundance of triangles) [15], community structure (higher order structure of tightly connected nodes) [16], node and edge centralities [2] and so forth. Some of the aforementioned metrics and their definitions can be found in Table 1.1. Hereafter we use the words graph and network interchangeably.

After describing the structure of the underlying graph of a complex system, one can raise questions about the dynamics of and on the system [6, 17]. The main goal of this step is to understand how a complex network —its structure, the process which is taking place on top of it or both— evolves in time. It is crucial to realize the difference between the dynamic of a network and the dynamic on a network. While the latter refers to a dynamical process e.g. random walk, taking place on the network, the former refers to the dynamic of the network itself. Throughout the thesis we are dealing with static networks, i.e. their underlying graphs are not evolving in time. We study dynamical processes that are taking place on top of static networks. Ultimately, we unravel the interplay between the structure of a graph and the dynamical process taking place on top of it.

An important point of view in the analysis of complex systems is controllability [18]. Controllability can be defined as follows. Given the structure, dynamic and initial state of the system under study \mathcal{S}_0 , controllability is defined as finding the required input signals or interventions in order to drive the system from the initial state \mathcal{S}_0 to the desired final state \mathcal{S}_1 [19, 20]. In the thesis we are not involved in this topic.

In this chapter I briefly describe the structure of complex networks and dynamics taking place on them. Regarding the structure, I specifically consider the networks that are used in my thesis, i.e. monoplex and multiplex networks and regarding the dynamics I consider

percolation process (or phenomenon as it is called in physics) and its variants as candidate processes that are taking place on top of these networks. It is worth to mention that the topic is broad and I will limit myself to the structures and models that are used in my thesis. The structure of the chapter is as follows. First basic definitions in complex networks are presented. Then random graph models are presented. Multiplex structures are defined and the need of this specific representation is discussed and justified. Then I briefly define percolation models on both monoplex and multiplex networks. Generating function method as a theoretical tool to study percolation on complex networks is presented. Finally the brief summary of the results of my thesis are presented in the following order: a) bond percolation on monoplex networks [2] and how it is used to model neuronal avalanches [21], b) generalized core percolation as a new layout for networked structures [22], c) depth-one percolation or observability transition on multiplex networks [23], d) robustness of community correlated multiplex networks [24] under targeted attack, e) optimal percolation or dismantling of multiplex networks [14] and f) k -core structure of real multiplex networks [25].

1.1 Networks and Percolation Models

1.1.1 Basic definitions

A network or a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is a set of nodes V connected by a set of links E . Nodes represent the elementary units of the system and links capture the connection between these units. For instance, in the case of a social network individuals can be represented by nodes and their social connections as links" [26], in a brain network neurons can be represented by nodes and their synaptic connections as links" [27] and so forth. In a weighted network each link has a weight which captures intensity of the interaction between the two nodes or simply the strength of the link. If all the links have the same weight $w = 1$ we call the network unweighted. A network is called directed if each link is associated with a direction.

Indirected networks can be considered as directed networks if each undirected link $i - j$ is replaced by two directed links $i \rightarrow j$ and $j \rightarrow i$ [2]. There is a handy generalization of networks in which existence of different types or colors of edges between nodes are allowed. This special case of networks can be modelled by multiplex (multi-layer) framework [28, 29].

Beyond the simple notation $\mathcal{G}(\mathcal{V}, \mathcal{E})$, one needs mathematically consistent definitions which can be defined and measured on graphs. These definitions become useful in analyzing random and real networks. Table 1.1 lists the basic definitions needed in order to follow the materials presented in the rest of this chapter.

Table 1.1 Basic Definitions

Adjacency Matrix (A)	Adjacency matrix A captures the connectivity pattern of a graph; $A_{ij} = 1$ when there is a link between node i and node j and $A_{ij} = 0$ otherwise.
Degree (k)	In an undirected network, k_i is number of links connected to node i : $k_i = \sum_j A_{ij}$. In a directed network one should differentiate between incoming and outgoing links. Then indegree and outdegree are define as $k_i^{in} = \sum_j A_{ij}$ and $k_j^{out} = \sum_i A_{ij}$ respectively.
Degree distribution ($p(k)$)	$p(k) = \frac{N_k}{N}$ is the fraction of nodes with degree k . The same definition can be generalized to indegree and outdegree distributions, where N_k is the number of nodes with degree k . $P(k)$ can be interpreted as the probability that if we choose a node at random it has degree k .

Excess Degree (q)	If one chooses a random edge and reaches to one of its endpoints, then the number of edges of that node except the one we came from is the excess degree of that node.
Excess Degree distribution ($q(k)$)	$q(k) = \frac{(k+1)p(k+1)}{\langle k \rangle}$ is the probability of the endpoint of a randomly chosen edge having excess degree k . $\langle k \rangle$ is the average degree of the whole graph.
Clustering Coefficient(c)	$c_i = \frac{N_\Delta}{\binom{k_i}{2}}$, where numerator is the number of triangles connected to the node i and denominator shows the whole number of possible triangles of node i . Clustering of the whole graph is $\bar{C} = \frac{1}{N} \sum_i c_i$.
Path	A set of consecutive edges makes a path. A closed path is called a loop. Triangle is the shortest possible loop (with length 3) in undirected graphs.
Component	It is a maximal subset of nodes in which each pair of the nodes in the set are connected with at least one path. A biggest component of a graph is called largest cluster or giant component (GC).
k -core	A maximal subset of graph where each node has at least degree k within the subset. k -cores are hierarchically nested with the highest k in the center.

1.1.2 Monoplex Networks: Random v.s. Real

There are families (ensemble) of random graph models in which one can create a random graph with specific parameters and specifications [30]. One of important random graph

models is Erdős-Rényi graph (sometimes simply called random graph or Poisson graph) [31]. Erdős-Rényi random graph model $\mathcal{G}(N, p)$ comes with two model parameters. N is a number of nodes and p is a probability of connection. One can make a realization of the model (create a random graph) by connecting each pair of nodes with probability p . $\mathcal{G}(N, p)$ features Poisson degree distribution, zero clustering and phase transition in the emergence of giant component [2, 31].

It is legitimate to assume that connections or links in the real-world networks are not random and hence their structural properties should be different from the ones of the random graphs [32]. With increase of computational powers and access to large scale real networks, people realized that these networks feature properties which differ significantly from the properties of the random graphs [2]. Most of the real networks are characterized by scale free degree distribution [33], high clustering [34] and community structure [35] only to mention few. The discrepancy between the structure of observed real networks and synthetic random graphs, led to the following problem. Often synthetic random graphs fail to possess the topological structures and features associated with the real world networks which introduces errors in modelling of the latter utilizing the former. In order to capture topological properties of the real world networks more complicated models beyond simple Erdős-Rényi model have been introduced. The graphs created by these models possess almost the same structural properties as the ones in their real world counterparts [2, 33, 36]. The analysis of the full set of these network models is beyond the scope of this thesis. I briefly describe only the models which are used in my thesis: configuration model [2, 37] and hyperbolic model [38].

1.1.3 Configuration Model

The main disadvantage of a Poisson random graph is its limitations to mimic the degree distribution of the real networks. While the former is only reproducing Poisson degree

distribution, the latter comes in variety of shapes from power-law degree distribution to exponential one and so forth [32, 39]. Thus, in order to properly model a real network, one needs a generative random graph model which is able to capture any desired degree distribution. Configuration model is a simple model which enables one to create maximally random graph out of a given degree sequence, meaning that beyond the given degree sequence, the whole structure of the graph is randomly created [2]. This degree sequence can be a sample degree sequence from a degree distribution or a degree sequence of a real network. The configuration model associates to each node i degree k_i from the candidate degree sequence. Then each node is given k_i stubs (half links). Finally pair of stubs are chosen randomly and coupled into a link. This process continues until no unmatched stubs are left. It is worth to mention that initially the total number of stubs needs to be an even number in order to prevent ending up with a single unmatched stub. Stub matching process may create self-links (links connecting nodes to themselves) and multi-links (more than one edges between two nodes). However, it is easy to show that density of these structures goes to zero in the limit of large network size. We utilize random networks created using configuration model [2, 37] and its extended version [21] in section 2.1 where we study neuronal avalanches on directed and undirected networks.

1.1.4 Hyperbolic Networks

While configuration model resolves the issue of degree distribution, it cannot create random instances of graphs with specific properties resembling the ones of the real networks. An important feature of real networks is non-zero clustering which occurs due to the abundance of triangles. Huge number of triangles and their self-similar behavior [40] accompanied by small-worldness and heterogeneous degree distribution makes it plausible to consider real networks as objects embedded in hidden metric space [38, 40]. Metric property of the hidden space supports triangle inequality; if a distance between two nodes in the underlying hidden

space is interpreted as the probability of connection between them, then high clustering is nothing more than topological counterpart of triangle inequality of metric space in the network structure [38, 40]. More details about these networks can be found in chapters five and seven.

1.1.5 Multiplex networks

Most of the real complex networks surrounding us are not just single isolated networks but collection of interacting networks e.g. social networks [41, 42], transportation networks [43] and so forth. Multi-layer networks framework enables one to generalize the tools utilized in the study of monoplex networks to analyze these high level structures [44, 45]. In my thesis, I contain myself to a special class of multi-layer or interdependent networks called multiplex networks [46].

An l -layer multiplex network is composed of l layers and N nodes where all the nodes exist simultaneously in all of the layers of the multiplex. Nodes of the same identity are called replicas. Replicas represent the same entities in each layer and they are connected with inter-layer links to each other. Inter-layer links between the replicas are interpreted as one to one correspondence between them. Each layer of a multiplex network corresponds to a different type or flavor of interaction between the nodes. For example a multiplex social network of Facebook and Twitter is composed of people who have interactions through one or both of these platforms. A multiplex biological network is composed of nodes (neurons, proteins, etc) where physical, chemical or any other kind of interactions potentially can exist between them. In multiplex framework each type of interaction is mapped onto a separate layer. $\mathcal{G}(N, A^{[1]}, A^{[2]})$ briefly captures the structure of a two layer multiplex network, where N is the number of nodes and $A^{[1]}$ ($A^{[2]}$) captures the interaction pattern in first (second) layer. Without losing generality we assume that labels of the nodes in both layer are enumerated in a way that node i of the two layers are replicas; $i = 1, 2, \dots, N$. An example of a multiplex

network is illustrated in Fig. 1.1.

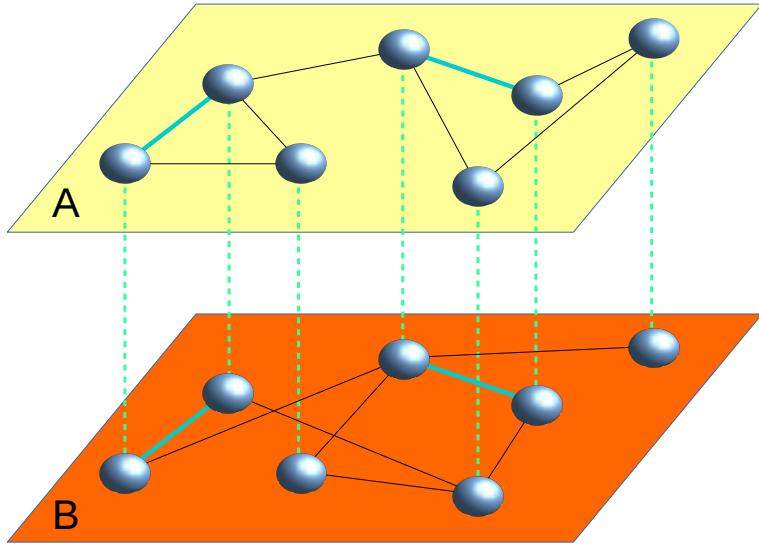


Figure 1.1 Multiplex networks. Schematic representation of a multiplex network composed of two layers A and B . Each node simultaneously exists in both layers of the multiplex network. Inter-layer links are shown as green dashed lines and exhibit one to one correspondence. Each layer represents a different type of interaction between nodes, i.e. each layer is composed of different intra-layer link structure. The links in cian color are called overlapping links. A link is called an overlapping link if it simultaneously exists in all the layers of the multiplex network.

Monoplex networks and their properties have been intensively and deeply studied [2]. It is a legitimate question now to ask how one should characterize a multiplex network [47]. Definitions of degree, degree distribution and other properties should be consistently generalized to multiplex networks. Here we mention four important properties observed in real multiplex networks which hugely impact the dynamics running on top of them: Degree correlation [48], edge overlap [49], community-correlated structure [24] and geometric correlations [50, 51]. For comprehensive and detailed information on the structural properties of random and real multiplex networks see [28, 29, 47, 52].

1.1.6 Why Multiplexity?

The need to increase the complexity of analysis of a system by using a multiplex structure is justified by many studies [28]. Intuitively the following question is raised that why a complicated multiplex framework is needed while there exists a much simpler monoplex framework [53]? The primitive suggestion can be reducing a multiplex structure to a monoplex structure \mathcal{G} described by $A = A^{[1]} \oplus A^{[2]}$. Then tools from monoplex theory can be applied to this aggregated network conveniently. While appealing, many studies showed that this representation can be faulty and lead to a maimed description of the system [28, 54]. By reducing multiplex structure to a monoplex structure (which is dubbed by the term “forgetting multiplexity”) one may not be able to describe the system properly [9, 14, 46, 55]. This is the nature of multiplexity that leads to new physics and phenomena.

An example of how a multiplex structure gives rise to new physics, is the emergence of a giant cluster in random Poisson multiplex networks [9, 56]. Consider a two-layer multiplex network (called duplex) in which each layer is a random Poisson network with N nodes and average degree $\langle k \rangle$. In analogy to giant connected component (GCC) in monoplex networks, one can define Giant Mutually Connected Components (GMCC). GMCC is a maximal subset of graph where each pair of nodes is connected at least by a path in the first layer and by a path in the second layer with a constraint that all these paths should be inclusive, meaning that all nodes on the paths also should belong to GMCC [9] (maximal subset is a subset of a set with a specific property. The property cannot be held by adding any other member to this subset). One can show analytically that GMCC undergoes a first order phase transition (with a jump) in contrast to second order phase transition of random Poisson networks [56]. This is an example of how existence of multiplex interactions can change the physics of the phenomena observed in the system. Numerous studies proved that considering multiplexity is needed in order to capture proper physics of dynamical and structural processes running on top of multiplex networks [9, 13, 14, 55, 57–62].

As discussed previously, real networks have properties that differ from random graphs. The same can also be stated for random and real multiplex networks. Random multiplex networks are created by random placement of inter-layer links between two random (or even real) isolated monoplex graphs. However, as one expects, real multiplex networks are not a random composition of two monoplex networks. It is not surprising that one to one correspondence between replicas (inter-layer links) in reality is not formed randomly. There are correlations between replica nodes and these correlations play an important role considering the dynamic running on top of a multiplex network [14, 24, 48–51]. There are four important inter-layer correlations in multiplex networks which are discussed and utilized in my thesis, namely, degree correlation [48, 63], edge overlap [14, 49], community correlation [24] and geometric correlations [50, 51]. It is worth to mention that another type of correlations also exists in the realm of monoplex networks [4]. These correlations are referred to as intra-layer correlations. The interest of my thesis is limited to inter-layer correlations since they are emergent due to multiplexity [47]. These correlations play major role in the percolation theory of multiplex networks [9, 14, 23, 64].

1.1.7 Degree Correlation

Degree correlation is the simplest correlation in a multiplex network. In a random multiplex network the inter-layer links (one-to-one correspondence between replicas) are placed randomly between the layers. However, generally it is possible that instead of a random one to one correspondence between nodes, one expects correlation between degrees of replicas in different layers, e.g. with a high degree in one layer has also a high degree in the other layer [47]. Inter-layer degree correlation in a multiplex network refers to the correlation between the degrees of the replicas in different layers of the multiplex network. Hereafter we simply call these correlations as degree correlations. The importance of degree correlations originates from the implication they have in the physics of the multiplex networks. For

instance, degree correlations (either positive or negative) can change the type of phase transition of GMCC [48, 63]. More importantly recent studies showed that degree correlations are not only synthetically implanted in random multiplex networks, but they also exist in real multiplex networks [50].

1.1.8 Edge Overlap

Edge overlap in multiplex networks refers to the simultaneous existence of an edge (i, j) in both layers of a multiplex network [28, 49]. In general a specific kind of correlation is considered to be important as long as it is not caused by randomness e.g. it is not observed in the synthetic random multiplex network. For instance, in the case of degree correlations we mentioned that random synthetic multiplex networks come with zero degree correlations due to the random placement of inter-layer links. Since a random multiplex network is a random one to one coupling of two networks, the probability that an edge between two nodes simultaneously exists in both layers of the multiplex is vanishingly small. However, in real multiplex networks a large amount of overlapping fractions have been observed [49]. For instance, considering online social platforms, it is highly probable that if two individuals are connected in the first platform, they are also connected in the second platform. It is crucial to understand the implications that edge overlap has on the physics of multiplex networks. We will discuss this in the result section, chapters four, five and six.

1.1.9 Community Correlation

In multiplex networks correlations exist on different scales. Degree correlations capture the node-level correlations and to what extent degrees of the replicas are correlated. Edge overlap captures correlations at a higher level of two-body interactions. Here we introduce a correlation that exists at macro level; multiplex networks with correlated community structure [24]. Community in a network is a group of nodes which have a tendency to have connections

within themselves than to other nodes of the network [16, 35]. Community detection consists of analysis aimed to discern these large scale structures [16, 35].

A community correlated multiplex network is a multiplex network in which there are correlations between the community structure of the layers. In other words, communities within separate layers of multiplex networks overlap [65]. More specifically these correlations imply that nodes belonging to a community in one layer, also form a community in the other layer. Random graphs do not contain statistically observable community structure [66] and hence community correlated structures cannot be defined for synthetic random multiplex networks. On the other hand, most of the real networks feature rich community structure and these community structures overlap between the layers of multiplex networks [24, 65]. This phenomenon and its implications are studied in the results section and more in-depth in chapter five.

1.1.10 Geometric Correlation

We previously introduced hyperbolic networks and mentioned that most of the real networks have underlying hidden metric space [40]. Broad degree distribution, high clustering and small-world property of real networks imply that the hidden metric space underlying most of the real networks is hyperbolic [38]. The hyperbolic embedding of complex networks associates coordinates (r, θ) to each node of the network. Coordinate r is called popularity and is related to the degree of a node. $\Delta\theta_{ij}$ abstracts similarity between the two nodes i and j [67]. A hyperbolic distance between two nodes i and j is defined as d_{ij} and is a combination of popularity and similarity. Intuitively one expects that the distance between connected nodes should be smaller than the distance between disconnected ones. The topic of hyperbolic geometry in complex networks is a decade-old and successful field and caught attention due to its success in shedding light on link-prediction, navigability, discovering of soft communities, geometric renormalization of networks and so forth [38, 67–70].

In order to generate a random graph with hyperbolic geometry (referred to \mathcal{H}^2 model) [38] coordinates of nodes are sampled from predefined density functions $\rho_r(r)$ and $\rho_\theta(\theta)$. Then each pair of the nodes is connected with a probability that depends on the hyperbolic distance between the nodes. Nodes with smaller distance between them have higher probability to be connected than the nodes with larger distance. Networks created with \mathcal{H}^2 model and with proper model parameters are able to mimic real networks in a sense that they feature power-law degree distribution, high clustering, small-worldness and realistic community structure [38, 40, 67, 69].

Let's assume we have a two-layer multiplex network. Embedding each layer of the multiplex network separately in a separate hyperbolic disk provides two sets of coordinates. By looking at the coordinates of replica nodes of the multiplex network one observes non-trivial correlation between the coordinates [50]. These correlations are called hidden, since they are not noticeable from the structure of the graph. One needs to embed layers of a multiplex network in metric space in order to unravel hidden correlations. Correlation between r coordinates of nodes is basically the degree correlation which we discussed previously. However, correlation between the angular coordinates of nodes is a novel observation [50] with huge physical implications [24, 50, 51].

Angular correlation between layers can be explained in the language of community structure. Correlation between angular coordinates of nodes in a multiplex network means that $\Delta\theta$ between pair of nodes (similarity between nodes) persists between the layers [50]. Recently an analogy between hyperbolic embedding and community structure has been suggested. Community structure captures the similarity between the nodes [24] or one can say that community structure and similarity are two equivalent descriptions. Utilizing this observation, one can conclude that geometric correlations in multiplex networks finds its counterpart as aforementioned community-correlated structures in multiplex networks [24]. Detailed description of these findings are provided in the results section and chapter five. Finally it is worth to mention that geometric correlations are very important when considering dynamics

of multiplex networks. Recently it was shown that these correlations boost the efficiency of greedy navigation in multiplex networks [50] and they are the mysterious reason of non-trivial robustness of real multiplex networks [24, 51] and their rich k -core structure [25].

1.1.11 Phase Transition

Understanding the different phases of matter and transitions between them, e.g., liquid-gas phase transition or paramagnetic-ferromagnetic phase transition was of great interest of statistical physicists [71]. As an example, consider a system composed of magnetic dipoles. How much do the local magnetic dipoles feel each other and are aligned? The celebrated Ising model captures the basic physics of this problem [72]. To quantify different phases one needs to define an order parameter for the system under study. For instance, in the Ising model of N spins one can define order parameter as $S = \frac{\sum_{i=1}^N s_i}{N}$ where $s_i \in \{-1, 1\}$. Now one can differentiate between different phases by separating the cases with $S > 0$ or $S = 0$, respectively. A transition from one phase to the other is usually driven by a control parameter, e.g., the temperature in the Ising model. Transition happens at some specific value of control parameters defining the transition point. It is important to notice that this simple picture may break down in some cases like 2D XY model, manifested in bound to unpaired transition of vortices and anti-vortices [73]. The concepts of order parameter and phase transition are applicable if one considers a thermodynamic limit where the number of constituent elements and/or volume of the system is going to infinite.

In this thesis, depending on the system under investigation we defined different order parameters, control parameters and phase transitions. Regarding the order parameter, one needs to differentiate between the concept of giant connected component (GCC) and largest connected component (LCC). In a network with N nodes you always can find a component with the maximum number of nodes which is called LCC of the network and has a constant size and independent of N . If in a graph of size N the size of the LCC is growing as N

(an extensive quantity), then one calls that component a GCC. Hereafter, when we mention order parameter or phase transition we refer to GCC or a relevant extensive quantity. No need to mention that GCC and phase transitions in our systems happen in thermodynamic limit.

Another important concept is the order of the phase transition. Classically the discontinuity of the first or the second derivative of a free energy shows whether the phase transition is first order or second order. However, we use less rigorous definition that if a change of order parameter from one phase to another, i.e., from $S = 0$ to $S > 0$ is discontinuous (continuous) then the phase transition is discrete or first order (continuous or second order). Here, we list different models considered in this thesis and the relevant phase transition parameters.

- **Bond percolation.** In bond percolation one randomly removes a fraction q of links from the network, then finds the size of the GCC as a function of q . Normalized size of GCC is the order parameter and fraction of removed links q plays the role of a control parameter. The phases can be distinguished considering presence or absence of an extensive GCC ($S > 0$ or $S = 0$).
- **Site percolation on monoplex network.** This is the same as bond percolation with a difference that fraction q of nodes (instead of links) are removed randomly from the network.
- **Percolation on multiplex (duplex) networks.** The order parameter is normalized size of GMCC and the control parameter is the fraction of nodes removed randomly from the both layers of a multiplex network. The two phases of presence or absence of GMCC are distinguished by the order parameter.
- **Optimal percolation on multiplex (duplex) networks.** This is the same as percolation on multiplex (duplex) networks though with an important difference that nodes are not removed randomly anymore and instead an optimal strategy is deployed

such that the maximum decrease in the size of GMCC is achieved.

- **Observability transition of monoplex networks.** In observability model a fraction of nodes are directly observed (host a sensor) and all their immediate neighbors are also observed(indirectly). Then one can find the giant connected component of observed nodes (either directly or indirectly). The normalized size of that GCC is the order parameter and the fraction of the nodes that are directly and randomly observed is the control parameter. The two phases of the problem corresponds to presence or absence of an extensive observable component.
- **Observability transition of multiplex networks.** The same as observability transition of monoplex networks except that the definition of observable component now extends since we have two layers of nodes.
- **k -core, core and G k -core.** Here, one is interested in the specific subset of nodes that make core, k -core or G k -core. These structures are obtained by specific greedy leaf removal algorithms which are explained in later subsections. The order parameter is defined as normalized size of that subset. Usually a random graph has that specific subset or structure when its average degree is bigger than a threshold or a critical average degree, i.e., control parameter is the average degree of the random graph. There are cases that other free parameters in a random graph rather than average degree can be chosen as control parameter. For instance, in power-law networks one can fix the average degree and take the exponent of the degree distribution as a control parameter and then study the phases of presence or absence of a specific structure in the graph.

1.1.12 Generating Function

In network theory the generating functions are heavily used as the most common toolbox to obtain theoretical results in studying different phenomena [2, 74]. A short description of the method and its usage to find the distribution of the size of the components is presented here. A more basic method to calculate the size of GCC is presented in the next sections.

A generating function (GF) can be considered as a condensed representation of a probability distribution function. For a general probability distribution function $S(n)$ with positive n one can define as follows, $G(z) = \sum_n S(n)z^n$. In a particular network we are given the degree and the excess degree distribution $p(k)$ and $q(k)$ as well as their generating functions $G_0(z) = \sum_k p(k)z^k$ and $G_1(z) = \sum_k q(k)z^k$, respectively. The goal is to obtain an analytical result for $\pi(s)$ the probability of a randomly chosen node belonging to a small component of size s and corresponding GF $h_0(z) = \sum_s \pi(s)z^s$. At this point an assumption and a property of GFs are introduced and utilized:

- **The local tree-like assumption.** This assumption states that small loops are not frequent in the graph and local neighborhood of a node can be considered as a tree. This assumption implies that a given random node is a hinge point of different branches emanating from its neighbors and thus, the overall size of the component that it belongs to is the summation of size of each of the branches plus one (counting the node itself).
- **Product property of generating functions.** GF of a summation of random integer numbers is product of the GF of the corresponding distributions. For instance, sampling T integer numbers n_1, n_2, \dots, n_T from $P_1(n), \dots, P_T(n)$ gives a random number $N = n_1 + \dots + n_T$ which has a GF that is product of the GFs: $G_N(z) = \prod_{i=1}^T G_i(z)$.

To go further, we introduce the probability of a node at the endpoint of a random edge to be in a small component of size s and corresponding GF $h_1(z)$ which we denote as ρ_s . Locally tree-likeness of the graph implies that a randomly selected node with degree k is in a

component of size s if the summation of the k outgoing branches of that node adds up to $s-1$ which happens with the probability $P(s-1|k)$. Averaging over different degrees yields $\pi(s) = \sum_{k=0} p(k)P(s-1|k)$. Transformation to GF yields $h_0(z) = \sum_{k=0} p(k) \sum_{s=1} P(s-1|k)z^s$. Now in the second summation we use the product property of GF and the fact that GF of component size of neighbors follow $h_1(z)$ and obtain $h_0(z) = zg_0(h_1(z))$. One can derive similar equation for the GF of the distribution of component size of a nodes at the end of a randomly selected edge, $h_1(z) = zg_1(h_1(z))$. Solving the equation for $h_1(z)$ and substituting it in the equation for $h_0(z)$ gives the GF for $\pi(s)$.

As an example, let's consider the specific case of calculating size of the GCC of a given graph with given $G_0(z)$ and $G_1(z)$. Let's define S as the probability of a randomly chosen node belongs to GCC. It is easy to see $S = 1 - h_0(1)$ because $h_0(1) = \pi(1) + \pi(2) + \dots$ is the probability of a random node belongs to a small component irrespective of the component size. The two equations for component sizes now reduces to $h_0(1) = g_0(h_1(1))$ and $h_1(1) = G_1(h_1(1))$. For a specific case of random graphs with Poisson degree distribution where $p(k) = q(k) = \frac{e^{-\langle k \rangle} \langle k \rangle^k}{k!}$ and $h_0(z) = h_1(z) = e^{\langle k \rangle(z-1)}$, the two equations reduce to a single self-consistent equation $h_0(1) = e^{\langle k \rangle(h_0(1)-1)}$. Utilizing $h_0(1) = 1 - S$ one obtains the equation for GCC, $S = 1 - e^{-\langle k \rangle S}$. In next section we derive a similar equation for the percolation on a graph.

1.1.13 Percolation on monoplex networks

Percolation aims to characterize the emergence of large scale connectivity in complex networks or lattices [75–77]. Percolation has many applications in transport in conductors [78], transport in random media [79], modeling of porous media [80], polymer science [81, 82], epidemic spreading [8, 83] and networks [10, 84]. In site percolation model one marks each site or node as occupied with probability p or unoccupied with probability $1-p$. In bond percolation model bonds or links are occupied or unoccupied with probability p or $1-p$

respectively. The order parameter in percolation process is the size of the giant connected component (GCC). Phase diagram of percolation phase transition illustrates order parameter as a function of a microscopic occupation probability p . Let's consider two extreme cases of site percolation process for the values of $p = 0$ and $p = 1$. In a network composed of one connected component, when $p = 0$ all the nodes of the networks are unoccupied and size of the GCC is 0. In the other extreme when $p = 1$, all the nodes are present in the network and size of the GCC is N (assuming that the initial graph is connected). It is clear from this example that there are two different regimes or phases. Size of GCC is an extensive quantity (growing with the size of the system) when $p = 1$, while it is an intensive quantity when $p = 0$. One can guess that there should be a phase transition point p_c where the large scale connectivity (GCC) emerges [2] and the order parameter changes its nature. Percolation phase transition on monoplex networks generally is of second order type while the phase transition of percolation on multiplex networks mainly is of first order type (in the absence of inter-layer correlations) [28, 56]. Here I briefly explain the generating function approach [2, 84] which is successfully used in the study of percolation model on networks. The theoretical approach introduced here is heavily used in the thesis; in the bond percolation description of neuronal avalanches [21], generalized core percolation [22] and observability phase transition or depth-one percolation [23].

Site percolation on simple monoplex networks. Site percolation is defined as follows. Each node in the network is occupied with probability p and then the fraction of the nodes that belongs to the GCC, $S = \frac{|GCC|}{N}$, is recorded. The goal of the percolation theory is to develop a theoretical framework to obtain the full phase diagram of the percolation $((S, p)$ diagram) and phase transition point p_c [10]. Let's assume we are considering a locally tree-like network with degree distribution $p(k)$ and introduce also a new quantity, excess degree distribution $q(k) = \frac{(k+1)p(k+1)}{\langle k \rangle}$. We define the generating functions of $p(k)$ and $q(k)$ by $G_0(z) = \sum_k p(k)z^k$ and $G_1(z) = \sum_k q(k)z^k$ respectively. A random node with a degree k is connected to GCC if a) it is present in the network and b) at least one of its neighbors is

connected to GCC. Let's define u as the probability that an end node of an edge (neighbor) is not connected to GCC. Thus, the probability that a randomly chosen node with k neighbors belongs to GCC is $s_k = p(1 - u^k)$. Now in order to obtain S , i.e. fraction of nodes in the GCC or the probability that a randomly chosen node belongs to GCC, one needs to average s_i over all possible values of the degree, $S = \sum_k p(k)s_k = p \sum_k p(k)(1 - u^k)$ which leads to $S = p[1 - G_0(u)]$. We need another equation for u in order to close the set of equations. A neighbor (an end point of an edge) with excess degree k does not belong to GCC if a) it is not present in the network which happens with probability $1 - p$ or b) it is present in the network but none of its neighbors belongs to GCC which happens with probability pu^k . For a neighbor with excess degree k the overall probability is $u = 1 - p + pu^k$. However, we need to average over all possible values of excess degrees $u = \sum_k q(k)u^k$ which results in $u = 1 - p + pG_1(u)$. Using these two equations one can solve equation for u iteratively and substitute the result in the equation for S and calculate size of GCC.

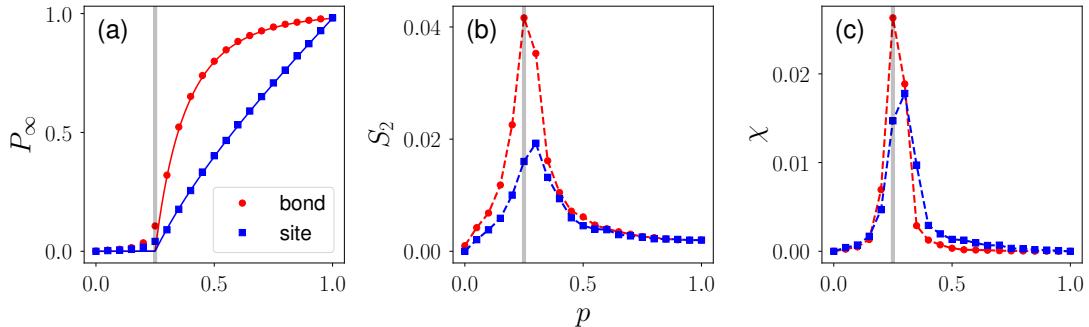


Figure 1.2 Site and bond percolation on monoplex networks. Site and bond percolation process are carried out on a random graph created by Erdős-Rényi model with $N = 1000$ nodes and average degree $\langle k \rangle = 4$. (a) Phase diagram of site and bond percolation. (b) Size of the second largest cluster as a function of occupation probability p . (c) Susceptibility of the order parameter is defined as $\chi = \frac{\langle S^2 \rangle - \langle S \rangle^2}{\langle S \rangle}$ [85]. In panel (a) markers are simulation results and solid lines are theoretical prediction. Vertical grey lines indicate the critical point.

As an example we show the case of Erdős-Rényi graphs with N nodes and average degree $\langle k \rangle$. In this specific case, we know that $p(k) = q(k) = \frac{e^{-\langle k \rangle} \langle k \rangle^k}{k!}$ and $G_0(z) = G_1(z) =$

$e^{<k>(z-1)}$. The equation for u and S in this case reduces to one equation $S = p(1 - e^{-<k>S})$. This equation can be solved iteratively starting from an initial guess for S . In order to obtain the phase diagram of percolation one needs to calculate S for the whole range of $p \in [0, 1]$. Phase transition point can be calculated by expanding the equation for S around the critical point where $S = 0$ which results in $p_c = \frac{1}{<k>}$. Fig. 1.2 shows the result of site and bond percolation on a random Erdős-Rényi graph. Divergence of size of the second largest cluster and susceptibility of the order parameter are two mostly used quantities in order to detect the critical point [85]. The susceptibility is defined as $\chi = \frac{<S^2> - <S>^2}{<S>}$, where angular brackets $< \dots >$ indicate average of the S over many different independent realizations of the process. The use of the divergence of the susceptibility as an indication of the phase transition point has deep roots in the statistical mechanics of phase transitions [72]. We used the same formalism of generating function in chapter two to analytically describe neuronal avalanches in the language of bond percolation model [21]. More details of generating function can be found there.

1.1.14 Percolation on multiplex networks

In this section we define site percolation on multiplex networks and briefly discuss how it differs from monoplex case [56]. The theory presented here is heavily used in the chapters four, five and six. In a site percolation process taking place on top of a multiplex network one occupies each node (in all of the layers simultaneously) with probability p and then measures the order parameter which is defined as fraction of the nodes that belong to the Giant Mutually Connected Component (GMCC) [9, 56]. As in monoplex case we are interested in the whole phase diagram of (S, p) where S is the fraction of nodes of multiplex network which belongs to GMCC and p is the microscopic occupation probability of nodes. Besides phase diagram, locating the exact phase transition point and understanding the type of phase transition can tell us more about the role of multiplexity in changing the physics of

percolation process [9].

Let's consider a two-layer multiplex network with N nodes. The degree distributions and excess degree distributions of the layers are denoted by $p_1(k)$, $p_2(k)$ and $q_1(k)$, $q_2(k)$ respectively. In analogy to the percolation on monoplex networks we can define generating functions of these distributions as $G_0^{[1]}(z) = \sum_k p_1(k)z^k$, $G_0^{[2]}(z) = \sum_k p_2(k)z^k$, $G_1^{[1]}(z) = \sum_k q_1(k)z^k$ and $G_1^{[2]}(z) = \sum_k q_2(k)z^k$. In order to be able to analytically derive an equation for the size of GMCC, one needs to deploy the locally tree-like assumption. When a network is locally a tree it does not have short loops or density of short loops is negligible. In locally tree-like graphs neighborhood of any random node looks like a tree. A randomly chosen node belongs to GMCC if it is a) present in the network which happens with probability p , b) belongs to the GMCC through first layer (at least one of its neighbors in the first layer is connected to GMCC) which happens with probability $1 - G_0^{[1]}(u_1)$ and c) connected to GMCC through the second layer (at least one of its neighbors in the second layer is connected to GMCC) which happens with probability $1 - G_0^{[2]}(u_2)$. Combining these terms one obtains $S = p[1 - G_0^{[1]}(u_1)][1 - G_0^{[2]}(u_2)]$. With the same procedure one can obtain an equation for u_1 and u_2 , $u_1 = p[1 - G_1^{[1]}(u_1)][1 - G_0^{[2]}(u_2)]$ and $u_2 = p[1 - G_0^{[1]}(u_1)][1 - G_1^{[2]}(u_2)]$. One can iteratively solve the equation for u_1 and u_2 and finally obtain S .

As an example lets consider a two-layer multiplex networks where each layer is a random graph created by Erdős-Rényi random graph model with N nodes and average degree $\langle k \rangle$. In this specific case, we know that $p_1(k) = p_2(k) = q_1(k) = q_2(k) = \frac{e^{-\langle k \rangle} \langle k \rangle^k}{k!}$ and $G_0^{[1]}(z) = G_0^{[2]}(z) = G_1^{[1]}(z) = G_1^{[2]}(z) = e^{\langle k \rangle(z-1)}$. The equations for u_1 , u_2 and s in this case reduce to one equation $S = p(1 - e^{-\langle k \rangle S})^2$. This equation can be solved iteratively for the whole range of $p \in [0, 1]$ and lead to the phase diagram of percolation. After some algebra we can obtain the phase transition point $p_c = \frac{2.4554}{\langle k \rangle}$. It is clear that the minimum average degree that a random Poisson duplex needs to acquire GMCC is $\langle k \rangle = 2.4554$. We used the same formalism in chapter four to analytically describe depth-one percolation or observability phase transition on multiplex networks. More details of generating function formalism on

multiplex networks can be found there.

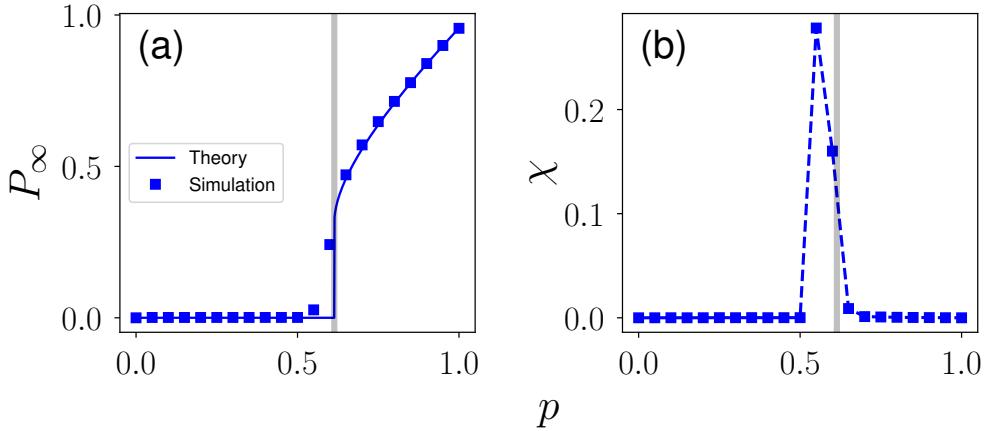


Figure 1.3 Site percolation on multiplex networks. Site percolation on random Poisson multiplex networks where each layer is a random Erdős-Rényi graph with $N = 1000$ nodes and average degree $\langle k \rangle = 4$. Each node is occupied (activated) in both layers with probability p , and then fraction of the nodes in GMCC is measured using the algorithm introduced here [9]. **(a)** Percolation phase diagram. **(b)** Susceptibility. In panel **(a)** markers show the simulation results and solid curve is the theoretical prediction. Grey vertical lines show the critical point.

Fig. 1.3 illustrates percolation phase diagram of a multiplex network where a random Erdős-Rényi graphs are used as the topology of the layers. Comparing Fig. 1.2 and Fig. 1.3 one can see that phase transition in multiplex networks in contrast to its single layer counterpart is first order where the order parameter undergoes sudden jump instead of smooth transition. Finally, one can see that the theoretical approach successfully predicts a first order phase transition for GMCC and the critical transition point which matches with the simulation results.

1.1.15 Implications of multiplexity

We mentioned previously that structure of a network can significantly influence and change the dynamics running on top of it. For instance, in the case of random monoplex networks created by configuration model, it has been proven that the phase transition of percolation process occurs at $p_c = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle}$ [37, 84]. The transition point p_c for networks with degree distributions that feature diverging first and second moment becomes zero which implies that these networks are robust against random failure. On the other hand, we proved that Poisson random graphs have non-zero transition probability $p_c = \frac{1}{\langle k \rangle}$. This example simply shows the effect of degree distribution on percolation of monoplex networks. With the same reasoning one will not be surprised that numerous factors play role in the percolation phase diagram of multiplex networks [49]. Degree correlations [48], edge-overlap [49] and geometric correlations [51] can change the type of phase transition from the first to the second order in multiplex networks.

We intuitively explain how existence of non-trivial edge overlap as one of the possible inter-layer correlations can change the physics of percolation on multiplex networks [49]. Let's consider a multiplex network with two layers. Furthermore, initially, the structure of the both layers is the same, and the labels of the nodes in the both layers ordered in a way that all the links of the multiplex initially are overlapping. Fully overlapping multiplex network behaves as a single monoplex network and in this case GMCC and GCC are the same. From the percolation theory of monoplex networks one knows that percolation phase transition of this special multiplex is of second order type, since its structure has already been reduced to a single layer structure. Now we perform the following randomization procedure. We pick each node of the first layer and change its identity or label with the label of another randomly chosen node from the same layer with probability α [49]. When $\alpha = 0$ both layers of multiplex fully overlap and reduce to a single monoplex network and obtain continuous phase transition for percolation process. When $\alpha = 1$ the overlap between

the two layers reduces to its minimum and it becomes a random multiplex network. Theory of percolation on multiplex networks predicts a first order phase transition for $\alpha = 1$. This simple observation shows that how an inter-layer correlation (in this case edge overlap) can change the physics of a process running on top of a multiplex network.

In chapter four we study the effect of edge overlap on observability phase transition of multiplex networks. As in the case of site percolation, observability phase transition of real networks with high edge overlap becomes smooth in contrast to synthetic multiplex networks with abrupt phase transition. In chapter six, we study the effect of edge overlap in the context of optimal percolation. We investigate robustness of multiplex networks in the presence of inter-layer correlations including the edge overlap. Apart from edge overlap other types of inter-layer correlations can immensely change the physics of percolation on multiplex networks. In chapter five we show that geometric correlations and community correlations cause most of the real multiplex networks to behave counter-intuitively robustly when facing not only random failure (site percolation) but also extreme structural stress.

1.2 Results

1.2.1 Bond percolation and neuronal avalanches

Very often in physics a criticality of a system is manifested through power laws [86, 87]. One of the contexts where power laws emerge is neuronal systems [88]. Neuronal system is a network of connected neurons in which a firing mechanism activates neighboring neurons and gives rise to collective behavior. A cluster of nodes composed of connected activated neurons is called an avalanche [89]. In neuronal systems power law distribution of avalanche sizes has been reported in experiments and simulations [88–90]. Power law distribution of avalanche sizes has been considered an indication that a neuronal system is at its critical point and benefits from the criticality [91–94]. However, power law distribution of avalanche

sizes is associated with criticality if the system passes other criticality tests like collapsing temporal profiles and finite size scaling [95, 96].

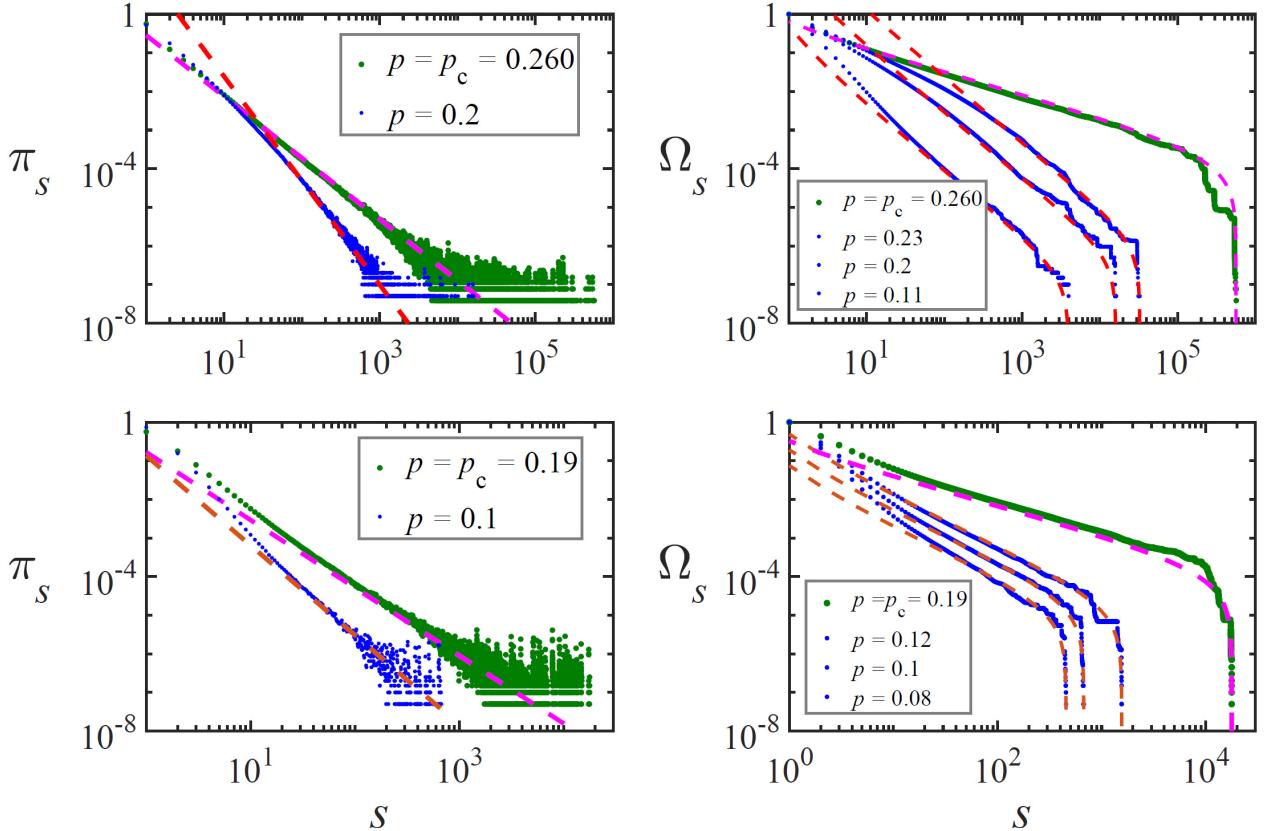


Figure 1.4 Probability and cumulative distributions of avalanche sizes. π_s (Ω_s) is the probability(cumulative) distribution of avalanche sizes of neuronal dynamics described in the text and here [21], obtained from theory (dashed lines) and numerical simulations (dots). Top row corresponds to the uncorrelated directed network topology with distribution of indegrees and outdegrees following scale free distribution with exponent $\lambda = 3.7$ and $\lambda = 2.7$, respectively. Bottom panels row corresponds to the correlated directed network topology with distribution of indegrees and outdegrees following scale free distribution with exponent $\lambda = 3.7$ and $\lambda = 3.1$, respectively. In all the panels lines and dots from right to left are in decreasing order of p . It is evident from the plot that emergence of power-law distributions of avalanche sizes can occur even in non-critical states of the system. (Extracted from [21])

We use a simplified model of neuronal avalanches called independent cascade model (ICM) [97]. ICM was designed and utilized in the context of social networks to investigate

the dynamics of opinion spreading [97, 98]. ICM starts from a single activated node, then at each time step, each activated node activates its inactive neighbors with probability p . The avalanche stops when no new activation occurs in the network. Then one is usually interested in measuring the distribution of sizes of active clusters (components of connected activated nodes) or avalanches. It has been proven that ICM is equivalent to bond percolation model [97]. This analogy maps the problem of distribution of avalanche sizes to finding of the distribution of connected components of the network under bond percolation process. The analogy roots in the fact that probabilistic activation of neighboring neurons can be decided beforehand, before firing of any neurons. At the beginning, each link is checked for activation with probability p . Then connected components using active links are obtained. In an active component, firing of a single neuron can activate all the nodes in that component and thus each of active components corresponds to a single avalanche and therefore, ICM can be captured by bond percolation model.

We come up with a unifying and general framework to study aforementioned neuronal dynamics on monoplex networks. Our framework is able to describe avalanche properties in undirected networks, uncorrelated directed networks and correlated directed networks. In correlated directed networks correlation between indegrees and outdegrees of nodes are allowed. In uncorrelated directed networks there is no correlation between indegrees and outdegrees of the nodes. Using generating function approach developed in the percolation theory of complex networks and techniques from branching processes, we were able to solve the model on random graphs [21] with the main assumption of locally tree-likeness of the underlying networks. Our theoretical analysis and simulation results show that in a non-critical system, power law distribution of avalanche sizes can arise due to the topological features of the underlying neuronal network such as degree distribution [21]. Fig. 1.4 illustrates the main finding of our study. We applied the theoretical framework on the described neuronal dynamics for correlated and uncorrelated network topologies. As obvious from the Fig. 1.4 power law distributions of neuronal avalanches arise even when the system in not in

its critical state.

1.2.2 Core percolation and k-core decomposition

Structural decomposition of a network sheds light on its structural properties. It can be described as grouping nodes of a graph in different categories with respect to a property. k -core decomposition is one of useful structural decompositions of networks [99]. k -core of a graph is a sub-graph in which each of its nodes has at least degree k . k -core of a graph can be obtained easily by a simple pruning algorithm. Removing nodes of a graph that have degree less than k , and continuing this pruning process until no node with a degree less than k is left, yields the k -core of a graph. k -core of a graph does not need to be a single connected component and it is potentially composed of disconnected sub-graphs [100]. k -core decomposition has many applications including, but not limited to, network visualization [100], locating influential spreaders [101], predicting robustness of mutualistic ecosystems [102], spectral properties [103] and so forth.

The largest k -core component is called as giant k -core. Analogous to emergence of GCC of random graphs, giant k -core of random graphs also undergoes a phase transition, i.e. below critical point there is not a giant k -core and beyond critical point giant k -core appears. However, the order parameter behaves differently in k -core phase transition in comparison to GCC phase transition. The behavior of the order parameter is discontinuous in the former whilst it is continuous in the latter. At the transition point $\langle k_c \rangle$ (an average degree of the random graph) giant k -core emerges and type of the phase transition and critical point depends on the number k . For $k = 1, 2$ the transition is of continuous type, similar to the phase transition of GCC of random graphs. For $k \geq 3$, phase transition of giant k -core is a hybrid type; discontinuous jump of the order parameter followed by critical singularity [11].

Another important sub-graph is called core. Core of a graph is obtained by applying a pruning procedure called Greedy Leaf Removal (**GLR**) which is introduced in the context

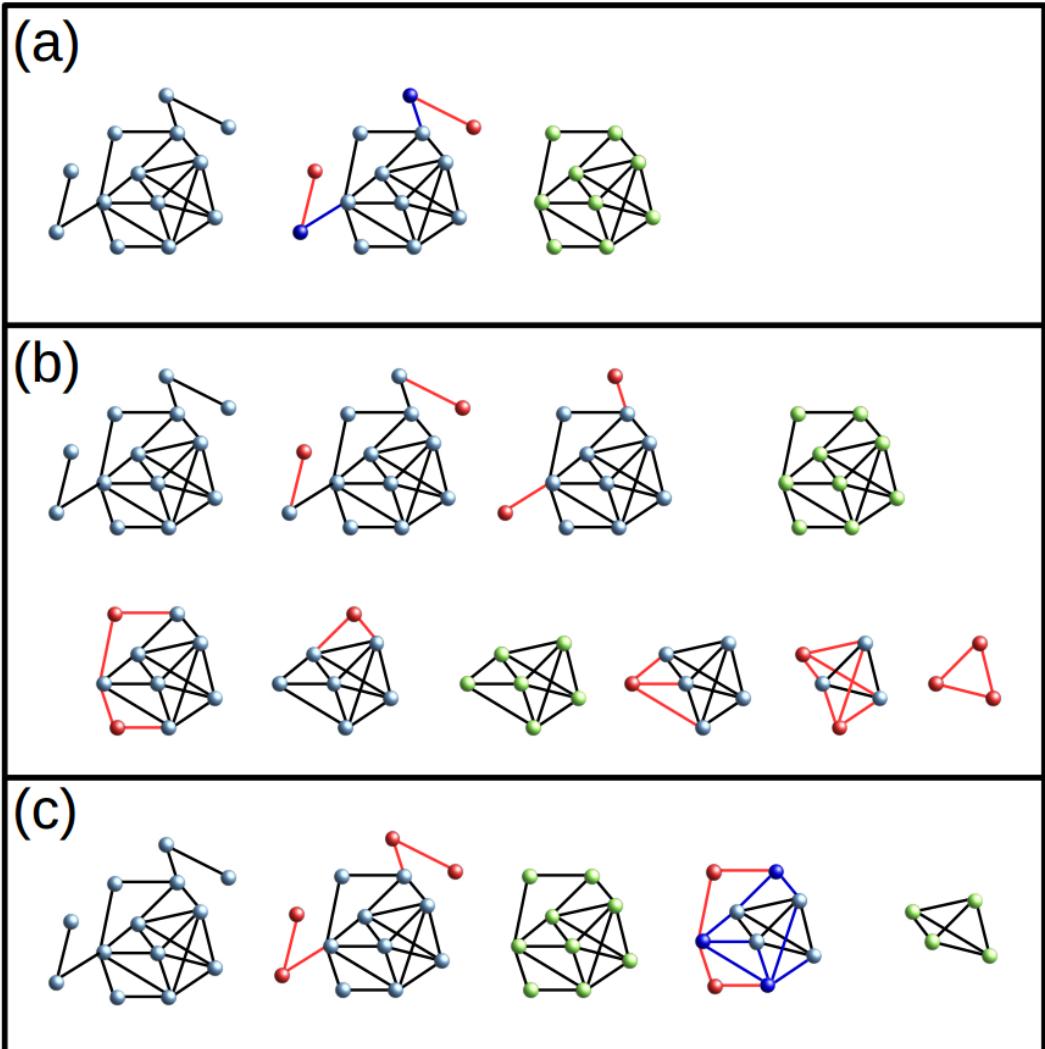


Figure 1.5 Greedy leaf removal. Schematic illustration of core, k -core and G_k -core decomposition of a small network. Panel (a) shows GLR of core percolation, panel (b) corresponds to the k -core decomposition of the network and panel (c) illustrates GLR of G_k -core. Nodes and edges in the red (blue) color are removed directly (indirectly) at each step and stable cores are colored in green. The network has a non-zero core, k -core and G_k -core for $k = 1, 2, 3$.

of maximum matching of graphs [104]. Leaves (nodes with degree one) of the graph are chosen randomly one by one. Then the chosen leaf and its immediate neighbor are removed from the graph. GLR is applied repeatedly until no leaf is left in the graph. This procedure yields the core of the graph. As in the case of GCC and giant k -core of random graphs, giant

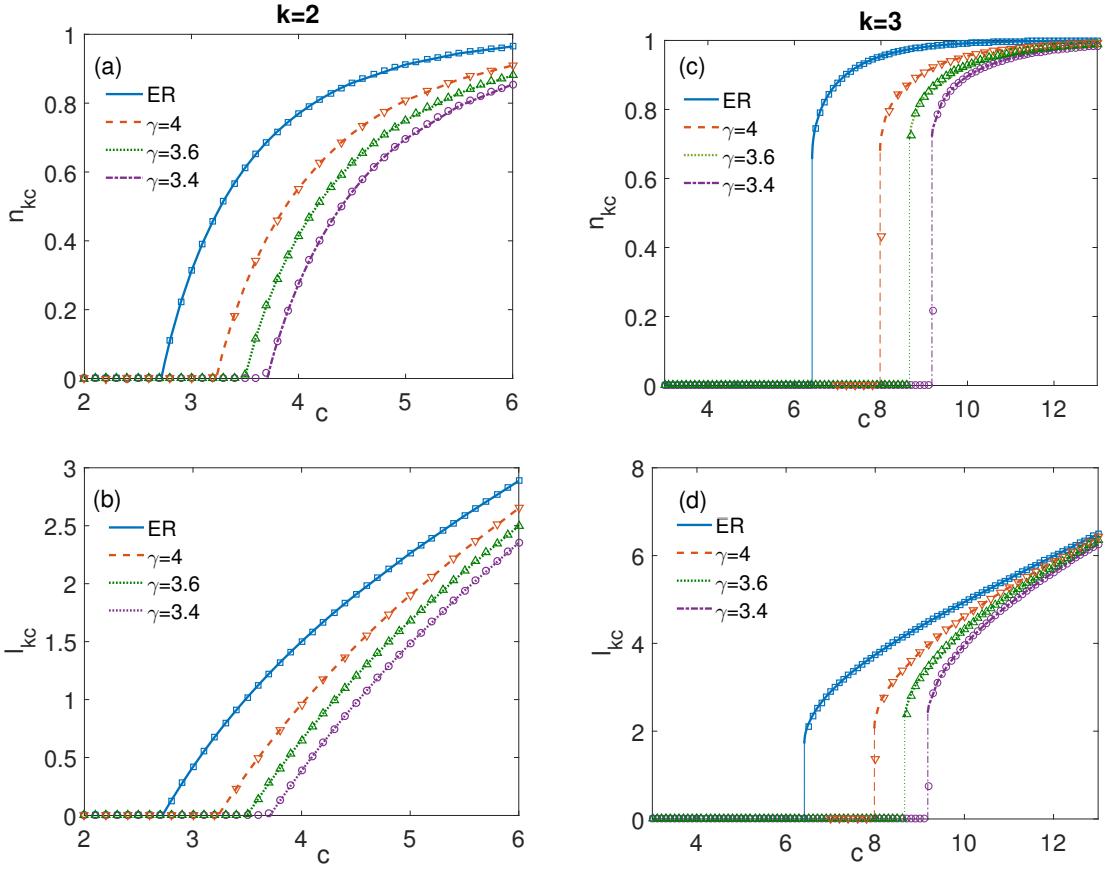


Figure 1.6 G_k-core percolation. The G_k-core phase diagram are obtained for Erdős–Rényi and scale-free networks with $N = 10^6$ nodes and for two values of $k = 2$ and 3. Top row (a and c) corresponds to the size of the giant G_k-core and bottom row (b and d) corresponds to the normalized number of links in the G_k-core. Markers are simulation results averaged over 10 realizations of network instances and lines are the theoretical results.(Extracted from [22])

core of random graphs undergoes phase transition. The giant core emerges continuously at the critical point $\langle k \rangle = e$ [104, 105]. Core of graph plays an important role in several optimization problems including: maximum matching [99, 106], minimum vertex cover [107], network controllability [20, 108], etc.

We generalize the definition of a core in chapter three. Our definition is based on the generalization of leaf nodes. In a graph, leaf refers to a node with degree one. We defined “ k -leaf” as generalized leaf. k -leaf is defined as a node with degree less than k . Then we

followed the GLR pruning process of core percolation [104]: a randomly chosen k -leaf and all of its immediate neighbors are removed. The process is done iteratively until no k -leaf is left in the network and consequently the generalized core of the graph is attained. Notice that when we remove a node all the links connected to that node are consequently removed. We call this new core as generalized k -core (G_k -core) of a network. G_k -core like k -core decomposes the network into nested sub-graphs [22]. It is worth to mention that in G_k -core notation, G_2 -core is the same as core of the network. Fig. 1.5 shows the GLR of core, k -core and G_k -core for a small graph.

As in the case of GCC, giant k -core and giant core we also observe phase transition in the emergence of giant G_k -core. The type of phase transition associated with appearance of giant G_k -core of a graph depends on the k -number. We studied the G_k -core percolation on random graphs using generating function and rate equation methods [22, 64]. As expected G_2 -core undergoes a continuous phase transition, since G_2 -core percolation is the same as ordinary core percolation [105]. However, giant G_k -core for $k \geq 3$ emerges in a discontinuous manner similar to k -core percolation phase transition. Fig. 1.6 illustrates the phase transitions diagram of G_k -core on random Erdős-Rényi and scale-free networks for two values of $k = 2$ and 3.

1.2.3 Depth-one percolation or observability phase transition

Here we introduce another variant of percolation model observability or depth-one percolation and generalize it to multiplex networks. Observability captures the idea of observing or monitoring state of a system. Let's assume there is a dynamical process running on top of a network. By placing a sensor or an observer on a node, one can record the states of the node and its immediate neighbors [109, 110]. Monitoring power-grid networks is the well-known example of observability [109]. Observability of monoplex networks has been defined and solved analytically on monoplex random [109] and real networks [111].

Observability phase transition of monoplex networks is defined as follows. As in the case of simple site percolation each node hosts a sensor with probability ϕ . However, there is a subtle difference between the site percolation and observability. When a node is observed (occupied) all of its immediate neighbors become observable (occupied) too. Hence, each node of the network falls in one of the following categories:

- directly observable, if the node is hosting a sensor.
- indirectly observable, if the node is not hosting a sensor but at least one of its immediate neighbors is hosting a sensor.
- not observable at all.

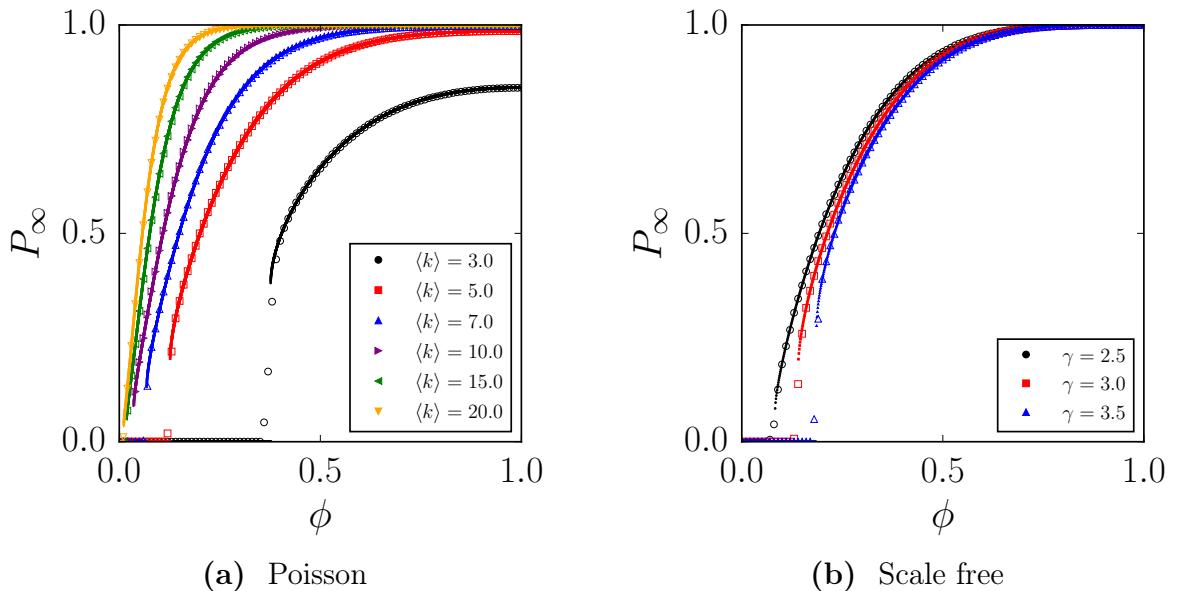


Figure 1.7 Observability phase transition of random multiplex networks. The observability model is applied on random multiplex networks. **(a)** We considered two-layer multiplex networks where each layer is created by Erdős–Rényi model with average degree $\langle k \rangle$. **(b)** The underlying multiplex networks are duplexes with random scale free networks with degree exponent γ . In both panels sizes of the networks are fixed to $N = 10^4$. Small markers are analytical results and big markers correspond to simulations.(Extracted from [23])

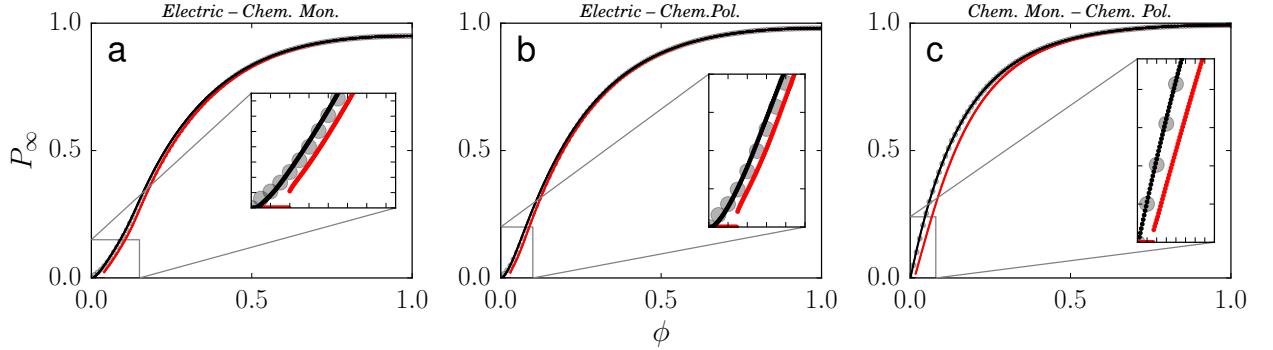


Figure 1.8 Observability phase transition of real multiplex networks. The underlying multiplex network corresponds to the *C.Elegans* connectome. Each layer of this multiplex represent different interaction type between neurons: electrical, chemical monadic, and chemical polyadic. Three duplex composed of different combination of layers have been considered (Panels (a), (b) and (c)). Observability model has been applied on these multiplexes. Grey circles are the results of the simulations with averaging done over 10000 realizations of the model. Small red circles are the predictions of message passing without overlap and small black circles are the predictions of message passing with overlap. Insets of the panels zoom in the part of the phase diagrams close to the critical points.(Extracted from [23])

Either directly or indirectly observable, connected observable nodes make clusters of observable nodes. The goal of observability phase transition is to study the emergence of giant observable cluster and the critical point of the phase transition. For synthetic random graphs and in the thermodynamic limit this phase transition model has been solved analytically. Observability has a smooth second order phase transition with a very small value of ϕ [109]. The same approach is generalized and extended to the real networks using message-passing or belief propagation methods [111]. Critical point of observability transition on real networks are very close to 0 [111], meaning that most of the real networks are already in observable regime.

In chapter four we study observability phase transition on multiplex networks. Generalization of observability model to multiplex networks is straightforward [23]. In the case of a two-layer multiplex network, placing an observer or a sensor on a node, makes it observable in both layers of the multiplex network (directly observable). A node of a multiplex net-

work is indirectly observable if it is indirectly observable in each of the layers, i.e. in each layer at least one of its immediate neighbors is indirectly observable. Similar to monoplex observability, observable nodes either directly or indirectly yield components of observable nodes. In the case of multiplex networks we are interested in the largest mutually observable cluster (LMOC). LMOC is a maximal subset of nodes that are observable and there is a path between them in both layers through the nodes inside the subset.

We studied observability phase transition on two-layer random multiplex networks. Fig. 1.7 illustrates the phase diagrams for random Poisson and scale-free networks. The quantity of interest is P_∞ which is a normalized fraction of the nodes in the LMOC. As expected the phase transition is of the first order type and there is a good agreement between theory and simulation.

Fig. 1.8 shows the phase diagram of observability phase transition of real multiplex networks. We utilized message passing tools in order to derive the full phase diagram of the model [23]. Similar to generating function approach our message passing framework is based on locally tree-like assumption. However, the presence of higher order correlations like edge overlap introduces errors in message passing method. Utilizing the recent method to deal with overlap we designed a message passing algorithms for observability phase transition in the presence of edge overlap [49, 112]. Fig. 1.8 shows that real multiplex networks have a critical point of observability transition with small value of ϕ , i.e. they are in observable regime. It is worth to mention that message passing framework with overlap successfully captures the continuity of the phase transition. However, message passing without overlap predicts discontinuous phase transition. Difference in the observability phase diagrams of random and real multiplex networks suggest that inter-layer correlations like edge overlap change the type of the phase transition [23].

1.2.4 Community-correlated multiplex

In previous sections we defined percolation on multiplex networks and we discussed how inter-layer correlations in multiplex network induce new physics. In my thesis I consider targeted percolation [113], optimal percolation [114] and depth-one percolation or observability phase transition (previous section) [109] on multiplex networks. As explained in introductory section, it is important to know how structural properties of a network can affect the dynamics running on top of it [6]. For instance, in the case of monoplex networks a homogeneous degree distribution causes a finite percolation phase transition point, while a heterogeneous degree distribution leads to $p_c = 0$, i.e. phase transition of percolation occurs at 0 [115]. When considering multiplex networks one usually is interested in topological factors beyond single layer structures [47]. There are many studies analyzing the effect of different topological structures within monoplex networks and how these structural properties affect the dynamics [2]. In multiplex networks we focus mainly on inter-layer correlations. In chapter five we analyze community-correlated structures [24] and how these high level correlations between layers can change the physics of percolation process on a two-layer multiplex network.

The motivation of this study came from a recent study by Kleinberg et al. [50] which revealed that most of the real multiplex networks contain some hidden inter-layer geometric correlations. In Ref [50] the authors mapped layers of a multiplex network into a hidden metric space (hyperbolic in this case). The mapping consists in association of coordinates to each of the nodes. Then they observed that there are non-trivial correlations between the coordinates of replica nodes of the layers. The same group later showed that these geometric correlations are the main reason of the emergent robustness of real multiplex networks under a targeted attack [51].

In chapter five we revisit the findings of these papers in the language of community structure. First we show that there is a one to one correspondence between hyperbolic embedding and community structure of a network. In previous section we discussed that

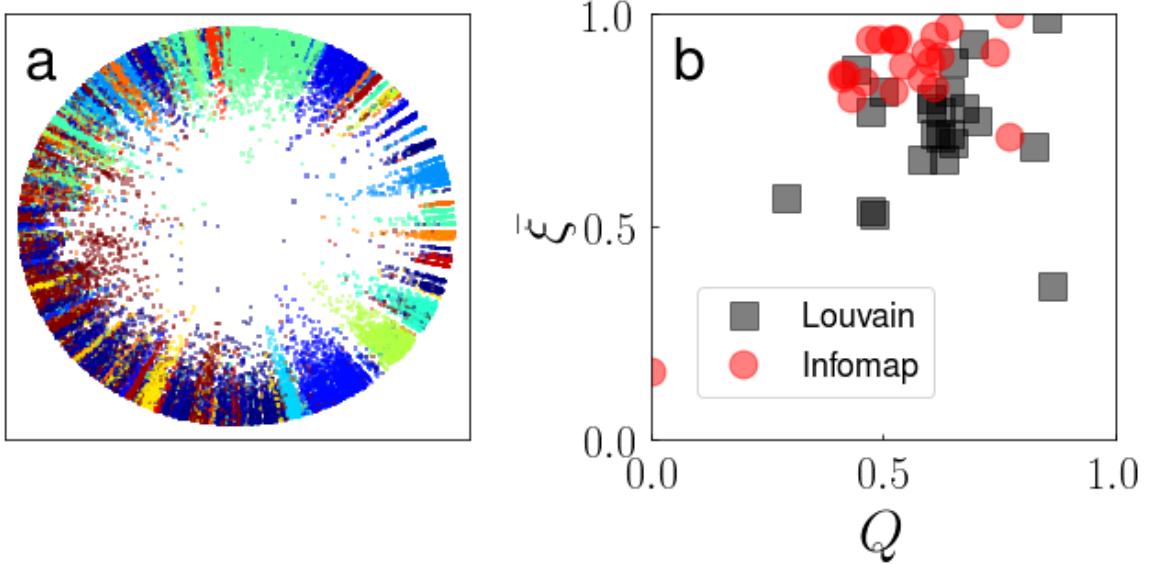


Figure 1.9 Hyperbolic embedding community structure of networks. Panel **a** shows the hyperbolic embedding of IPv4 Internet and its corresponding community structure. Position of the nodes are coded with (r, θ) resulting from hyperbolic embedding of the network. Community structure of the network is color coded, i.e. nodes belonging to one community have same color. Panel **b** is the scatter plot of 39 real-world network in $(\bar{\xi}, Q)$ -plane. Each point represents a network, its modularity and angular coherence.(Extracted from [24])

hyperbolic embedding of a network is set of coordinates (r, θ) . Radial coordinates abstracts popularity of the nodes and are related to their degrees. On the other hand $\Delta\theta_{ij}$ captures the similarity between nodes i and j [67]. The smaller the hyperbolic distance between a pair, the higher is the probability that they are connected [38].

Network science offers community structure as an alternative representation of networks. Nodes belonging to community are tightly connected and the probability of finding a connection between the nodes is higher than the probability of establishing connection between the nodes belonging to different communities. This analogy suggests that if a network with community structure is embedded in hyperbolic space nodes belonging to the same community tend to appear close to each other in the hyperbolic representation. We tested this idea by applying hyperbolic embedding on IPv4 Internet structure. Fig. 1.9a shows that

communities (nodes with same color) are located tightly in sectors of hyperbolic disk. One needs a proper way to quantify the analogy between hyperbolic embedding and community structure and support the claim that nodes within a community are located coherently in the hyperbolic embedding. To this end, we define the angular coherence ξ_g of a community g as $\xi_g e^{i\phi_g} = \frac{1}{n_g} \sum_{j \in g} e^{i\theta_j}$, where n_g is the number of the nodes of the community g . $\xi_g \in [0, 1]$ properly captures the idea of localization of the nodes of the community within a slice of hyperbolic disk. $\xi_g = 0$ is the extreme case where the nodes of the community g is scattered in the whole hyperbolic disk. The other extreme is $\xi_g = 1$ where nodes of the community g is localized in a narrow slice of the hyperbolic disk. Then the overall angular coherence of a network is obtained by weighted averaging over all of its communities: $\bar{\xi} = \frac{1}{N} \sum_{g \in C} n_g \xi_g$. Fig. 1.9b shows the scatter plot of angular coherence (from hyperbolic embedding) and modularity (from community detection) of 39 real world networks. This figure supports our claim that the hyperbolic embedding and community structure of networks can be considered as two sides of the same coin.

In the next step we deployed this analogy and showed that geometric correlations and, specifically correlations between similarity of the nodes in a multiplex network can be considered as a correlation of community structure between the layers. If (a) there is one to one corresponding between hyperbolic embedding and community structure and (b) angular correlations are the reason behind robustness of multiplex networks then one can conclude that community correlation can increase the robustness of multiplex networks [24]. In the remainder of this section I briefly introduce the percolation model, multiplex networks with community correlation and the results that support this statement.

In order to inspect the robustness of a multiplex network we use the following percolation model which is called a targeted attack strategy (an optimal version of percolation) on a multiplex network. First the score $s_i = \max(k_i^{[1]}, k_i^{[2]})$ of each node i of the multiplex network is calculated and then the node with the highest score is removed from the both layers of the multiplex network [51]. While removing nodes one after another one keeps track of the

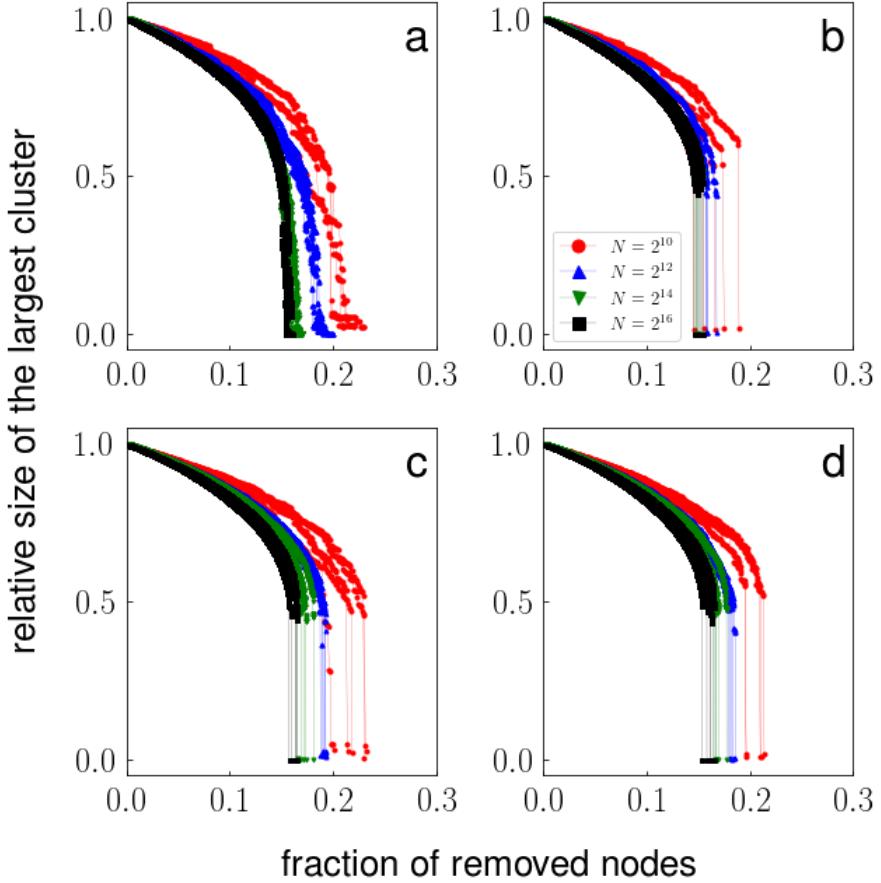


Figure 1.10 Robustness of community correlated multiplex networks. We considered random multiplex network with community correlated structures. The networks are created using LFR method [116] as explained in the text. The targeted attack algorithm described in the text is applied to these multiplex networks and relative size of GMCC is plotted versus fraction of removed nodes. Continuous phase transition of GMCC here indicates the robustness of the multiplex network. Panel **a** illustrates size of GMCC versus fraction of removed nodes for multiplex networks with community correlation. **b** is the same as **a** but for randomized multiplex networks. Panels **c** and **d** are the same as panels **a** and **b** but for multiplex networks with weaker community structure.(Extracted from [24])

size of the GMCC. It is worth to mention that this algorithm is adaptive and at each step of the algorithm the degrees of the nodes and consequently the scores of the nodes should be updated.

To analyze the effect of community correlations on the robustness of multiplex networks

we follow these steps. First we create a single monoplex network with inherent community structure using LFR model. LFR model creates networks with scale-free degree distributions and implanted community structures [116]. Then we use this layer as the structure of the second layer of the multiplex network. Up to this point, this multiplex is a fully overlapping multiplex network with a community structure which can be considered as a monoplex network. This monoplex network as expected undergoes a continuous second order phase transition in the percolation model. Then two randomization procedures are applied to this multiplex network. In the first randomization procedure one goes through the communities of the second layer and in each community and changes the labels of the nodes within community and shuffles their labels. This randomization process keeps the inter-layer community correlation while removing inter-layer degree correlation and edge overlap. In the second randomization procedure, one picks nodes of the second layer one after another and relabels its label with the label of another randomly chosen node from the same layer. This randomization procedure removes all inter-layer correlations including community correlation between the two layers. Fig. 1.10 shows that how multiplex networks with community correlated structures behave robustly against the targeted attack.

1.2.5 Optimal percolation

Percolation deals with emergence of large scale or extensive connectivity by modulating microscopic occupation probability p [75]. Optimized version of percolation refers to finding the minimum number of nodes or links in order to dismantle a graph into small connected components with sub-extensive size of the original graph [117, 118]. Optimal percolation is the optimized targeted attack on networks in which one tries to destroy the GCC (GMCC) of a monoplex (multiplex) network by removing minimum number of nodes [114]. Obviously there is a trivial solution to destroy a GCC of a graph which is removing all of its nodes. However, we are not interested in this solution due to the optimization nature of the optimal

percolation problem. There are better approximate solutions to this optimization problem than the trivial one.

Graph dismantling or optimal percolation can be defined mathematically in a proper way [117, 118]. In a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ a C -dismantling set is defined as a subset of nodes, whose removal upper-bounds the size of GCC by C . C -dismantling number is the minimum size of all C -dismantling sets. In real applications instead of C -number usually we are interested in scenarios where one can reduce the size of GCC to 0.01 of its original size. However, in physics, thermodynamic limit (large N) implies that $0.01 \times N$ is still an extensive quantity. Thus, conventionally the threshold $GCC_{th} = \frac{1}{N} \times \sqrt{N}$ is used as the condition to check if a graph is dismantled.

Optimal percolation or graph dismantling is of a huge interest in network science mainly due to the fact that many of interesting optimization problems can be mapped into the optimal percolation problem. For instance, influence maximization (maximizing an effect in the social network) [97, 114] or effective epidemic containment (finding optimal nodes to vaccinate in order to contain an epidemic) [119, 120] are two well-known examples in this respect.

Optimal percolation is an NP-complete problem [117]. It means that the time needed to solve this problem grows exponentially with the size of the system. However, recently fruitful attempts have been made in order to solve this problem in life-scenario situations which involve real networks [114, 117, 120–122]. There are heuristic algorithms which attain approximate solutions for an optimal percolation problem. These algorithms can be categorized in three main categories: greedy algorithms, score-based algorithms and, finally, many body algorithms.

- Greedy algorithms: These algorithms at each step remove the node that leads to the largest decrease in the size of GCC (or any other quantity of our interest). [97]
- Score-based algorithms: These algorithms are based on defining a score for nodes. The

score abstracts importance of a node in the structural robustness of the graph. Simply one can imagine attributing structural centralities to nodes. Then starting from the nodes with the highest score nodes are removed one by one until the size of GCC (GMCC) reaches the predefined threshold value [114].

- Many-body approaches: These approaches consider the fact that optimal percolation is related to the large scale connectivity and thus it is a many-body problem. Then a proper definition of Hamiltonian or a cost function is needed in order to capture the many-body aspect of the problem. However, as mentioned previously, optimal percolation is an NP-complete problem and these algorithms are not able to solve them exactly, however, in most cases they outperform heuristic algorithms [117, 122].

In chapter six I study optimal percolation on multiplex networks [14, 123]. The goals of this chapter are **a)** to generalize optimal percolation to multiplex networks and design heuristic algorithms or generalize algorithms of optimal percolation of monoplex networks to approximate the solution of optimal percolation on multiplex networks and **b)** to study the effect of multiplexity and inter-layer correlations. We consider only two layer random and real multiplex networks. First, optimal percolation problem is solved approximately on these multiplex networks. Then the optimal percolation is solved approximately on the constituent monoplex layers of the multiplex networks separately. After that the relation between the structural set of multiplex networks and its constituent layers are quantified. Finally the effect of inter-layer correlations such as edge overlap on the optimal percolation problem is investigated.

In the problem of optimal percolation of multiplex networks the goal is to reduce the relative size of GMCC to the predefined threshold value $GMCC_{th} = \frac{1}{\sqrt{N}}$. We extended algorithms of optimal percolation of monoplex networks to multiplex networks. This generalization can lead to some confusion. For instance, consider score-based dismantling algorithms. Generalizing score-based algorithms to the case of multiplex networks on its own

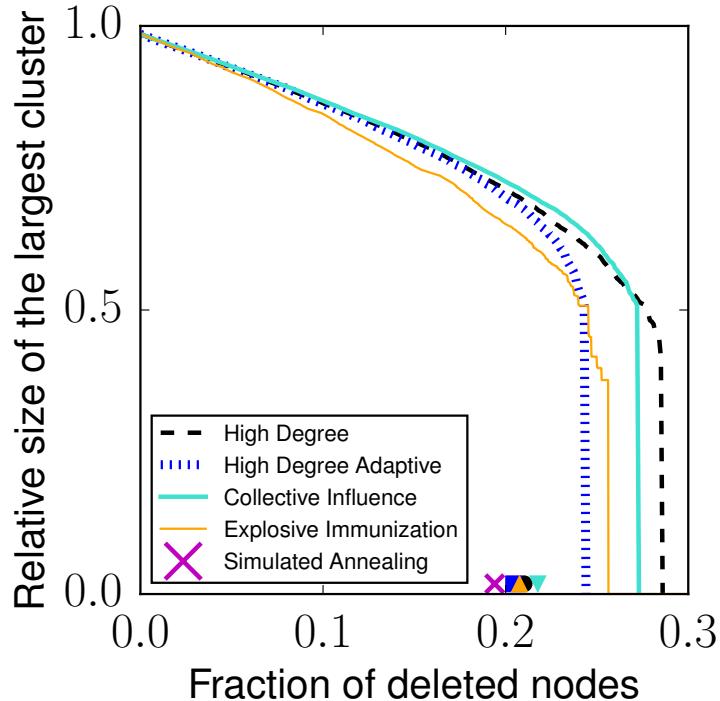


Figure 1.11 Optimal percolation on a random multiplex network. We considered a two layer multiplex network where each of the layers is created separately by Erdős-Rényi model with $N = 10^4$ nodes and average degree $\langle k \rangle = 5$. The plot illustrates the size of the GMCC as function of fraction of removed nodes. Markers show the improved efficiency of dismantling algorithms when combined by post processing greedy reinsertion. The cross marker identifies the approximate solution found by SA.(extracted from [14])

can be challenging. Let's consider a two layer multiplex network. A score based algorithm associates to node i $\sigma_i^{[1]}$ in the first layer and $\sigma_i^{[2]}$ in the second layer. Now it is legitimate to ask what is the score of the node i in the whole multiplex network, i.e. functional form of $\sigma_i = f(\sigma_i^{[1]}, \sigma_i^{[2]})$, where σ_i is the score associated to node i in the multiplex network. Here we briefly describe high degree (HD) algorithm and its generalization. This algorithm works by gradually removing nodes with highest degree until the GMCC becomes smaller than the threshold. There are many possibilities of merging HD scores of a node in layers of a multiplex network. Representing $\sigma_i = k_i$ (here the score is the same as degree) we can come up with the following candidates $k_i = k_i^{[1]} + k_i^{[2]}$ or $k_i = k_i^{[1]} * k_i^{[2]}$ or any other combination. More

details about generalizing scores could be found in the Supplementary Material of chapter six.

A drawback of most of score-based algorithms is that they are not considering the many-body aspect of the optimal percolation problem [117]. Therefore, these algorithms overestimate the size of the structural set. There will be some nodes within the dismantling sets found by these algorithms that can be removed from the dismantling sets and inserted back to the network while keeping size of the GMCC smaller than the threshold. Most of the heuristic algorithms are accompanied by this post processing step in order to get rid of the nodes that are not structural nodes.

Beyond the aforementioned score-based algorithms we also generalized and redesigned the simulated annealing (SA) algorithm previously designed for dismantling monoplex networks [117]. SA is a well known approach in tackling optimization problems. One defines a Hamiltonian for the system which in its essence captures the cost function of the optimization. Now the algorithm treats the system as a thermodynamical system. It is known that low temperature behavior of a physical system is associated with the lowest energy of the Hamiltonian. Thus, SA algorithm cools down the system until it reaches a minimum and then reads out the state of the system as a candidate solution of the Hamiltonian (cost function). The possibility of being trapped in local minimas of the Hamiltonian is the reason that SA is not a reliable method to find the ground state (lowest energy). However, with slow cooling there is a high probability to reach good solutions that are not attainable by other heuristic algorithms.

Fig. 1.11 illustrates the effectiveness of the designed algorithms in approximating the solution of the optimal percolation on a random multiplex network. Performance of the score based algorithms is boosted when they are combined by post processing greedy reinsertion. However, SA still offers the best approximate solution to the problem.

Finally the goal is to understand how multiplexity changes the physics of optimal percolation. As mentioned previously in the introduction section, the importance of multiplexity

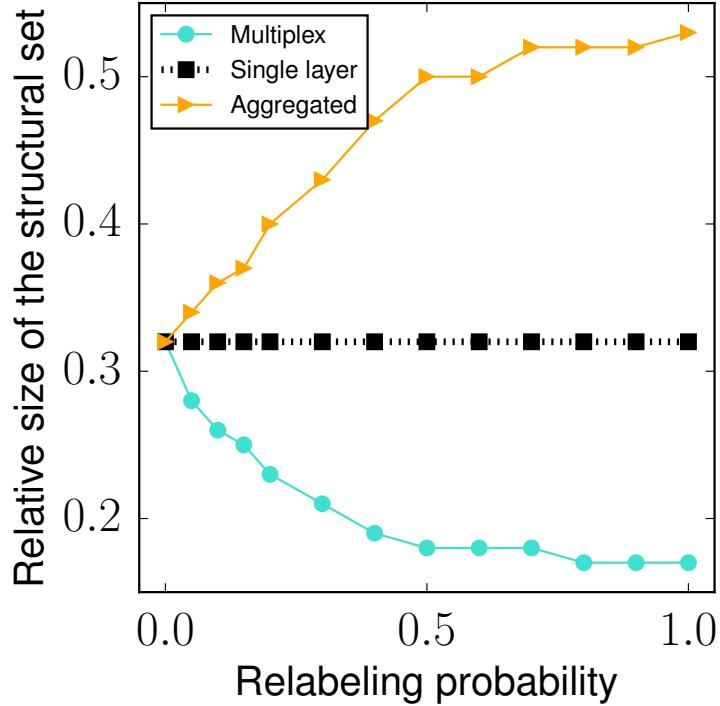


Figure 1.12 Effect of edge overlap on the size of structural sets. We consider a random multiplex network where initially both layers are identical and a copy of a graph generated by Erdős–Rényi model with $N = 1000$ nodes and average degree $\langle k \rangle = 5$. We select one of the layers, go through each node and switch the label of the node with another randomly chosen node with probability α . This randomization process explores the whole regime of multiplexity from full-overlapping to non-overlapping extremes. Then we apply SA algorithm on the randomized multiplex, their constituent layers and aggregated layers and obtain the average size of structural set for 100 realizations of SA algorithm. (extracted from [14])

emanates from the fact that most of the well-known phenomena studied on networks undergo drastic changes when the underlying network is of multiplex type. For instance, percolation, k -core percolation, super-diffusion, etc. are phenomena that radically change when considering multiplexity [9, 28, 55, 64].

The same question arises that is it important to consider multiplexity in the optimal percolation process and what are the consequences if one forgets multiplexity. Forgetting multiplexity means aggregating all kinds of links and interactions in one layer and treating the network as a single layer monoplex. In this regard we define structural nodes (SN) as

the set of nodes which are the solution of the optimal percolation problem. We obtain SNs of a multiplex network and SNs of each of the monoplex layers of the multiplex network. The latter is attained by applying optimal percolation on the monoplex layers separately, i.e. when one forgets about multiplexity. Having the SNs of multiplex network and SNs of its monoplex constituents, one can compare them to decipher the effect of multiplexity.

We consider two layer random synthetic multiplex network where the first layer is created using Erdős-Rényi random graph model. Then we copy the second layer from the first layer. At this point we have a monoplex network. By using the randomization technique introduced in previous section we can switch from monoplex to multiplex network and monitor the size of the SNs. Fig. 1.12 shows the result of the analysis of SNs on this multiplex network for different values of randomization parameter α . We briefly summarize the effect of multiplexity on optimal percolation:

- Size of the set of structural nodes of a multiplex network is smaller than sets of SNs of corresponding monoplex layers.
- Beyond the size of the SN sets, one realizes that identities of the nodes detected as SNs also differ in multiplex scenario from monoplex scenario.
- Inter-layer correlations as degree correlations and edge overlap justifies the discrepancy between SN set of multiplex network and SN sets of its associated single layer monoplex networks.

1.2.6 k -core of real multiplex networks

k -core of a graph refers to a subgraph in which each node has degree at least k . However, this definition is limited to monoplex networks. Azimi. et al. [124] generalized definition of k -core to multiplex networks. In the case of an L -layer multiplex network \mathbf{k} -core is defined as subset of nodes in which at each layer, degree of nodes satisfies the threshold condition

$k_i^l \geq k_l$ where threshold vector is $\mathbf{k}=(k_1, k_2, \dots, k_L)$. \mathbf{k} -core of multiplex networks can be attained by the generalized pruning process. In this pruning process one checks degree of each node and if at least in one of the layers the degree in that layer is smaller than the threshold of the layer, the node should be removed. The pruning procedure is iteratively applied until no node is left or we are left with \mathbf{k} -core of the multiplex network. More detail about the definitions of k -core, k -shell and corresponding algorithms can be found in [25, 124]. Azimi. et al. [124] introduced this model and solved them analytically using the method of generating function for locally tree-like random multiplex networks. However, they reported a discrepancy between theoretical prediction and simulation results for the case of real multiplex networks. The discrepancy is attributed to the underlying inter-layer correlations existing in the multiplex networks.

As explained in previous sections, different type of inter-layer correlations exist in multiplex networks. In order to systematically analyze effects of inter-layer correlations on the \mathbf{k} -core structure of multiplex networks, we used the framework of hyperbolic networks, i.e. each layer of the multiplex network is considered separately as an object embedded in hyperbolic space. Then each node i is represented by its coordinates (r_i, θ_i) . In this framework we are able to detect, control and remove existing inter-layer geometric correlations in multiplex networks. In this regard, as we explained in previous sections the simple approach to investigate the effects of inter-layer correlations on the dynamics or process taking place on top of a multiplex network is to use randomization technique. Randomization enables us to reduce or completely remove inter-layer correlations and consequently observe its effect on the \mathbf{k} -core structure of a multiplex network. For instance, a simple randomization of labels of the nodes in one of the layers of the multiplex while removes all inter-layer correlation between the layers, keeps the structure of the constituent layers unchanged. This technique allows us to separate the effect of inter-layer correlations from the one of the intra-layer correlations.

Fig. 1.13 illustrates the k -core structure of the multiplex networks of arXiv and its ran-

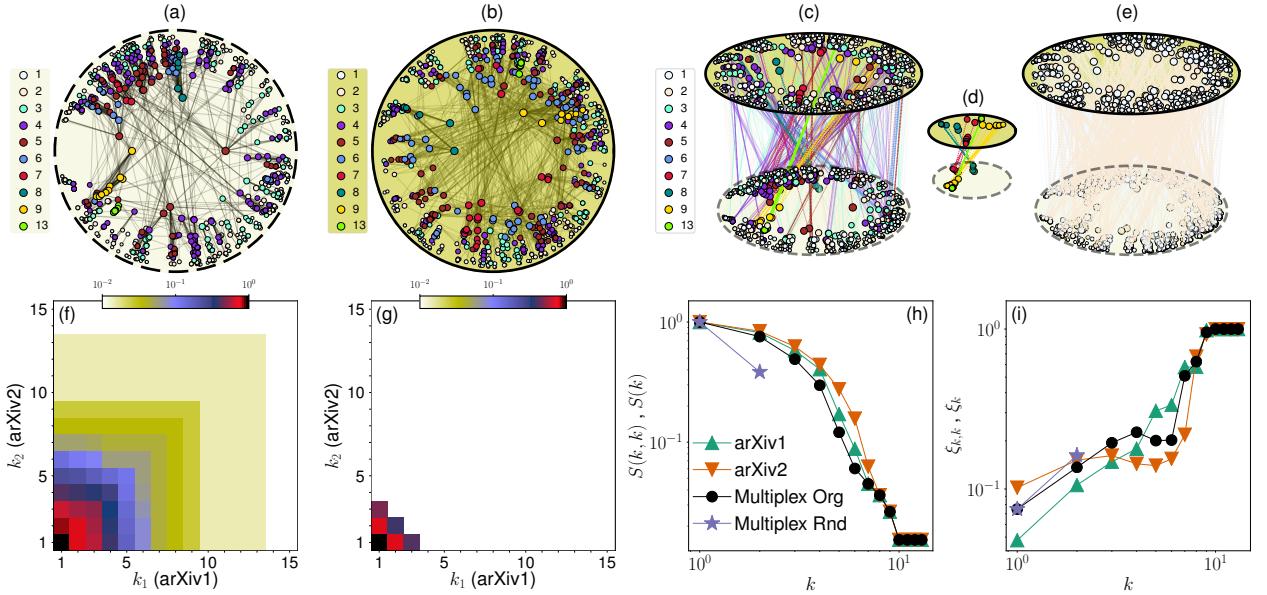


Figure 1.13 k -core of multiplex network of arXiv. Multiplex network of arXiv has been considered in this figure. Panels (a) and (b) illustrate the hyperbolic embedding of the two layers of the multiplex network. Positions of the nodes in the hyperbolic space correspond to their hyperbolic coordinates (obtained by embedding). Nodes are color coded with respect to their location in the k -shell decomposition of the network. Panel (c) shows the diagonal \mathbf{k} -shell decomposition of the arXiv multiplex network. (d) is the same as (c) for the few inner shells. (e) shows the shell decomposition of the multiplex network after randomization. Panels (f) and (g) illustrate the full \mathbf{k} -core spectrum of the original and randomized multiplex networks of arXiv. (h) shows the size of the k -core of each of the layers of the multiplex and size of the \mathbf{k} -core of the original and randomized multiplex networks. Panel (i) is the same as (h) but for angular coherence metric. (extracted from [25])

domized counterpart. We used hyperbolic embedding framework to visualize the networks. Here full randomization takes place where we randomize labels of the nodes in one of the layers. Randomization destroys all kind of inter-layer correlations. Fig. 1.13(h) shows the size of the k -core of each of the layers of the multiplex and size of the \mathbf{k} -core of the original and randomized multiplex networks. As clear from the Fig. 1.13(g), the randomization weakens the k -core spectrum. Moreover, we also defined coherence metric for nodes in k -cores. This metric is defined exactly in the same way as the one defined for community structure

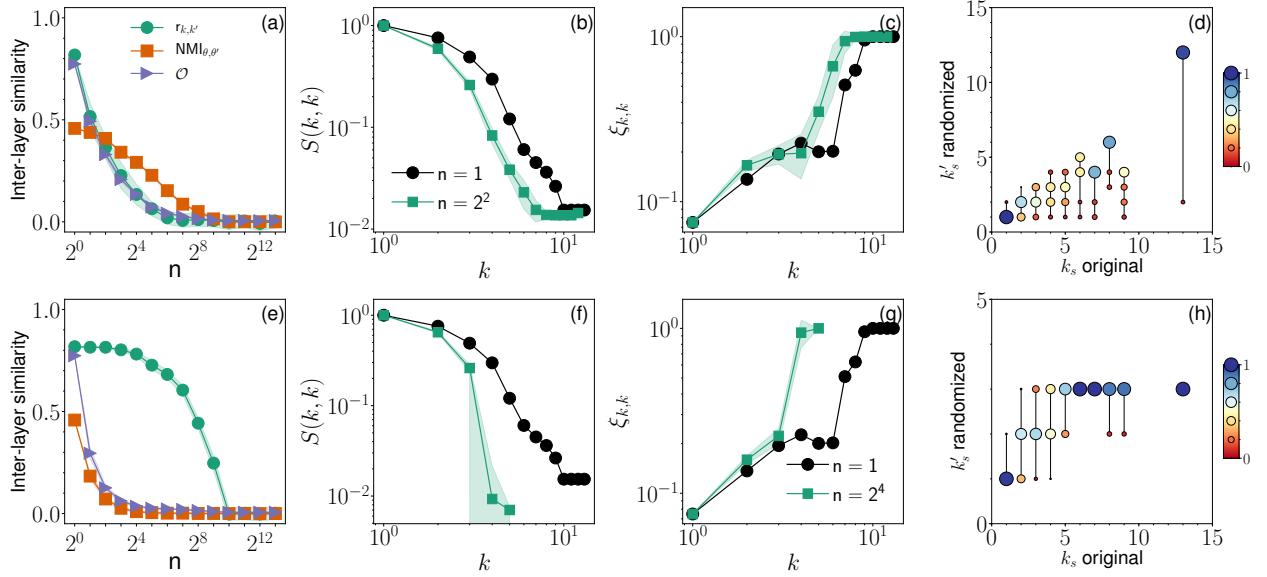


Figure 1.14 Randomization of multiplex network of arXiv In order to remove radial or degree correlations, radial randomization (top row) has been carried out. Panel (a) illustrates the inter-layer correlation metrics such as degree correlation, normalized mutual information of the angular coordinates and edge overlap as a function of groups size n of the randomization procedure (more detail can be found in chapter seven). It is obvious that increasing the group size n increases randomization effects. (b) shows size of the diagonal k -core of the original multiplex and its randomized version. While in the former all original inter-layer correlations are untouched, in the latter degree correlations are reduced (but angular correlations are preserved). (c) the same as (b) but for angular coherence metric. (d) illustrates the effect of randomization and relocation of the nodes from the inner shells of the original multiplex to the outer shells of the randomized multiplex. Each vertical line shows that how the original diagonal k -shell is redistributed in the new k -shells of the randomized multiplex network. Bottom row (panels (e), (f), (g) and (h)) are the same as top row (panels (a), (b), (c) and (d)) but for the angular randomization case in which angular correlations are removed while radial correlations are preserved. (extracted from [25])

in previous section. Fig. 1.13(i) shows that inner k -cores of both monoplex and multiplex networks are more coherent. While this analysis shows that inter-layer correlations play a big role in the k -core structure of the network, it does not give us the detailed information on how angular or radial correlations play role separately. Moreover, we want to know the role of the inter-layer correlations with respect to heterogeneity of the degree distribution.

In order to answer these questions we need a generalized randomization technique, i.e. a randomization technique which enables us to reduce each type of the correlations separately without destroying the other types.

To analyze the effect of geometric correlation (radial and angular correlations) separately, the following algorithm is introduced. Let's consider the case in which one wants to destroy angular correlations between the nodes of the layers. The following algorithm can serve this purpose. First we choose one of the layers and sort the nodes with respect to their angular coordinates. Now we group these nodes in small groups of size n . Then we relabel the labels of the nodes within each group. n is size of the groups and enables us to control the severity of randomization. The extreme case in which $n = 1$ corresponds to no randomization (relabeling) of the nodes and consequently preserves angular correlations (and other correlations) between the layers. The other extreme $n = N$ where N is the number of the common nodes of the multiplex network, corresponds to the complete relabeling of the nodes in which all kind of correlations are destroyed. The same algorithm is used in order to keep angular correlations and remove radial correlations. However, in this case one needs to sort the common nodes with respect to their degrees.

Fig. 1.14 shows the effect of different kind of randomization described above on arXiv multiplex network. We apply radial randomization (decreasing degree correlations while keeping angular correlations) and angular randomization (decreasing angular correlations while preserving degree correlations) on this multiplex network. As Fig. 1.14(b),(f) show the k -core structure of the arXiv is mostly affected (reduced) by removing inter-layer angular correlations. However, in the case of multiplex network of the Internet the opposite is observed 1.15. Considering Fig. 1.15(b),(f) one realizes that diagonal k -core of the Internet is mostly affected when one destroys inter-layer degree correlations. These simple experiments suggest that in general, both degree and angular correlations are important in order a multiplex to have a rich k -core structure. However, in the case of multiplex networks with more homogeneous degree distribution (arXiv) angular correlation between the nodes of the layers

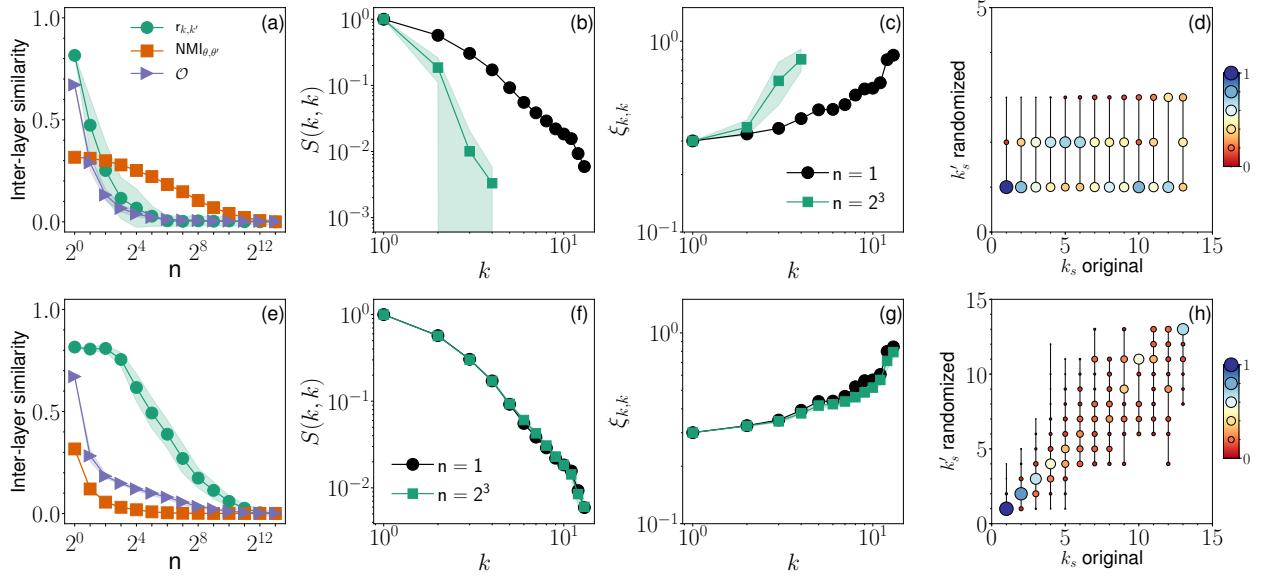


Figure 1.15 Randomization of multiplex network of the Internet. The same as Fig. 1.14 but for the multiplex network of the Internet.(extracted from [25])

is the main actor. In the case of multiplex networks with heterogeneous degree distributions (Internet) radial or degree correlation plays the most important role.

Chapter Two

Emergence of power laws in noncritical
neuronal systems

Contribution

I contributed to the modelling part of this research. My main contribution was performing the simulation of bond percolation model of neuronal avalanches and analysis of the results. Writing of the manuscript was mostly done by the main author Ali Fafeeh. All the authors contributed to the editing of the paper.

Emergence of power laws in noncritical neuronal systems

Ali Faqeeh,^{1,2,*} Saeed Osat,³ Filippo Radicchi,² and James P. Gleeson¹

¹*Mathematics Consortium for Science and Industry, Department of Mathematics and Statistics, University of Limerick, Limerick V94 T9PX, Ireland*

²*Center for Complex Networks and Systems Research, School of Informatics, Computing, and Engineering, Indiana University, Bloomington, Indiana 47408, USA*

³*Deep Quantum Labs, Skolkovo Institute of Science and Technology, Moscow 143026, Russia*



(Received 26 February 2019; published 2 July 2019)

Experimental and computational studies provide compelling evidence that neuronal systems are characterized by power-law distributions of neuronal avalanche sizes. This fact is interpreted as an indication that these systems are operating near criticality, and, in turn, typical properties of critical dynamical processes, such as optimal information transmission and stability, are attributed to neuronal systems. The purpose of this Rapid Communication is to show that the presence of power-law distributions for the size of neuronal avalanches is not a sufficient condition for the system to operate near criticality. Specifically, we consider a simplistic model of neuronal dynamics on networks and show that the degree distribution of the underlying neuronal network may trigger power-law distributions for neuronal avalanches even when the system is not in its critical regime. To certify and explain our findings we develop an analytical approach based on percolation theory and branching processes techniques.

DOI: [10.1103/PhysRevE.100.010401](https://doi.org/10.1103/PhysRevE.100.010401)

I. INTRODUCTION

In neuronal systems, but also in many other apparatuses crucial to living organisms, the emergence of power-law distributions [1–6] has a remarkable importance. The unique form of distribution may indicate that the system is operating near a critical point [7–9] and is therefore benefiting from a series of potential advantages of critical systems [10–12], such as optimum information transmission [1,13], dynamical range and sensitivity to sensory stimuli [14], information capacity [15,16], and stability [13,17].

In neural systems, power-law distribution of avalanche sizes and durations (lifetimes) have been observed both in experiments [1,4,8] and computational models [17–19]. To validate that power-law distributions are indeed due to criticality, one needs to perform other tests [4], including testing finite-size scaling relations [4,20] and performing collapse of temporal profiles [4,20,21]. However, these techniques cannot determine what mechanisms or conditions keep or pose the system in the critical regime or if/how the system may lose its criticality. Nevertheless, other approaches can be used to demonstrate mechanisms or conditions that can lead to criticality of biological systems [22] or mechanisms (especially in living systems) other than criticality that can lead to power-law distributions [23–29]. In particular, Friedman and Landsberg [22] considered a simplistic model for neuronal dynamics and introduced a mechanism through which the hierarchical structure of neuronal networks can generate power-law distributions even far from criticality. The importance of network structure underlying neuronal dynamics for the

generation of power laws has been also reported in other studies [18,19].

In this Rapid Communication, we demonstrate that the degree distribution of the network underlying neural dynamics plays a fundamental role in the emergence of power-law distributions of avalanche sizes. To do so, we consider a simplified model of neural dynamics on networks, and show that, for some scale-free networks, avalanche sizes obey power-law distributions even in subcritical dynamical regime. Moreover, in other cases in which the avalanche size distribution is a power law with exponential cut-off, we disclose what structural parameters determine the cut-off size and show that even in such cases it is possible to observe distributions that are approximately power law over several orders of magnitude. In addition to numerical evidence, we provide an analytical description of the phenomenon relying on techniques borrowed from the theory of percolation [30] and branching processes [31–33]. We believe that our findings may have important implications in understanding properties of dynamics on real-world networks that have heavy-tailed degree distributions [2,5,34–37].

As mentioned above, we consider a simplistic model of neural avalanches for which we can show lucidly the impact of the network structure. In our model, an avalanche starts with a single activated neuron and, at each time step, every one of the active neurons fires a signal that stimulates all of their neighbors. This stimulus activates with a probability p each neighbor that has not been already activated. The avalanche of activities continues until no new neuron can be activated. This model is identical to the so-called independent cascade model, often considered in the context of opinion spreading in social networks [38–40]. For neural dynamics, it is a more realistic version of the Friedman-Landsberg model (FLM) [22] as in our model each time a neuron receives a stimulus it has the

*afaqeeh@iu.edu

chance to become activated while in the FLM the activation does not depend on the number of stimulations. In spite of its simplicity, our model captures the fast timescale behavior of integrate-and-fire models [41,42]. This fact follows from the simplifying assumptions that repetitive activation is neglected and the stimulations that activate a neuron (by increasing its potential to above its firing threshold) are set at random [19,22,43,44].

An advantage of this simplification is that our model is equivalent to a bond percolation model, thus, the avalanche size distribution is identical to the probability distribution π_s that a randomly chosen node belongs to a percolation cluster of size s . This analogy enables us to consider a set of well-established techniques developed for percolation models and branching processes. In the following, we first describe our analytical calculations. Then, we show that our theoretical predictions are in very good agreement with the results of numerical simulations.

We provide a unifying framework that can describe the avalanche properties on both undirected and directed networks. We consider networks with negligible source-target correlation, i.e., the correlation between the degree values at the ending points of an edge. Nevertheless, for directed networks (DNs), we include analysis for networks with and without input-output correlation, i.e., the correlation between the values of indegree j and outdegree k of a node. Thus we consider three network types: undirected networks (UNs), uncorrelated directed network (UDNs), and input-output correlated directed networks (CDNs).

To generate UNs with specific degree distribution we use the configuration model [45–47] and for DNs we use an extended version of this model [48]. In particular, if the number of stubs of indegree and outdegree distributions are unbalanced, we remove, from a fraction u of nodes, some stubs of the distribution with more stubs such that its tail conserves its form [48].

II. RESULTS

A. Relevant theoretical findings

Our analytical calculations are built on techniques originated from studies that shed light on structural properties of networks [49], spread of epidemics [50], properties of site percolation on undirected [51] and directed [52] networks, branching processes [32,53], spread of online information on Twitter [33], and relevant methods for obtaining the properties of generating functions [54,55]. The findings most related to our calculations correspond to those of Refs. [51,52] in which analytical results for the functional form of the distribution of cluster sizes in a site percolation process were reported. To improve upon the findings of these references, we substitute parts of the approaches they employed with our own techniques developed on the basis of Refs. [33,49,50,53–55].

B. Constructing the governing equations

For UNs, we consider the degree distribution p_k of the network and for DNs we consider the indegree distribution P_j , the outdegree distribution P_k (note that we use k for the degree in UNs as well as for the outdegree in DNs as, we will later

show that they play the same role in describing the avalanche sizes), and the joint degree distribution P_{jk} which equals the fraction of nodes with indegree j and outdegree k . From the degree distributions we can obtain the excess degree distribution functions $q_k = kp_k/\langle k \rangle$ for UNs or $q_{jk} = jP_{jk}/\langle j \rangle$ for DNs which describe the probability that following a random edge we find a node with, respectively, degree $k + 1$ (UNs) or indegree j and outdegree k (DNs).

We are going to calculate the distribution of avalanche sizes π_s . This quantity depends on ρ_s , the probability that following an edge of the network we reach an avalanche (cluster) with size s [30,49,50,56]. To calculate these quantities we will need to work with their generating functions defined as, respectively, $H_0(z) = \sum_{s=1}^{\infty} \pi_s z^s$ and $H_1(z) = \sum_{s=0}^{\infty} \rho_s z^s$. We will also need the generating functions for degree k and the excess degree distributions, defined as

$$G_0(z) = \begin{cases} \sum_{k=0}^{\infty} p_k z^k, & \text{UN} \\ \sum_{j,k=0}^{\infty} P_{jk} z^k, & \text{DN}, \end{cases} \quad (1)$$

$$G_1(z) = \begin{cases} \sum_{k=0}^{\infty} q_k z^k, & \text{UN} \\ \sum_{j,k=0}^{\infty} q_{jk} z^k, & \text{DN}. \end{cases} \quad (2)$$

A UN on which a bond percolation process with occupation probability p is applied should be described by $\widehat{G}_n(z) = G_n(1 - p + pz)$ instead [50,57], where $n = 0, 1$; it is straightforward to show that this property holds also for DNs. We use this fact to extend the governing equations that Newman derived for H_0 and H_1 in the absence of percolation [50] to our case; thus we get

$$H_1(z) = zG_1[1 - p + pH_1(z)], \quad (3)$$

$$H_0(z) = zG_0[1 - p + pH_1(z)]. \quad (4)$$

The first difference between our calculations and the method of Refs. [51,52] for calculation of cluster sizes is that we use the accurately derived Eqs. (3) and (4) instead of equations derived from heuristics [58].

The next steps of our approach include (i) calculation of the leading order nonanalytic behavior of $H_0(z)$ by finding the behavior of G_1 and G_0 around $\bar{\eta} = 1 - p + pH_1(1)$, and (ii) using the asymptotic properties of generating functions [48,54,55] to obtain π_s for large avalanche sizes ($s \gg 1$) using the results of (i). To do so, we integrate the above equations with the methods described in [51,52] and improve upon parts of these methods by combining them with techniques and ideas, including branching processes methods [33,53].

C. Solution methods for different regimes of dynamics

1. The critical and subcritical regimes

In these regimes, $H_0(1)$, which equals the probability that a randomly chosen node is in a finite cluster, can be set to 1 [for the supercritical regime, we can instead assume that

$H_0(1) \approx 1$, if p is not much larger than the critical occupation probability p_c ; thus, according to Eq. (4) $H_1(1) = 1$ too. Accordingly, to obtain the leading order behavior of $H_1(z)$ [from Eq. (3)] and $H_0(z)$ [from Eq. (4)], we can assume $H_1(1-w) \sim 1-\phi$, where $\phi \ll 1$ and $z \doteq 1-w$ for $w \ll 1$, and then expand the degree-dependent generating functions around $z=1$ to get [48]

$$\begin{aligned} G_1(1-p\phi) &\sim G_1(1) - G'_1(1)p\phi + G''_1(1)(p\phi)^2 + D\phi^{\tilde{\lambda}-1} \\ &+ o(\phi^2, \phi^{\tilde{\lambda}-1}) = 1 - \frac{p}{p_c}\phi + B\phi^2 + D\phi^{\tilde{\lambda}-1} \\ &+ o(\phi^2, \phi^{\tilde{\lambda}-1}), \end{aligned} \quad (5)$$

$$\begin{aligned} G_0(1-p\phi) &\sim G_0(1) - G'_0(1)p\phi + M\phi^{\tilde{\lambda}-1} + \mathcal{O}(\phi^2) \\ &= 1 - E\phi + M\phi^{\tilde{\lambda}-1} + \mathcal{O}(\phi^2), \end{aligned} \quad (6)$$

as $\phi \rightarrow 0$, where p_c is obtained analytically according to the results of Refs. [47,51,52], and the terms with $\tilde{\lambda}$ or $\bar{\lambda}$ are present only for scale-free networks; these effective exponents are

$$\tilde{\lambda} = \begin{cases} \lambda & (\text{UN}) \\ \lambda_o & (\text{DN}) \end{cases} \quad \text{and} \quad \tilde{\lambda} = \begin{cases} \lambda - 1 & (\text{UN}) \\ \lambda_o & (\text{UDN}) \\ \lambda_o - \frac{\lambda_o - 1}{\lambda_i - 1} & (\text{CDN}), \end{cases}$$

where λ , λ_o , and λ_i are the exponents for the tail of the distribution of, respectively, the degrees k in a UN, the outdegrees k in a DN, and the indegrees j in that DN. The other coefficients in Eqs. (5) and (6) depend on the network degree distribution [48]. Note that $p_c > 0$ for the range of $\tilde{\lambda}$ values we considered.

We keep up to the third (second) leading order term of G_1 (G_0) and substitute the result in Eq. (3) [Eq. (4)] to get

$$\begin{aligned} H_1(1-w) &\doteq 1-\phi \sim (1-w)\left(1 - \frac{p}{p_c}\phi + B\phi^2 + D\phi^{\tilde{\lambda}-1}\right) \\ &\Rightarrow w \sim -\frac{\delta}{p_c}\phi + B\phi^2 + D\phi^{\tilde{\lambda}-1} \end{aligned} \quad (7)$$

and

$$H_0(1-w) \sim 1 - E\phi + M\phi^{\tilde{\lambda}-1}, \quad (8)$$

where, in Eq. (7), $\delta = p - p_c$. Using Eqs. (7) and (8) we can show that the leading order nonanalytic term of $H_0(z)$, depending on the dynamical regime, has either the form Rw^β or $R\sqrt{1+s^*w}$ [48], where β is a noninteger number and s^* and R are constant. According to the asymptotic properties of generating functions [48,54,55], the first form gives a π_s with a power-law tail and the second form results in a power law with exponential decay. In particular, for $\tilde{\lambda} > 3$ and non-scale-free networks, $\pi_s = R_1 s^{-3/2}$ at the critical point (p_c) and $\pi_s = R_2 s^{-3/2} e^{-s/s^*}$ in the noncritical phases; however, for $2 < \tilde{\lambda} < 3$, $\pi_s = R_2 s^{-[1+1/(\tilde{\lambda}-1)]}$ at p_c (see Supplemental Material [48] for the definitions obtained for R_1 and R_2 [59]). For these cases, Refs. [51,52] reported the same results for the functional form of π_s ; however, the equations they used [instead of Eqs. (3) and (4)] underestimate the prefactors [48]. We also retrieve the result $s^* \propto \delta^{-2}$ calculated previously for UNs (using another method [51]) and we discover that a similar relation also holds for DNs; furthermore, we find that the exponential decay factor s^* is also controlled by

the skewness of the degree distribution according to $s^* \approx \frac{2p^2(k)\langle k^3 \rangle}{\delta^2\langle k^2 \rangle^2}$ for UNs and $s^* \approx \frac{2p^2(k)\langle jk^2 \rangle}{\delta^2\langle jk \rangle^2}$ for DNs. This indicates that even for skewed non-scale-free networks it is possible that a power-law distribution of avalanche sizes, expanded for several orders of magnitudes (i.e., as long as $s \ll s^*$), emerges.

On the other hand, despite the expectations of Refs. [51,52], at the subcritical regime of $2 < \tilde{\lambda} < 3$, we get pure power-law distribution in the form

$$\pi_s \sim \begin{cases} \widehat{R}(1 + \frac{pp_c(k)}{-\delta})(\frac{pp_c}{-\delta})^{\lambda_o-1} s^{-\lambda_o} & (\text{UDN}) \\ \widehat{R}(\frac{pp_c}{-\delta})^{\tilde{\lambda}-1} \left[(\frac{pp_c}{-\delta})^{\tilde{\lambda}-\bar{\lambda}+1} s^{-\tilde{\lambda}} + s^{-\bar{\lambda}} \right] & (\text{CDN/UN}), \end{cases} \quad (9)$$

where $\widehat{R} = a^*(1-\bar{u})$, and $a^* \doteq 1/\sum_k \bar{k}^{-\bar{\lambda}}$ for \bar{k} in the tail of the corresponding k distribution (degree or outdegree distribution) and $\bar{u} < 1$ depends on the mass of that tail. To obtain this result we assumed that the solution of Eq. (7) for $2 < \tilde{\lambda} < 3$ has the form $\phi \sim a_1 w^{\alpha_1} + a_2 w^{\alpha_2} + \dots$, where $\alpha_1 < \alpha_2 < \dots$, and used the dominant balance method [33,44,60] to obtain the correct form for the leading order terms [48]. As demonstrated by Eq. (9), the exponent of this distribution in UNs is $\tilde{\lambda} = \bar{\lambda} = \lambda_o$, and in CDs and UNs is $\tilde{\lambda}$; in CDs the leading order term can be corrected with a $\bar{\lambda}$ order term if the difference between the exponents $\tilde{\lambda}$ and $\bar{\lambda}$ is considerably small. Figure 1 shows that our predictions for the subcritical and critical regimes of $2 < \tilde{\lambda} < 3$ capture very well the power-law behaviors of the tail of π_s .

In the procedure for deriving Eq. (9) we made no assumption about the sign of δ ; however, we immediately notice that Eq. (9) is only valid for the subcritical regime since in the supercritical regime (that $\delta > 0$) the prefactor and hence the probability π_s will not be a real non-negative value. In

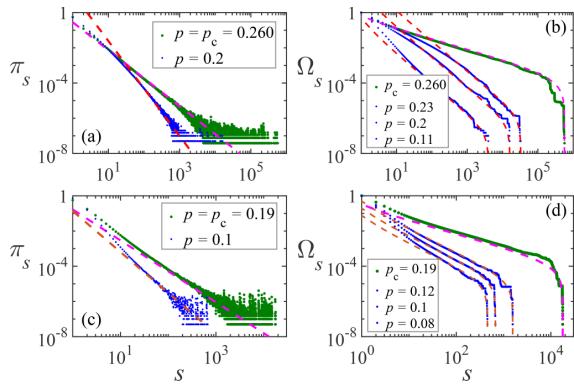


FIG. 1. The probability π_s and its cumulative distribution Ω_s for a UDN with $\lambda_o = 2.7$ and $\lambda_i = 3.7$ (top panels) and a CDN with $\lambda_o = 3.1$ and $\lambda_i = 3.7$ (bottom panels) both with 2×10^7 nodes. The green or blue dots are the numerical results and the dashed lines correspond to the theoretical power law π_s [Eq. (9) for $p < p_c$] with the same maximum s as that of the numerical data. The nearly vertical cut-offs in Ω_s are caused by the cumulative sum on data with finite maximum s and not by the shape of π_s . In each panel, the numerical results for lower p values are those located at lower positions. The networks are constructed using our extension of the configuration model [48].

the next section, we show that a set of equations other than Eqs. (5) and (6) and a modified method should be used to obtain valid results for the supercritical regime of $2 < \tilde{\lambda} < 3$.

2. The supercritical regime of $2 < \tilde{\lambda} < 3$

We first consider the governing Eqs. (3) and (4). Then as we know that in the supercritical phase $H_1(1) < 1$, for z values close to 1 (or $z \doteq 1 - w$ with $w \ll 1$) we can write

$$H_1(z) = \tilde{h} + \epsilon = 1 - \eta + \epsilon, \quad (10)$$

where $\tilde{h}, \eta < 1$ and $\epsilon \ll 1$. Thus, in the right-hand side of Eqs. (3) and (4),

$$1 - p + pH_1(z) = 1 - p\eta + p\epsilon \quad (11)$$

$$\doteq \bar{\eta} + p\epsilon, \quad (12)$$

where $\bar{\eta} = 1 - p\eta = 1 - p(1 - \tilde{h}) = 1 - p + p\tilde{h}$. Now, we can write

$$H_1(z) = zG_1(\bar{\eta} + p\epsilon) \quad (13)$$

$$= z[G_1(\bar{\eta}) + G'_1(\bar{\eta})p\epsilon + \frac{1}{2}G''_1(\bar{\eta})p^2\epsilon^2 + \dots]. \quad (14)$$

Therefore, according to Eqs. (10) and (14),

$$\tilde{h} + \epsilon = z[a_0 + a_1\epsilon + a_2\epsilon^2 + \dots], \quad (15)$$

where $a_0 = G_1(\bar{\eta}) = \tilde{h}$ [see Eqs. (3) and (10)], $a_1 = G'_1(\bar{\eta})p$, and $a_2 = \frac{1}{2}G''_1(\bar{\eta})p^2$. Equation (15) gives

$$\epsilon \sim \frac{1 - a_1 z}{2a_2 z} \pm \frac{1}{2a_2 z} \sqrt{(1 - a_1 z)^2 + 4a_0 a_2 z(1 - z)}. \quad (16)$$

Now we consider that the supercritical properties of $H_1(z)$ can be well approximated using the behavior of ϵ near the branch point $z = 1$; around this point we have

$$\epsilon \sim \frac{1 - a_1}{2a_2} \pm \frac{1 - a_1}{2a_2} \sqrt{1 + s^*(1 - z)}, \quad (17)$$

where $s^* = \frac{4a_0 a_2}{(1 - a_1)^2}$. Now,

$$H_0(z) = zG_0(\bar{\eta} + p\epsilon) \quad (18)$$

$$\sim (1 - w)[G_0(\bar{\eta}) + G'_0(\bar{\eta})p\epsilon + o(\epsilon^2)] \quad (19)$$

$$\sim \text{analytical terms} \pm b\sqrt{1 + s^*(1 - z)}, \quad (20)$$

where $b = \frac{pG'_0(\bar{\eta})(1 - a_1)}{2a_2}$. Then, according to the asymptotic properties of generating functions [48,54,55],

$$\pi_s \sim \pm \frac{bs^{*1/2}}{2\sqrt{\pi}} s^{-3/2} e^{-s/s^*} \quad \text{as } s \rightarrow \infty \quad (21)$$

$$\sim \pm \frac{G'_0(\bar{\eta})\sqrt{G_1(\bar{\eta})}}{\sqrt{2\pi} G'_1(\bar{\eta})} s^{-3/2} e^{-s/s^*} \quad \text{as } s \rightarrow \infty, \quad (22)$$

where $s^* \doteq \frac{2p^2 G_1(\bar{\eta}) G'_1(\bar{\eta})}{(1 - p G'_1(\bar{\eta}))^2}$ and $\bar{\eta}$ is calculated using Eqs. (3) and (12) according to the prescription described in Sec. S3.2.d of [48]. Figures 2(e) and 2(f) show that Eq. (22) performs well in describing the distribution of avalanches with finite (nonextensive) sizes in the supercritical regime of $2 < \tilde{\lambda} < 3$. It is worth noting that, in the supercritical regime, extremely large avalanches do also exist; such avalanches have a size that scales linearly with the network size. Hence, in the thermodynamic limit where the network size $N \rightarrow \infty$, their

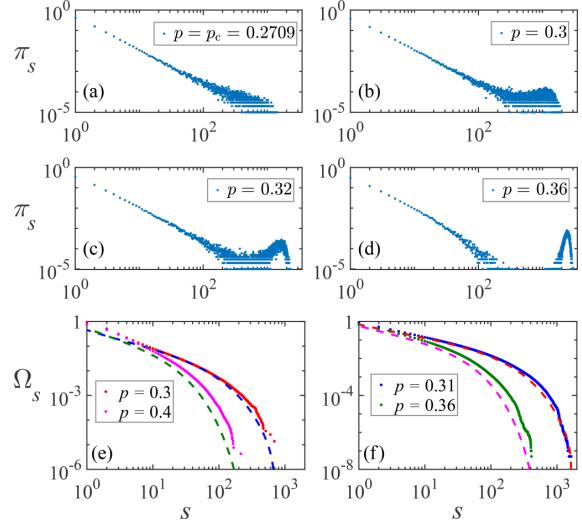


FIG. 2. (a)–(d) The full cluster size distribution at the critical and supercritical regimes of our neuronal dynamics model for a UDN with 5000 nodes and the other parameters identical to those of Fig. 1(a). At the supercritical phase, a bump appears at the tail of the distribution which corresponds to the percolating clusters whose sizes diverge at the thermodynamic limit (i.e., as $N \rightarrow \infty$). (e), (f) The distribution Ω_s (the cumulative distribution of π_s) of the finite clusters (i.e., excluding the bump at the tail) for the supercritical regime of (e) a UN with $\lambda = 3.3$, $p_c = 0.23$, and 5×10^7 nodes and (f) the UDN of Fig. 1(a). The dots represent numerical simulations and the dashed lines are the theoretical results. Closer to the critical point a better agreement between theory and numerics is observed. In panels (e) and (f), the results for the lower p value are those located at a lower position.

size also diverges. In finite networks, the effect of such avalanches on the distribution of avalanche sizes can be observed as a bump (in DNs) or a single point (in UNs) in the tail of the distribution. As the system moves further from the critical point this bump (or point in UNs) separates and moves away from the rest of the distribution [Figs. 2(a)–2(d)].

An interesting outcome of Eq. (22) is that the dependence of s^* (the exponential decay parameter) on the inverse of δ is no longer purely quadratic; nonetheless, s^* is still determined by p and skewness of degree distributions through a function that depends on p and the properties of G_1 at $\bar{\eta}$ [61]. It is worth mentioning that, for analyzing supercritical avalanches, methods based on Ref. [53] are also possible; nonetheless, such methods produce rather poor results [48].

As we mentioned earlier, a prominent implication of our results for the noncritical cases is that even non-scale-free networks can produce avalanches distributed according to a power law for several orders of magnitude. This has significant implications for the experiments of neuronal dynamics that are commonly performed on small size samples [62,63]; this is because in such cases pure power laws and the power-law part of noncritical systems may be indistinguishable (see Fig. 3).

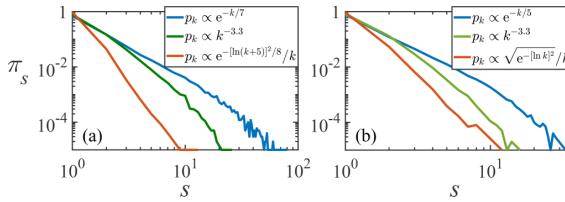


FIG. 3. The distribution π_s for the subcritical regime of (a) CDNs and (b) UNs with 5000 nodes for 10^5 avalanches in each network. The curves, from top to bottom, correspond to a CDN (UN) with, respectively, exponential, power-law, and log-normal outdegree (degree) distribution. For each of these networks an apparent power-law tail is observed for π_s . (Note that power laws observed in experimental setups have similar ranges of s [1,8].) The simulations are performed far from the critical point at occupation probabilities in the range $(0, p_c/2)$. In non-scale-free CDNs, the correlations are implemented by setting $j = k^{0.7}$.

III. CONCLUSIONS

In summary, we have provided analytical proofs, accompanied by numerical confirmations, that even in a simplified description of neuronal dynamics, because of structural heterogeneity, different types of critical and noncritical power-law avalanches with different exponents can be observed. This finding may help us to better explain the emergence

of power-law neuronal avalanches with exponents different from $3/2$ observed in experiments [64,65] and in realistic computational simulations [19]. Moreover, critical systems are known to have crucial advantages such as optimum information transmission, capacity, and stability [1,13,15–17] because of their power-law avalanches. Thus the existence of power laws in other dynamical regimes implies that some noncritical systems may benefit from similar advantages due to divergence of the mean values and scale invariance of their power-law distributions [13]. Furthermore, the emergence of such noncritical power laws introduces new challenges for accurate detection of criticality in experimental setups due to the finite size of the commonly used samples. In addition to the importance of these findings in the context of neuronal systems, the insights on percolation properties of networks that this Rapid Communication provides may find applications in topics such as network robustness [56,66], epidemic spreading [50,67,68], and stability of biological systems [69].

ACKNOWLEDGMENTS

The authors thank J. M. Beggs for helpful comments on the manuscript. A.F. and J.P.G. acknowledge support from the Science Foundation Ireland (Grants No. 16/IA/4470 and No. 16/RC/3918). F.R. acknowledges support from the National Science Foundation (CMMI-1552487) and the U.S. Army Research Office (W911NF-16-1-0104).

- [1] J. M. Beggs and D. Plenz, Neuronal avalanches in neocortical circuits, *J. Neurosci.* **23**, 11167 (2003).
- [2] D. Fraiman, P. Balenzuela, J. Foss, and D. R. Chialvo, Ising-like dynamics in large-scale functional brain networks, *Phys. Rev. E* **79**, 061922 (2009).
- [3] C. Adami, Self-organized criticality in living systems, *Phys. Lett. A* **203**, 29 (1995).
- [4] N. Friedman, S. Ito, B. A. W. Brinkman, M. Shimono, R. E. Lee DeVille, K. A. Dahmen, J. M. Beggs, and T. C. Butler, Universal Critical Dynamics in High Resolution Neuronal Avalanche Data, *Phys. Rev. Lett.* **108**, 208102 (2012).
- [5] A. Clauset, C. Shalizi, and M. Newman, Power-law distributions in empirical data, *SIAM Rev.* **51**, 661 (2009).
- [6] M. A. Muñoz, *Colloquium: Criticality and dynamical scaling in living systems*, *Rev. Mod. Phys.* **90**, 031001 (2018).
- [7] D. R. Chialvo, Emergent complex neural dynamics, *Nat. Phys.* **6**, 744 (2010).
- [8] J. M. Beggs and N. Timme, Being critical of criticality in the brain, *Front. Physiol.* **3**, 163 (2012).
- [9] D. Plenz and E. Niebur, *Criticality in Neural Systems* (Wiley-VCH, Weinheim, Germany, 2014).
- [10] C. G. Langton, Computation at the edge of chaos: Phase transitions and emergent computation, *Phys. D (Amsterdam, Neth.)* **42**, 12 (1990).
- [11] P. Bak, *How Nature Works: The Science of Self-Organized Criticality* (Springer & Business Media, New York, 1996).
- [12] D. R. Chialvo, Critical brain networks, *Phys. A (Amsterdam, Neth.)* **340**, 756 (2004).
- [13] J. M. Beggs, The criticality hypothesis: How local cortical networks might optimize information processing, *Philos. Trans. R. Soc. A* **366**, 329 (2008).
- [14] O. Kinouchi and M. Copelli, Optimal dynamical range of excitable networks at criticality, *Nat. Phys.* **2**, 348 (2006).
- [15] W. L. Shew, H. Yang, S. Yu, R. Roy, and D. Plenz, Information capacity and transmission are maximized in balanced cortical networks with neuronal avalanches, *J. Neurosci.* **31**, 55 (2011).
- [16] N. Bertschinger and T. Natschläger, Real-time computation at the edge of chaos in recurrent neural networks, *Neural Comput.* **16**, 1413 (2004).
- [17] C. Haldeman and J. M. Beggs, Critical Branching Captures Activity in Living Neural Networks and Maximizes the Number of Metastable States, *Phys. Rev. Lett.* **94**, 058101 (2005).
- [18] T. Tanaka, T. Kaneko, and T. Aoyagi, Recurrent infomax generates cell assemblies, neuronal avalanches, and simple cell-like selectivity, *Neural Comput.* **21**, 1038 (2009).
- [19] M. Rubinov, O. Sporns, J.-P. Thivierge, and M. Breakspear, Neurobiologically realistic determinants of self-organized criticality in networks of spiking neurons, *PLoS Comput. Biol.* **7**, e1002038 (2011).
- [20] J. P. Sethna, K. A. Dahmen, and C. R. Myers, Crackling noise, *Nature (London)* **410**, 242 (2001).
- [21] J. P. Gleeson and R. Durrett, Temporal profiles of avalanches on networks, *Nat. Commun.* **8**, 1227 (2017).
- [22] E. J. Friedman and A. S. Landsberg, Hierarchical networks, power laws, and neuronal avalanches, *Chaos* **23**, 013135 (2013).

- [23] M. E. J. Newman, Power laws, Pareto distributions, and Zipf's law, *Contemp. Phys.* **46**, 323 (2005).
- [24] M. P. H. Stumpf and M. A. Porter, Critical truths about power laws, *Science* **335**, 665 (2012).
- [25] M. Martinello, J. Hidalgo, A. Maritan, S. di Santo, D. Plenz, and M. A. Muñoz, Neutral Theory and Scale-Free Neural Dynamics, *Phys. Rev. X* **7**, 041071 (2017).
- [26] M. Benayoun, J. D. Cowan, W. van Drongelen, and E. Wallace, Avalanches in a stochastic model of spiking neurons, *PLOS Comput. Biol.* **6**, e1000846 (2010).
- [27] J. Touboul and A. Destexhe, Power-law statistics and universal scaling in the absence of criticality, *Phys. Rev. E* **95**, 012413 (2017).
- [28] J. Touboul and A. Destexhe, Can power-law scaling and neuronal avalanches arise from stochastic dynamics? *PLoS One* **5**, e8982 (2010).
- [29] V. Priesemann and O. Shriki, Can a time varying external drive give rise to apparent criticality in neural systems? *PLoS Comput. Biol.* **14**, e1006081 (2018).
- [30] M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Oxford, 2010).
- [31] K. B. Athreya and P. E. Ney, *Branching Processes* (Springer-Verlag, Berlin, 1972).
- [32] K.-I. Goh, D.-S. Lee, B. Kahng, and D. Kim, Sandpile on Scale-Free Networks, *Phys. Rev. Lett.* **91**, 148701 (2003).
- [33] J. P. Gleeson, J. A. Ward, K. P. O'Sullivan, and W. T. Lee, Competition-Induced Criticality in a Model of Meme Popularity, *Phys. Rev. Lett.* **112**, 048701 (2014).
- [34] A.-L. Barabási, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek, Evolution of the social network of scientific collaborations, *Phys. A (Amsterdam, Neth.)* **311**, 590 (2002).
- [35] M. Shimono and J. M. Beggs, Functional clusters, hubs, and communities in the cortical microconnectome, *Cereb. Cortex* **25**, 3743 (2014).
- [36] A.-L. Barabási, Scale-free networks: A decade and beyond, *Science* **325**, 412 (2009).
- [37] A. D. Broido and A. Clauset, Scale-free networks are rare, *Nat. Commun.* **10**, 1017 (2019).
- [38] J. Goldenberg, B. Libai, and E. Muller, Talk of the network: A complex systems look at the underlying process of word-of-mouth, *Marketing Lett.* **12**, 211 (2001).
- [39] J. Goldenberg, B. Libai, and E. Muller, Using complex systems analysis to advance marketing theory development: Modeling heterogeneity effects on new product growth through stochastic cellular automata, *Acad. Marketing Sci. Rev.* **9**, 1 (2001).
- [40] D. Kempe, J. Kleinberg, and É. Tardos, Maximizing the spread of influence through a social network, in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, New York, 2003), pp. 137–146.
- [41] A. N. Burkitt, A review of the integrate-and-fire neuron model: I. Homogeneous synaptic input, *Biol. Cybern.* **95**, 1 (2006).
- [42] A. N. Burkitt, A review of the integrate-and-fire neuron model: II. Inhomogeneous synaptic input and network properties, *Biol. Cybern.* **95**, 97 (2006).
- [43] A. Levina, J. M. Herrmann, and T. Geisel, Dynamical synapses causing self-organized criticality in neural networks, *Nat. Phys.* **3**, 857 (2007).
- [44] A. Faqeeh, Percolation and its relations to other processes in networks, Ph.D. thesis, University of Limerick, 2016.
- [45] E. A. Bender and E. Canfield, The asymptotic number of labeled graphs with given degree sequences, *J. Comb. Theory, Ser. A* **24**, 296 (1978).
- [46] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, Complex networks: Structure and dynamics, *Phys. Rep.* **424**, 175 (2006).
- [47] M. Molloy and B. Reed, A critical point for random graphs with a given degree sequence, *Random Struct. Algor.* **6**, 161 (1995).
- [48] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevE.100.010401> for more details of calculations and/or further relevant results.
- [49] 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).
- [50] M. E. J. Newman, Spread of epidemic disease on networks, *Phys. Rev. E* **66**, 016128 (2002).
- [51] R. Cohen, D. ben-Avraham, and S. Havlin, Percolation critical exponents in scale-free networks, *Phys. Rev. E* **66**, 036113 (2002).
- [52] N. Schwartz, R. Cohen, D. ben-Avraham, A.-L. Barabási, and S. Havlin, Percolation in directed scale-free networks, *Phys. Rev. E* **66**, 015104(R) (2002).
- [53] D. Lee, J. Kim, B. Kahng, and D. Kim, Scale-free random branching trees in supercritical phase, *J. Phys. A: Math. Theor.* **40**, 7139 (2007).
- [54] H. S. Wilf, *Generating Functionology* (A K Peters, Ltd., Wellesley, MA, 2006).
- [55] G. Weiss, *Aspects and Applications of the Random Walk, Random Materials and Processes* (North-Holland, New York, 2005).
- [56] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Network Robustness and Fragility: Percolation on Random Graphs, *Phys. Rev. Lett.* **85**, 5468 (2000).
- [57] M. E. J. Newman, Component sizes in networks with arbitrary degree distributions, *Phys. Rev. E* **76**, 045101(R) (2007).
- [58] In fact the property $\tilde{G}(z) = G_1(1 - p + p + pz)$ (that holds for site percolation too) is used in Refs. [51,52] only to calculate the location of the critical point of site percolation and not to construct the governing equations.
- [59] Note that obtaining the prefactors is important as the full functional form is needed to reliably compare the analytical predictions with the simulations results; moreover, we will later show that in some cases the prefactors can indicate the range of validity of the theoretical predictions.
- [60] P. D. Miller, *Applied Asymptotic Analysis*, Graduate studies in mathematics (American Mathematical Society, Providence, RI, 2006).
- [61] Note that in other cases s^* depended on the properties of G_1 at 1 which we could describe them in terms of averages of the different powers of k (and also j in DNs).
- [62] G. W. Gross, B. K. Rhoades, H. M. Azzazy, and M.-C. Wu, The use of neuronal networks on multielectrode arrays as biosensors, *Biosens. Bioelectron.* **10**, 553 (1995).
- [63] J. Müller, M. Ballini, P. Livi, Y. Chen, M. Radivojevic, A. Shadmani, V. Viswam, I. L. Jones, M. Fiscella, R. Diggelmann, A. Stettler, U. Frey, D. J. Bakkum, and A. Hierlemann, High-resolution CMOS MEA platform to study neurons at subcellular, cellular, and network levels, *Lab Chip* **15**, 2767 (2015).
- [64] T. L. Ribeiro, M. Copelli, F. Caixeta, H. Belchior, D. R. Chialvo, M. A. L. Nicolelis, and S. Ribeiro, Spike avalanches

- exhibit universal dynamics across the sleep-wake cycle, *PLoS One* **5**, e14129 (2010).
- [65] E. Tagliazucchi, P. Balenzuela, D. Fraiman, and D. Chialvo, Criticality in large-scale brain fMRI dynamics unveiled by a novel point process analysis, *Front. Physiol.* **3**, 15 (2012).
- [66] A. Fafeeh, S. Melnik, P. Colomer-de-Simón, and J. P. Gleeson, Emergence of coexisting percolating clusters in networks, *Phys. Rev. E* **93**, 062308 (2016).
- [67] A. Fafeeh, S. Melnik, and J. P. Gleeson, Network cloning unfolds the effect of clustering on dynamical processes, *Phys. Rev. E* **91**, 052807 (2015).
- [68] M. A. Muñoz, R. Juhász, C. Castellano, and G. Ódor, Griffiths Phases on Complex Networks, *Phys. Rev. Lett.* **105**, 128701 (2010).
- [69] S. Squires, E. Ott, and M. Girvan, Dynamical Instability in Boolean Networks as a Percolation Problem, *Phys. Rev. Lett.* **109**, 085701 (2012).

Supplemental Materials

Emergence of power-laws in noncritical neuronal systems

Ali Faqeeh¹, Saeed Osat, Filippo Radicchi, James P. Gleeson

Contents

S 1 Degree distributions and Network construction	2
S 1.1 Initial assignment of the degrees in directed networks	2
S 1.2 Balancing the total number of in- and out-stubs	3
S 1.3 The joint distributions	4
S 1.4 The generating functions for degree-based distributions	4
S 2 Asymptotics for the degree-based generating functions	5
S 2.1 Definitions	5
S 2.2 Examples of calculations of the asymptotic forms	6
S 2.2.a Case 1: UDNs and $X_{\text{init}} < 0$	6
S 2.2.b Case 2: CDNs and $X_{\text{init}} > 0$	6
S 3 Governing equations and the use of asymptotic expansions	7
S 3.1 Non-scale-free and scale-free with $\tilde{\lambda} > 3$ networks	8
S 3.2 Scale-free networks with $2 < \tilde{\lambda} < 3$	9
S 3.2.a At the critical point ($\delta = 0$)	9
S 3.2.b Subcritical regime ($\delta < 0$)	9
S 3.2.c Supercritical regime ($\delta > 0$); method 1	10
S 3.2.d Calculation of \tilde{h} and $\bar{\eta}$	13
S 3.3 A note on the effect of percolation on governing equations	13
S 4 Summary of the results for π_S and its parameters	14
S 5 More on the sample size effects	14
S 6 $(a - z)^\beta$ non-analyticity (as in Eqs. (S31a) and (S108))	15
S 7 $\sqrt{1 + s^*(\tilde{z} - z)}$ non-analyticity (as in Eqs. (S31b), (S52) and (S100))	16
S 8 Simpler equations to numerically calculate $G_{0/1}$ at a specific point h	18
S 9 Other methods for the supercritical regime	19
Method 2	19
Method 3	21

¹ali.faqeeh@ul.ie

S 1 Degree distributions and Network construction

In undirected networks (UNs), for a given degree distribution p_k we use configuration model to generate our benchmark networks. In directed networks (DNs), we follow the recipe described in Secs. S 2.1 and S 1.1. Note that we use k to denote the degree of nodes in UNs and also the out-degree of nodes in DNs; this use of notations facilitates drawing an analogy between the equations obtained for the undirected and for the directed networks.

S 1.1 Initial assignment of the degrees in directed networks

In DNs, two types of degree-degree correlations are possible: the input-output (IO) correlation and the source-target (ST) correlation. IO is the correlation between the indegree and outdegree values of each of the nodes; ST, instead, is the correlation between the outdegree of a source (i.e., the node from which a link started) and the indegree of a target (i.e., the ending point (at the arrow side) of that link). It is worth noting that undirected networks can be assumed to have full IO correlation as the degree of a node is a measure for both the number of possible inputs and the number of possible outputs. On the other hand, similar to the directed case, in undirected networks we may or may not observe ST correlation and, in fact, in undirected networks people only measure ST correlation (as the IO correlation is not a varying factor).

We consider networks with negligible ST correlation (we see in Sec. S 1.3 why this consideration is important for our calculations). For synthesizing our undirected scale-free networks, we generate networks with no assortativity [1] (a measure of the ST correlation) by imposing a cutoff on the tail of the degree distribution [2], i.e., we consider a degree distribution $p_k \propto k^{-\lambda}$, for $k_{\text{start}} \leq k \leq k_{\text{max}}$, and $p_k = 0$ everywhere else. It is also known [3, 4] that random directed scale-free networks can be disassortative (i.e., they may have negative ST correlation); thus, we imposed cutoffs on the in-degree and out-degree distribution and checked that the generated networks do not have an effective correlation between pairs of connected nodes, such that the probability that a node with out-degree k is connected to a node with in-degree j does not depend on k .

Nevertheless, for DNs, we include in our study networks with correlations between the in-degree and out-degree of each node (the IO correlation). In terms of this type of correlation, we consider uncorrelated and fully correlated directed networks and denote the out-degree and in-degree of a node with, respectively, k and j . In case of scale-free networks, we first draw the out-degree sequence from a power-law distribution

$$P_k^{(\text{tail})} = c_o k^{-\lambda_o} \quad \text{for } k_{\text{start}} \leq k \leq k_{\text{max}}. \quad (\text{S1})$$

The upper cutoff, that is the largest possible out-degree k_{max} , is set to $N^{\frac{1}{\lambda_o - 1}}$. For uncorrelated directed networks (UDNs), we then draw the sequence of in-degrees from the distribution

$$P_j^{(\text{tail})} = c_i j^{-\lambda_i} \quad \text{for } j_{\text{start}} \leq j \leq j_{\text{max}}, \quad (\text{S2})$$

where j_{max} is set to $N^{\frac{1}{\lambda_i - 1}}$. For fully correlated directed networks (CDNs) with power-law degree distribution, in order to preserve the power-law shapes of the input degree sequences, the in-degree of each node is set to a value determined by the relation [5]

$$j = J(k) = k^{\frac{\lambda_o - 1}{\lambda_i - 1}}, \quad (\text{S3})$$

thus,

$$P_j^{(\text{tail})} = c_0 k^{-\lambda_0} \delta(k, J^{-1}(j)) = c_i j^{-\lambda_i} \delta(j, J(k)). \quad (\text{S4})$$

In non-scale-free networks that we do not need to impose a restriction on the type of the in-degree distribution we just set $J(k) = k^\alpha$, where α is a constant number. When generating the CDNs, to assign the appropriate value of j to nodes with out-degree k , we obtain $[J(k)]$, the integer part of $J(k)$; then with probability $J(k) - [J(k)]$, we assign the in-degree $[J(k)]$ to a node with out-degree k ; otherwise, we set its in-degree to $[J(k)] + 1$.

S 1.2 Balancing the total number of in- and out-stubs

For generating CDNs, after obtaining the degree sequences of the nodes, we want to connect them at random to generate the network. However, before that, we need to balance the total number of out-stubs and the total number of in-stubs so that no extra stub is left after connecting the nodes. Depending on the values of λ_0 , λ_i , k_{start} and j_{start} , the total number of out-stubs may be smaller or larger than the total number of in-stubs. We define $X_{\text{init}} = N \left(\sum_{k \geq k_{\text{start}}} k P_k^{(\text{tail})} - \sum_{j \geq j_{\text{start}}} j P_j^{(\text{tail})} \right)$, which is the initial number of out-stubs minus the initial number of in-stubs. Let's consider the case where the total number of out-stubs is larger than the total number of in-stubs, i.e., $X_{\text{init}} > 0$. In this case, we consider the out-degrees of the nodes. Then we perform the following steps

1. Select a $k_{\text{min}} \leq k_{\text{start}}$ (We considered $k_{\text{min}} = k_{\text{start}} - 1$).
2. Select a node at random and reduce its out-degree to k_{min} .
3. Calculate $X =$ the total number of out-stubs – the total number of in-stubs.
4. If $X > 0$, return to step 2. Otherwise, put back the extra stubs removed by adding them to the last node selected; then return the new sequence of out-stubs.

The resulting out-degree distribution is

$$P_k = \begin{cases} u & \text{for } k = k_{\text{min}} \\ (1-u)P_k^{(\text{tail})} & \text{for } k \geq k_{\text{start}}, \end{cases} \quad (\text{S5})$$

where u is the fraction of nodes with out-degree k_{min} . As the in-degree distribution is not altered $P_j = P_j^{(\text{tail})}$.

In the case where there are more in-stubs than out-stubs (i.e., when $X_{\text{init}} < 0$), we perform a similar set of steps and decrease the number of in-degrees of a set of randomly selected nodes to a predefined value j_{min} . This procedure results in a new distribution of in-degrees

$$P_j = \begin{cases} u & \text{for } j = j_{\text{min}} \\ (1-u)P_j^{(\text{tail})} & \text{for } j \geq j_{\text{start}}, \end{cases} \quad (\text{S6})$$

where, here, u is the fraction of nodes with in-degree j_{min} . The out-degree distribution remains to be $P_k = P_k^{(\text{tail})}$.

S 1.3 The joint distributions

The joint degree distribution for the UDNs is

$$P_{jk} = P_j P_k \quad \text{for UDNs} \quad (\text{S7})$$

For the CDNs, if $X_{\text{init}} > 0$, then the joint degree distribution is

$$P_{jk} = \begin{cases} u P_j^{(\text{tail})} & \text{for } k = k_{\min} \\ (1-u) P_k^{(\text{tail})} & \text{for } k \geq k_{\text{start}}. \end{cases} \quad (\text{S8})$$

On the other hand, if, $X_{\text{init}} < 0$,

$$P_{jk} = \begin{cases} u P_k^{(\text{tail})} & \text{for } j = j_{\min} \\ (1-u) P_k^{(\text{tail})} & \text{for } j \geq j_{\text{start}}. \end{cases} \quad (\text{S9})$$

As we consider networks with negligible ST correlation, the “excess” joint degree distribution q_{jk} is

$$q_{jk} = \frac{j}{\langle j \rangle} P_{jk}. \quad (\text{S10})$$

Similarly, in our UNs the ST correlation is negligible; hence, their excess degree distribution is simply $q_k = \frac{k}{\langle k \rangle} p_k$, where p_k is the degree distribution.

S 1.4 The generating functions for degree-based distributions

The generating functions (GFs) for degree and excess degree distributions in UNs and for joint degree distribution and joint excess degree distribution in DNs are defined as

$$G_0(h) = \begin{cases} \sum_{k=0}^{\infty} p_k h^k & \text{undirected network} \\ \sum_{j,k=0}^{\infty} P_{jk} h^k & \text{directed network,} \end{cases} \quad (\text{S11})$$

$$G_1(h) = \begin{cases} \sum_{k=0}^{\infty} q_k h^k & \text{undirected network} \\ \sum_{j,k=0}^{\infty} q_{jk} h^k & \text{directed network.} \end{cases} \quad (\text{S12})$$

In a bond percolation with occupation probability p , the generating functions will be transferred to $\widehat{G}_{0/1}(h) = G_{0/1}(1 - p + ph)$ (see Ref. [6] for the derivation of this result for undirected networks; it is straightforward to show that this result also holds for DNs). We will later see that we would need to obtain the behavior of $\widehat{G}_{0/1}(h)$ for h close to 1, i.e. we would need the behavior of $\widehat{G}_{0/1}(1 - \phi) = G_{0/1}(1 - p\phi)$ where $h \doteq 1 - \phi$ and $\phi \ll 1$.

S 2 Asymptotics for the degree-based generating functions

In our calculations we need the asymptotic forms of the degree dependent generating functions when they are expanded around 1 [5, 7, 8], i.e.,

$$G_1(1 - p\phi) \sim G_1(1) - G'_1(1)p\phi + G''_1(1) [p\phi]^2 + D\phi^{\tilde{\lambda}-1} + o(\phi^2 | \phi^{\tilde{\lambda}-1}), \quad (\text{S13})$$

$$G_0(1 - p\phi) \sim G_0(1) - G'_0(1)p\phi + M\phi^{\bar{\lambda}-1} + O(\phi^2). \quad (\text{S14})$$

We find that, for a bond percolation with occupation probability p , there exist a general form for the asymptotic expansion of each of the degree dependent generating functions:

$$G_1(1 - p\phi) = 1 - Ap\phi + B\phi^2 + D\phi^{\tilde{\lambda}-1} + \dots, \quad (\text{S15})$$

$$G_0(1 - p\phi) = 1 - E\phi + M\phi^{\bar{\lambda}-1} + \dots, \quad (\text{S16})$$

where the definitions of the parameters depend on the type of the network (note that the $\tilde{\lambda} - 1$ and $\bar{\lambda} - 1$ terms exist only in scale-free networks). We provide these definitions in Sec. S 2.1. In Sec. S 2.2, we provide details of the calculations for two different cases. A similar approach can be employed for any of the other cases.

S 2.1 Definitions

We denote by k either the degree in undirected networks (UNs) or the out-degree in DNs (and their two sub-classes UDNs and CDNs). Below we summarize the definitions of various parameters appeared in Eqs. (S15) and (S16); these parameters will be used in the rest of this supplemental material as well.

$$\begin{aligned} \bar{\lambda} &= \begin{cases} \lambda & (\text{UN}) \\ \lambda_o & (\text{UDN}) \\ \lambda_o & (\text{CDN}) \end{cases}, \quad \tilde{\lambda} = \begin{cases} \lambda - 1 & (\text{UN}) \\ \lambda_o & (\text{UDN}) \\ \lambda_o - \frac{\lambda_o - 1}{\lambda_i - 1} & (\text{CDN}) \end{cases}, \quad a^* \doteq \frac{1}{\sum_{k_{\text{start}}}^{k_{\text{max}}} k^{-\bar{\lambda}}}, \quad Q \doteq \frac{1}{\sum_{k_{\text{start}}}^{k_{\text{max}}} k^{-\tilde{\lambda}}} = \begin{cases} \frac{a^*}{\langle k \rangle} & (\text{UN}) \\ a^* & (\text{UDN}), \\ \frac{a^*}{\langle k \rangle} & (\text{CDN}) \end{cases} \\ A &= \frac{1}{p_c} = \begin{cases} \frac{\langle k(k-1) \rangle}{\langle k \rangle} & (\text{UN}) \\ \frac{\langle jk \rangle}{\langle k \rangle} & (\text{DN}) \end{cases}, \quad B = \begin{cases} \frac{1}{2} \frac{\langle k(k-1)(k-2) \rangle}{\langle k \rangle} p^2 & (\text{UN}) \\ \frac{1}{2} \frac{\langle jk(k-1) \rangle}{\langle k \rangle} p^2 & (\text{DN}) \end{cases}, \quad \bar{u} = \begin{cases} 0 & (\text{UN}) \\ 0 & (\text{UDN with } X_{\text{init}} < 0) \\ u & (\text{UDN with } X_{\text{init}} > 0) \\ u & (\text{CDN}) \end{cases}, \end{aligned}$$

$$D = (1 - \bar{u})Q\Gamma(1 - \tilde{\lambda})p^{\tilde{\lambda}-1}, \quad M = (1 - \bar{u})a^*\Gamma(1 - \bar{\lambda})p^{\bar{\lambda}-1} \quad \text{and} \quad E = \langle k \rangle p.$$

S 2.2 Examples of calculations of the asymptotic forms

S 2.2.a Case 1: UDNs and $X_{\text{init}} < 0$

$$G_0(H_1) = \sum_{j,k} P_j P_k H_1^k \equiv \sum_k P_{j_{\min},k} H_1^k + \sum_{(j \geq j_{\text{start}}), k} P_{j,k} H_1^k \quad (\text{S17})$$

$$= \sum_k P_{j_{\min}} P_k H_1^k + \sum_{(j \geq j_{\text{start}}), k} P_j P_k H_1^k$$

$$= \sum_k u P_k^{(\text{tail})} H_1^k + \sum_{(j \geq j_{\text{start}}), k} (1-u) P_j^{(\text{tail})} P_k^{(\text{tail})} H_1^k$$

$$= u \sum_k P_k^{(\text{tail})} H_1^k + (1-u) \sum_k P_k^{(\text{tail})} H_1^k$$

$$= \sum_k P_k^{(\text{tail})} H_1^k$$

$$= \sum_k P_k^{(\text{tail})} (1-p\phi)^k$$

$$= \sum_k c_o k^{-\lambda_o} (1-p\phi)^k \quad (\text{S18})$$

$$\sim 1 - \langle k \rangle p\phi + M\phi^{\lambda_o-1}, \quad (\text{S19})$$

where $M = c_o \Gamma(1 - \lambda_o) p^{\lambda_o-1}$, c_o and λ_o are defined in Eq. (S1), and in line 4 we used the fact that $\sum_{(j \geq j_{\text{start}})} P_j^{(\text{tail})} = 1$. To obtain (S19) from (S18) we used generating function techniques (see Sec. S 6 and Refs. [5, 7–9]). Now,

$$\begin{aligned} G_1(H_1) &= \sum_k \frac{1}{\langle j \rangle} \left[u j_{\min} + (1-u) \sum_{j \geq j_{\text{start}}} j P_j \right] P_k H_1^k \\ &= \sum_k \frac{1}{\langle j \rangle} [\langle j \rangle] P_k H_1^k \\ &= \sum_k P_k H_1^k = \sum_k P_k^{(\text{tail})} H_1^k \\ &\sim 1 - \frac{\langle jk \rangle}{\langle j \rangle} p\phi + \frac{\langle jk(k-1) \rangle}{2\langle j \rangle} p^2\phi^2 + D\phi^{\tilde{\lambda}-1}, \end{aligned} \quad (\text{S20})$$

where $D = c_o \Gamma(1 - \tilde{\lambda}) p^{\tilde{\lambda}-1}$ (see Sec. S 6 and Refs. [5, 7–9]).

S 2.2.b Case 2: CDNs and $X_{\text{init}} > 0$

$$\begin{aligned} G_0(1-p\phi) &= u(1-p\phi)^{k_{\min}} + (1-u) \sum_{j,k \geq k_{\text{start}}} P_k^{(\text{tail})} (1-p\phi)^k \\ &= u [1 - k_{\min} p\phi + \dots] + (1-u) \left[1 - \bar{k} p\phi + \frac{1}{2} \bar{k}(\bar{k}-1) p^2 \phi^2 + M\phi^{\lambda_o-1} + \dots \right] \\ &\sim 1 - \langle k \rangle p\phi + (1-u) M\phi^{\lambda_o-1}, \end{aligned} \quad (\text{S21})$$

where $\overline{\overline{\circlearrowright}}$ denotes averaging over $P_k^{(\text{tail})}$ and $M = c_o \Gamma(1 - \lambda_o) p^{\lambda_o - 1}$.

$$\begin{aligned}
G_1(H_1) &= u H_1^{k_{\min}} + (1-u) \sum_{j,k \geq k_{\text{start}}} \frac{j}{\langle j \rangle} P_k^{(\text{tail})} H_1^k \\
&= u (1-p\phi)^{k_{\min}} + (1-u) \sum_{k \geq k_{\text{start}}} \frac{c_o}{\langle j \rangle} k^{-\tilde{\lambda}} (1-p\phi)^k \\
&= u \left[1 - k_{\min} p\phi + \frac{k_{\min}(k_{\min}-1)}{2} p^2 \phi^2 + \dots \right] + (1-u) \left[1 - \frac{\overline{\overline{j}}}{\langle j \rangle} p\phi + \frac{1}{2} \frac{\overline{\overline{jk(k-1)}}}{\langle j \rangle} p^2 \phi^2 + \overline{D}\phi^{\tilde{\lambda}-1} + \dots \right] \\
&\sim 1 - \frac{\langle jk \rangle}{\langle j \rangle} p\phi + \frac{\langle jk(k-1) \rangle}{2\langle j \rangle} p^2 \phi^2 + (1-u) D\phi^{\tilde{\lambda}-1},
\end{aligned} \tag{S22}$$

where $\tilde{\lambda} = \lambda_o - \frac{\lambda_o - 1}{\lambda_i - 1}$ and $D = \frac{c_o}{\langle j \rangle} \Gamma(1 - \tilde{\lambda}) p^{\tilde{\lambda}-1}$.

S 3 Governing equations and the use of asymptotic expansions

Let's denote by π_s the probability that an avalanche has a size s (or equivalently the probability that the size of the avalanche of a (randomly selected) seed is s . For our neural dynamics model (described in the main text), the size of avalanche of a node in a directed network is equivalent to the size of its outcomponent in bond percolation process with occupation probability p . On an undirected network, π_s is the probability that, in the bond percolation process, a randomly selected node is in a component with size s . This equivalence enables us to employ the findings for percolation properties of networks. In that regard, we consider the generating function $H_0(z)$ corresponding to π_s . Also we consider an initial seed (root) of an avalanche and the (sub-)avalanches created at each of its neighbors. Accordingly, we denote by ρ_s the probability that one of this sub-avalanche has a size s and by $H_1(z)$ its corresponding generating function. It has been shown that in a network with no percolation (i.e., with $p = 1$) we can write $H_1(z) = z G_1(H_1(z))$ and $H_0(z) = z G_0(H_1(z))$ [6, 10–12]. Also it has been proved that for a percolation process with probability p an undirected network, $G_0(z)$ is transformed to $G_0(1 - p + pz)$ and $G_1(z)$ is transferred to $G_1(1 - p + pz)$ [6]. It is straightforward to show that the same relations hold for percolation on directed networks [9]. The above findings conclude that in a bond percolation process the generating functions $H_0(z)$ and $H_1(z)$ satisfy the pair of equations below:

$$H_1(z) = z \widehat{G}_1(H_1(z)), \tag{S23}$$

$$H_0(z) = z \widehat{G}_0(H_1(z)). \tag{S24}$$

For $z \sim 1 - w$ and $w \ll 1$, $H_1 \sim 1 - \phi$ with $\phi \ll 1$, Eqs. (S15) and (S23) together yield:

$$\begin{aligned}
H_1(1-w) &= 1 - \phi \\
&\sim (1-w)(1 - Ap\phi + B\phi^2 + D\phi^{\tilde{\lambda}-1})
\end{aligned}$$

$$\Rightarrow w \sim -A\delta\phi + B\phi^2 + D\phi^{\tilde{\lambda}-1}. \quad (\text{S25})$$

On the other hand, combining Eq. (S16) with Eq. (S24) gives:

$$H_0(1-w) \sim 1 - E\phi + M\phi^{\bar{\lambda}-1}. \quad (\text{S26})$$

Using Eq. (S25), we obtain the leading order behavior of ϕ . Then we substitute the results in Eq. (S26) to obtain the leading order nonanalytic behavior of H_0 . The result is then used to calculate the probability π_s of having an avalanche with size s (see Secs. S 3.1 and S 3.2).

S 3.1 Non-scale-free and scale-free with $\tilde{\lambda} > 3$ networks

In non-scale-free networks the $\tilde{\lambda} - 1$ term in Eq. (S25) and the $\bar{\lambda} - 1$ term in Eq. (S26) do not exist; hence their results are identical to the results of scale-free networks with $\tilde{\lambda} > 3$ and $\bar{\lambda} > 2$, or equivalently $\tilde{\lambda} > 3$, as $\tilde{\lambda} \leq \bar{\lambda}$. In such (non-scale-free or $\tilde{\lambda} > 3$) networks, Eq. (S25) can be written as

$$w \sim -A\delta\phi + B\phi^2. \quad (\text{S27})$$

Thus at the critical point ($\delta = 0$)

$$\phi \sim \left(\frac{w}{B}\right)^{1/2}, \quad (\text{S28})$$

and in the noncritical regimes

$$\phi \sim \frac{A\delta}{2B} \pm \frac{1}{2B} \sqrt{(A\delta)^2 + 4Bw} \quad \text{as } w \rightarrow 0 \quad (\text{S29})$$

$$\sim \frac{A\delta}{2B} \pm \frac{A\delta}{2B} \sqrt{1 + s^*w}, \quad (\text{S30})$$

where $s^* \doteq \frac{4B}{(A\delta)^2}$. Then,

$$H_0(1-w) \sim \text{analytic part} + \frac{-E}{B^{1/2}} w^{1/2} \quad (\text{for } \delta = 0), \quad (\text{S31a})$$

$$H_0(1-w) \sim \text{analytic part} + \pm \frac{AE\delta}{2B} \sqrt{1 + s^*w} \quad (\text{for } \delta \neq 0), \quad (\text{S31b})$$

as $w \rightarrow 0$, and using generating functions techniques (see Secs. S 6 and S 7)

$$\pi_s \sim \frac{-E}{B^{1/2} \Gamma(-1/2)} s^{-3/2} \quad (\delta = 0), \quad (\text{S32a})$$

$$\pi_s \sim \pm \frac{E}{\sqrt{4\pi B}} s^{-3/2} e^{-s/s^*} \quad (\delta \neq 0), \quad (\text{S32b})$$

as $s \rightarrow \infty$ (see the simplified definition of parameters in Sec. S 4).

S 3.2 Scale-free networks with $2 < \tilde{\lambda} < 3$

In this case, according to Eq. (S25), we have

$$w + A\delta\phi - D\phi^{\tilde{\lambda}-1} \sim 0. \quad (\text{S33})$$

The solutions of this equation for different cases are derived in the following.

S 3.2.a At the critical point ($\delta = 0$)

In this case, equation (S33) gives

$$\begin{aligned} w &\sim D\phi^{\tilde{\lambda}-1} \\ \Rightarrow \phi &\sim \left(\frac{w}{D}\right)^{\frac{1}{\tilde{\lambda}-1}} \\ \Rightarrow H_0 &\sim -E \left(\frac{w}{D}\right)^{\frac{1}{\tilde{\lambda}-1}} \end{aligned} \quad (\text{S34})$$

$$\Rightarrow \pi_s \sim \frac{-E}{\Gamma\left(\frac{-1}{\tilde{\lambda}-1}\right)} \left(\frac{1}{D}\right)^{\frac{1}{\tilde{\lambda}-1}} s^{-\left(\frac{1}{\tilde{\lambda}-1}+1\right)}. \quad (\text{S35})$$

For UDNs, the prefactor equals $-\langle k \rangle / \left[\left(1 - \bar{u}\right) c_o \Gamma(1 - \tilde{\lambda}) \right]^{1/(\tilde{\lambda}-1)} \Gamma(1/(1 - \tilde{\lambda}))$ (see Sec. S 2.1); for CDNs and UNs, the prefactor is obtained by multiplying this answer by $\langle k \rangle^{1/(\tilde{\lambda}-1)}$. The results of Eq. (S35) matches very well the numerical results for UDNs (the results for $p = p_c$ in the top panels of Fig. S1); for CDNs the theoretical prediction for $p = p_c$ slightly underestimates the numerics (Fig. S1 bottom panels).

S 3.2.b Subcritical regime ($\delta < 0$)

To find the leading order behavior of (S33) which states

$$w + A\delta\phi - D\phi^{\tilde{\lambda}-1} \sim 0,$$

we try $\phi \sim b_1 w^{\beta_1} + b_2 w^{\beta_2} + \dots$ and using the dominant balance method [13] we get $\beta_1 = 1$, $b_1 = \frac{-1}{A\delta}$, $\beta_2 = \tilde{\lambda} - 1$ and $b_2 = \frac{Db_1^{\tilde{\lambda}-1}}{A\delta} = \frac{D}{A\delta} \left(\frac{-1}{A\delta}\right)^{\tilde{\lambda}-1}$. Thus,

$$H_0 \sim \frac{-ED}{A\delta} b_1^{\tilde{\lambda}-1} w^{\tilde{\lambda}-1} + M b_1^{\tilde{\lambda}-1} w^{\tilde{\lambda}-1} \quad (\text{S36})$$

$$\Rightarrow \pi_s \sim \frac{-ED b_1^{\tilde{\lambda}-1}}{A\delta \Gamma(1 - \tilde{\lambda})} s^{-\tilde{\lambda}} + \frac{M b_1^{\tilde{\lambda}-1}}{\Gamma(1 - \tilde{\lambda})} s^{-\tilde{\lambda}}. \quad (\text{S37})$$

Using the definitions of Sec. S 2.1 we get

$$\pi_s \sim \begin{cases} a^*(1 - \bar{u}) \left(1 + \frac{pp_c(k)}{-\delta}\right) \left(\frac{pp_c}{-\delta}\right)^{\lambda_o-1} s^{-\lambda_o} & (\text{UDN}) \\ a^*(1 - \bar{u}) \left[\left(\frac{pp_c}{-\delta}\right)^{\tilde{\lambda}} s^{-\tilde{\lambda}} + \left(\frac{pp_c}{-\delta}\right)^{\tilde{\lambda}-1} s^{-\tilde{\lambda}}\right] & (\text{CDN and UN}) \end{cases} \quad (\text{S38})$$

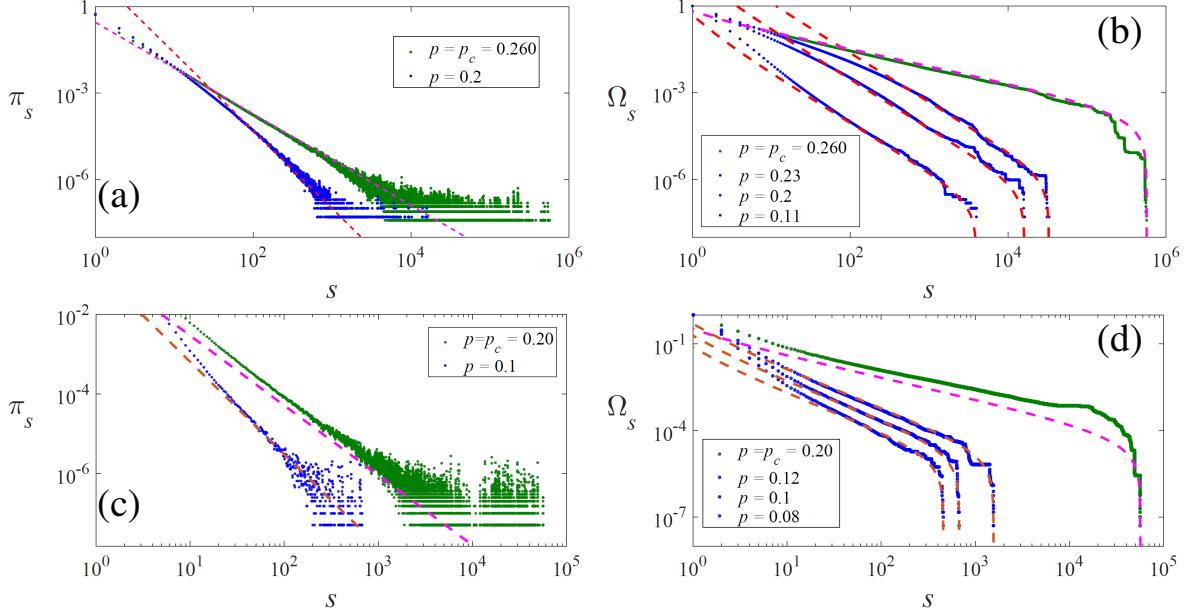


Figure S1: The probability π_s and its cumulative distribution Ω_s for a UDN (top panels) with $\lambda_o = 2.7$, $\lambda_i = 3.7$, and $X_{\text{init}} < 0$ and a CDN (bottom panels) with $\lambda_o = 3.1$ and $\lambda_i = 3.7$, and $X_{\text{init}} > 0$. The green or blue dots are the numerical results and the dashed lines are the theoretical results.

The results of Eq. (S37) are in very good agreements with the numerical results (Fig. S1) as long as p is not too close to p_c or too far from p_c (i.e., too close to 0). We immediately notice that the result we obtain is only valid for the subcritical regime ($\delta < 0$) since in supercritical regime that $\delta > 0$ the prefactor and hence the probability π_s won't be a real nonnegative value. Thus we see that we should not use exactly the same method for the supercritical regime. This point was neglected in Refs. [5, 7]; in the following we provide the correct way of calculating π_s in the supercritical regime.

S 3.2.c Supercritical regime ($\delta > 0$); method 1

Method 1 that we introduce here is the method used for generation of the results of Fig. 2 in the main text. In this method, we first consider the governing equations (S23) and (S24) written in terms of G_1 and G_2 instead of \widehat{G}_1 and \widehat{G}_2 :

$$\begin{aligned} H_1(z) &= z G_1(1 - p + pH_1(z)), \\ H_0(z) &= z G_0(1 - p + pH_1(z)). \end{aligned}$$

Then as we know that in the supercritical phase $H_1(1) < 1$, for z values close to 1, or $z = 1 - w$ with $w \ll 1$, we can write

$$H_1(z) = \tilde{h} + \epsilon = 1 - \eta + \epsilon. \quad (\text{S39})$$

Thus,

$$1 - p + pH_1(z) = 1 - p + p(1 - \eta + \epsilon) \quad (\text{S40})$$

$$= 1 - p + p - p\eta + p\epsilon \quad (\text{S41})$$

$$= 1 - p\eta + p\epsilon \quad (\text{S42})$$

$$\doteq \bar{\eta} + p\epsilon, \quad (\text{S43})$$

where $\bar{\eta} = 1 - p\eta = 1 - p(1 - \tilde{h}) = 1 - p + p\tilde{h}$. Now, we can write

$$H_1(z) = z G_1(\bar{\eta} + p\epsilon) \quad (\text{S44})$$

$$= z \left[G_1(\bar{\eta}) + G'_1(\bar{\eta})p\epsilon + \frac{1}{2}G''_1(\bar{\eta})p^2\epsilon^2 + \dots \right]. \quad (\text{S45})$$

Therefore, according to Eqs. (S39) and (S45)

$$\tilde{h} + \epsilon = z [a_0 + a_1\epsilon + a_2\epsilon^2 + \dots], \quad (\text{S46})$$

where $a_0 = G_1(\bar{\eta}) = \tilde{h}$ (see Eqs. (S23) and (S39)), $a_1 = G'_1(\bar{\eta})p$, and $a_2 = \frac{1}{2}G''_1(\bar{\eta})p^2$. Equation (S46) gives

$$\epsilon \sim \frac{1 - a_1 z}{2a_2 z} \pm \frac{1}{2a_2 z} \sqrt{(1 - a_1 z)^2 + 4a_0 a_2 z(1 - z)} \quad (\text{S47})$$

Now we consider that the supercritical properties of $H_1(z)$ can be well approximated using the behavior of ϵ near the pole $z = 1$; around this pole we have

$$\epsilon \sim \frac{1 - a_1}{2a_2} \pm \frac{1 - a_1}{2a_2} \sqrt{1 + s^*(1 - z)}, \quad (\text{S48})$$

where $s^* = \frac{4a_0 a_2}{(1 - a_1)^2}$. Now, using Eqs. (S48), (S43) and (S24) we get

$$H_0(z) = z G_0(\bar{\eta} + p\epsilon) \quad (\text{S49})$$

$$\sim (1 - w) \left[G_0(\bar{\eta}) + G'_0(\bar{\eta})p\epsilon + \frac{1}{2}G''_0(\bar{\eta})p^2\epsilon^2 + \dots \right] \quad (\text{S50})$$

$$\sim \text{analytic terms} \pm G'_0(\bar{\eta})p \frac{1 - a_1}{2a_2} \sqrt{1 + s^*(\tilde{z} - z)} \quad (\text{S51})$$

$$\sim \text{analytic terms} \pm b \sqrt{1 + s^*(\tilde{z} - z)}, \quad (\text{S52})$$

where $b = \frac{p G'_0(\bar{\eta})(1 - a_1)}{2a_2}$. Then, Eq. (S52) yields (see Sec. S 7),

$$\pi_s \sim \pm \frac{bs^{1/2}}{2\sqrt{\pi}} s^{-3/2} e^{-s/s^*} \quad \text{as } s \rightarrow \infty \quad (\text{S53})$$

$$\sim \pm \frac{p G'_0(\bar{\eta})(1 - a_1)}{2a_2} \frac{2\sqrt{a_0 a_2}}{2\sqrt{\pi}} s^{-3/2} e^{-s/s^*} \quad \text{as } s \rightarrow \infty \quad (\text{S54})$$

$$\sim \pm \frac{p G'_0(\bar{\eta}) \sqrt{a_0}}{2\sqrt{\pi a_2}} s^{-3/2} e^{-s/s^*} \quad \text{as } s \rightarrow \infty. \quad (\text{S55})$$

The $\bar{\eta}$ in Eq. (S55) is calculated according to the prescription described in Sec. S 3.2.d. Figure S2 shows

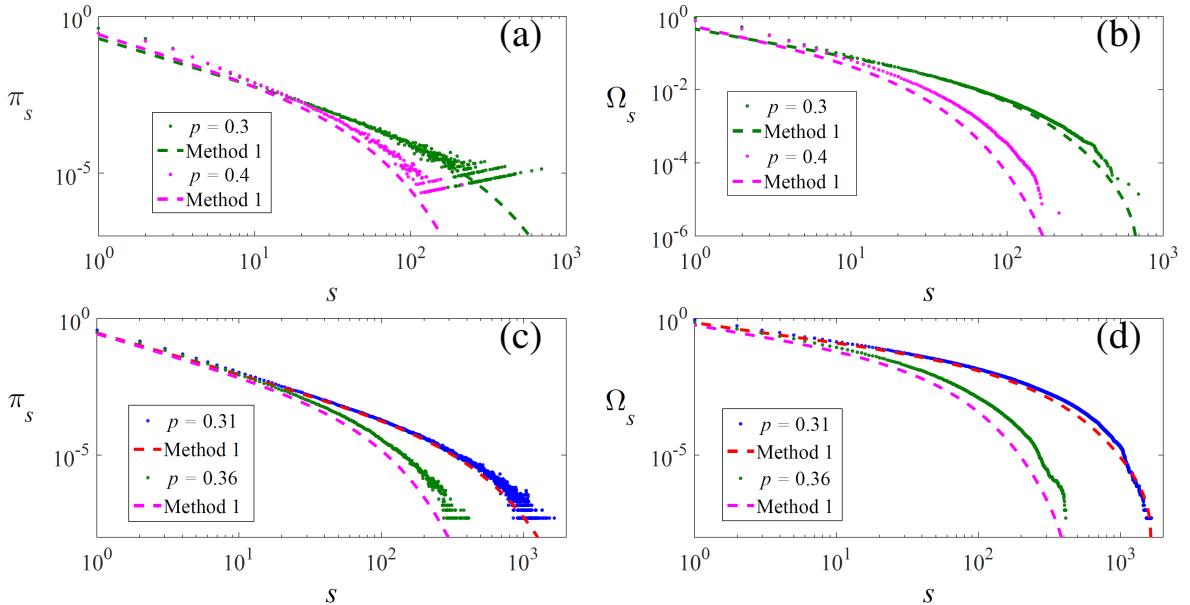


Figure S2: The probability π_s and its cumulative distribution Ω_s for the supercritical regime of an undirected network with $\lambda = 3.3$ (top panels) and the UDN of Fig. S1 (bottom panels). The dots are the numerical results and the dashed lines are the theoretical results. Closer to the critical point a better agreement between theory and numerics is observed.

that the results of Eq. (S55) are in good agreement with the numerics as long as p is not too far from p_c .

In Appendix S9, we describe two other methods. **Method 3**, at first, expands the generating functions around $H_1 = 1$ (as opposed to the expansions in this section which considered the fact that at $z = 1$, H_1 takes a value appreciable smaller than 1). Then z is expanded around the point z^* at which its derivative is zero! This operation cannot be justified and, as we will see in Appendix S9, this operation will result in inaccurate predictions; nonetheless it gives a π_s which depends on better known network parameters such as the average degree and not on the rather elaborate functions of $\bar{\eta}$.

Method 2 is our attempt to relate method 1 and method 3; in **method 2**, we assume that although $H_1(1)$ is smaller than 1, we can still approximate $H_1(1)$ with 1. Using this approximation, the parameters that appear in the expansions of the generating functions depend on well-known quantities such as the average degree, etc., instead of the quantity $\bar{\eta}$. On the other hand, in contrast to method 3, in method 2 we do not expand z around an unjustifiable value z^* ; rather we expand z around a value \tilde{z} determined by our governing equation for $H_1(z)$ (see Eq. (S56)).

In Appendix S9, we show that method 1 described here is more accurate and more reliable than the other two methods. It is worth noting that, a finding of method 3 is that s^* , the exponent of the exponential decay of π_s , is proportional to $\delta^{-(\tilde{\lambda}-1)/(\tilde{\lambda}-2)}$; this result was obtained before (Ref. [7]) using another set of techniques. But now we know that this result may not be correct, as method 3 does not accurately describe the behavior of π_s . This indicates that even for the supercritical regime of $2 < \tilde{\lambda} < 3$, the methods of preexisting studies may have significant inaccuracies.

S 3.2.d Calculation of \tilde{h} and $\bar{\eta}$

According to Eqs. (S23) and (S39) (and similarly Eqs. (S23) and (S96)), at $z = \tilde{z}$,

$$\tilde{h} = \tilde{z} \tilde{G}_1(\tilde{h}) \quad (\text{S56})$$

$$= \tilde{z} G_1(1 - p + p\tilde{h}) \quad (\text{S57})$$

We assume $\tilde{z} \approx 1$. Then we can solve Eq. (S57) recursively. Alternatively, if we are interested in obtaining quickly the whole \tilde{h} - p behavior, we can consider $\bar{\eta} \doteq 1 - p + p\tilde{h}$ as a new generating function to get

$$\bar{\eta} = 1 - p + p\tilde{h} \quad (\text{S58})$$

$$= 1 - p + p\tilde{z} G_{[1]}(\bar{\eta}) \quad (\text{S59})$$

$$= 1 - p (1 - \tilde{z} G_{[1]}(\bar{\eta})) \quad (\text{S60})$$

$$\Rightarrow \quad (\text{S61})$$

$$p = \frac{1 - \bar{\eta}}{1 - \tilde{z} G_{[1]}(\bar{\eta})}. \quad (\text{S62})$$

Then for each value of $\bar{\eta}$ we can calculate directly the corresponding p (using the assumption $\tilde{z} \approx 1$), and then obtain from Eq. (S58) the corresponding \tilde{h} .

S 3.3 A note on the effect of percolation on governing equations

As mentioned in the main text the governing equations used in Refs. [5, 7] cannot give correct prefactors for π_s ; this is because they do not apply correctly the effect of percolation on generating functions, while such effect is correctly captured by Eqs. (S23) and (S24) that incorporate the analytical results that Ref. [6] obtained for the effect of percolation on generating functions G_0 and G_1 . In fact the π_s obtained using Eqs. (S23) and (S24) equals one over p times the π_s that can be obtained using the governing equations of Refs. [5, 7]. One way to show this is comparing the two different sets of equations for the generating function employed in each method (see also Ref. [9]): Let us define the argument of G_1 in Eq. (S23) as another generating function \tilde{H}_1 , i.e., $\tilde{H}_1 = 1 - p + pH_1$. Now, writing Eq. (S23) in terms of \tilde{H}_1 , we get $\tilde{H}_1(z) = 1 - p + pz G_1(\tilde{H}_1(z))$; thus, \tilde{H} satisfies the governing equation

$$\tilde{H}_1(z) = 1 - p + pz G_1(\tilde{H}_1(z)), \quad (\text{S63})$$

used in [7] and [5]. On the other hand, whether we define $\phi = 1 - H_1$ or $\phi = 1 - \tilde{H}_1$, both of the corresponding equations for H_0 (i.e., the equation $\tilde{H}_0(z) = 1 - p + pz G_0(\tilde{H}_1(z))$ used in [5, 7] and Eq. (S24)) lead to the same form for H_0 in terms of ϕ , up to the linear order (compare Eq. (5.53) of [9] and Eq. (S26)). Hence, the difference between π_s values obtained from the two methods lies in the difference between H_1 and \tilde{H}_1 . Now, as $H_1 = (1 - p)/p + \tilde{H}_1/p$, therefore, except for $s = 0$, all the coefficients ρ_s of the generating function $H_1(z) = \sum_s \rho_s z^s$ are equal to $\tilde{\rho}_s$, the coefficients of $\tilde{H}_1(z)$, divided by p . Consequently, π_s obtained using H_1 is equal to one over p times the π_s obtained using \tilde{H}_1 , for $s \neq 0$.

S 4 Summary of the results for π_S and its parameters

$$\pi_s \sim \begin{cases} R_3 s^{-\tilde{\lambda}} & (\text{subcritical, } 2 < \tilde{\lambda} < 3) \\ R_4 s^{-3/2} e^{-s/s^*} & (\text{supercritical, } 2 < \tilde{\lambda} < 3) \\ R_2 s^{-(1+\frac{1}{\tilde{\lambda}-1})} & (\text{critical, } 2 < \tilde{\lambda} < 3) \\ R_1 s^{-3/2} e^{-s/s^*} & (\text{noncritical, } \tilde{\lambda} > 3) \\ R_1 s^{-3/2} & (\text{critical, } \tilde{\lambda} > 3) \end{cases}, \text{ where } s^* = \begin{cases} \frac{2p^2 \langle k \rangle \langle k^3 \rangle}{\delta^2 \langle k^2 \rangle^2} & (\tilde{\lambda} > 3, \text{ UN}) \\ \frac{2p^2 \langle k \rangle \langle jk^2 \rangle}{\delta^2 \langle jk \rangle^2} & (\tilde{\lambda} > 3, \text{ DN}), \\ \frac{2p^2 G_1(\bar{\eta}) G_1''(\bar{\eta})}{(1 - pG_1'(\bar{\eta}))^2} & (2 < \tilde{\lambda} < 3) \end{cases}$$

$$R_3 = \begin{cases} (1 - \bar{u}) a^* \left(1 + \frac{pp_c}{|\delta|} \langle k \rangle \right) \left(\frac{pp_c}{-\delta} \right)^{\lambda_o - 1} & (\text{UDN}) \\ (1 - \bar{u}) a^* \left(\frac{pp_c}{-\delta} \right)^{\tilde{\lambda}} & (\text{CDN \& UN}) \end{cases}, \quad R_4 = \pm \frac{G_0'(\bar{\eta}) \sqrt{G_1(\bar{\eta})}}{\sqrt{2\pi G_1''(\bar{\eta})}},$$

$$R_1 = \begin{cases} \sqrt{\langle k \rangle^3 / (2\pi \langle jk^2 \rangle)} & (\text{DN}) \\ \sqrt{\langle k \rangle^3 / (2\pi \langle k^3 \rangle)} & (\text{UN}) \end{cases}, \text{ and } R_2 = \begin{cases} \frac{-\langle k \rangle}{\Gamma\left(\frac{1}{1-\lambda_o}\right) [(1 - \bar{u})c_o \Gamma(1 - \lambda_o)]^{\frac{1}{\lambda_o - 1}}} & (\text{UDN}) \\ \frac{-\langle k \rangle^{1+\frac{1}{\tilde{\lambda}-1}}}{\Gamma\left(\frac{1}{1-\tilde{\lambda}}\right) [(1 - u)a^* \Gamma(1 - \tilde{\lambda})]^{\frac{1}{\tilde{\lambda}-1}}} & (\text{CDN \& UN}) \end{cases}.$$

S 5 More on the sample size effects

The finite size of a sample can impose exponential-like cut-offs on power-law distributions even at the critical point. Figure S3(a) demonstrates that such exponential cut-offs disappear (are in fact displaced to larger values) as larger sample sizes are used and the numerics approach the power-law distribution predicted by the theory. Another effect of the finite size that we described in the main text (see Fig. 3 of the main text) was that systems with skewed degree distribution can produce power-law distributions that expand for several orders of magnitudes and that such power-laws may be indistinguishable from pure power-laws. Such power-laws can be observed if a sufficiently large number of experiments (e.g. 10^5) are performed (Fig. S3(a)); on the other hand, in case a huge number of experiments are performed exponential cut-offs produced by the finite size can be detected. However, such exponential decays have a similar effect on pure or non-pure power-laws and power-laws with different nature may still be indistinguishable (see Fig. S3(a)).

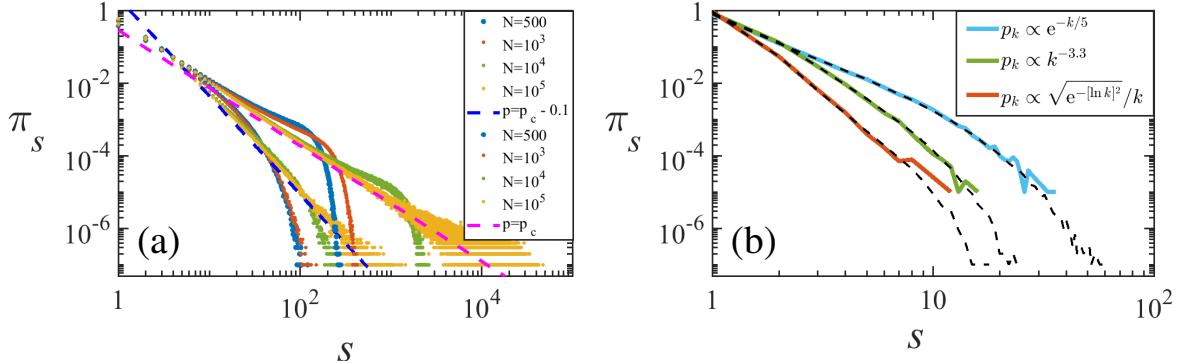


Figure S3: (a) The exponential cut-offs imposed on power-law avalanches by finite size effects disappear as the network size is increased. The dots show simulation results for subcritical and critical regime of UDNs with $\lambda_o = 2.7$ and $\lambda_i = 3.3$ with different sizes N ; the numerics approach the theoretical description (dashed lines) as N increases. (b) In finite sample sizes that pure power laws can be indistinguishable from power-laws that have an exponential cut-off because of their noncritical avalanches (compare the bold lines and also see Fig. 3 of the main text). Although a huge number of experiments can disclose the exponential cut-off produced by finite size effects; still the source of the power-law behaviour is undecidable. For bold lines the number of numerical experiments was 10^5 and for the dashed lines 10^7 experiments are performed.

S 6 $(a - z)^\beta$ non-analyticity (as in Eqs. (S31a) and (S108))

Consider a function $H(z) = \sum_n^\infty \pi_n z^n$ that has a branch point at $z^* = a$; hence its expansion around a has a leading order behaviour in the form

$$H(z) \sim \text{analytic part} + R(a - z)^\beta \quad \text{as } z \rightarrow a. \quad (\text{S64})$$

This is the same format Eqs. (S31a) and (S108) have considering $\beta = 1/2$, $w \doteq 1 - z$ and $a = 1$ (Eq. (S31a)) or $a = z^*$ (Eq. (S108)). The sequence π_n , for large n , is then determined by the leading order nonanalytic term² $R(a - z)^\beta$ [14]. Hence, to obtain π_n , we need to find the coefficients of z^n in the expansion of $(a - z)^\beta = \sum_n^\infty \eta_n z^n$. To do so, we can use binomial expansion [14]:

$$(a - z)^\beta = \sum_{n=0}^{\infty} \binom{\beta}{n} (-z)^n a^{\beta-n} \quad (\text{S65a})$$

²If the function has more than one singular point, for example two singularities, the behaviour of π_n for the largest values of n is determined by the leading order term for the branch point closest to the origin; but for some smaller n (which are still sufficiently large) the behaviour may be determined by the leading order term for the branch point further from the origin [8]. The accuracy of approximations for π_n is increased by the number of singularities considered according to a non-increasing order of the modulus of the singular points [14].

$$= \sum_{n=0}^{\infty} \binom{n-\beta-1}{n} e^{(\ln a)(\beta-n)} z^n. \quad (\text{S65b})$$

Accordingly, η_n , which are the coefficients of z^n in the above expansion, are equal to

$$\eta_n \doteq \binom{n-\beta-1}{n} e^{(\ln a)(\beta-n)} \quad (\text{S66a})$$

$$= \frac{\Gamma(n-\beta)}{\Gamma(-\beta)\Gamma(n+1)} e^{(\ln a)(\beta-n)} \quad (\text{S66b})$$

$$\sim \frac{e^{(\ln a)\beta}}{\Gamma(-\beta)} n^{-(\beta+1)} e^{-(\ln a)n} \quad \text{as } n \rightarrow \infty, \quad (\text{S66c})$$

where in S66c we used Stirling's approximation $\Gamma(n+1) = n! = \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$. Hence,

$$\pi_n \sim \frac{R e^{(\ln a)\beta}}{\Gamma(-\beta)} n^{-(\beta+1)} e^{-(\ln a)n} \quad \text{as } n \rightarrow \infty. \quad (\text{S67})$$

For the generating function $H(z)$ in the form of Eq. (S34), $a = 1$ and for that in the form of Eq. (S108), $a = z^*$. For $a = 1$, Eq. (S35) is obtained, that is,

$$\pi_n \sim \frac{R}{\Gamma(-\beta)} n^{-(\beta+1)} \quad \text{as } n \rightarrow \infty. \quad (\text{S68})$$

On the other hand, for $a = z^* \sim 1 + \xi$ (which is the case for Eq. (S108)), and $\xi < 1$ we get

$$\pi_n \sim \frac{R e^{\xi\beta}}{\Gamma(-\beta)} n^{-(\beta+1)} e^{-\xi n} \quad \text{as } n \rightarrow \infty. \quad (\text{S69})$$

S7 $\sqrt{1 + s^*(\tilde{z} - z)}$ non-analyticity (as in Eqs. (S31b), (S52) and (S100))

Using the Cauchy theorem, π_s can be written as [11]

$$\pi_s = \frac{1}{2\pi i} \oint_C \frac{H_0(z)}{z^s} dz, \quad (\text{S70})$$

where C can be any closed contour around the origin that does not enclose any poles in H_0 . We choose a unit circle and deform it into the keyhole contour $C_R \cup l_1 \cup C_\epsilon \cup l_2$ shown in Fig. S4. In Fig. S4, μ is a branch point, and the branch cut is located on the real axis from α to ∞ . It is straightforward to show that the integrals along the circular contours C_R and C_ϵ limit to zero [15, 16]. Hence, for $H_0(z) \sim$ analytic part $\pm b \sqrt{1 + s^*(\tilde{z} - z)}$ as $z \rightarrow \tilde{z}$, we will have:

$$\pi_s = \pm \frac{b}{2\pi i} \int_{l_1 \cup l_2} \frac{\sqrt{1 + s^*(\tilde{z} - z)}}{z^s} dz \quad (\tilde{z} - z \doteq w) \quad (\text{S71a})$$

$$= \mp \frac{b}{2\pi i} \int_{l_1 \cup l_2} \frac{\sqrt{1 + s^* w}}{z^s} dw \quad (z \doteq e^\rho \text{ and } w \doteq \tilde{z} - e^\rho \sim -\rho \text{ [assuming } \tilde{z} \approx 1]) \quad (\text{S71b})$$

$$= \pm \frac{b}{2\pi i} \int_{l_1 \cup l_2} \frac{\sqrt{1 - s^* \rho}}{e^{s\rho}} d\rho \quad (u \doteq 1 - s^* \rho) \quad (\text{S71c})$$

$$= \pm \frac{b}{2\pi i} \int_{l_1 \cup l_2} \frac{\sqrt{u}}{e^{s(1-u)/s^*}} \left(-\frac{du}{s^*}\right) \quad (\text{S71d})$$

$$= \mp \frac{b e^{-s/s^*}}{2\pi i s^*} \int_{l_1 \cup l_2} u^{1/2} e^{(s/s^*)u} du, \quad (\text{S71e})$$

where in Eq. (S71a), we used the fact that the net contribution from the analytic part of H_0 on $l_1 \cup l_2$ is zero. In Eq. (S71d), we made the substitution $z \doteq e^\rho$, and $w \doteq \tilde{z} - e^\rho \sim -\rho$ as $\rho \rightarrow 0$. To obtain Eq. (S71e), we employed the substitution $u \doteq 1 - s^* \rho$. Now consider that the location of an arbitrary point on the l_1 path can be written as $z = c + i d$, or, equivalently, $w = (\tilde{z} - c) - i d$, $\rho \simeq (c - \tilde{z}) + i d$, and $u = (1 + \tilde{z} - c) - i d$. Thus, in the complex u -plane, l_1 is located in the third quadrant. Moreover, note that, along l_1 , z in Eq. (S71a) varies from $1/s^* + \tilde{z}$ to ∞ ; this corresponds to a variation in w from $-1/s^*$ to $-\infty$. Hence, ρ in Eq. (S71d) varies from $1/s^*$ to ∞ , and u varies from 0 to $-\infty$. Similarly, along l_2 , u is located in the second quadrant and varies from $-\infty$ to 0. Therefore, we can use another change of variable: $u \doteq r e^{i\theta}$, where

$$\begin{cases} \text{along } l_1 : & \theta = -\pi, \ u = -r, \text{ and } u^{1/2} = r^{1/2} e^{-i\pi/2} \\ \text{along } l_2 : & \theta = \pi, \ u = -r, \text{ and } u^{1/2} = r^{1/2} e^{i\pi/2}. \end{cases} \quad (\text{S72})$$

Thus, along l_1 , r varies from 0 to ∞ , and along l_2 , it varies from ∞ to 0. Therefore,

$$\pi_s = \mp \frac{b e^{-s/s^*}}{2\pi i s^*} \int_{l_1 \cup l_2} r^{1/2} e^{i\theta/2} e^{-(s/s^*)r} du, \quad (\text{S73a})$$

$$= \mp \frac{b e^{-s/s^*}}{2\pi i s^*} \left[\int_0^\infty e^{-ir/2} + \int_\infty^0 e^{i\pi/2} (-dr) \right] r^{1/2} e^{-(s/s^*)r} (-dr) \quad (\text{S73b})$$

$$= \pm \frac{b e^{-s/s^*}}{\pi s^*} \left[\frac{e^{i\pi/2} - e^{-i\pi/2}}{2i} \right] \int_0^\infty r^{1/2} e^{-(s/s^*)r} dr \quad (\text{S73c})$$

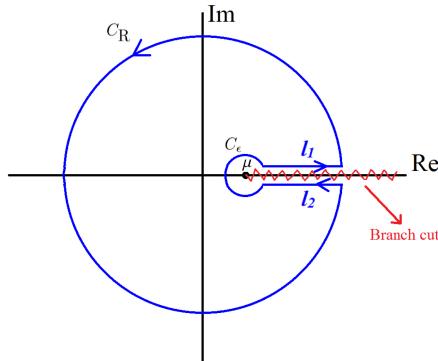


Figure S4: The keyhole contour C in the complex z -plane for the integral of Eq. (S70). A branch cut extends from $\mu = 1/s^* + \tilde{z}$ (see Eq. (S71b)) to ∞ . The circular arcs have radii ϵ and R .

$$= \pm \frac{b e^{-s/s^*}}{\pi s^*} (s^*/s)^{3/2} \int_0^\infty t^{3/2-1} e^{-t} dt, \quad (\text{S73d})$$

where the change of variable $r = (s^*/s)t$ is performed in Eq. (S73d). Now, the integral in Eq. (S73d) equals $\Gamma(3/2) = \sqrt{\pi}/2$. Hence,

$$\pi_s \sim \pm \frac{b s^{*1/2}}{2\sqrt{\pi}} s^{-3/2} e^{-s/s^*}. \quad (\text{S74})$$

S 8 Simpler equations to numerically calculate $G_{0/1}$ at a specific point h

In uncorrelated directed networks

$$G_0(h) = \sum_{j,k=0}^\infty P_{jk} h^k = \sum_{j,k=0}^\infty P_j P_k h^k = \sum_{j=0}^\infty P_j \sum_{k=0}^\infty P_k h^k = \sum_{k=0}^\infty P_k h^k, \quad (\text{S75})$$

$$G_1(h) = \frac{1}{\langle j \rangle} \sum_{j,k=0}^\infty j P_{jk} h^k = \left(\frac{1}{\langle j \rangle} \sum_{j=0}^\infty j P_j \right) \sum_{k=0}^\infty P_k h^k = \sum_{k=0}^\infty P_k h^k \quad (\text{S76})$$

However, for correlated directed networks of Eq. (S8)

$$G_1(h) = \sum_{j,k=0}^\infty q_{jk} h^k \quad (\text{S77})$$

$$= \frac{1}{\langle j \rangle} \sum_{j=0}^\infty \left[j u P_j^{(\text{tail})} h^{k_{\min}} + \sum_{k=k_{\text{start}}}^\infty j (1-u) P_k^{(\text{tail})} h^k \right] \quad (\text{S78})$$

$$= \frac{1}{\langle j \rangle} \left[u \langle j \rangle h^{k_{\min}} + (1-u) \sum_{j=0}^\infty \sum_{k=k_{\text{start}}}^\infty j P_k^{(\text{tail})} \delta_{k,K(j)} h^k \right] \quad (\text{S79})$$

$$= \frac{1}{\langle j \rangle} \left[u \langle j \rangle h^{k_{\min}} + (1-u) \sum_{j=0}^\infty j P_j^{(\text{tail})} h^{K(j)} \right] \quad (\text{S80})$$

$$= u h^{k_{\min}} + \frac{1}{\langle j \rangle} (1-u) \sum_{j=0}^\infty j P_j^{(\text{tail})} h^{K(j)} \quad (\text{S81})$$

$$= u \frac{\langle j h^{k_{\min}} \rangle}{\langle j \rangle} + (1-u) \frac{1}{\langle j \rangle} \langle j h^{K(j)} \rangle \quad (\text{S82})$$

$$= \frac{\langle j h^k \rangle}{\langle j \rangle}, \quad (\text{S83})$$

where in Eq. (S79) we used the fact that in our fully correlated networks, according to Eq. (S3), k is determined by j such that $k = K(j) = j^{\frac{A_j-1}{A_j-1}}$. To write Eq. (S80), we considered that for a j and a k that satisfy Eq. (S3), P_k and P_j are actually identical. It is worth noting that in Eqs. (S81)-(S82), the averages are over all the in-degree values (compared to Eq. (S77) which averages over all in-degree-out-degree pairs).

Similarly, for correlated directed networks of Eq. (S9) we can write

$$G_1(h) = \sum_{j,k=0}^{\infty} q_{jk} h^k \quad (\text{S84})$$

$$= \frac{1}{\langle j \rangle} \sum_{j,k=0}^{\infty} j P_{jk} h^k \quad (\text{S85})$$

$$= \frac{1}{\langle j \rangle} \sum_{k=0}^{\infty} \left[j_{\min} u P_k^{(\text{tail})} h^k + \sum_{j=j_{\text{start}}}^{\infty} j (1-u) P_k^{(\text{tail})} h^k \right] \quad (\text{S86})$$

$$= \frac{1}{\langle j \rangle} \sum_{k=0}^{\infty} \left[u j_{\min} P_k^{(\text{tail})} h^k + (1-u) \sum_{j=j_{\text{start}}}^{\infty} j P_k^{(\text{tail})} \delta_{j,J(k)} h^k \right] \quad (\text{S87})$$

$$= \frac{1}{\langle j \rangle} \left[u j_{\min} \sum_{k=0}^{\infty} P_k^{(\text{tail})} h^k + (1-u) \sum_{k=0}^{\infty} \sum_{j=j_{\text{start}}}^{\infty} j P_k^{(\text{tail})} \delta_{j,J(k)} h^k \right] \quad (\text{S88})$$

$$= \frac{1}{\langle j \rangle} \left[u j_{\min} \sum_{k=0}^{\infty} P_k^{(\text{tail})} h^k + (1-u) \sum_{k=0}^{\infty} J(k) P_k^{(\text{tail})} h^k \right] \quad (\text{S89})$$

$$= \frac{1}{\langle j \rangle} [u j_{\min} \langle h^k \rangle + (1-u) \langle J(k) h^k \rangle] \quad (\text{S90})$$

$$= \frac{1}{\langle j \rangle} [u \langle j_{\min} h^k \rangle + (1-u) \langle J(k) h^k \rangle] \quad (\text{S91})$$

$$= \frac{\langle j h^k \rangle}{\langle j \rangle}, \quad (\text{S92})$$

where the averages are over all the out-degree values.

S 9 Other methods for the supercritical regime

Method 2

This method is a variation of method 3 (see Sec. Method 3) and provides a much better description of numerical results than method 3; however its performance is still worse than that of method 1 (Sec. S 3.2.c). We start with Eq. (S23) which states that

$$z = \frac{1-\phi}{G_1(1-p\phi)}. \quad (\text{S93})$$

Then, for $2 < \tilde{\lambda} < 3$, we have

$$z \sim \frac{1-\phi}{1 - Ap\phi + D\phi^v + o(v)} \quad \text{as } \phi \rightarrow 0 \quad (\text{S94a})$$

$$\sim (1-\phi)(1 + Ap\phi - D\phi^v) \quad \text{as } \phi \rightarrow 0 \quad (\text{S94b})$$

$$\sim 1 + A\delta\phi - D\phi^\nu \quad \text{as } \phi \rightarrow 0, \quad (\text{S94c})$$

where $\nu \doteq \tilde{\lambda} - 1$. Equation (S94c) has the same leading order behavior as Eq. (S25) when $2 < \tilde{\lambda} < 3$; hence, obviously, it has the same solutions for $\delta \leq 0$. To obtain a valid solution $\delta > 0$, we first define $h \doteq H_1(z) = 1 - \phi$; accordingly, $z = H_1^{-1}(h)$. According to Eq. (S94c),

$$z = 1 + A\delta(1 - h) - D(1 - h)^\nu, \quad (\text{S95a})$$

The taylor expansion of z around an arbitrary point $\tilde{h} < 1$ has the form:

$$z \sim \tilde{z} - \beta(h - \tilde{h}) - \alpha(h - \tilde{h})^2 \quad \text{as } h \rightarrow \tilde{h}, \quad (\text{S96})$$

where $\tilde{z} = z(\tilde{h})$, $\beta = -\frac{dz}{dh}\Big|_{h=\tilde{h}} = A\delta - D\nu(1 - \tilde{h})^{\nu-1}$, and $\alpha = -\frac{1}{2}\frac{d^2z}{dh^2}\Big|_{h=\tilde{h}} = \frac{D\nu(\nu-1)(1 - \tilde{h})^{\nu-2}}{2}$. It is worth mentioning that we know that at $\tilde{h} = 1$ the correct behavior of z is described by Eq. (S104a) and not Eq. (S96). The solution to Eq. (S96) is

$$h - \tilde{h} \sim \frac{-\beta}{2\alpha} \pm \frac{1}{2\alpha} \sqrt{\beta^2 - 4\alpha(\tilde{z} - z)}; \quad (\text{S97})$$

Therefore,

$$\phi = 1 - h \sim -\tilde{h} + \frac{\beta}{2\alpha} \pm \frac{1}{2\alpha} \sqrt{\beta^2 + 4\alpha(\tilde{z} - z)}, \quad (\text{S98})$$

and according to Eqs. (S24) and (S16)

$$H_0(\tilde{z}) \sim \text{analytic terms} \pm \frac{\tilde{z}E}{2\alpha} \sqrt{\beta^2 + 4\alpha(\tilde{z} - z)} \quad (\text{S99})$$

$$\sim \text{analytic terms} \pm b \sqrt{1 + s^*(\tilde{z} - z)}, \quad (\text{S100})$$

where, $b = \frac{\tilde{z}E\beta}{2\alpha} = \frac{\tilde{z}E(A\delta - D\nu(1 - \tilde{h})^{\nu-1})}{2\nu(\nu-1)(1 - \tilde{h})^{\nu-2}}$ and $s^* = \frac{4\alpha}{\beta^2} = \frac{2D\nu(\nu-1)(1 - \tilde{h})^{\nu-2}}{(A\delta - D\nu(1 - \tilde{h})^{\nu-1})^2}$.

Using Eq. (S99) and the result of S 7, for $2 < \tilde{\lambda} < 3$ and $\delta > 0$, we obtain

$$\pi_s \sim \pm \frac{bs^{*1/2}}{2\sqrt{\pi}} s^{-3/2} e^{-s/s^*} \quad (\text{S101})$$

$$\pi_s \sim \pm \frac{\tilde{z}E}{2\sqrt{\pi\alpha}} s^{-3/2} e^{-s/s^*} \quad \text{as } s \rightarrow \infty \quad (\text{S102})$$

$$\sim \pm \frac{\tilde{z}E}{\sqrt{2\pi D\nu(\nu-1)(1 - \tilde{h})^{\nu-2}}} s^{-3/2} e^{-s/s^*} \quad \text{as } s \rightarrow \infty. \quad (\text{S103})$$

In Eq. (S103)³, we should substitute a suitable pair of values (\tilde{z}, \tilde{h}) , such that they satisfy the condition that \tilde{h} is not equal to 1 but also satisfy our assumptions that \tilde{h} is close to 1. This imposes a degree of arbitrariness and unreliability in our method. In Fig. S5 we used $\tilde{h} = 0.95$ and obtained the corresponding \tilde{z} using Eq. (S57); we see that for this pair of values method 2 performs better than Method 3 (see the following section). Figure S5 also shows that method 1 (Sec. S 3.2.c) is more accurate than Method 3 and more reliable than method 2 (this section).

³It is worth mentioning that the above approach can also be used for $\lambda > 4$; in this case the term $D\phi^\nu$ in Eq. (S94a) is replaced with $B\phi^2$. Consequently, π_s is determined by Eq. (S101), but with different s^* and b values.

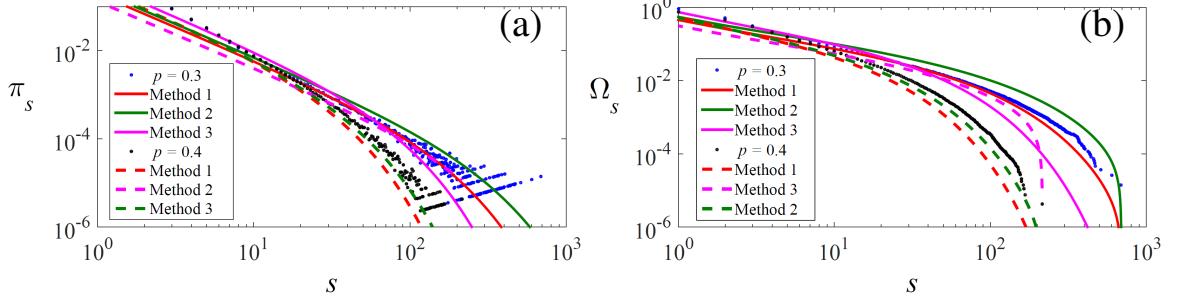


Figure S5: The probability π_s and its cumulative distribution Ω_s for the undirected network of Fig. S2. The dots are the numerical results and the lines are the theoretical results obtained from different methods.

Method 3

We developed this method based on the method of Lee *et al.* [8]. Consider Eq. (S95a) and its derivative:

$$z = 1 + A\delta(1 - h) - D(1 - h)^\nu, \quad (\text{S104a})$$

$$\frac{dz}{dh} = -A\delta + D\nu(1 - h)^{\nu-1}. \quad (\text{S104b})$$

Also consider the point h^* for which $\left.\frac{dz}{dh}\right|_{h^*} = 0$. According to Eq. (S104b), $1 - h^* = \left(\frac{A\delta}{D\nu}\right)^{\frac{1}{\nu-1}}$ and $z^* \doteq z(h^*) = 1 + A\delta(1 - h^*) \left[1 - \frac{D}{A\delta}(1 - h)^{\nu-1}\right] = 1 + A\delta(1 - h^*)(1 - 1/\nu)$ is obtained from Eq. (S104a). Then for the expansion of z around h^* we get [8]

$$z = z^* + \sum_{n=2}^{\infty} \frac{M_n}{n!} (h - h^*)^n. \quad (\text{S105a})$$

To the leading order,

$$z - z^* \sim \frac{M_2}{2} (h - h^*)^2 \quad \text{as } h \rightarrow h^*, \quad (\text{S106})$$

where $M_2 = -\nu(\nu-1)D(1 - h^*)^{\nu-2} = -\nu(\nu-1)D \left(\frac{A\delta}{D\nu}\right)^{(\nu-2)/(\nu-1)}$. Thus, the leading order behaviour of $H_1(z)$ is determined by

$$1 - \phi = h \sim h^* \pm \sqrt{\frac{2}{M_2}} (z - z^*)^{1/2} \quad \text{as } z \rightarrow z^*; \quad (\text{S107})$$

as a result, the expansion of $H_0(z)$ around z^* has a singularity at $z = z^*$:

$$H_0(z) \sim \text{analytic part} \pm \langle k \rangle p \sqrt{\frac{2}{-M_2}} (z^* - z)^{1/2} \quad \text{as } z \rightarrow z^*, \quad (\text{S108})$$

as opposed to the critical and subcritical case, where the singularity (closest to the origin) is located at $z = 1 - w = 1$. It is worth noting that M_2 in Eq. (S108) is real only for nonnegative δ ; hence, as we will show in a moment, for $3 < \gamma < 4$, π_s obtained from Eq. (S108) is valid only for $p > p_c$.

Using Eq. (S108) and the results of Appendix S 6, for $3 < \gamma < 4$ and $\delta > 0$, we obtain

$$\pi_s \sim \frac{\langle k \rangle p e^{\ln(z^*)/2}}{\sqrt{-2\pi M_2}} s^{-3/2} e^{-\ln(z^*) s} \quad \text{as } s \rightarrow \infty. \quad (\text{S109})$$

Now, it is obvious that as, for $3 < \gamma < 4$, M_2 is real when $\delta > 0$, Eq. (S109) is valid only for the supercritical phase. The rate of exponential decay is determined by $\ln(z^*)$ which is proportional to $\delta^{1+1/(\gamma-1)} = \delta^{(\tilde{\lambda}-1)/(\tilde{\lambda}-2)}$; this result is consistent with the result obtained using finite size scaling [7] for undirected networks with $3 < \lambda < 4$. Fig. S5 illustrates that method 3 is not accurate for describing the supercritical properties of networks with $2 < \tilde{\lambda} < 3$ and that methods 1 and 2 provide better descriptions for such networks.

It is worth mentioning that the above approach can also be used for $\gamma > 4$; in this case the term $D\phi^\gamma$ in Eq. (S94a) is replaced with $B\phi^2$. Consequently, π_s is determined by Eq. (S109), but with a different z^* and M_2 values. This result is valid for both the supercritical and subcritical region, and matches (not shown) the result of Sec. 5.5.4 of Ref. [9] (the so called method (IV) in [9]) for $\gamma > 4$.

References

1. M. E. J. Newman, “Assortative Mixing in Networks”, *Phys. Rev. Lett.* **89**, 208701 (2002).
2. M. Catanzaro, M. Boguñá & R. Pastor-Satorras, “Generation of uncorrelated random scale-free networks”, *Phys. Rev. E* **71**, 027103 (2005).
3. O. Williams & C. I. Del Genio, “Degree Correlations in Directed Scale-Free Networks”, *PLOS ONE* **9**, 1–6 (Oct. 2014).
4. D. Yang, L. Pan & T. Zhou, “Lower bound of assortativity coefficient in scale-free networks”, *Chaos* **27**, 033113 (2017).
5. N. Schwartz, R. Cohen, D. ben-Avraham, A.-L. Barabási & S. Havlin, “Percolation in directed scale-free networks”, *Phys. Rev. E* **66**, 015104 (2002).
6. M. E. J. Newman, “Spread of epidemic disease on networks”, *Phys. Rev. E* **66**, 016128 (2002).
7. R. Cohen, D. ben-Avraham & S. Havlin, “Percolation critical exponents in scale-free networks”, *Phys. Rev. E* **66**, 036113 (2002).
8. D. Lee, J. Kim, B Kahng & D Kim, “Scale-free random branching trees in supercritical phase”, *Journal of Physics A: Mathematical and Theoretical* **40**, 7139 (2007).
9. A. Faqeeh, *Percolation and its relations to other processes in networks; University of Limerick* (2016).
10. M. E. J. Newman, S. H. Strogatz & D. J. Watts, “Random Graphs With Arbitrary Degree Distributions and Their Applications”, *Phys. Rev. E* **64**, 026118 (2001).
11. M. E. J. Newman, “Component Sizes in Networks With Arbitrary Degree Distributions”, *Phys. Rev. E* **76**, 045101(R) (2007).
12. M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Oxford, 2010).
13. P. D. Miller, *Applied Asymptotic Analysis* (American Mathematical Soc., 2006).

14. H. S. Wilf, *Generatingfunctionology* (A K Peters, Ltd., Wellesley, MA, 2006).
15. J. P. Gleeson, J. A. Ward, K. P. O’Sullivan & W. T. Lee, “Competition-Induced Criticality in a Model of Meme Popularity”, *Phys. Rev. Lett.* **112**, 048701 (2014).
16. J. P. Keener, *Principles of Applied Mathematics: Transformation and Approximation* (Addison-Wesley, Boston, MA, 1988).

Chapter Three

Generalization of core percolation on complex networks

Contribution

I performed all the simulation part of this research which includes algorithmic design to obtain k-cores and generalized k-cores of the random and real networks and corresponding lay-outs. My contribution in the research design was the idea to locate the transition points of the Gk-core percolation using divergence of the iteration. I also contributed partially in editing of the manuscript.

Generalization of core percolation on complex networks

N. Azimi-Tafreshi

Physics Department, Institute for Advanced Studies in Basic Sciences, 45195-1159 Zanjan, Iran

S. Osat

Quantum Complexity Science Initiative, Skolkovo Institute of Science and Technology, Skoltech Building 3, Moscow, 143026, Russia

S. N. Dorogovtsev

*Departamento de Física da Universidade de Aveiro and I3N, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal
and A. F. Ioffe Physico-Technical Institute, 194021 St. Petersburg, Russia*



(Received 1 August 2018; revised manuscript received 9 December 2018; published 22 February 2019)

We introduce a k -leaf removal algorithm as a generalization of the so-called leaf removal algorithm. In this pruning algorithm, vertices of degree smaller than k , together with their first nearest neighbors and all incident edges, are progressively removed from a random network. As the result of this pruning the network is reduced to a subgraph which we call the Generalized k -core (Gk -core). Performing this pruning for the sequence of natural numbers k , we decompose the network into a hierarchy of progressively nested Gk -cores. We present an analytical framework for description of Gk -core percolation for undirected uncorrelated networks with arbitrary degree distributions (configuration model). To confirm our results, we also derive rate equations for the k -leaf removal algorithm which enable us to obtain the structural characteristics of the Gk -cores in another way. Also we apply our algorithm to a number of real-world networks and perform the Gk -core decomposition for them.

DOI: [10.1103/PhysRevE.99.022312](https://doi.org/10.1103/PhysRevE.99.022312)

I. INTRODUCTION

Structural decomposition of complex networks providing classification of the vertices into different subsets is one of the effective approaches for studying the structural properties of networks. As a primary and well-known example, one can indicate k -core decomposition, which is an efficient technique for uncovering structural properties of large networks [1,2]. The k -core of a network is defined as the largest subgraph whose vertices have degree at least k [3]. There is a pruning algorithm enabling one to obtain k -core subgraphs for a given network: at each step, a vertex of degree less than k is randomly chosen and removed. The pruning is continued until no further removal is possible. As the result of this pruning the network is decomposed to a set of enclosed k -cores. The vertices belonging to higher (more central) cores are more strongly connected. It was also shown that the vertices of the inner core are more influential spreaders in epidemic processes [4]. A giant k -core emerges above a percolation threshold [5]. The most remarkable result is that for $k \geq 3$ the giant k -core shows a discontinuous hybrid phase transition combining discontinuity and a critical singularity [5,6]. Furthermore, generalized models for k -core percolation have been studied on interdependent and multiplex networks, which reveal more features than the ordinary k -core percolation problem on single networks [7,8].

Another key subgraph of a random network is simply called its core. These subgraphs significantly differ from the k -cores. A core of an undirected network is obtained only by a pruning algorithm in contrast to the k -core, which is, in addition, defined by a specific constraint on the connectivity of

its vertices. The pruning algorithm producing a core is called the leaf removal algorithm and was introduced by Karp and Sipser [9]. In this pruning algorithm, a vertex of degree one (a leaf) is randomly chosen and removed together with its neighbor and all incident edges. The algorithm is continued until no leaves remain. The resulting subgraph is formed by some isolated subgraphs and the giant one, which is called the core. For the Erdős-Rényi (ER) random graphs, Bauer and Golinelli showed that the core percolation threshold is located at the mean degree $\langle q \rangle = e = 2.718\dots$, so that above this point the network contains the giant core, while below the threshold the size of the giant core is zero [10]. The core structure and the phase transition at $\langle q \rangle = e$ is related to a number of phenomena in physics such as conductor-insulator transitions [11] and replica symmetry breaking in the minimal vertex covers [12]. Moreover it was shown that the formation of the core is related to controllability robustness [13,14] and some combinatorial optimization problems such as the maximum matching and minimum vertex cover [9,12,15]. Also a generalized leaf removal process, which is applicable in the minimum dominating set problem, has been introduced in Ref. [16]. Using a time-dependent analysis, people have studied the core percolation related to this generalized leaf-removal algorithm.

In this paper, we generalize definition of the leaf to the “ k -leaf,” defined as a vertex of degree less than k . In this algorithm we remove recursively a k -leaf together with all its first neighbors and their incident edges. Following this pruning algorithm, the network is decomposed to a hierarchy of nested cores, similarly to the ordinary k -core decomposition. We call this structure the Generalized k -core (Gk -core). In this

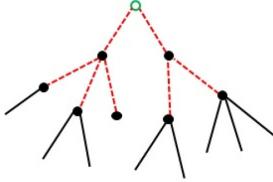


FIG. 1. The open green dot shows a k -leaf. Once a k -leaf is selected, the k -leaf together with the dashed red edges are removed.

notation, the ordinary core is represented by the G_2 -core. The vertices belonging to inner G_k -cores and their first neighbors are of high degree and well connected. Analytical calculation is possible only for the networks with a locally treelike structure. For these kind of networks, and using the generating function technique, we study the structural transitions and emergence points of the G_k -core subgraphs.

The k -leaf removal algorithm can be also considered as the inducing effect, introduced by Zhao *et al.* [17]. In the inducing process, a collapsed vertex i will induce its remaining neighbors, to be collapsed if vertex i has fewer than k remaining neighbors. In Ref. [17] the inducing effect together with the spontaneous collapsing process leads to the emergence of other subgraphs, called protected cores.

The leaf removal algorithm is a Markovian process. We describe evolution of the network structure during the pruning process by applying rate equations, which have been derived for the ordinary leaf removal algorithm on undirected and directed graphs [15,18,19]. This approach provided the size and the emergence point of the ordinary core. In this paper, we also derive rate equations for the degree distribution of a network during the execution of the k -leaf algorithm, which enables us to obtain the structure of the G_k -cores in an alternative way.

The paper is organized as follows. In Sec. II we present an analytical framework to study G_k -core percolation for random networks with arbitrary degree distributions. We apply our general results to the ER and scale-free networks. We compare our results with numerical simulations. In Sec. III we derive the rate equations for the k -leaf removal algorithm, and using these equations we find in another way how G_k -cores are organized. In Sec. IV a set of real-world networks are analyzed in the framework of our approach.

II. ANALYTICAL FRAMEWORK

Let us consider an uncorrelated network with an arbitrary degree distribution $P(k)$. To produce a generalization of the core subgraph, we use the following pruning algorithm: at each step we randomly choose a k -leaf (i.e., a vertex of degree less than k) and remove it together with its neighbors and all incident edges to the neighbors. Figure 1 shows a k -leaf (open green dot) and the k -leaf removal process. As a result of the pruning, the degrees of some vertices change. The procedure is iterated until no vertices of degree less than k remain in the network. The residual network, if it exists, is called the G_k -core.

To find the size of the G_k -core, we classify the vertices into three groups: (1) α -removable: the vertices that can become a



FIG. 2. Schematic representation of the probabilities α and β .

leaf; (2) β -removable: the vertices that can become a neighbor of a leaf; (3) the vertices that are neither α -removable nor β -removable and hence belong to G_k -core. Using the assumption that the network has a locally treelike structure, we can write self-consistency equations for probabilities that a random neighbor of a random vertex is α -removable, β -removable, or a nonremovable vertex. We call these probabilities α , β , and $1 - \alpha - \beta$, respectively. These probabilities are represented graphically in Fig. 2. Note that the definition of these probabilities is the same as that already defined in Ref. [20]. The difference is in the definition of the leaves.

At least one of the neighbors of a β -removable vertex must be α -removable. Furthermore, an end vertex of a randomly chosen edge belongs to the G_k -core, if it has at least $k - 1$ neighbors which belong to the G_k -core and none of its neighbors are of type α . Taking into account these facts, we write the following two self-consistent equations:

$$\begin{aligned} 1 - \alpha - \beta &= \sum_q \frac{qP(q)}{\langle q \rangle} \\ &\times \sum_{s=k-1}^{q-1} \binom{q-1}{s} (1 - \alpha - \beta)^s \beta^{q-1-s}, \\ \beta &= 1 - \sum_q \frac{qP(q)}{\langle q \rangle} (1 - \alpha)^{q-1}. \end{aligned} \quad (1)$$

The first equation represents the probability that an end vertex of a randomly chosen edge belongs to the G_k -core. $qP(q)/\langle q \rangle$ is the probability that the end vertex of a uniformly randomly chosen edge has degree q , and the combinatorial multiplier $\binom{q-1}{s}$ gives the number of ways one can choose n edges from a sample of m edges. At least $k - 1$ edges of $q - 1$ edges (other edges than the starting one) must lead to the G_k -core. Equation (1) also shows the probability that an end vertex of a randomly chosen edge is β -removable. At least one of the neighbors of a β -removable vertex must be a leaf, i.e., an α -removable vertex. These two equations are schematically represented in Fig. 3.

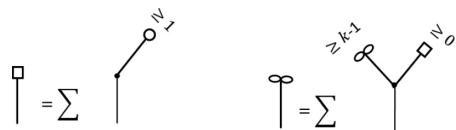


FIG. 3. Graphical representation of the self-consistency equations for the probabilities β and $1 - \alpha - \beta$.

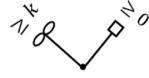


FIG. 4. Schematic representation of the probability that a vertex belongs to the G_k -core, which is the relative size n_{kc} of the G_k -core.

From Eq. (1), one can derive the following self-consistency equation for α :

$$\alpha = \sum_q \frac{qP(q)}{\langle q \rangle} \sum_{s=0}^{k-2} \binom{q-1}{s} (1-\alpha-\beta)^s \beta^{q-1-s}. \quad (2)$$

The probabilities α and β enable us to obtain the probability n_{kc} that a randomly chosen vertex belongs to the G_k -core, which is also the relative size of the G_k -core. Figure 4 shows a schematic representation of this probability. A vertex is in the G_k -core if the vertex has at least k neighbors which belong to the G_k -core. Hence we can write the following equation for the relative size of the G_k -core:

$$n_{kc} = \sum_{q>k} P(q) \sum_{s=k}^q \binom{q}{s} (1-\alpha-\beta)^s \beta^{q-s}. \quad (3)$$

To be able to solve Eqs. (1)–(3) analytically, we rewrite these equations using generating functions [21]. For a network with a given degree distribution $P(q)$, the generating function $G(x)$

is defined as

$$G(x) \equiv \sum_q P(q)x^q. \quad (4)$$

Hence, we obtain the following equations for α , β and n_{kc} in terms of the generating function:

$$\begin{aligned} \alpha &= \frac{1}{\langle q \rangle} \sum_{s=0}^{k-2} \frac{(1-\alpha-\beta)^s}{s!} G^{(s+1)}(\beta), \\ \beta &= 1 - \frac{G^{(1)}(1-\alpha)}{\langle q \rangle}, \\ n_{kc} &= G(1-\alpha) - \sum_{s=0}^{k-1} \frac{(1-\alpha-\beta)^s}{s!} G^{(s)}(\beta), \end{aligned} \quad (5)$$

where we used the notation $G^{(s)}(x)$ for the s th derivatives of $G(x)$.

Furthermore, the probability that both end vertices of an edge in the network belong to the G_k -core is $(1-\alpha-\beta)^2$. Hence, the fraction of edges in the G_k -core, denoted by l_{kc} , is obtained as

$$l_{kc} = \frac{c}{2}(1-\alpha-\beta)^2. \quad (6)$$

Let us first consider ER networks with Poisson degree distributions, $P(q) = c^q e^{-c}/q!$, where c is the vertex mean degree for the network. For the Poisson distribution, the generating function and its s th derivative are $G(x) = e^{-c(1-x)}$ and $G^{(s)}(x) = c^s e^{-c(1-x)}$, respectively. One can easily find the

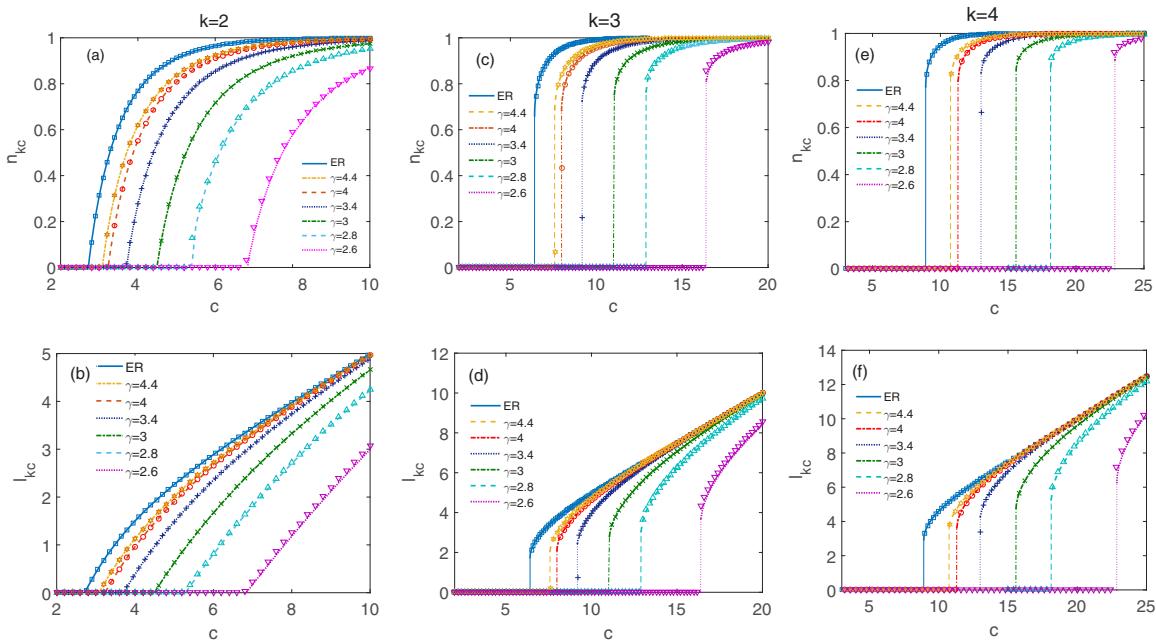


FIG. 5. The relative sizes and the normalized number of edges of the G_k -core for $k = 2, 3, 4$. The points are the results of numerical simulation for the ER and asymptotically scale-free networks of size $N = 10^6$, averaged over 10 realizations. The lines are analytical results obtained from Eqs. (5) and (6). As γ approaches 2, finite size effects become more important, and a deviation between theoretical results and simulations is observed.

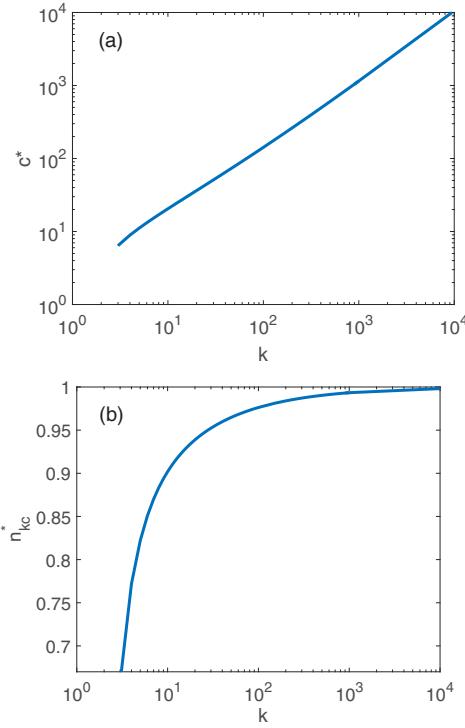


FIG. 6. The behavior of (a) the transition point c^* and (b) the size of the G_k -core at the transition point, n_{kc}^* , vs k for ER random networks.

relation between α and β probabilities as $\beta = 1 - e^{-c\alpha}$, which is independent of the value of k . For ER networks with Poisson degree distributions, one can write a closed form for α and n_{kc} from Eqs. (1) and (2):

$$\alpha = e^{-c\alpha} \frac{\Gamma[k-1, c(e^{-c\alpha}-\alpha)]}{(k-2)!},$$

$$n_{kc} = e^{-c\alpha} \left\{ 1 - \frac{\Gamma[k, c(e^{-c\alpha}-\alpha)]}{(k-1)!} \right\}, \quad (7)$$

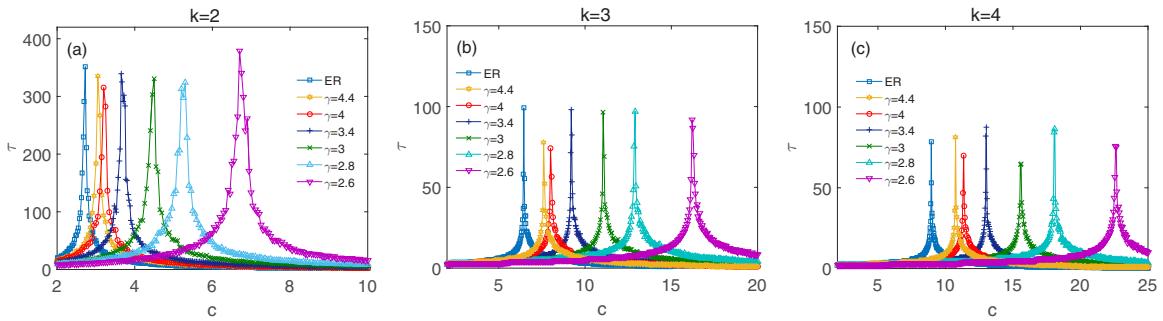


FIG. 7. Total number of pruning steps τ vs mean degree c . The curves shows diverging of $\tau(c)$ at the emergence point of (a) G_2 -core, (b) G_3 -core, and (c) G_4 -core for the ER and asymptotically scale-free networks of size $N = 10^6$, averaged over 10 realizations.

where $\Gamma(s, x)$ is the upper incomplete Γ function. The relative size and the normalized number of edges of the G_k -core for $k = 2, 3$, and 4 are shown in Fig. 5. The analytic results (curves) are compared with numerical simulations (symbols). As we can see in the figure, in contrast to the ordinary core ($k = 2$), for $k \geq 3$ a G_k -core emerges discontinuously at the percolation threshold.

Equations (7) enable us to obtain the transition point c^* and the size of the G_k -core at the transition point, n_{kc}^* , for each k . From numerical data, we estimate the asymptotic representations for c^* and n_{kc}^* as the following:

$$c^* \approx k + C\sqrt{k} \ln \ln k,$$

$$n_{kc}^* \approx 1 - \frac{1}{C\sqrt{k} \ln \ln k}, \quad (8)$$

$$C = 2.413\dots$$

Figure 6 shows the behavior of c^* and n_{kc}^* , in which the curves asymptotically coincide to Eqs. (8).

Next we consider scale-free networks. It was shown that for the purely power-law scale-free networks the ordinary core does not exist [20]. Hence we consider the asymptotically scale-free, uncorrelated networks generated by the static model with the degree distribution $P(q) = [\frac{c(\gamma-2)}{2(\gamma-1)}]^{q-1} \Gamma[q-\gamma+1, \frac{c(\gamma-2)}{2(\gamma-1)}]/\Gamma(q+1) \cong q^{-\gamma}$, where $\Gamma(s)$ is the Γ function [22,23]. For this degree distribution the generating function is $G(x) = (\gamma-1)E_n[(1-x)\frac{c(\gamma-2)}{2(\gamma-1)}]$, where $E_n(x) = \int_1^\infty dy e^{-xy} y^{-n}$ is the exponential integral. Figure 5 shows the relative size and the normalized number of the general 2-, 3-, and 4-cores for different values of γ versus c . With decreasing γ , the emergence point is shifted towards higher values of c . For scale-free networks when $\gamma \rightarrow 2$, finite-size effects become more significant. By imposing the proper degree cutoffs, one can eliminate the finite-size effects and the intrinsic degree correlations [24,25]. In Fig. 5 we compare the emergence of cores for asymptotically scale-free and ER networks. As one can see, the dependence of the cores on c for these networks is similar and, as expected, the curves with larger γ approach the result for ER networks.

We define pruning time steps in a way that enables us to classify the vertices of the network into a set of layers for a given k . At time step $t' = 1$, we select the vertices of

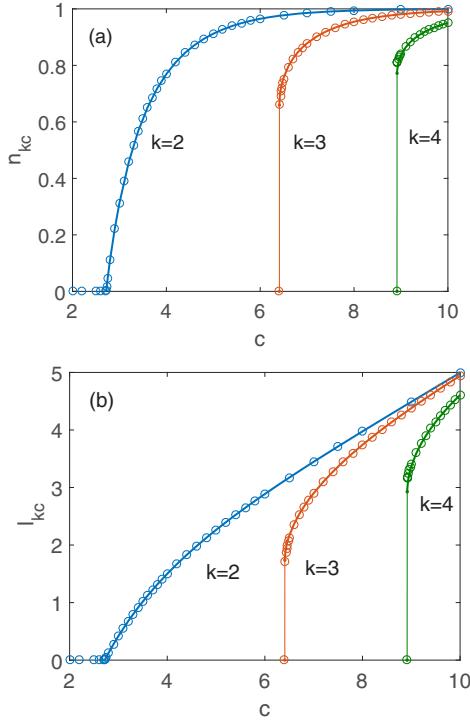


FIG. 8. (a) The relative sizes and (b) normalized number of edges of the G_k -core in ER networks for $k = 2, 3$, and 4 , vs the mean degree c of the network. The points show the results obtained by the rate equation approach and lines show the results obtained using the formalism of Sec. II [Eqs. (5) and (6)].

degree less than k (k -leaves) and remove these vertices and their neighbors by applying the k -leaf algorithm. Removing the vertices in the first step may produce new k -leaves, which will be removed at $t' = 2$ and so on. The vertices removed

at each step t' form a layer of the network. In other words, the network is pruned layer by layer until there is no k -leaf left. We denote the total number of pruning steps as τ so that $t' = \{1, 2, \dots, \tau\}$. After τ steps, the network consists of finite components or a giant G_k -core. For different networks we obtain $\tau(c)$ using numerical simulation; see Fig. 7. As we can see, the dependencies $\tau(c)$ diverge at the birth points of the cores.

III. RATE EQUATIONS

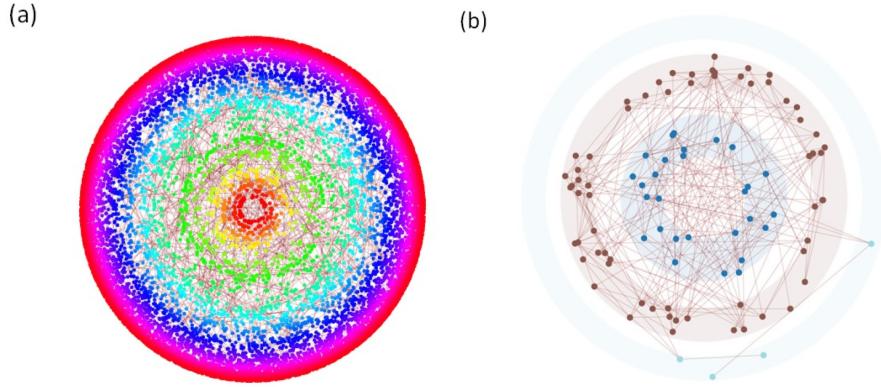
The structural evolution of the network during pruning processes is described by the so-called rate equations for the degree distribution of the remaining network [15,18]. Here we derive rate equations for the k -leaf removal algorithm. Let us consider a network of N vertices and L edges. For simplicity we remove only the edges during the pruning process. In other words, at each time step t we choose randomly a k -leaf and remove all k edges incident to it, together with all edges incident to its k neighbors. In this way, the number of vertices of the network remain constant. Note that the time steps t differ from t' . The algorithm is iterated until $P(q) = 0$ for all $q < k$. The important point of this approach is that the dynamics is self-averaging in the thermodynamic limit: $N \rightarrow \infty$. After a certain number of time steps, almost all random networks have the same degree distribution, which is independent of the (random) order of the removal of leaves [10]. Hence this approach can be used as a confirmation of the results obtained in the previous section.

We introduce the rescaled time $t = \frac{T}{N}$, where T is the total number of steps of the pruning algorithm, so $\Delta t = 1/N$ is the rescaled time of one iteration. Let $N(q, t)$ be the average number of vertices with degree q at time t . Since the total number of vertices is constant, i.e., $N(t) = N$, we have $N(q, t) = NP(q, t)$. We can write the change of $N(q, t + \Delta t) - N(q, t)$ after one iteration. In the large network limit, we can pass from the discrete difference to the time derivative of the degree distribution and obtain the following evolution equation for the degree distribution:

$$\begin{aligned} N(q, t + \Delta t) - N(q, t) &= \dot{P}(q, t) \\ &= -\frac{\theta(k-q)P(q, t)}{\sum_q \theta(k-q)P(q, t)} + \delta_{q,0} \left[1 + \frac{\sum_q q\theta(k-q)P(q, t)}{\sum_q \theta(k-q)P(q, t)} \right] - \frac{\sum_q q\theta(k-q)P(q, t)}{\sum_q \theta(k-q)P(q, t)} \frac{qP(q, t)}{\langle q \rangle_t} \\ &\quad + \frac{\sum_q q\theta(k-q)P(q, t)}{\sum_q \theta(k-q)P(q, t)} \frac{\sum_q q(q-1)P(q, t)}{\sum_q qP(q, t)} \left[\frac{(q+1)P(q+1, t) - qP(q, t)}{\langle q \rangle_t} \right]. \end{aligned} \quad (9)$$

Let us explain different terms on the right-hand side of Eq. (9). First, we choose a random vertex of degree less than k and remove all edges incident to it. The probability that a vertex has degree less than k is $\frac{\theta(k-q)P(q, t)}{\sum_q \theta(k-q)P(q, t)}$, where $\theta(i)$ is defined for integers: $\theta(i \geq 0) = 1$ and $\theta(i < 0) = 0$. Thus with this probability, the number of vertices with $q < k$ decreases by 1. This gives the first term. After removing the edges incident to the leaf and all edges incident to its neighbors, the

leaf and all its neighbors become vertices of degree zero. The average number of neighbors of a vertex of degree less than k is $\frac{\sum_q q\theta(k-q)P(q, t)}{\sum_q \theta(k-q)P(q, t)}$. Hence the second term shows the number of vertices whose degrees become zero. On the other hand, the degree distribution of the end vertices of a randomly chosen edge is $\frac{qP(q)}{\langle q \rangle_t}$. When we remove the edges incident to the nearest neighbors of the leaf, the number of vertices of degree q is decreased by the mean degree of the leaf with probability

FIG. 9. Graphical visualization of the G_k -core decomposition of (a) astrophysics and (b) transcriptional regulation networks.

$\frac{qP(q)}{\langle q \rangle}$. Finally the last contribution results from modification of degrees of the second neighbors of the leaf. After removal of all edges incident to the leaf and its nearest neighbors, the number of connections of the second nearest neighbors of the leaf decreases by one. The average number of the second neighbors is equal to the mean degree of the nearest neighbors except one (connection to the leaf), multiplied by the average number of the nearest neighbors of the leaf. Equation (9) is a set of differential equations, describing the evolution of a network during the pruning. For $k = 2$, these equations coincide with the known ones [18]. Solving Eq. (9) iteratively, we can obtain the degree distribution of the network at each time step t .

As we already mentioned, we do not remove the vertices during the leaf removal algorithm, and so the total number of the vertices remains constant. However, at each time step all edges incident to the leaf and the edges incident to all its nearest neighbors are removed. Hence, at each time step the average number of removed edges is equal to the mean number of nearest neighbors multiplied by their mean degree. This results to the following evolution equation for the aver-

age number of remained edges in the network:

$$\frac{\dot{L}(t)}{N} = -\frac{\langle q^2 \rangle_t}{\langle q \rangle_t} \frac{\sum_q q\theta(k-q)P(q,t)}{\sum_q \theta(k-q)P(q,t)}. \quad (10)$$

We apply the leaf removal algorithm to an uncorrelated network with a degree distribution $P(q, t=0)$ and a vertex mean degree equal to c_0 as the initial conditions. For each value of k , the algorithms are iterated until no vertices of degree less than k remain. To find the G_k -core, the algorithm must continue until time t_k^* at which $P(1, t_k^*) = P(2, t_k^*) = \dots = P(k-1, t_k^*) = 0$. Our numerical results for different networks show that $P(1, t)$ is the last probability to become zero; that is, the vertices of degree 1 disappear after all other leaves. This is why during iteration we look at the behavior of $P(1, t)$, and the algorithm stops at time t_k^* for a given k . The remaining subgraph is the G_k -core. For $k = 2$ the algorithm coincides with the ordinary leaf-removal algorithm and the remaining subgraph is the G_2 -core or simply the core. After we find t_k^* , the size and the number of edges of the G_k -core

TABLE I. G_k -core decomposition of real networks with the number of vertices N and the number of edges L . k_{\max} is the label of the innermost core. $n_{k_{\max}-\text{core}}$ and $n_{2-\text{core}}$ show the relative size of the innermost and outermost cores, respectively. Similarly, $l_{k_{\max}-\text{core}}$ and $l_{2-\text{core}}$ show the normalized number of edges of the innermost and outermost cores, respectively.

Name	N	L	Ref.	k_{\max}	$n_{k_{\max}-\text{core}}$	$l_{k_{\max}-\text{core}}$	$n_{2-\text{core}}$	$l_{2-\text{core}}$
<i>E. coli</i> , transcription	97	212	[26]	3	0.319	0.793	0.917	2.051
AS Oregon	6474	12 572	[27]	2	0.001	0.001	0.001	0.001
Astrophysics	16 046	121 251	[28]	31	0.002	0.045	0.769	5.980
<i>C. elegans</i> , neural	297	2148	[29]	3	0.885	6.037	0.915	6.447
Cond-Mat	16 264	47 594	[28]	10	0.006	0.003	0.618	1.884
Dolphins	62	159	[30]	3	0.161	0.290	0.645	1.322
Email-Enron	36 692	183 831	[31]	7	0.0004	0.001	0.389	1.052
Linux	30 834	213 217	[32]	5	0.0003	0.0008	0.147	0.375
petster-friendship-hamster	1858	12 534	[32]	8	0.010	0.047	0.584	2.664
Sociopatterns-Infectious	410	2765	[33]	9	0.056	0.443	0.912	6.090
PGPgiantcompo	10 680	24 316	[34]	17	0.001	0.014	0.158	0.483
US Air Transportation	500	2980	[35]	3	0.008	0.012	0.260	0.494
Yeast-protein	2284	6646	[36]	3	0.003	0.011	0.025	0.052

can be obtained from the following relations:

$$N_{kc} = N[1 - P(0, t_k^*)], \quad (11)$$

$$L_{kc} = L(t_k^*). \quad (12)$$

We apply this approach to the ER random graphs. The Poisson degree distribution rapidly decays, and it is sufficient to consider $q_{\max} = 30$, i.e., we solve the set of the first 31 equations. Figure 8 shows the size and the number of edges calculated from Eqs. (9)–(12) for the ER networks. In this figure we compare the results obtained by solving the rate equations (points) with the analytic results of the previous section (lines) for the general 2-, 3-, and 4-cores.

IV. REAL-WORLD NETWORKS

We apply the k -leaf removal to a number of real-world networks and find cores of these networks. The characteristics of real-world networks, analyzed in the paper, are listed in Table I. The outermost core is the largest core, which corresponds to $k = 2$ and includes other cores. As we increase the value of k , the size of cores is decreased, and the core corresponding to maximum k (k_{\max}) is the smallest and innermost core. We present the relative size and number of edges of the outermost and innermost Gk -cores in Table I. We find that many real social networks are decomposed to a large hierarchy of the Gk -cores. For instance, the layers of arXiv networks, e.g., cond-mat, astro-ph, or hep-th, have the highest numbers of the Gk -cores nested into each other among networks analyzed in this paper. In contrast, the food webs and biological networks have a small number of cores. Using the visualization algorithm proposed in Ref. [1], visualization of the astrophysics network in 2005 [28] and transcriptional regulation network [26] are presented as two examples in Fig. 9. The regulation network has a few cores, while the astrophysics network has around 30 cores in our proposed network decomposition scheme. Comparing with the random networks, the real networks have more cores. Similarly to what was found in the ordinary core problem, this difference reveals that other structural features such as correlations and

clustering may be significant for the sizes and organization of the Gk -cores.

V. CONCLUSION

In this work we have generalized the ordinary core subgraph to the Gk -cores. We proposed the k -leaf removal algorithm as a generalization of the ordinary leaf removal. The k -leaf pruning algorithm enables us to decompose large random networks into a hierarchical set of progressively nested subgraphs which we called the Gk -cores. Our approach can also be considered as a generalization of the ordinary k -core decomposition. In our pruning at each time step, not only the vertices of degree less than k but also their nearest neighbors are removed. Following this pruning, the network is decomposed into a hierarchy of progressively nested Gk -cores such that the vertices, belonging to the inner cores, and also their first neighbors are of higher degree and well connected. Using the generating function technique, we found the structural characteristics and the emergence point of the Gk -cores for the Erdős-Rényi and scale-free random networks. Similarly to the ordinary k -core percolation, Gk -cores show a discontinuous phase transition for $k \geq 3$. We compared our results with numerical simulations and observed a complete agreement. In addition, we used the rate equation approach to describe the evolution of degree distribution of random networks during the k -leaf pruning algorithm. We checked that the result of the application of this approach to the ER graph completely coincides with the exact result obtained by the analytical calculations. We have applied the k -leaf removal algorithm to a number of real-world networks. Among the real networks explored, the social networks have a large k_{\max} .

We emphasize that in contrast to the k -core decomposition, the Gk -cores are not about the classification of vertices in a network according to their properties but rather about the characterization of a specific robustness of this network. Suppose that a network is attacked by a virus infecting and removing weak vertices (of degree less than k) and their nearest neighbors. The Gk -cores show what will remain of the network after this epidemic. The resilience and robustness of a network against this kind of epidemic is characterized by the size of its Gk -core. This may explain why the social networks that we explored have a large k_{\max} .

-
- [1] J. I. Alvarez-Hamelin, L. Dell'Asta, A. Barrat, and A. Vespignani, k -core decomposition of internet graphs: Hierarchies, self-similarity and measurement biases, [arXiv:cs/0511007](https://arxiv.org/abs/cs/0511007) (2005).
 - [2] S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, and E. Shir, A model of internet topology using k -shell decomposition, *Proc. Natl. Acad. Sci. U S A* **104**, 11150 (2007).
 - [3] S. B. Seidman, Network structure and minimum degree, *Social Netw.* **5**, 269 (1983).
 - [4] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, Identification of influential spreaders in complex networks, *Nat. Phys.* **6**, 888 (2010).
 - [5] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, k -Core Organization of Complex Networks, *Phys. Rev. Lett.* **96**, 040601 (2006).
 - [6] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, Critical phenomena in complex networks, *Rev. Mod. Phys.* **80**, 1275 (2008).
 - [7] N. K. Panduranga, J. Gao, X. Yuan, H. E. Stanley, and S. Havlin, Generalized model for k -core percolation and interdependent networks, *Phys. Rev. E* **96**, 032317 (2017).
 - [8] N. Azimi-Tafreshi, J. Gómez-Gardeñes, and S. N. Dorogovtsev, k -core percolation on multiplex networks, *Phys. Rev. E* **90**, 032816 (2014).
 - [9] R. M. Karp and M. Sipser, Maximum matching in sparse random graphs, in *Proceedings of the 22nd Annual IEEE Symposium on Foundations of Computer Science* (IEEE, New York, 1981), p. 36.

- sium on Foundations of Computer Science, Nashville, TN, USA (IEEE, Piscataway, NJ, 1981), pp. 364–375.
- [10] M. Bauer and O. Golinelli, Core percolation in random graphs: A critical phenomena analysis, *Eur. Phys. J. B* **24**, 339 (2001).
- [11] M. Bauer and O. Golinelli, Exactly Solvable Model with Two Conductor-Insulator Transitions Driven by Impurities, *Phys. Rev. Lett.* **86**, 2621 (2001).
- [12] M. Weigt and A. K. Hartmann, Number of Guards Needed by a Museum: A Phase Transition in Vertex Covering of Random Graphs, *Phys. Rev. Lett.* **84**, 6118 (2000).
- [13] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, Controllability of complex networks, *Nature (London)* **473**, 167 (2011).
- [14] T. Jia, and M. Pósfai, Connecting Core Percolation and Controllability of Complex Networks, *Sci. Rep.* **4**, 5379 (2014).
- [15] M. Weigt and A. K. Hartmann, *Phase Transitions in Combinatorial Optimization Problems* (Wiley-VCH, Weinheim, 2005).
- [16] J. H. Zhao, Y. Habibulla, and H. J. Zhou, Statistical mechanics of the minimum dominating set problem, *J. Stat. Phys.* **159**, 1154 (2015).
- [17] J. H. Zhao, H. J. Zhou, and Y. Y. Liu, Inducing effect on the percolation transition in complex networks, *Nat. Commun.* **4**, 2412 (2013).
- [18] M. Weigt, Dynamics of heuristic optimization algorithms on random graphs, *Eur. Phys. J. B* **28**, 369 (2002).
- [19] N. Azimi-Tafreshi, S. N. Dorogovtsev, and J. F. F. Mendes, Core organization of directed complex networks, *Phys. Rev. E* **87**, 032815 (2013).
- [20] Y.-Y. Liu, E. Csóka, H. Zhou, and M. Pósfai, Core Percolation on Complex Networks, *Phys. Rev. Lett.* **109**, 205703 (2012).
- [21] 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).
- [22] M. Catanzaro and R. Pastor-Satorras, Analytic solution of a static scale-free network model, *Eur. Phys. J. B* **44**, 241 (2005).
- [23] K.-I. Goh, B. Kahng, and D. Kim, Universal Behavior of Load Distribution in Scale-Free Networks, *Phys. Rev. Lett.* **87**, 278701 (2001).
- [24] M. Boguñá, R. Pastor-Satorras, and A. Vespignani, Cut-offs and finite size effects in scale-free networks, *Eur. Phys. J. B* **38**, 205 (2004).
- [25] J.-S. Lee, K.-I. Goh, B. Kahng, and D. Kim, Intrinsic degree-correlations in the static model of scale-free networks, *Eur. Phys. J. B* **49**, 231 (2006).
- [26] S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon, Network motifs in the transcriptional regulation network of *Escherichia coli*, *Nat. Genet.* **31**, 64 (2002).
- [27] J. Leskovec, J. Kleinberg, and C. Faloutsos, Graphs over time: Densification laws, shrinking diameters and possible explanations, in *KDD '05: Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, Chicago, Illinois, USA* (ACM New York, NY, USA, 2005), pp. 177–187.
- [28] M. E. J. Newman, The structure of scientific collaboration networks, *Proc. Natl. Acad. Sci. U.S.A.* **98**, 404 (2001).
- [29] J. D. Watts, and S. H. Strogatz, Collective dynamics of small-world networks, *Nature (London)* **393**, 440 (1998).
- [30] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations, *Behav. Ecol. Sociobiol.* **54**, 396 (2003).
- [31] M. W. Mahoney, A. Dasgupta, K. J. Lang, and J. Leskovec, Community structure in large networks: natural cluster sizes and the absence of large well-defined clusters, *Internet Math.* **6**, 29 (2009).
- [32] J. Kunegis, KONECT: The Koblenz Network Collection, in *WWW'13 Companion Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil* (ACM New York, NY, USA, 2013), p. 1343.
- [33] L. Isella, J. Stehlé, A. Barrat, C. Cattuto, J.-F. Pinton, and W. Van den Broeck, What's in a crowd? Analysis of face-to-face behavioral networks, *J. Theor. Biol.* **271**, 166 (2011).
- [34] M. Boguñá, R. Pastor-Satorras, A. Diáz-Guilera, and A. Arenas, Models of social networks based on social distance attachment, *Phys. Rev. E* **70**, 056122 (2004).
- [35] V. Colizza, R. Pastor-Satorras, and A. Vespignani, Reaction-diffusion processes and metapopulation models in heterogeneous networks, *Nat. Phys.* **3**, 276 (2007).
- [36] D. Bu, Y. Zhao, L. Cai, H. Xue, X. Zhu, H. Lu, J. Zhang, S. Sun, L. Ling, N. Zhang, G. Li, and R. Chen, Topological structure analysis of the protein-protein interaction network in budding yeast, *Nucleic Acids Res.* **31**, 2443 (2003).

Chapter Four

Observability transition in multiplex networks

Contribution

As main author in this paper I carried out both theory and simulation of the research. In the theoretical part the message-passing equations governing the observability transition were derived for both real and random multiplex networks and for both cases of the multiplex networks with and without link-overlap. The writing and editing part of the manuscript was done in collaboration with Prof. Radicchi.



Observability transition in multiplex networks

Saeed Osat^a, Filippo Radicchi^{b,*}

^a Quantum Complexity Science Initiative, Skolkovo Institute of Science and Technology, Skoltech Building 3, Moscow 143026, Russia

^b Center for Complex Networks and Systems Research, School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN 47408, USA



HIGHLIGHTS

- We extend the observability model from isolated to multilayer networks.
- We introduce a theory able to characterize the observability phase diagram.
- We perform a systematic analysis of several real-world multiplex networks.

ARTICLE INFO

Article history:

Received 25 September 2017

Received in revised form 17 November 2017

Available online 9 March 2018

Keywords:

phase transition

Multiplex networks

Observability

Message passing

ABSTRACT

We extend the observability model to multiplex networks composed of two network layers. We present mathematical frameworks, valid under the treelike ansatz, able to describe the emergence of the macroscopic cluster of mutually observable nodes in both synthetic and real-world multiplex networks. We show that the observability transition in synthetic multiplex networks is discontinuous. In real-world multiplex networks instead, edge overlap among layers is responsible for the disappearance of any sign of abruptness in the emergence of the macroscopic cluster of mutually observable nodes.

© 2018 Published by Elsevier B.V.

1. Introduction

Complex systems where elementary units have different types of interactions can be conveniently modeled as multiplex networks [1–3]. This is a very generic representation, where elements of a system are organized in multiple network layers, each standing for a specific color or flavor of interaction. Systems that can be represented in this way are abundant in the real world. Examples include, among others, social networks sharing the same actors [4,5], and multimodal transportation graphs sharing common geographical locations [6,7].

Several analyses of multiplex networks have been performed recently [1–3]. A common result, shared by the vast majority of these studies, is that a process defined on a multiplex network is characterized by features radically different from those observed for the same type of process when this is applied to an isolated network. Examples regard dynamical processes taking place on multiplex networks, such as diffusion [8,9], epidemic spreading [10–13], synchronization [14], and controllability [15]. Examples include also structural processes as those typically framed in terms of percolation models. For instance in their seminal paper, Buldyrev et al. considered a site-percolation model aimed at understanding the role of interdependencies among two network layers [16]. The macroscopic cluster of mutually connected nodes in a multiplex network emerges discontinuously, at odds with what instead observed for the same process on an isolated network where the percolation transition is always continuous. A large number of subsequent studies have then analyzed in detail the

* Corresponding author.

E-mail address: filiradi@indiana.edu (F. Radicchi).

features of the percolation transition in multiplex networks [17–28]. Several variants of the percolation model have been also considered, including redundant site percolation [29], k-core percolation [30], weak percolation [31], bond percolation [32], and optimal percolation [33].

In this paper, we focus our attention on an additional variant of the percolation model usually named as the observability model [34,35]. In isolated networks, the model finds its motivation in the study of some dynamical processes where the state of the system can be determined by monitoring or dominating the states of a limited number of nodes in the network [36]. Examples include, among others, real-time monitoring of power-grid networks [34], and mobile ad-hoc networks [37]. The observability model has been considered in synthetic models [34] and real-world topologies [35]. In the simplest version of the model, every node in the network can host an observer with probability ϕ . Placing an observer on one node can make the node itself and all its nearest neighbors observable. Nodes in the network can therefore assume three different states: (i) directly observable, if hosting an observer; (ii) indirectly observable, if being the first neighbor of an observer; (iii) or not observable, otherwise. Observable, either directly or indirectly, nearest-neighbor nodes form clusters of connected observable nodes. As in the case of percolation, the question of interest in network observability is understanding the macroscopic formation of observable clusters in the network on the basis of microscopic changes in the state of its individual nodes. In synthetic infinite graphs, the macroscopic cluster of observable nodes obeys a continuous phase transition as a function of the probability ϕ , and the critical threshold is generally very small [34]. In most real-world networks also, the largest cluster grows smoothly, and the transition point is generally very close to zero [35].

In our extension of the observability model to multiplex networks, we focus our attention on the emergence of clusters of mutually observable nodes. The definition of these clusters is a straightforward combination of the notions of clusters of observable nodes in isolated networks [34] and clusters of mutually connected nodes in multiplex networks [16]. The extension of the observability model to multiplex networks finds its rationale in any situation where the goal is monitoring the structure or controlling the dynamics of a system structured in multiple layers of interactions. A genuine example could be understanding the process of partial reconstruction of the structure of two coupled social media, such as Facebook and Twitter, from random sampling. Actors are shared by the two social networks. Observing an individual means getting full access to her/his account thus being able to get information about the identity of her/his friends on both platforms. If one does not have any prior knowledge of the connections among nodes in the system, but has access only to the list of all its nodes, a naive way to reconstruct part of the topology of the multiplex would be to randomly sample a fraction of nodes, and aggregate structural information from their egocentric graphs [38]. Clusters of mutually observable nodes in this situation would correspond to mutually connected portions of the system that emerge from such a sampling strategy.

The paper is organized as follows. In Section 2, we define the observability model in multiplex networks. In Section 3, we study the model in ensembles of multiplex networks whose layers are generated according to the configuration model. In Section 4, we introduce a message-passing method to deal with the observability model in multiplex networks with specified topology, such as real-world multiplex networks. The framework described in this section is based on the assumption that the number of edges shared by the different layers of the multiplex is negligible. The message-passing method valid for multiplex networks in presence of edge overlap is much more cumbersome, and, for this reason, presented only in the Appendices. Finally, in Section 5, we summarize the main findings of the paper.

2. Observability model in multiplex networks

We consider a multiplex network composed of N nodes structured in two layers, namely α and β . Node labels take integer values from 1 to N in both layers. Nodes with identical labels correspond to copies (or replicas) of the same individual or unit in the two layers. The observability model we consider here is a natural extension to multiplex networks of the same model already considered on isolated networks [34,35]. Observers or sensors are placed at random with probability ϕ on every node in the system. If an observer is placed on a node i , the node i is directly observable in both layers. A node i , that is not directly observable, but is attached to at least one directly observable node j in layer α and at least one directly observable node k in layer β , with j not necessarily equal to k , is indirectly observable. We focus our attention on mutually observable clusters of nodes. The way these clusters are defined is identical to the one in which clusters of mutually connected nodes are defined in site percolation [16]. The only difference comes from the fact that a node in order to be “occupied” can be either in the directly or indirectly observable state. Note that mutually connected and mutually observable clusters coincide for $\phi = 1$. In particular, two nodes belong to the same cluster of mutually observable nodes if they are observed, either directly or indirectly, and they are connected by at least by one path (internal to the cluster itself) in both layers. Our model can be seen as a depth-one percolation model on a multiplex network. As we are interested in understanding the extent of the system that can be monitored by placing random points of observation, the focus of our analysis is centered on quantifying how the size of the Largest Mutually Observable Cluster (LMOC) changes as a function of the microscopic probability ϕ of nodes to be directly observable. In the following, we discuss the details of how the problem can be solved exactly under two conditions: (i) absence of link overlap among layers, and (ii) validity of the locally treelike ansatz. In the Appendix A, we report the full mathematical framework valid when condition (i) is removed.

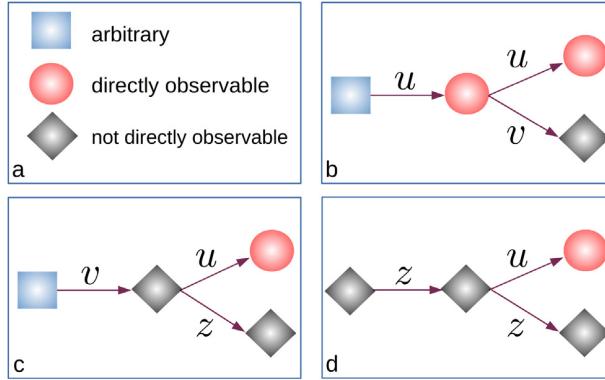


Fig. 1. (a) The different variables used in the single-layer equations are defined depending on the state of the nodes, here denoted by different shapes and colors. Schematic illustrations for the derivation of the equations for the probabilities u (panel b), v (panel c) and z (panel d).

3. Observability transition in random multiplex networks

Let us consider the case of a multiplex composed of two network layers generated independently according to the configuration model [39]. If the layers are sufficiently sparse, the fact that the layers are generated independently allow us to consider the overlap among layers (i.e., the simultaneous existence of the same edge in both layers happens with vanishing probability) negligible. The only inputs required to study such a system are the degree distributions of the two layers, namely $P^{[\alpha]}(k)$ and $P^{[\beta]}(k)$.

The analytic treatment for this special types of networks is similar to one presented in Ref. [34] for isolated networks. To generate a self-consistent set of equations able to describe how the LMOC changes as a function of ϕ in this special type of multiplex networks, we define the following conditional probabilities valid for a randomly selected edge in layer α (see Fig. 1):

1. $u^{[\alpha]}$, that is the probability to reach the LMOC if we start from a node in arbitrary state and we follow one of its edges having a node in the directly observable state on the other end of the edge.
2. $v^{[\alpha]}$, that is the probability to reach the LMOC if we start from a node in arbitrary state and we follow one of its edges having, on the other end of the edge, a node that is not directly observable.
3. $z^{[\alpha]}$, that is the probability to reach the LMOC if we start from a node that is not directly observable and we follow one of its edges having, on the other end of the edge, a node that is also not directly observable.

The same exact definitions are valid for the conditional probabilities $u^{[\beta]}$, $v^{[\beta]}$, and $z^{[\beta]}$ for layer β .

As mentioned above, a generic node is part of the LMOC if observable, either directly or indirectly, and attached to at least one other node that is part of the LMOC. The way this happens depends however on the state of the node. We make therefore a distinction between (i) a directly observable node and (ii) a not directly observable node.

Let us consider first case (i). If one of the nodes at the end of a generic edge is directly observable, the probability that the node is not connected to the LMOC is $\phi(1 - u^{[\alpha]})$ if the other node is directly observable, or $(1 - \phi)(1 - v^{[\alpha]})$ if the other node is not directly observable. If the node we are considering has degree k , then the probability that the node is connected to the LMOC in layer α is

$$q_k^{[\alpha]} = 1 - \sum_{m=0}^k \binom{k}{m} [\phi(1 - u^{[\alpha]})]^m [(1 - \phi)(1 - v^{[\alpha]})]^{k-m}. \quad (1)$$

The sum on the r.h.s. of the equation above quantifies the probability that none of the neighbors of a node with degree k is part of the LMOC. This probability is then discounted from 1 to estimate the probability that at least one neighbor of the node is part of the LMOC. If we consider all nodes, we have that

$$q^{[\alpha]} = \sum_k P^{[\alpha]}(k) q_k^{[\alpha]} = 1 - G_0^{[\alpha]} (1 - \phi u^{[\alpha]} - (1 - \phi)v^{[\alpha]}), \quad (2)$$

where $G_0^{[\alpha]}(x) = \sum_k P^{[\alpha]}(k)x^k$ is the generating function of the degree distribution for layer α . We can repeat the same exact arguments for layer β . In particular, the probability that a generic node is part of the LMOC is given by the product that the node is attached to the LMOC simultaneously in both layers, that is

$$q = \phi q^{[\alpha]} q^{[\beta]}, \quad (3)$$

where the extra factor ϕ comes from the fact that we are considering the case of a node that is directly observable.

For case (ii), we can proceed in a similar way as above. If our node is not directly observable, and we select one of its edges at random, then the probability that this node is not connected to the LMOC is $\phi(1 - u^{[\alpha]})$ if the other node at the end of the edge is directly observable, or $(1 - \phi)(1 - z^{[\alpha]})$ if the other node is not directly observable. If the node we are considering has degree k , the probability that this node is connected to the LMOC in layer α is

$$\begin{aligned} r_k^{[\alpha]} = & 1 - (1 - \phi)^k + \\ & - \sum_{m=1}^k \binom{k}{m} [\phi(1 - u^{[\alpha]})]^m [(1 - \phi)(1 - z^{[\alpha]})]^{k-m}. \end{aligned} \quad (4)$$

The term $(1 - \phi)^k$ comes from the fact that if none of the nodes at the end of the k edges departing from our node are directly observable, then necessarily our node will be not indirectly observable and thus surely out of any cluster. The remaining part of the r.h.s. instead quantifies the probability that none of the neighbors of our node is part of the LMOC, assuming that at least one of them is directly observable. Both these probabilities are discounted from 1 to compute the probability that our indirectly observable node is part of the LMOC. If we consider all nodes, we have that

$$\begin{aligned} r^{[\alpha]} = & \sum_k P^{[\alpha]}(k) r_k^{[\alpha]} = \\ & 1 - G_0^{[\alpha]} (1 - \phi u^{[\alpha]} - (1 - \phi) z^{[\alpha]}) \\ & + G_0^{[\alpha]} ((1 - \phi)(1 - z^{[\alpha]})) - G_0^{[\alpha]} (1 - \phi). \end{aligned} \quad (5)$$

For layer β , the arguments are identical. The probability that a generic node that is not directly observable is part of the LMOC is given by the product that the node is attached to the LMOC simultaneously in both layers, that is

$$r = (1 - \phi) r^{[\alpha]} r^{[\beta]}, \quad (6)$$

where the extra factor $1 - \phi$ comes from the fact that we are considering the case of a node that is not directly observable. Combining cases (i) and (ii) together, we can finally write

$$P_\infty = q + r, \quad (7)$$

for the average size of the LMOC. To compute Eq. (7), we still require a way to estimate properly the conditional probabilities $u^{[\alpha]}$, $u^{[\beta]}$, $v^{[\alpha]}$, $v^{[\beta]}$, $z^{[\alpha]}$, and $z^{[\beta]}$. Following a similar approach as the one described above, we have that

$$\begin{aligned} u^{[\alpha]} = & [1 - G_1^{[\alpha]} (1 - \phi u^{[\alpha]} - (1 - \phi) v^{[\alpha]})] \times \\ & [1 - G_0^{[\beta]} (1 - \phi u^{[\beta]} - (1 - \phi) v^{[\beta]})], \end{aligned} \quad (8)$$

$$\begin{aligned} v^{[\alpha]} = & [1 - G_1^{[\alpha]} (1 - \phi u^{[\alpha]} - (1 - \phi) z^{[\alpha]})] \times \\ & [1 - G_0^{[\beta]} (1 - \phi u^{[\beta]} - (1 - \phi) z^{[\beta]}) + \\ & - G_0^{[\beta]} (1 - \phi) + G_0^{[\beta]} ((1 - \phi)(1 - z^{[\beta]}))], \end{aligned} \quad (9)$$

and

$$\begin{aligned} z^{[\alpha]} = & [1 - G_1^{[\alpha]} (1 - \phi u^{[\alpha]} - (1 - \phi) z^{[\alpha]}) + \\ & - G_1^{[\alpha]} (1 - \phi) + G_1^{[\alpha]} ((1 - \phi)(1 - z^{[\alpha]}))] \times \\ & [1 - G_0^{[\beta]} (1 - \phi u^{[\beta]} - (1 - \phi) z^{[\beta]}) + \\ & - G_0^{[\beta]} (1 - \phi) + G_0^{[\beta]} ((1 - \phi)(1 - z^{[\beta]}))]. \end{aligned} \quad (10)$$

The intuition behind the previous equations is straightforward. When looking at an edge on layer α , the probability that this edge will bring to the LMOC will depend on the generating function of the excess degree distribution of the layer, i.e., $G_1^{[\alpha]}(x) = [d/dx G_0^{[\alpha]}(x)]/[d/dx G_0^{[\alpha]}(1)]$. On the other hand, the probability that a node at the end of this edge is also attached to the LMOC on layer β will depend only on the degree of the node itself on layer β , accounted by the generating function of the degree distribution, namely $G_0^{[\beta]}(x)$. Note that equations for $u^{[\beta]}$, $v^{[\beta]}$, and $z^{[\beta]}$ can be obtained by simply swapping α and β in Eqs. (8), (9), and (10).

At this point, we have everything necessary to estimate Eq. (7). We have first to solve Eqs. (8), (9), and (10), and the analogous ones for layer β , by iteration. Then, we can plug the obtained values of the conditional probabilities for generic edges into the equations for the nodes.

Fig. 2 shows a comparison between the results of direct numerical simulations of the observability model and the numerical solutions of our equations for duplex networks formed by layers obeying Poisson degree distributions, i.e., $P^{[\alpha]}(k) = P^{[\beta]}(k) = \frac{(k)^k e^{-\langle k \rangle}}{\langle k \rangle!}$. The figure provides evidence of a perfect agreement between theory and simulations.

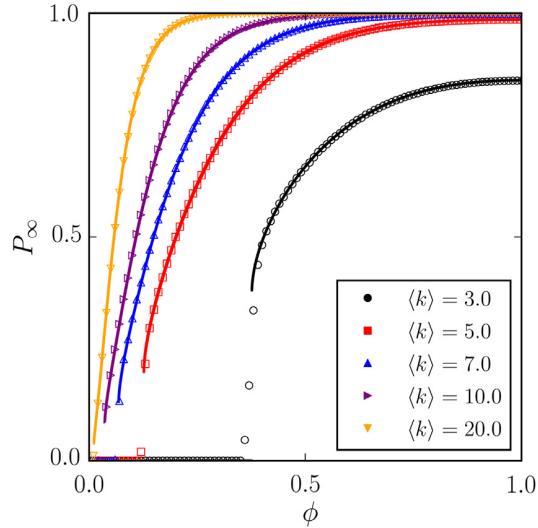


Fig. 2. Observability transition in random Poisson multiplex networks. We consider multiplex networks with layers generated independently and given by realizations of the configuration model with $N = 10,000$ nodes and degree sequence generated according to a Poisson distribution with average degree $\langle k \rangle$. Results of our theoretical method (small symbols) are compared with those of numerical simulations of the observability model (large symbols). Symbols correspond to average values over 100 realizations of the network model and a single simulation of the observability model for each of them. We consider different values of $\langle k \rangle$, $\langle k \rangle = 3.0, 5.0, 7.0, 10.0$, and 20.0 . Values of P_∞ for the various cases drop to zero in the same order if the figure is read from right to left.

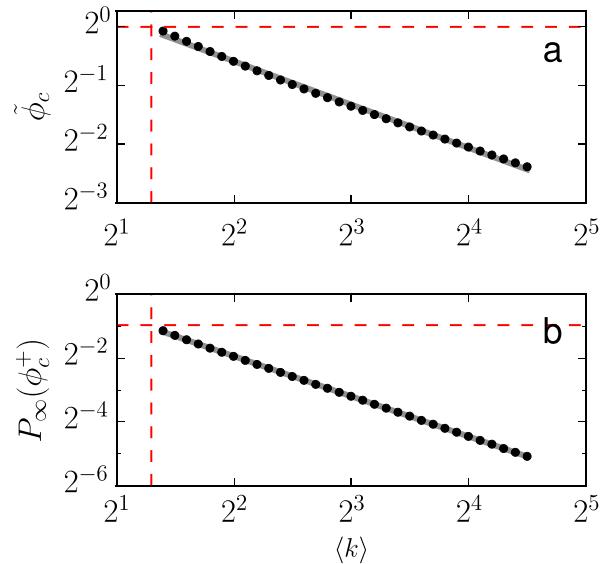


Fig. 3. Observability transition in random Poisson multiplex networks. (a) We plot the rescaled value of the critical threshold $\tilde{\phi}_c = \phi_c + (1 - \phi_c)[1 - (1 - \phi_c)^{\langle k \rangle}]$ (black circles) as a function of the average degree $\langle k \rangle$ for Poisson multiplex networks of infinite size. $\tilde{\phi}_c$ is the effective probability of a generic node to be observed, either directly or indirectly. We find that $\tilde{\phi}_c \propto \langle k \rangle^{-3/4}$ (gray full line). The vertical red line indicates the value $\langle k \rangle_c \simeq 2.4554$ where $\tilde{\phi}_c = 1$ (horizontal red line). (b) Height of the discontinuous jump $P_\infty(\phi_c^+)$ as a function of the average degree $\langle k \rangle$ (black circles) for the same multiplex networks of panel (a). We find that $P_\infty(\phi_c^+) \propto \langle k \rangle^{-7/4}$ (gray full line). The vertical red line indicates the value $\langle k \rangle_c \simeq 2.455$ where $P_\infty(\phi_c^+) \simeq 0.511$ (horizontal red line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

As the generating functions $G_0^{[\alpha]}, G_1^{[\alpha]}, G_0^{[\beta]},$ and $G_1^{[\beta]}$ can be written in a closed form for the Poisson distribution, we can also study the numerical solution of the equations for infinitely large networks (Fig. 3). As long as the average degree $\langle k \rangle$ of the multiplex network model is finite, the transition between the non-observable and the observable phases is always discontinuous, exactly as in the case of the standard percolation model [16]. We note that the system is always in the

non-observable regime for $\langle k \rangle < 2.4554$, i.e., the same value found for the minimum average degree necessary for the emergence of the percolating phase in the standard percolation model [16]. This is expected from the equivalence between the observability model and the standard percolation model for $\phi = 1$. For $\langle k \rangle \geq 2.4554$, the transition point and height of the discontinuous jump in the observability model become different from those typical of site percolation. There, we have that the critical occupation probability p_c and the height of the discontinuous jump $P_\infty(p_c^+)$ decay to zero as a function of the average degree $\langle k \rangle$ as $p_c \propto P_\infty(p_c^+) \propto \langle k \rangle^{-1}$ [16]. For the observability model, we find instead that the rescaled critical threshold $\tilde{\phi}_c = \phi_c + (1 - \phi_c)[1 - (1 - \phi_c)^{\langle k \rangle}]$ does not decrease as $\langle k \rangle^{-1}$ as a naive mapping between the two models would predict. Instead, the scaling is compatible with $\tilde{\phi}_c \propto \langle k \rangle^{-3/4}$ (Fig. 3a). The height of the discontinuous jump $P_\infty(\phi_c^+)$ also does not go to zero as $\langle k \rangle^{-1}$ as predicted in the standard percolation model, but instead as $P_\infty(\phi_c^+) \propto \langle k \rangle^{-7/4}$ (Fig. 3b). On the basis of these numerical findings, we conclude that observability transition cannot be trivially explained in terms of the standard percolation transition.

4. Observability transition in real multiplex networks

In this section, we develop an analytic framework able to approximate the phase diagram of the observability transition for a duplex with given adjacency matrices for the two layers. This information is encoded in the sets $\partial_i^{[\alpha]}$ and $\partial_i^{[\beta]}$ containing the neighbors of node i respectively in layers α and β for every node i . We indicate the sizes of these sets respectively as $k_i^{[\alpha]}$ and $k_i^{[\beta]}$. We make use of two assumptions: (i) layers have null overlap in the sense that set of neighbors of the same node in the two layers have null intersection; (ii) the network is locally treelike.

Suppose we are interested in estimating the probability s_i that node i is part of the LMOC. This will happen if node i is receiving at least one message in layer α and one message in layer β about the belonging to the LMOC. We consider conditional probabilities, or average message values, over the ensemble of random placements of observers in the network. In particular, we define three different messages for every edge $j \rightarrow i$ in layer α dependent on the states of the nodes j and i :

1. $u_{j \rightarrow i}^{[\alpha]}$ is the probability that node j is in the LMOC, irrespective of whether node i is in the LMOC or not, given that node j is directly observable.
2. $v_{j \rightarrow i}^{[\alpha]}$ is the probability that node j is in the LMOC, irrespective of whether node i is in the LMOC or not, given that node j is not directly observable.
3. $z_{j \rightarrow i}^{[\alpha]}$ is the probability that node j is in the LMOC, irrespective of whether node i is in the LMOC or not, given that neither j nor i are directly observable.

$u_{j \rightarrow i}^{[\beta]}$, $v_{j \rightarrow i}^{[\beta]}$, and $z_{j \rightarrow i}^{[\beta]}$ represent the same quantities as above but for layer β . Note that messages traveling on the same edge but in opposite direction are different.

The generic node i is part of the LMOC if one of these two conditions are met: (i) the node is directly observable and attached to at least one other node in layer α and layer β that are part of the LMOC; (ii) the node is indirectly observable and attached to at least one other node in layer α and layer β that are part of the LMOC. The rationale behind the derivation of the following equations is still graphically described by the illustration of Fig. 1.

Considering only layer α , for case (i), we can write

$$q_i^{[\alpha]} = \left[1 - \prod_{j \in \partial_i^{[\alpha]}} (1 - \phi u_{j \rightarrow i}^{[\alpha]} - (1 - \phi) v_{j \rightarrow i}^{[\alpha]}) \right]. \quad (11)$$

For case (ii), we have instead

$$\begin{aligned} r_i^{[\alpha]} = & \left[1 - \prod_{j \in \partial_i^{[\alpha]}} (1 - \phi u_{j \rightarrow i}^{[\alpha]} - (1 - \phi) z_{j \rightarrow i}^{[\alpha]}) + \right. \\ & \left. - (1 - \phi)^{k_i^{[\alpha]}} \left[1 - \prod_{j \in \partial_i^{[\alpha]}} (1 - z_{j \rightarrow i}^{[\alpha]}) \right] \right]. \end{aligned} \quad (12)$$

The same type of equations are valid for layer β . As the node must be part of the LMOC in both layer simultaneously, we can write

$$q_i = \phi q_i^{[\alpha]} q_i^{[\beta]}, \quad (13)$$

and

$$r_i = (1 - \phi) r_i^{[\alpha]} r_i^{[\beta]}, \quad (14)$$

where the factors ϕ and $1 - \phi$ come from the fact that we are considering the cases of a directly observable and a non directly observable node, respectively. The size of the average LMOC is given by

$$P_\infty = \frac{1}{N} \sum_i [q_i + r_i]. \quad (15)$$

The self-consistent equations for the messages are instead given by

$$\begin{aligned} u_{j \rightarrow i}^{[\alpha]} &= \left[1 - \prod_{k \in \partial_j^{[\alpha]} \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) v_{k \rightarrow j}^{[\alpha]}) \right] \times \\ &\quad \left[1 - \prod_{k \in \partial_j^{[\beta]}} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) v_{k \rightarrow j}^{[\beta]}) \right], \end{aligned} \quad (16)$$

$$\begin{aligned} v_{j \rightarrow i}^{[\alpha]} &= \left[1 - \prod_{k \in \partial_j^{[\alpha]} \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] \times \\ &\quad \left[1 - \prod_{k \in \partial_j^{[\beta]}} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) + \right. \\ &\quad \left. - (1 - \phi)^{k_j^{[\beta]}} \left[1 - \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \right] \right], \end{aligned} \quad (17)$$

and

$$\begin{aligned} z_{j \rightarrow i}^{[\alpha]} &= \left[1 - \prod_{k \in \partial_j^{[\alpha]} \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) + \right. \\ &\quad \left. - (1 - \phi)^{k_j^{[\alpha]} - 1} \left[1 - \prod_{k \in \partial_j^{[\alpha]} \setminus i} (1 - z_{k \rightarrow j}^{[\alpha]}) \right] \right] \times \\ &\quad \left[1 - \prod_{k \in \partial_j^{[\beta]}} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) + \right. \\ &\quad \left. - (1 - \phi)^{k_j^{[\beta]}} \left[1 - \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \right] \right], \end{aligned} \quad (18)$$

for a generic edge $j \rightarrow i$ belonging to layer α . Equations with the same structure allow to compute the probabilities defined for edges in layer β . We remark that the equations above make use of the locally treelike approximation, hence the products appearing on their right-hand sides. Further, backtracking terms are excluded in the products.

The analytic framework is now completed. To estimate the average size of the LMOC as a function of ϕ , one needs to: first, solve Eqs. (16), (17), and (18) by iteration; second, plug these solutions into Eqs. (11), and (12); third, estimate in sequence Eqs. (13), (14), and (15).

We performed comparisons between numerical solutions of the framework and results of numerical simulations for a single multiplex networks formed by random scale-free network layers which have negligible overlap (see Fig. 4). The agreement between the two approaches is remarkable.

We stress the fact that the framework presented above is valid under the assumption that the two layers that compose the multiplex do not share any edge. This is a very strong assumption, often violated by real-world networks. For instance, edge overlap may have a dramatic consequence on the properties of the percolation transition [40–42,26,43,28]. We developed a mathematical framework valid in case of edge overlap. Given its length and complexity, we present the details only in the Appendix A.

Indeed, results from the application of the two methods (with or without edge overlap) to real-world multiplex networks provide different scenarios and degree of accuracy. In Fig. 5 for example, we show a comparisons between numerical simulations and numerical solutions of the frameworks when applied to a multiplex transportation network [26]. The framework that accounts for edge overlap well approximates the ground-truth results obtained with numerical simulations. The LMOC grows smoothly for any value of ϕ . The framework that does not account for edge overlap instead tends to underestimate the size of LMOC obtained from numerical simulations. It also provides evidence of an abrupt emergence of the LMOC at $\phi > 0$. Similar considerations are valid also for the multiplex network representing the *C. Elegans* connectome

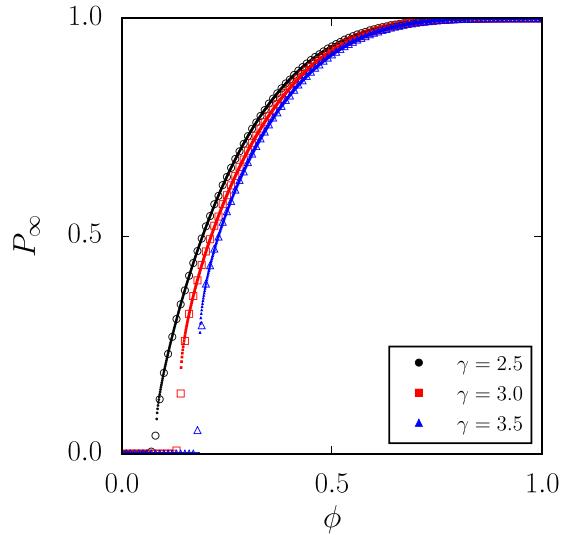


Fig. 4. Observability transition in random scale-free multiplex networks. We consider single instances of multiplex networks with layers generated independently, and given by realizations of the configuration model with $N = 10,000$ nodes and degree sequence generated according to a power-law degree distribution $P^{(\alpha)}(k) = P^{(\beta)}(k) \sim k^{-\gamma}$ with degree exponent γ (minimal and maximal degrees are respectively set $k_{min} = 3$ and maximal degree $k_{max} = \sqrt{N}$, so that edge overlap between layers is negligible). Results of the message-passing framework without overlap (small symbols) are compared with those of numerical simulations of the observability model (large symbols). Results of numerical simulations are obtained on a single instance of the network model and 100 simulations of the observability model. We consider different values of γ , $\gamma = 2.5, 3.0$ and 3.5 . Values of P_∞ for the various cases drop to zero in the same order if the figure is read from right to left.

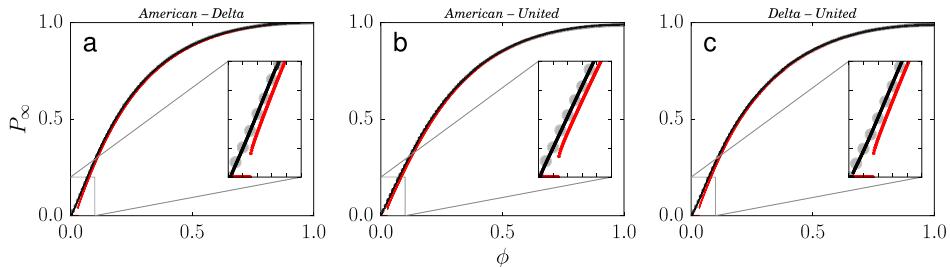


Fig. 5. Observability transition in the US air transportation multiplex network. (a) The system is obtained by combining American Airlines and Delta routes. We consider only US domestic flights operated in January, 2014, and construct the duplex network where airports are nodes and connections on the layers are determined by the existence of at least a flight between the two locations. In the diagram, the gray big circles represent results of numerical simulations, the red small circles stand for results from the framework that does not account for edge overlap, and the black small circles represent results obtained from the mathematical framework that accounts for edge overlap. Gray circles correspond to average values over 10,000 simulations of the observability model. The inset shows a zoom of a specific part of the diagram. (b) Same as in panel (a), but for the combination of American Airlines and United flights. (c) Same as in panel (a), but for the combination of Delta and United flights. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(Fig. 6) [44,45], and all other real-world multiplex networks considered in Appendix E. In Table 1, we summarize results from the analysis of various real-world networks. We considered systems of very different nature, including transportation, biological, and social multiplex networks. Most of the real-world multiplex networks analyzed here appear to be always in observable regime, as the critical value ϕ_c estimated from the theoretical frameworks is very close to zero. This second observation is apparent from the solution of the framework accounting for edge overlap. Instead the framework that does not account for edge overlap seems to provide a distorted vision of the true nature of the phase transition in several cases. Based on this finding, we believe that edge overlap represents the simplest cause of the smoothness of the observability transition in real-world multiplex networks.

5. Conclusions and discussion

In this paper, we extended the observability model, originally considered on isolated networks, to multiplex networks. In particular, we focused our attention on the emergence of the largest cluster of mutually observed nodes as a function

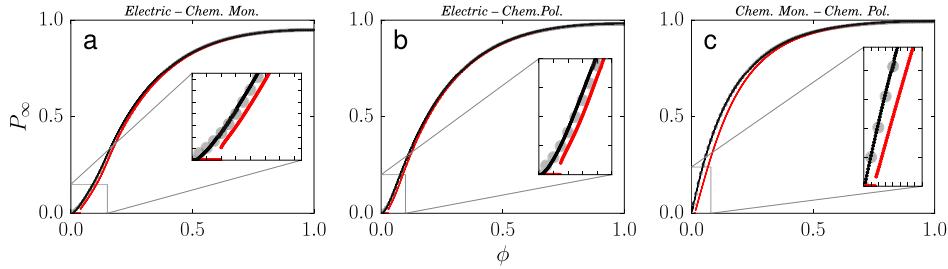


Fig. 6. Observability transition in the *C. Elegans* connectome multiplex network. Edges in different layers represent different types of synaptic junctions among the neurons: electrical, chemical monadic, and chemical polyadic. (a) Analysis of the multiplex obtained by combining together the layers of electrical and chemical monadic interactions. In the diagram, the gray big circles represent results of numerical simulations, the red small circles stand for results from the framework that does not account for edge overlap, and the black small circles represent results obtained from the mathematical framework that accounts for edge overlap. Gray circles correspond to average values over 10,000 simulations of the observability model. The inset shows a zoom of a specific part of the diagram. (b) Same as in panel (a), but for the combination of electrical and chemical polyadic interactions. (c) Same as in panel (a), but for the combination of chemical monadic and polyadic interactions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1

Observability transition in real multiplex networks. From left to right, we report the following information: name of the system and references of the papers where the system has been studied as a multiplex network, name of the layers used to generate the multiplex network, number of nodes in the network, critical probability ϕ_c and height of the jump $P_\infty(\phi_c^+)$ estimated with the Message-Passing (MP) framework that does not account for edge overlap, critical probability ϕ_c and height of the jump $P_\infty(\phi_c^+)$ estimated with the Message-Passing (MP) framework that accounts for edge overlap. Values of the critical probability ϕ_c are computed with precision equal to 0.001, therefore entries in the table showing exactly this numerical value indicate that true values are actually smaller than it.

Network	Layers	N	MP		MP with overlap	
			ϕ_c	$P_\infty(\phi_c)$	ϕ_c	$P_\infty(\phi_c)$
Air Transportation [26]	American Air. – Delta	84	0.030	0.040	0.004	0.005
	American Air. – United	72	0.027	0.035	0.002	0.003
	United – Delta	81	0.033	0.041	0.005	0.007
<i>C. Elegans</i> [44,45]	Electric – Chem. Mon.	226	0.039	0.021	0.006	0.001
	Electric – Chem. Pol.	247	0.029	0.020	0.004	0.001
	Chem. Mon. – Chem. Pol.	257	0.016	0.016	0.002	0.002
Arxiv [46,47]	cond-mat.dis-nn – physics.data-an	916	0.017	0.006	0.001	0.001
	physics.bio-ph – cond-mat.dis-nn	790	0.026	0.003	0.001	0.002
	physics.bio-ph – physics.data-an	564	0.023	0.002	0.001	0.000
	cond-mat.dis-nn – cs.SI	521	0.030	0.005	0.001	0.001
	physics.soc-ph – physics.data-an	50	0.057	0.082	0.005	0.004
Sacc. Pom. [48,49,47]	Direct – Phys. Assoc.	510	0.086	0.004	0.011	0.000
	Phys. Assoc. – Synth. Gen.	426	0.077	0.026	0.013	0.000
	Direct – Synth. Gen.	289	0.087	0.033	0.022	0.000
	Phys. Assoc. – Association	54	0.074	0.067	0.010	0.007
Physicians [50,47]	Advice – Discussion	104	0.097	0.076	0.016	0.006
	Advice – Friendship	99	0.176	0.130	0.078	0.042
	Discussion – Friendship	106	0.159	0.130	0.057	0.026
Drosophila M. [48,49,47]	Supp. Gen. – Addit. Gen.	449	0.019	0.011	0.001	0.000
India Transportation [51,47]	Airports – Train Stat.	67	0.022	0.031	0.001	0.000
Human Brain [52,47]	Anatomical – Functional	74	0.090	0.150	0.021	0.002

of the microscopic probability of individual nodes to host observers. We developed mathematical frameworks able to well describe the observability diagram in synthetic and real-world multiplex networks. The framework allows us to approximate the true observability diagram of a given multiplex network with a number of operations that scales as $E \ln E$, where E is the total number of edges in the system (see Appendix D). Our results indicate that the features of this model cannot be trivially deduced from those valid for percolation. This statement is true both for randomly generated multiplex networks, as well as for multiplex networks representing real-world systems. Interestingly, real-world multiplex networks seem to be always in the observable regime, as long as the fraction of nodes that is directly observed is larger than zero. This fact seems due to the natural, and ubiquitously observed, presence of edge overlap among the layers that compose real multiplex networks.

The approach described in this paper is valid only for multiplex networks composed of two layers. The generalization of the framework (accounting for edge overlap) to an arbitrary number of layers seems not as straightforward. Additional theoretical work is required to devise a complete theory of the observability model in arbitrary multiplex networks.

Acknowledgments

The authors thank N. Azimi for discussions on the early stages of this research. FR acknowledges support from the National Science Foundation, United States (Grant CMMI-1552487) and the US Army Research Office (W911NF-16-1-0104).

Appendix A. Message passing with edge overlap

This section is devoted to the description of the message-passing algorithm to approximate the behavior of the Largest Mutually Observable Cluster (LMOC) in multiplex networks. The formalism differ from the one already presented in the main text for the fact that the only approximation used is the locally treelike ansatz. Edge overlap among layers is instead accounted by this framework. The method presented here is a generalization of the algorithm proposed in Ref. [28] for standard site percolation on multiplex networks. The method relies on the definition of the multilink $\vec{m}_{ji} = (a_{ji}^{[\alpha]}, a_{ji}^{[\beta]})$ for every pairs of nodes j and i in the duplex, where $a_{ji}^{[\alpha]} = 1$ if the nodes are connected in layer α , and $a_{ji}^{[\alpha]} = 0$, otherwise. The same definition applies for layer β . Messages considered in the approach are:

1. $u_{j \rightarrow i}^{\vec{m}_{ji}, \vec{n}}$, valid when node j is directly observable.
2. $v_{j \rightarrow i}^{\vec{m}_{ji}, \vec{n}}$, valid when node j is not directly observable.
3. $z_{j \rightarrow i}^{\vec{m}_{ji}, \vec{n}}$, valid when neither node i nor node j are directly observable.

In the definition of the messages, we use $\vec{n} = (n^{[\alpha]}, n^{[\beta]})$ to maintain the notation as compact as possible. Different messages are indicated by different values of \vec{n} , namely $(0, 0)$, $(1, 0)$, $(0, 1)$, and $(1, 1)$. Note that if $n^{[\alpha]}(1 - a_{ji}^{[\alpha]}) + n^{[\beta]}(1 - a_{ji}^{[\beta]}) \neq 0$, the corresponding message is automatically zero. Further, normalization implies that $u_{j \rightarrow i}^{\vec{m}_{ji}, (0, 0)} = 1 - u_{j \rightarrow i}^{\vec{m}_{ji}, (1, 0)} - u_{j \rightarrow i}^{\vec{m}_{ji}, (0, 1)} - u_{j \rightarrow i}^{\vec{m}_{ji}, (1, 1)}$. It is further convenient to cumulate messages over layers as:

$$u_{j \rightarrow i}^{[\alpha]} = u_{j \rightarrow i}^{\vec{m}_{ji}, (1, 0)} + u_{j \rightarrow i}^{\vec{m}_{ji}, (1, 1)}, \quad (\text{A.1})$$

$$u_{j \rightarrow i}^{[\beta]} = u_{j \rightarrow i}^{\vec{m}_{ji}, (0, 1)} + u_{j \rightarrow i}^{\vec{m}_{ji}, (1, 1)}, \quad (\text{A.2})$$

and

$$u_{j \rightarrow i}^{[\alpha, \beta]} = u_{j \rightarrow i}^{\vec{m}_{ji}, (1, 0)} + u_{j \rightarrow i}^{\vec{m}_{ji}, (0, 1)} + u_{j \rightarrow i}^{\vec{m}_{ji}, (1, 1)}. \quad (\text{A.3})$$

The same definitions are valid for v - and z -type messages.

The probability that node i is in the LMOC is calculated in different manner depending on whether node i is (i) directly observable or (ii) not directly observable. For case (i), we have

$$\begin{aligned} q_i &= \phi \left[1 - \left[\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\alpha]} - (1 - \phi) v_{j \rightarrow i}^{[\alpha]}) \right] - \left[\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\beta]} - (1 - \phi) v_{j \rightarrow i}^{[\beta]}) \right] + \right. \\ &\quad \left. \left[\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\alpha, \beta]} - (1 - \phi) v_{j \rightarrow i}^{[\alpha, \beta]}) \right] \right]. \end{aligned} \quad (\text{A.4})$$

Essentially, the probability for node i to be directly observable and part of the LMOC is given by the product of the probability to be directly observable and attached at least to another node that is part of the LMOC. The latter is estimated as one minus the probability that none of the nodes connected to i are part of the LMOC.

To account for the overlap among edges in the two layers, we need to make a distinction among neighbors of a node. We define three sets of neighbors for node i : $\partial_i^{[\alpha]}$, that is set of nodes that are neighbors of node i just in layer α ; $\partial_i^{[\beta]}$, that is set of nodes that are neighbors of node i just in layer β ; $\partial_i^{[\alpha, \beta]}$, that is the set of nodes that are neighbors of node i at the same time in both layers. In particular, we indicate with $k_i^{[\alpha]}$, $k_i^{[\beta]}$, $k_i^{[\alpha, \beta]}$ the size of the sets $\partial_i^{[\alpha]}$, $\partial_i^{[\beta]}$, $\partial_i^{[\alpha, \beta]}$, respectively. We also define degree of each node in layer α , layer β and total as follows: $k_i^{[1]} = k_i^{[\alpha]} + k_i^{[\alpha, \beta]}$, $k_i^{[2]} = k_i^{[\beta]} + k_i^{[\alpha, \beta]}$, $k_i^{[1, 2]} = k_i^{[\alpha]} + k_i^{[\beta]} + k_i^{[\alpha, \beta]}$.

$$r_i = (1 - \phi)[1 - A - B + C] \quad (\text{A.5})$$

$$\begin{aligned} A &= \left[\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\alpha]} - (1 - \phi) z_{j \rightarrow i}^{[\alpha]}) \right] + (1 - \phi)^{k_i^{[1]}} \left[1 - \prod_{j \in \partial_i} (1 - z_{j \rightarrow i}^{[\alpha]}) \right] \\ B &= \left[\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\beta]} - (1 - \phi) z_{j \rightarrow i}^{[\beta]}) \right] + (1 - \phi)^{k_i^{[2]}} \left[1 - \prod_{j \in \partial_i} (1 - z_{j \rightarrow i}^{[\beta]}) \right] \end{aligned}$$

$$\begin{aligned}
C = & \left[\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\alpha, \beta]} - (1 - \phi) z_{j \rightarrow i}^{[\alpha, \beta]}) \right] + (1 - \phi)^{k_i^{[1,2]}} \left[1 - \prod_{j \in \partial_i} (1 - z_{j \rightarrow i}^{[\alpha, \beta]}) \right] + \\
& \left[\prod_{j \in \partial_i^{[\alpha]}} (1 - \phi u_{j \rightarrow i}^{[\alpha]} - (1 - \phi) z_{j \rightarrow i}^{[\alpha]}) - (1 - \phi)^{k_i^{[\alpha]}} \prod_{j \in \partial_i^{[\alpha]}} (1 - z_{j \rightarrow i}^{[\alpha]}) \right] \times \\
& (1 - \phi)^{k_i^{[2]}} \left[\prod_{j \in \partial_i^{[\alpha, \beta]}} (1 - z_{j \rightarrow i}^{[\alpha]}) - \prod_{j \in \partial_i^{[\beta]}} (1 - z_{j \rightarrow i}^{[\beta]}) \prod_{j \in \partial_i^{[\alpha, \beta]}} (1 - z_{j \rightarrow i}^{[\alpha, \beta]}) \right] + \\
& \left[\prod_{j \in \partial_i^{[\beta]}} (1 - \phi u_{j \rightarrow i}^{[\beta]} - (1 - \phi) z_{j \rightarrow i}^{[\beta]}) - (1 - \phi)^{k_i^{[\beta]}} \prod_{j \in \partial_i^{[\beta]}} (1 - z_{j \rightarrow i}^{[\beta]}) \right] \times \\
& (1 - \phi)^{k_i^{[1]}} \left[\prod_{j \in \partial_i^{[\alpha, \beta]}} (1 - z_{j \rightarrow i}^{[\beta]}) - \prod_{j \in \partial_i^{[\alpha]}} (1 - z_{j \rightarrow i}^{[\alpha]}) \prod_{j \in \partial_i^{[\alpha, \beta]}} (1 - z_{j \rightarrow i}^{[\alpha, \beta]}) \right]. \tag{A.6}
\end{aligned}$$

The terms A and B are obvious. C instead contains a series of exceptions that must be handled to properly account for overlap. In particular,

1. The first term in C stands for the probability that node i is not connected to the LMOC in none of the layers. It accounts also for the exception where all of its neighbors are not directly observable.
2. Probability for node i of being not connected to LMOC through layer α and being not observable in layer β . Connections in layer β do not matter, but overlapping links, may connect node i to the LMOC in layer α .
3. Same as point 2, but swapping α with β .

The exceptions mentioned above are computed noting that

$$\begin{aligned}
\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\alpha, \beta]} - (1 - \phi) v_{j \rightarrow i}^{[\alpha, \beta]}) = & \left[\prod_{j \in \partial_i^{[\alpha]}} (1 - \phi u_{j \rightarrow i}^{(1,0),(1,0)} - (1 - \phi) v_{j \rightarrow i}^{(1,0),(1,0)}) \right] \times \\
& \left[\prod_{j \in \partial_i^{[\beta]}} (1 - \phi u_{j \rightarrow i}^{(0,1),(0,1)} - (1 - \phi) v_{j \rightarrow i}^{(0,1),(0,1)}) \right] \times \\
& \left[\prod_{j \in \partial_i^{[\alpha, \beta]}} (1 - \phi u_{j \rightarrow i}^{[\alpha, \beta]} - (1 - \phi) v_{j \rightarrow i}^{[\alpha, \beta]}) \right]. \tag{A.7}
\end{aligned}$$

We can now derive self-consistent equations for the messages. For u - and v -type messages, these are provided by the following equations:

$$\begin{aligned}
u_{j \rightarrow i}^{(1,1),(1,1)} = u_{j \rightarrow i}^{(1,0),(1,0)} = u_{j \rightarrow i}^{(0,1),(0,1)} = 1 - & \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) v_{k \rightarrow j}^{[\alpha]}) \right] \\
& - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) v_{k \rightarrow j}^{[\beta]}) \right] + \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha, \beta]} - (1 - \phi) v_{k \rightarrow j}^{[\alpha, \beta]}) \right], \tag{A.8}
\end{aligned}$$

$$u_{j \rightarrow i}^{(1,1),(1,0)} = \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) v_{k \rightarrow j}^{[\beta]}) \right] - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha, \beta]} - (1 - \phi) v_{k \rightarrow j}^{[\alpha, \beta]}) \right], \tag{A.9}$$

$$u_{j \rightarrow i}^{(1,1),(0,1)} = \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) v_{k \rightarrow j}^{[\alpha]}) \right] - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha, \beta]} - (1 - \phi) v_{k \rightarrow j}^{[\alpha, \beta]}) \right], \tag{A.10}$$

$$\begin{aligned}
v_{j \rightarrow i}^{(1,1),(1,1)} = 1 - & \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) \right] + \\
& \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha, \beta]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha, \beta]}) \right], \tag{A.11}
\end{aligned}$$

$$v_{j \rightarrow i}^{(1,1),(1,0)} = \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) \right] - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha,\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha,\beta]}) \right], \quad (\text{A.12})$$

$$v_{j \rightarrow i}^{(1,1),(0,1)} = \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha,\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha,\beta]}) \right], \quad (\text{A.13})$$

and

$$\begin{aligned} v_{j \rightarrow i}^{(1,0),(1,0)} &= 1 - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] - \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) + \right. \\ &\quad \left. (1 - \phi)^{k_j^{[2]}} \times \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{k \rightarrow j}^{[\beta]}) \right] \right] + \\ &\quad \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha,\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha,\beta]}) + \left[\prod_{k \in \partial_j^{[\alpha]} \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] \times \right. \\ &\quad \left. (1 - \phi)^{k_j^{[2]}} \times \left[\prod_{k \in \partial_j^{[\alpha,\beta]}} (1 - z_{k \rightarrow j}^{[\alpha]}) - \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \prod_{k \in \partial_j^{[\alpha,\beta]}} (1 - z_{k \rightarrow j}^{[\alpha,\beta]}) \right] \right]. \end{aligned} \quad (\text{A.14})$$

For z -type message, the expressions are more complicated. We have that

$$\begin{aligned} I &= \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] + (1 - \phi)^{k_j^{[1]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{k \rightarrow j}^{[\alpha]}) \right] \\ II &= \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) \right] + (1 - \phi)^{k_j^{[2]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{j \rightarrow i}^{[\beta]}) \right] \\ (I, II) &= \left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha,\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha,\beta]}) \right] + (1 - \phi)^{k_j^{[1,2]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{j \rightarrow i}^{[\alpha,\beta]}) \right] + \\ &\quad \left[\left[\prod_{k \in \partial_j^{[\alpha]}} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) - (1 - \phi)^{k_j^{[\alpha]}} \prod_{k \in \partial_j^{[\alpha]}} (1 - z_{k \rightarrow j}^{[\alpha]}) \right] \times \right. \\ &\quad \left. (1 - \phi)^{k_j^{[2]} - 1} \left[\prod_{k \in \partial_j^{[\alpha,\beta] \setminus i}} (1 - z_{k \rightarrow j}^{[\alpha]}) - \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \prod_{k \in \partial_j^{[\alpha,\beta] \setminus i}} (1 - z_{k \rightarrow j}^{[\alpha,\beta]}) \right] \right] + \\ &\quad \left[\left[\prod_{k \in \partial_j^{[\beta]}} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) - (1 - \phi)^{k_j^{[\beta]}} \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \right] \times \right. \\ &\quad \left. (1 - \phi)^{k_j^{[1]} - 1} \left[\prod_{k \in \partial_j^{[\alpha,\beta] \setminus i}} (1 - z_{k \rightarrow j}^{[\beta]}) - \prod_{k \in \partial_j^{[\alpha]}} (1 - z_{k \rightarrow j}^{[\alpha]}) \prod_{k \in \partial_j^{[\alpha,\beta] \setminus i}} (1 - z_{k \rightarrow j}^{[\alpha,\beta]}) \right] \right] \end{aligned} \quad (\text{A.15})$$

$$\begin{aligned} z_{j \rightarrow i}^{(1,1),(1,1)} &= 1 - I - II + (I, II) \\ z_{j \rightarrow i}^{(1,1),(1,0)} &= II - (I, II) \\ z_{j \rightarrow i}^{(1,1),(0,1)} &= I - (I, II) \end{aligned} \quad (\text{A.16})$$

$$\begin{aligned} z_{j \rightarrow i}^{(1,0),(1,0)} &= 1 - \left[\left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] + (1 - \phi)^{k_j^{[1]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{k \rightarrow j}^{[\alpha]}) \right] \right] - \\ &\quad \left[\left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) \right] + (1 - \phi)^{k_j^{[2]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{j \rightarrow i}^{[\beta]}) \right] \right] + \end{aligned}$$

$$\begin{aligned}
& \left[\left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha, \beta]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha, \beta]}) \right] + (1 - \phi)^{k_j^{[1,2]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{j \rightarrow i}^{[\alpha, \beta]}) \right] \right] + \\
& \left[\left[\prod_{k \in \partial_j^{[\alpha]} \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) - (1 - \phi)^{k_j^{[\alpha]} - 1} \prod_{k \in \partial_j^{[\alpha]} \setminus i} (1 - z_{k \rightarrow j}^{[\alpha]}) \right] \times \right. \\
& \quad \left. (1 - \phi)^{k_j^{[2]}} \left[\prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\alpha]}) - \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\alpha, \beta]}) \right] \right] + \\
& \left[\left[\prod_{k \in \partial_j^{[\beta]}} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) - (1 - \phi)^{k_j^{[\beta]} - 1} \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \right] \times \right. \\
& \quad \left. (1 - \phi)^{k_j^{[1]} - 1} \left[\prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) - \prod_{k \in \partial_j^{[\alpha]}} (1 - z_{k \rightarrow j}^{[\alpha]}) \prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\alpha, \beta]}) \right] \right] \tag{A.17}
\end{aligned}$$

$$\begin{aligned}
z_{j \rightarrow i}^{(0,1),(0,1)} = 1 - & \left[\left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) \right] + (1 - \phi)^{k_j^{[1]}} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{k \rightarrow j}^{[\alpha]}) \right] \right] - \\
& \left[\left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) \right] + (1 - \phi)^{k_j^{[2]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{j \rightarrow i}^{[\beta]}) \right] \right] + \\
& \left[\left[\prod_{k \in \partial_j \setminus i} (1 - \phi u_{k \rightarrow j}^{[\alpha, \beta]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha, \beta]}) \right] + (1 - \phi)^{k_j^{[1,2]} - 1} \left[1 - \prod_{k \in \partial_j \setminus i} (1 - z_{j \rightarrow i}^{[\alpha, \beta]}) \right] \right] + \\
& \left[\left[\prod_{k \in \partial_j^{[\alpha]}} (1 - \phi u_{k \rightarrow j}^{[\alpha]} - (1 - \phi) z_{k \rightarrow j}^{[\alpha]}) - (1 - \phi)^{k_j^{[\alpha]} - 1} \prod_{k \in \partial_j^{[\alpha]}} (1 - z_{k \rightarrow j}^{[\alpha]}) \right] \times \right. \\
& \quad \left. (1 - \phi)^{k_j^{[2]} - 1} \left[\prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\alpha]}) - \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\alpha, \beta]}) \right] \right] + \\
& \left[\left[\prod_{k \in \partial_j^{[\beta]}} (1 - \phi u_{k \rightarrow j}^{[\beta]} - (1 - \phi) z_{k \rightarrow j}^{[\beta]}) - (1 - \phi)^{k_j^{[\beta]} - 1} \prod_{k \in \partial_j^{[\beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) \right] \times \right. \\
& \quad \left. (1 - \phi)^{k_j^{[1]} - 1} \left[\prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\beta]}) - \prod_{k \in \partial_j^{[\alpha]}} (1 - z_{k \rightarrow j}^{[\alpha]}) \prod_{k \in \partial_j^{[\alpha, \beta]}} (1 - z_{k \rightarrow j}^{[\alpha, \beta]}) \right] \right]. \tag{A.18}
\end{aligned}$$

In the Appendices B and C, we briefly discuss the reduction of the framework presented in Appendix A in two simple cases: no overlap and full overlap.

Appendix B. Reduction of the framework to the case of no overlap

If there is no overlap between the two layers of the multiplex, then $u_{j \rightarrow i}^{\tilde{m}_{ji},(1,1)} = v_{j \rightarrow i}^{\tilde{m}_{ji},(1,1)} = z_{j \rightarrow i}^{\tilde{m}_{ji},(1,1)} = 0$ and $k_i^{[1]} = k_i^{[\alpha]}$ and $k_i^{[2]} = k_i^{[\beta]}$. Further, any term that previously was separated into three multiplications, such as the r.h.s. of Eq. (A.4), will contain only two multiplications. Please note that multiplications over empty sets, such as $\prod_{j \in \partial_i^{[\alpha, \beta]}}$, will be, by definition, equal to one. Eq. (A.4) becomes

$$\begin{aligned}
\prod_{j \in \partial_i} (1 - \phi u_{j \rightarrow i}^{[\alpha, \beta]} - (1 - \phi) v_{j \rightarrow i}^{[\alpha, \beta]}) = & \left[\prod_{j \in \partial_i^{[\alpha]}} (1 - \phi u_{j \rightarrow i}^{(1,0),(1,0)} - (1 - \phi) v_{j \rightarrow i}^{(1,0),(1,0)}) \right] \times \\
& \left[\prod_{j \in \partial_i^{[\beta]}} (1 - \phi u_{j \rightarrow i}^{(0,1),(0,1)} - (1 - \phi) v_{j \rightarrow i}^{(0,1),(0,1)}) \right].
\end{aligned}$$

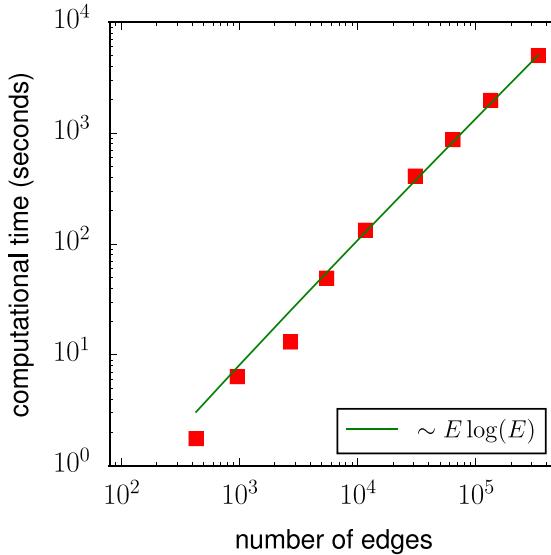


Fig. D.7. Estimate of the computational time τ required to draw the entire observability diagram for an interdependent network formed by two scale-free graphs with minimal degree $k_{min} = 3$, and degree exponent $\gamma = 2.5$ (red squares). Results have been obtained on an Intel(R) Xeon(R) CPU E5-2695 v2 2.40 GHz machine. Each point corresponds to the average value over 10 realizations of the network models. Standard deviation is compatible with the size of the symbols. We draw the percolation diagram by considering any value of the probability ϕ in the interval $[0, 1]$ with a precision $d\phi = 10^{-3}$. Numerical convergence of the iterative algorithm is achieved when values of the same variable differ by at maximum $\epsilon = 10^{-7}$ between two consecutive iterations. The computational time τ is plotted as a function of the number of edges E in the multiplex. We test the scaling $\tau \sim E \ln(E)$ (green line).

As there are no links shared by the two layers, the r.h.s of Eq. (A.1) becomes identical to Eq. (9) of the main text, with only difference of containing explicitly the factors $q_i^{[\alpha]}$ and $q_i^{[\beta]}$ of Eq. (7). We can apply the same procedure for the other equations. For example, Eq. (A.2) reduces to Eq. (10) as follows. Substitute $r_i^{[\alpha]}$ and $r_i^{[\beta]}$, as defined in Eq. (8), into Eq. (10). Define $r_i^{[\alpha]} = 1 - X$, $r_i^{[\beta]} = 1 - Y$, and $r_i^{[\alpha]} r_i^{[\beta]} = 1 - X - Y + XY$. When there is no overlap, the terms A and B of Eq. (A.3) are exactly given by X and Y . The last term C is nothing more than XY , as we know that $k_i^{[1]} = k_i^{[\alpha]}$ and $k_i^{[2]} = k_i^{[\beta]}$.

Appendix C. Reduction of the framework to the case of full overlap

When there is full overlap between the two layers of the multiplex, then $u_{j \rightarrow i}^{[\alpha]} = u_{j \rightarrow i}^{[\beta]} = u_{j \rightarrow i}^{[\alpha, \beta]}$ and $k_i^{[1]} = k_i^{[2]} = k_i^{[\alpha, \beta]}$. Further, any term that previously was separated into three multiplications, such as the r.h.s. of Eq. (A.4), will contain only one multiplication. Please note that multiplications over empty sets, such as $\prod_{j \in \partial_i^{[\alpha]}}$ and $\prod_{j \in \partial_i^{[\beta]}}$, will be, by definition, equal to one.

In the full overlap, case, the three terms on the r.h.s. of Eq. (A.1) become identical, thus two of them will cancel each other. Excluding the prefactor ϕ , Eq. (A.1) thus reduces to Eq. (7) of the main text.

When there is full overlap, the terms A and B , and first line of C of Eq. (A.3) are equivalent. In Eq. (A.2), two of the terms cancel each other, thus the equation, except for the prefactor $1 - \phi$, reduces to the single-layer version given by Eq. (8) of the main text. The remaining terms of the factor C (line 2, 3, 4 and 5) are all equal to zero. To understand this fact, it is enough to look at line 3: if we set multiplication over empty set ($\prod_{j \in \partial_i^{[\beta]}}$) equal to one, then the term inside the bracket is equal to zero, since for the full-overlap case, we have that $z_{j \rightarrow i}^{[\alpha]} = z_{j \rightarrow i}^{[\beta]} = z_{j \rightarrow i}^{[\alpha, \beta]}$. For the same reason, line 5 of the C is zero too.

We can reduce all other equations of the Appendix A to their single-layer version, similarly as we did above.

Appendix D. Computational complexity of the algorithm

See Fig. D.7.

Appendix E. Real-world multiplex networks

See Figs. E.8, E.9, E.10 and E.11.

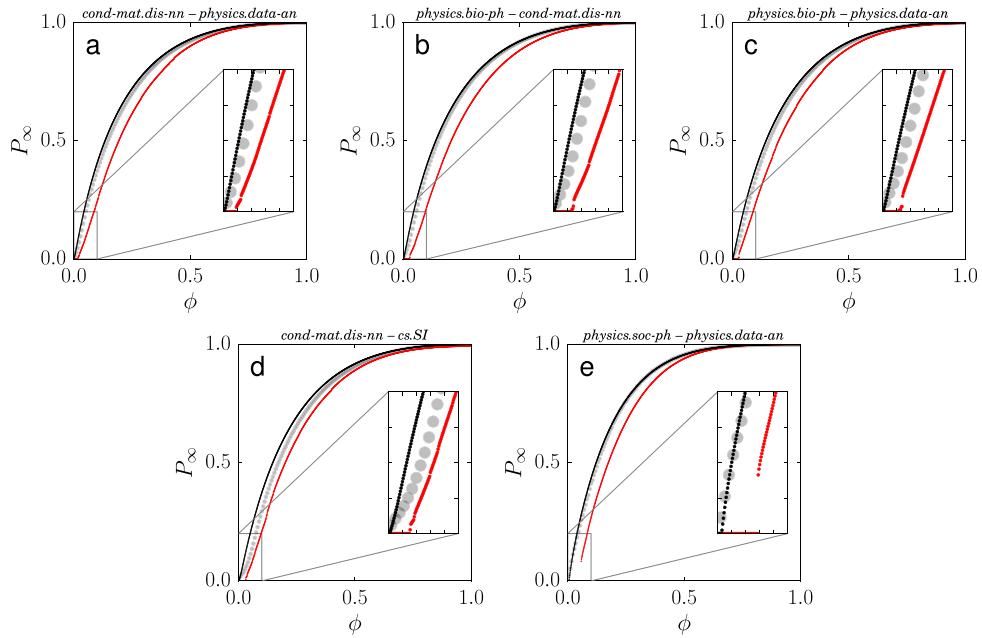


Fig. E.8. Observability transition in the Arxiv collaboration multiplex networks [46,47].

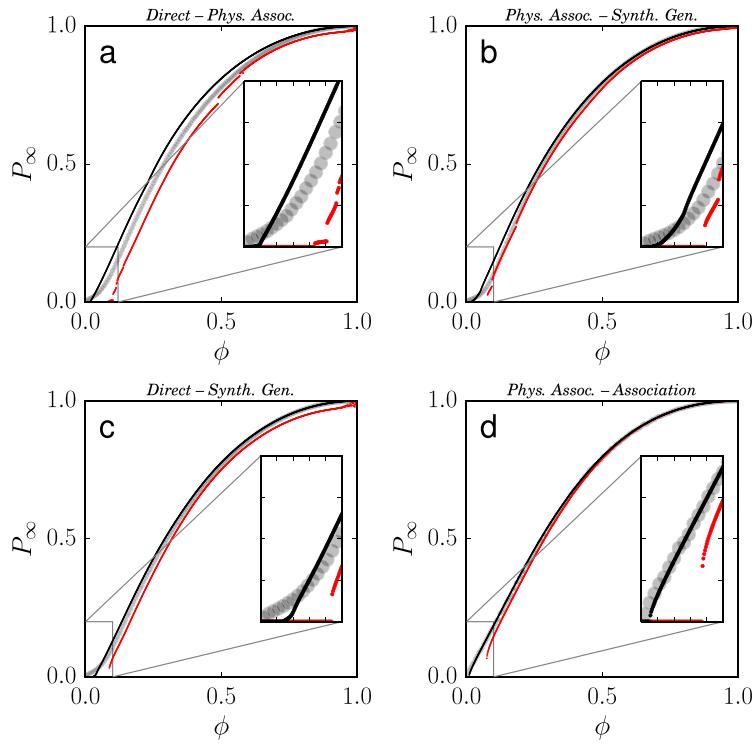


Fig. E.9. Observability transition in the *Saccharomyces Pombe* multiplex networks [48,49,47].

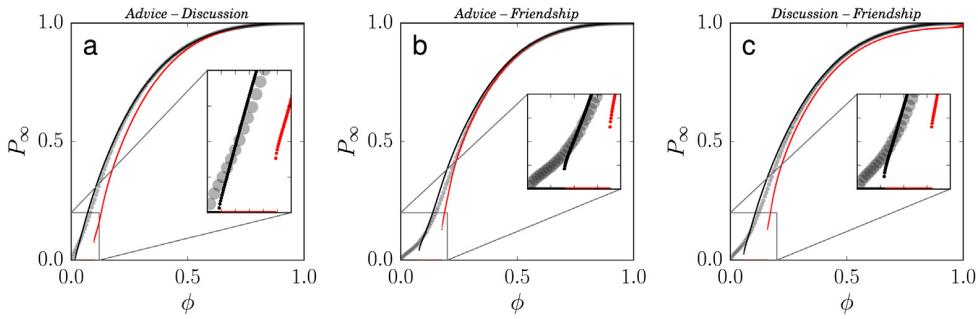


Fig. E.10. Observability transition in the Physician multiplex networks [50,47].

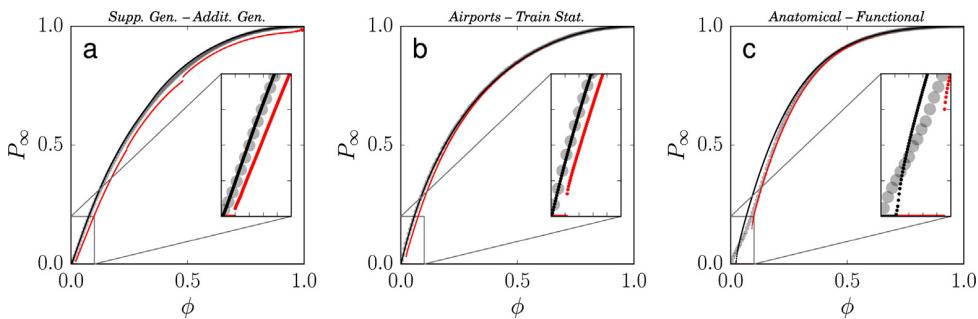


Fig. E.11. Observability transition in the (a) *Drosophila M.* [48,49,47], (b) *India Transportation* [51,47], and (c) *Human Brain* [52,47] multiplex networks.

References

- [1] S. Boccaletti, G. Bianconi, R. Criado, C.I. Del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendiña-Nadal, Z. Wang, M. Zanin, The structure and dynamics of multilayer networks, *Phys. Rep.* 544 (1) (2014) 1–122.
- [2] M. Kivelä, A. Arenas, M. Barthelemy, J.P. Gleeson, Y. Moreno, M.A. Porter, Multilayer networks, *J. Complex Netw.* 2 (3) (2014) 203–271.
- [3] K.-M. Lee, B. Min, K.-I. Goh, Towards real-world complexity: an introduction to multiplex networks, *Eur. Phys. J. B* 88 (2) (2015) 1–20.
- [4] M. Szell, R. Lambiotte, S. Thurner, Multirelational organization of large-scale social networks in an online world, *Proc. Natl. Acad. Sci. USA* 107 (31) (2010) 13636–13641.
- [5] P.J. Mucha, T. Richardson, K. Macon, M.A. Porter, J.-P. Onnela, Community structure in time-dependent, multiscale, and multiplex networks, *Science* 328 (5980) (2010) 876–878.
- [6] M. Barthélémy, Spatial networks, *Phys. Rep.* 499 (1) (2011) 1–101.
- [7] A. Cardillo, J. Gómez-Gardeñes, M. Zanin, M. Romance, D. Papo, F. del Pozo, S. Boccaletti, Emergence of network features from multiplexity, *Sci. Rep.* 3 (2013) 1344.
- [8] S. Gómez, A. Díaz-Guilera, J. Gómez-Gardeñes, C.J. Pérez-Vicente, Y. Moreno, A. Arenas, Diffusion dynamics on multiplex networks, *Phys. Rev. Lett.* 110 (2013) 028701.
- [9] M. De Domenico, A. Solé-Ribalta, S. Gómez, A. Arenas, Navigability of interconnected networks under random failures, *Proc. Natl. Acad. Sci. USA* 111 (23) (2014) 8351–8356.
- [10] M. Dickison, S. Havlin, H.E. Stanley, Epidemics on interconnected networks, *Phys. Rev. E* 85 (2012) 066109.
- [11] A. Saumell-Mendiola, M.A. Serrano, M. Boguñá, Epidemic spreading on interconnected networks, *Phys. Rev. E* 86 (2012) 026106.
- [12] C. Granell, S. Gómez, A. Arenas, Dynamical interplay between awareness and epidemic spreading in multiplex networks, *Phys. Rev. Lett.* 111 (2013) 128701.
- [13] M. De Domenico, C. Granell, M.A. Porter, A. Arenas, The physics of spreading processes in multilayer networks, *Nat. Phys.* 12 (2016) 901–906.
- [14] C.I. del Genio, J. Gómez-Gardeñes, I. Bonamassa, S. Boccaletti, Synchronization in networks with multiple interaction layers, *Sci. Adv.* 2 (11) (2016) e1601679.
- [15] M. Pósfai, J. Gao, S.P. Cornelius, A.-L. Barabási, R.M. D’Souza, Controllability of multiplex, multi-time-scale networks, *Phys. Rev. E* 94 (2016) 032316.
- [16] S.V. Buldyrev, R. Parshani, G. Paul, H.E. Stanley, S. Havlin, Catastrophic cascade of failures in interdependent networks, *Nature* 464 (7291) (2010) 1025–1028.
- [17] R. Parshani, S.V. Buldyrev, S. Havlin, Interdependent networks: reducing the coupling strength leads to a change from a first to second order percolation transition, *Phys. Rev. Lett.* 105 (4) (2010) 048701.
- [18] R. Parshani, C. Rozenblat, D. Ietri, C. Dueruet, S. Havlin, Inter-similarity between coupled networks, *Europhys. Lett.* 92 (6) (2011) 68002.
- [19] G. Baxter, S. Dorogovtsev, A. Goltsev, J. Mendes, Avalanche collapse of interdependent networks, *Phys. Rev. Lett.* 109 (24) (2012) 248701.
- [20] S. Watanabe, Y. Kabashima, Cavity-based robustness analysis of interdependent networks: Influences of intranetwork and internetwork degree-degree correlations, *Phys. Rev. E* 89 (2014) 012808.
- [21] S.-W. Son, G. Bizhani, C. Christensen, P. Grassberger, M. Paczuski, Percolation theory on interdependent networks based on epidemic spreading, *Europhys. Lett.* 97 (1) (2012) 16006.
- [22] B. Min, S. Do Yi, K.-M. Lee, K.-I. Goh, Network robustness of multiplex networks with interlayer degree correlations, *Phys. Rev. E* 89 (4) (2014) 042811.

- [23] G. Bianconi, S.N. Dorogovtsev, J.F.F. Mendes, Mutually connected component of networks of networks with replica nodes, *Phys. Rev. E* 91 (2015) 012804.
- [24] G. Bianconi, S.N. Dorogovtsev, Multiple percolation transitions in a configuration model of a network of networks, *Phys. Rev. E* 89 (6) (2014) 062814.
- [25] F. Radicchi, A. Arenas, Abrupt transition in the structural formation of interconnected networks, *Nat. Phys.* 9 (11) (2013) 717–720.
- [26] F. Radicchi, Percolation in real interdependent networks, *Nat. Phys.* 11 (2015) 597–602.
- [27] D. Cellai, G. Bianconi, Multiplex networks with heterogeneous activities of the nodes, *Phys. Rev. E* 93 (3) (2016) 032302.
- [28] G. Bianconi, F. Radicchi, Percolation in real multiplex networks, *Phys. Rev. E* 94 (2016) 060301.
- [29] F. Radicchi, G. Bianconi, Redundant interdependences boost the robustness of multiplex networks, *Phys. Rev. X* 7 (2017) 011013.
- [30] N. Azimi-Tafreshi, J. Gómez-Gardeñes, S.N. Dorogovtsev, *k*-core percolation on multiplex networks, *Phys. Rev. E* 90 (2014) 032816.
- [31] G.J. Baxter, S.N. Dorogovtsev, J.F.F. Mendes, D. Cellai, Weak percolation on multiplex networks, *Phys. Rev. E* 89 (2014) 042801.
- [32] A. Hackett, D. Cellai, S. Gómez, A. Arenas, J.P. Gleeson, Bond percolation on multiplex networks, *Phys. Rev. X* 6 (2016) 021002.
- [33] S. Osat, A. Faqeesh, F. Radicchi, Optimal percolation on multiplex networks, *Nature Commun.* 8 (1) (2017) 1540. <http://dx.doi.org/10.1038/s41467-017-01442-2>.
- [34] Y. Yang, J. Wang, A.E. Motter, Network observability transitions, *Phys. Rev. Lett.* 109 (2012) 258701.
- [35] Y. Yang, F. Radicchi, Observability transition in real networks, *Phys. Rev. E* 94 (2016) 030301.
- [36] Y.-Y. Liu, J.-J. Slotine, A.-L. Barabási, Controllability of complex networks, *Nature* 473 (7346) (2011) 167–173.
- [37] J. Wu, H. Li, On calculating connected dominating set for efficient routing in ad hoc wireless networks, in: Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, ACM, 1999, pp. 7–14.
- [38] M.E. Newman, Ego-centered networks and the ripple effect, *Social Networks* 25 (1) (2003) 83–95.
- [39] M. Molloy, B. Reed, A critical point for random graphs with a given degree sequence, *Random Struct. Algorithms* 6 (2–3) (1995) 161–180.
- [40] G. Bianconi, Statistical mechanics of multiplex networks: Entropy and overlap, *Phys. Rev. E* 87 (6) (2013) 062806.
- [41] G.J. Baxter, G. Bianconi, R.A. da Costa, S.N. Dorogovtsev, J.F. Mendes, Correlated edge overlaps in multiplex networks, *Phys. Rev. E* 94 (2016) 012303.
- [42] B. Min, S. Lee, K.-M. Lee, K.-I. Goh, Link overlap, viability, and mutual percolation in multiplex networks, *Chaos Solitons Fractals* 72 (2015) 49–58.
- [43] D. Cellai, S.N. Dorogovtsev, G. Bianconi, Message passing theory for percolation models on multiplex networks with link overlap, *Phys. Rev. E* 94 (2016) 032301.
- [44] B.L. Chen, D.H. Hall, D.B. Chklovskii, Wiring optimization can relate neuronal structure and function, *Proc. Natl. Acad. Sci. USA* 103 (12) (2006) 4723–4728.
- [45] M. De Domenico, M.A. Porter, A. Arenas, Muxviz: a tool for multilayer analysis and visualization of networks, *J. Complex Netw.* (2014) cnu038.
- [46] M. De Domenico, A. Lancichinetti, A. Arenas, M. Rosvall, Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems, *Phys. Rev. X* 5 (1) (2015) 011027.
- [47] K.-K. Kleineberg, M. Boguñá, M. Ángeles Serrano, F. Papadopoulos, Hidden geometric correlations in real multiplex networks, *Nat. Phys.* 12 (2016) 1076–1081.
- [48] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, M. Tyers, Biogrid: a general repository for interaction datasets, *Nucleic Acids Res.* 34 (suppl 1) (2006) D535–D539.
- [49] M. De Domenico, V. Nicosia, A. Arenas, V. Latora, Structural reducibility of multilayer networks, *Nat. Commun.* 6 (2015) 6864.
- [50] J. Coleman, E. Katz, H. Menzel, The diffusion of an innovation among physicians, *Sociometry* 20 (4) (1957) 253–270.
- [51] A. Halu, S. Mukherjee, G. Bianconi, Emergence of overlap in ensembles of spatial multiplexes and statistical mechanics of spatial interacting network ensembles, *Phys. Rev. E* 89 (2014) 012806.
- [52] T. Simas, M. Chavez, P. Rodriguez, A. Diaz-Guilera, An algebraic topological method for multimodal brain networks comparisons, *Front. Psychol.* 6 (2015) 904.

Chapter Five

Characterizing the analogy between
hyperbolic embedding and community
structure of complex networks

Contribution

I contributed to all aspects of this work including research design, simulation and writing the manuscript. In the numerical part my contributions were 1) designing a multiplex network with community-correlated structure and 2) optimal targeted attack on these networks. I had equal contribution in writing, editing and revising of the paper.

Characterizing the Analogy Between Hyperbolic Embedding and Community Structure of Complex Networks

Ali Faqeesh,^{1,2} Saeed Osat,³ and Filippo Radicchi^{2,*}

¹MACSI, Department of Mathematics and Statistics, University of Limerick, Limerick V94 T9PX, Ireland

²Center for Complex Networks and Systems Research, School of Informatics, Computing, and Engineering, Indiana University, Bloomington, Indiana 47408, USA

³Quantum Complexity Science Initiative, Skolkovo Institute of Science and Technology, Skoltech Building 3, Moscow 143026, Russia



(Received 9 April 2018; revised manuscript received 21 May 2018; published 29 August 2018)

We show that the community structure of a network can be used as a coarse version of its embedding in a hidden space with hyperbolic geometry. The finding emerges from a systematic analysis of several real-world and synthetic networks. We take advantage of the analogy for reinterpreting results originally obtained through network hyperbolic embedding in terms of community structure only. First, we show that the robustness of a multiplex network can be controlled by tuning the correlation between the community structures across different layers. Second, we deploy an efficient greedy protocol for network navigability that makes use of routing tables based on community structure.

DOI: 10.1103/PhysRevLett.121.098301

A wealth of recent publications provides evidence of the advantages that may arise from thinking of real-world networks as instances of random network models embedded in hidden metric spaces [1,2]. In this class of models, every node is represented by coordinates that identify its position in the underlying space, and the distance between pairs of nodes determines their likelihood of being connected. The most popular formulation of spatially embedded network models relies on hyperbolic geometry [3,4]. Hyperbolic network geometry emerges spontaneously from models of growing simplicial complexes [5]. Hyperbolic geometry appears the natural choice for networks with broad degree distributions, under the hypothesis that the generating mechanism for edges in the network is a compromise between the popularity of individual nodes and similarity among pairs of nodes [6]. Popularity is represented by the radial coordinate of nodes in the hyperbolic space, while similarity is accounted for by the difference between angular coordinates of pairs of nodes. Hyperbolic maps are useful in practical contexts, as generating efficient routing protocols in information networks [7], characterizing the hierarchical organization of biochemical pathways in cellular networks [8], and monitoring the evolution of the international trade network [9]. However, thinking of networks as embedded in the hyperbolic space is important from the theoretical point of view too. Growing network models that rely on hyperbolic geometry provide a genuine explanation for the emergence of power-law degree distributions from local optimization principles only [6]. Further, recent work show that the main features of the percolation transition in multiplex networks can be predicted by simply accounting

for interlayer correlation among hyperbolic coordinates of nodes [10,11].

Popularity and similarity are core features of models used in network hyperbolic embedding. They are, however, central in another heavily used model in network science: the degree-corrected stochastic block model (SBM) [12]. The SBM assumes a hidden cluster structure where nodes are divided into a certain number of groups. This classification accounts for similarity, as pairs of nodes have different likelihoods of being connected depending on their group memberships. Instead, the degree correction provides a natural way of accounting for the popularity of the individual nodes. The SBM is generally considered in the context of graph clustering, representing a generative network model with given mesoscopic structure [13]. The SBM is used in the formulation of principled community detection methods [14]. These methods, in turn, are equivalent to other well-established techniques for community detection, therefore giving a central role in the graph clustering business to the SBM [15].

At least superficially, the analogy between the ideas of hyperbolic embedding and community structure is apparent. In a recent paper, Wang *et al.* showed that information about community structure can be used to improve the accuracy and efficiency of standard algorithms for hyperbolic embedding [16]. Also, previous work was devoted to the development of network models embedded in hyperbolic geometry with the addition of a preimposed community structure [17–19]. We are not aware, however, of previous attempts to investigate the theoretical and practical similarity of the two approaches when applied

independently to the same network topology. This is the purpose of the present Letter.

We assume that the topology of an undirected and unweighted network G with N nodes is fully specified by its adjacency matrix A , whose element $A_{i,j} = A_{j,i} = 1$ if a connection between nodes i and j is present, or $A_{i,j} = A_{j,i} = 0$, otherwise. The hyperbolic embedding of the network G consists in a pair of coordinates (r_i, θ_i) for every node $i \in G$. The quantity r_i is the radial coordinate of node i ; θ_i is its angular coordinate. We assume that this information is at our disposal. The way we acquire such knowledge depends on whether the network analyzed is synthetic or real. For synthetic graphs, we consider single instances of the popularity-similarity optimization model (PSOM) [6], so that hyperbolic coordinates correspond to ground-truth values of the model. We also analyze several real networks, where coordinates of nodes are obtained by fitting graphs against the PSOM. In this second scenario, we either rely on publicly available embeddings [10,20] or we apply publicly available algorithms to the graphs [20]. Details are provided in [21]. We remark that the PSOM is the model of reference in most of the hyperbolic embedding techniques. It assumes the existence of an underlying hyperbolic space, and it consists of a random growing network model, where nodes are connected depending on their distance and the value of other model parameters, such as average degree $\langle k \rangle$, exponent γ of the power-law degree distribution $P(k) \sim k^{-\gamma}$, and temperature T . When a real network is fitted against the PSOM, the parameters $\langle k \rangle$ and γ of the model are determined on the basis of the observed network, while T is treated as a free parameter [20]. Its value may be set to the one that yields the best match between the theoretical and numerical results for the distance dependent connection probability [38]; when hyperbolic embedding is used in greedy routing, one may look for the T value that results in the highest success rate [20]. The radial coordinate r_i of every node i is uniquely identified by its degree k_i ; hence, r_i is not truly learned. Instead, the angular coordinate θ_i for every node $i \in G$ is treated as a fitting parameter. There are various techniques to perform the fit, including approximated optimization algorithms [20,38], and *ad hoc* heuristic methods [39,40].

In our analysis, we further assume to know the community structure of the graph G , consisting in a flat partition of the network into C total communities, where every node $i \in G$ is associated with a discrete-valued coordinate $\sigma_i = 1, \dots, C$. Algorithms for community detection are numerous [13]. Here, we rely on results obtained by three popular methods: the Louvain algorithm [41], Infomap [42], and the algorithm by Ronhovde and Nussinov [43]. We remark that, in the degree-corrected SBM, the probability for nodes i and j to be connected is a function of σ_i , σ_j , k_i , and k_j . Hence, the graph G can be thought of as embedded into a community structure, where

every node i is *de facto* represented by the coordinates (k_i, σ_i) .

A direct comparison between the hyperbolic embedding and the community structure of the graph G consists in a comparison between the coordinates of the individual nodes in the two representations. Further, as the degree of the nodes trivially matches in both representations, the comparison reduces only in matching angular coordinates θ s and group memberships σ s. From the numerous empirical tests we conducted on both real and synthetic networks, two main conclusions emerge. First, networks usually considered in hyperbolic embedding applications are highly modular, in the sense that partitions found by community detection algorithms correspond to very large values of the modularity function Q [44] (see Fig. 1 and [21]). Second, nodes within the same communities are likely to have similar angular coordinates. This second finding is in line with what already shown in Ref. [16]. To quantify coherence among angular coordinates of nodes within the same community g , we first define the variables ξ_g and ϕ_g with

$$\xi_g e^{i\phi_g} = \frac{1}{n_g} \sum_{j=1}^N \delta_{\sigma_j, g} e^{i\theta_j}. \quad (1)$$

$\delta_{x,y} = 1$ if $x = y$ and $\delta_{x,y} = 0$, otherwise. The rhs of Eq. (1) stands for the sums of vectors in the complex plane of the type $e^{i\theta} = \cos(\theta) + i \sin(\theta)$ of all nodes in group g . The vectorial sum is divided by the community

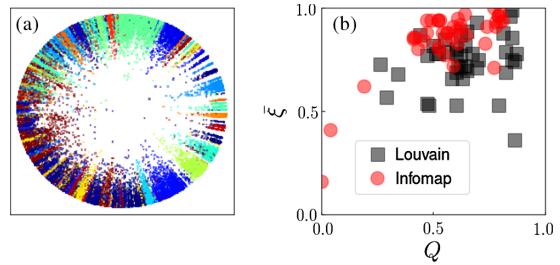


FIG. 1. Hyperbolic embedding and community structure for real and synthetic networks. (a) We compare the hyperbolic embedding of the IPv4 Internet with its community structure. Every point represents a node in the largest connected component of the graph. Positions are determined by the radial and angular coordinates of the nodes in the hyperbolic embedding of the network [10]. We use the best partition found by the Louvain algorithm to determine the community structure of the graph [41]. The partition consists of $C = 31$ communities. Colors of the points identify community memberships. The value of the modularity is $Q = 0.61$, while angular coherence is $\xi = 0.72$. (b) We consider 39 real-world networks and 2 instances of the PSOM, and we compare their community structure and hyperbolic embedding (see details in [21]). The plot displays each network on the (Q, ξ) -plane. We show results obtained using Louvain (black squares) and Infomap (red circles) [42].

size n_g to obtain an average vector for the community. ϕ_g is the angular coordinate of community g . The module $0 \leq \xi_g \leq 1$ indicates how coherent are the angular coordinates of the nodes within group g . Note that, the definition of Eq. (1) resembles the one used for the order parameter of the Kuramoto model [45]. We finally measure the angular coherence of a partition as the weighted average

$$\bar{\xi} = \frac{1}{N} \sum_{g=1}^C n_g \xi_g. \quad (2)$$

By definition, we have that $0 \leq \bar{\xi} \leq 1$. For all networks considered in our analysis (see Fig. 1 and [21]), angular coherence is typically large.

Our empirical tests demonstrate that strong angular coherence within communities of strongly modular networks is a quite robust feature of both synthetic and real systems. This finding tells us that the analogy between community structure and hyperbolic embedding may extend beyond the mere similarity among their ingredients. The following examples show that the analogy is useful also in the interpretation of physical properties of networks and the design of practical algorithms on networks.

Our first example regards the rephrasing, in terms of community structure only, of a result obtained by analyzing the hyperbolic embedding of multiplex networks. In two recent papers [10,11], Kleinberg and collaborators found that the interlayer correlation between hyperbolic coordinates of nodes in multiplex networks is a good predictor for the robustness of a system under targeted attack. Specifically, they found that, when correlation among angular coordinates is high, the percolation transition is smooth. Instead, multiplex networks characterized by a small value of interlayer correlation exhibit abrupt percolation transitions. The finding was initially obtained for real-world multiplex networks. A theoretical explanation was then given in terms of a synthetic network model [11]. To further support the analogy between hyperbolic embedding and community structure that we are arguing for in this Letter, we replicated all results of Ref. [11] using community structure only. First, we analyzed the same real-world multiplex networks considered in Ref. [11]. We found that their robustness can be predicted very well by the level of correlation among the community structures of the layers [21]. Then, we provided a theoretical explanation. We replaced the network model by Kleinberg *et al.* with a variant of the SBM known in the literature as the Lancichinetti-Fortunato-Radicchi (LFR) benchmark graph [46]. The LFR model mostly differs from the standard SBM for relying on heterogeneous distributions of node degrees and community sizes. In our model for multiplex networks [21], we first generate a single LFR graph that is used as the topology for both layers. We then exchange the node labels in one layer to destroy edge overlap and degree-degree

correlation. We consider two distinct scenarios. In the first case, we exchange the label of every node with the one of a randomly chosen node from the same community. This allows us to maintain perfect correlation between the community structure of the two layers. In the second case, we exchange the labels of a number of randomly sampled nodes, such that the edge overlap between the layers equals the value obtained in the first randomization scheme. This second recipe completely destroys the correlation between the community structures of the two layers. In Fig. 2(a), we show the phase diagrams for instances of the multiplex model when relabeling uses information about the community structure of the graph. Here, the community structure is strong, in the sense that the fraction of external connections per node is only $\mu = 0.1$. The transition

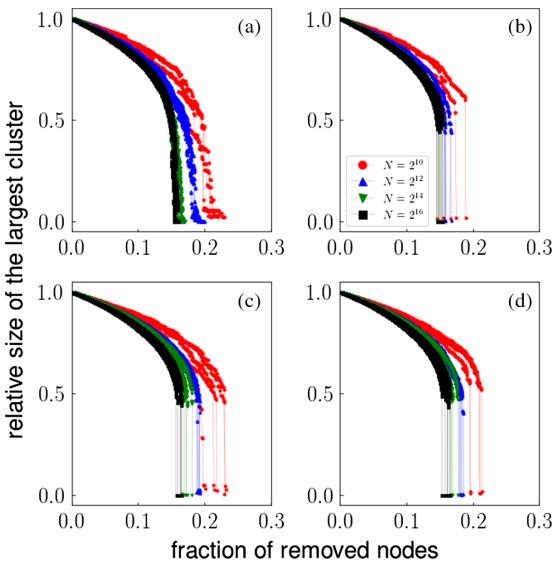


FIG. 2. Robustness of multiplex networks with correlated community structure. We measure the relative size of the largest mutually connected cluster as a function of the fraction of nodes removed from the system. The synthetic multiplex graphs are obtained using the recipe described in the text, where two Lancichinetti-Fortunato-Radicchi (LFR) networks with size N are coupled together. The LFR models are such that: the average degree is $\langle k \rangle = 6$, the maximum degree is $k_{\max} = \sqrt{N}$, node degrees k obey a power-law distribution $P(k) \sim k^{-\gamma}$ with exponent $\gamma = 2.6$, and there are $C = \sqrt{N}$ communities of identical size $S = \sqrt{N}$. For every N , we show the results for five distinct instances of the model. (a) LFR graphs are generated with $\mu = 0.1$. Labels are exchanged only among nodes within the same clusters. All nodes are considered for relabeling at least once. (b) Same as in panel (a). However, relabeling of nodes is not constrained by community structure. The number of nodes that are relabeled is such that the edge overlap among layers is the same as in panel (a) [21]. (c) and (d) Same as in panels (a) and (b), respectively, but for LFR graphs constructed using $\mu = 0.3$.

appears smooth, and it becomes smoother as the size of the model increases. This is an indication that, in the limit of infinitely large LFR graphs, the percolation transition is likely continuous. In Fig. 2(b), we consider the second relabeling scheme that does not account for community structure. The resulting diagrams indicate that the percolation transition is abrupt. The level of correlation among community structure of the two layers can be decreased by increasing μ , so that community structure itself becomes less neat. This is done in Figs. 2(c) and 2(d), where the transition appears abrupt no matter how the labels of the nodes are relabeled. In [21], we report results for different parameter values of the LFR model. Results confirm our claim that the extent of correlation between the community structure of the layers of a multiplex can be used to explain the robustness properties of the system under targeted attack.

Our second example focuses on greedy routing [2,7]. To be brief, the scenario considered is the following. A packet originated by node s must be delivered to node t . The packet can navigate the network by walking at each step on an edge. The packet moves on the network until it reaches its destination t , or it visits the same node twice. In the first case, the packet is correctly delivered. In the second case, the packet is considered lost, and it is discarded. The goal of a good routing strategy is to deliver packets with a high probability and with a small number of steps, for any randomly chosen pair of source and target nodes s and t . Hyperbolic embedding turns out to be very useful in the formulation of a greedy strategy, where individual steps are determined on the basis of the distance among nodes in the hyperbolic space. Specifically, if a message is at node i , then the next move will be on the node

$$j_{(\text{best})}^{(i)} = \arg \min_{j \in \mathcal{N}_i} d(j, t), \quad (3)$$

where \mathcal{N}_i is the set of neighbors of i , and $d(j, t)$ is the distance between nodes j and t . The greedy technique is computationally feasible as every node needs to know only the identity and the geometric coordinates of its neighbors. The regimes of effectiveness of the routing method have been systematically studied in artificial network models [2]. The technique has been proven to be extremely effective on some real-world topologies [2,7]. We devised a new greedy routing protocol that makes use of the cluster structure of a network instead of its hyperbolic embedding. Specifically, we replaced the definition of distance in the hyperbolic space between nodes with the fitness

$$d(j, t) = \beta D_{\sigma_j, \sigma_t} - (1 - \beta) \ln k_j, \quad (4)$$

where k_j is the degree of node j , and σ_j and σ_t are the indices of the communities of nodes j and t , respectively. D_{σ_j, σ_t} is the length of the shortest path between

communities σ_j and σ_t calculated on a weighted supernetwork in which supernodes are communities of the original network. Each pair of supernodes g and q is connected with a superedge with weight $1 - \ln \rho_{g,q}$; here, $\rho_{g,q}$ is the probability that, in the original network, a randomly chosen node in community g has an edge to community q [21]. The term $\ln k_j$ in Eq. (4) serves to perform degree correction. The factor $0 \leq \beta \leq 1$ serves to control the relative importance of one factor over the other. β plays a similar role as of the temperature T in hyperbolic routing protocols [20], and its value may be appropriately chosen with the goal of optimizing the success rate in the delivery of messages [21]. The routing protocol based on Eq. (4) is still computationally efficient as long as the total number of communities C grows sublinearly with the size of the graph N . In the extreme case, where every community is formed by a single node, so that $C = N$, the method will be 100% accurate in delivering packets, but it will also be computationally expensive. In Fig. 3, we display the performance of community-based greedy routing as a function of the mean size of the communities. We study the performance on both synthetic and real-world networks. The number of communities is tuned by changing the resolution parameter in

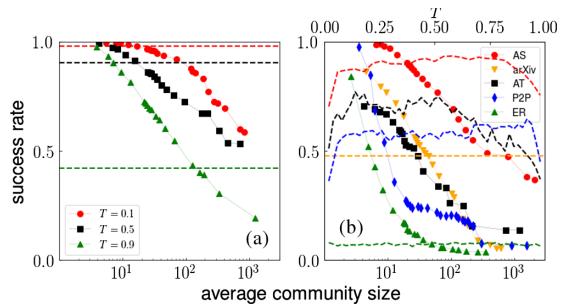


FIG. 3. Performance of community-based routing. (a) We consider single instances of the growing network model of Ref. [38] with $N = 5,000$ nodes, $\langle k \rangle = 5$, and degree exponent $\gamma = 2.1$. Different symbols and colors refer to different values of the temperature T . The plot shows how success rate of the community-based greedy routing strategy changes as a function of the average size of the communities. Communities are identified using the algorithm by Ronhovde and Nussinov [43]. Their number can be varied by changing the resolution level of the algorithm. Dashed lines are obtained on the same networks but using hyperbolic greedy routing. (b) Same as in panel a, but for real networks. We consider the following networks: the Internet at the level of autonomous systems (AS) [47]; the worldwide air transportation network (AT) [48]; the European road network (ER) [49]; the peer-to-peer network (P2P) [50]; the arXiv collaboration network [51]. For all the networks (except arXiv) the dashed lines are obtained by varying the temperature T in the algorithm for hyperbolic embedding introduced in Ref. [20]; for the arXiv network the dashed line shows the result for the optimum hyperbolic coordinates whose data were available in [10]. Details can be found in [21].

the algorithm by Ronhovde and Nussinov [43]. Success rates of the community-based greedy protocol are always very good, as long as communities are not too large.

In summary, we showed that looking at a network as embedded in a hyperbolic geometry is similar, both in theory and practice, to pretending that the network is organized into communities, provided that community structure is detected by a method that accounts for the degree of the nodes. Our finding provides evidence that the intercommunity structure in networks may have geometric organization, meaning that at the global level, geometry dominates, while at the local scale, community memberships prevail. Thus, real networks may be modeled by a graphon [52] consisting of a mixture of latent-spatial and blocklike structures. This fundamental model has the potential to generate further understanding of physical processes, such as spreading and synchronization, in real networks.

The authors thank G. Bianconi, C. V. Cannistraci, D. Krioukov, and M.Á. Serrano for comments on the manuscript. A. F. and F. R. acknowledge support from the U.S. Army Research Office (W911NF-16-1-0104). F. R. acknowledges support from the National Science Foundation (CMMI-1552487). A .F. acknowledges support from the Science Foundation Ireland (16/IA/4470).

* filiradi@indiana.edu

- [1] M. Á. Serrano, D. Krioukov, and M. Boguná, *Phys. Rev. Lett.* **100**, 078701 (2008).
- [2] M. Boguna, D. Krioukov, and K. C. Claffy, *Nat. Phys.* **5**, 74 (2009).
- [3] D. Krioukov, F. Papadopoulos, A. Vahdat, and M. Boguná, *Phys. Rev. E* **80**, 035101 (2009).
- [4] D. Krioukov, F. Papadopoulos, M. Kitsak, A. Vahdat, and M. Boguná, *Phys. Rev. E* **82**, 036106 (2010).
- [5] G. Bianconi and C. Rahmede, *Sci. Rep.* **7**, 41974 (2017).
- [6] F. Papadopoulos, M. Kitsak, M. Á. Serrano, M. Boguná, and D. Krioukov, *Nature (London)* **489**, 537 (2012).
- [7] M. Boguná, F. Papadopoulos, and D. Krioukov, *Nat. Commun.* **1**, 62 (2010).
- [8] M. Á. Serrano, M. Boguná, and F. Sagués, *Mol. Biosyst.* **8**, 843 (2012).
- [9] G. García-Pérez, M. Boguná, A. Allard, and M. Á. Serrano, *Sci. Rep.* **6**, 33441 (2016).
- [10] K.-K. Kleineberg, M. Boguná, M. Á. Serrano, and F. Papadopoulos, *Nat. Phys.* **12**, 1076 (2016).
- [11] K.-K. Kleineberg, L. Buzna, F. Papadopoulos, M. Boguná, and M. Á. Serrano, *Phys. Rev. Lett.* **118**, 218301 (2017).
- [12] B. Karrer and M. E. J. Newman, *Phys. Rev. E* **83**, 016107 (2011).
- [13] S. Fortunato, *Phys. Rep.* **486**, 75 (2010).
- [14] T. P. Peixoto, Chapter in *Advances in Network Clustering and Blockmodeling*, edited by P. Doreian, V. Batagelj, and A. Ferligoj (Wiley, New York, 2018).
- [15] M. E. J. Newman, *Phys. Rev. E* **94**, 052315 (2016).
- [16] Z. Wang, Q. Li, F. Jin, W. Xiong, and Y. Wu, *Physica A (Amsterdam)* **455**, 104 (2016).
- [17] K. Zuev, M. Boguná, G. Bianconi, and D. Krioukov, *Sci. Rep.* **5**, 9421 (2015).
- [18] G. García-Pérez, M. Á. Serrano, and M. Boguná, *J. Stat. Phys.*, DOI: 10.1007/s10955-018-2084-z (2018).
- [19] A. Muscoloni and C. V. Cannistraci, *New J. Phys.* **20**, 052002 (2018).
- [20] F. Papadopoulos, R. Aldecoa, and D. Krioukov, *Phys. Rev. E* **92**, 022807 (2015).
- [21] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevLett.121.098301> for the analysis of additional real and synthetic networks, which includes Refs. [22–37].
- [22] G. García-Pérez, M. Boguná, and M. Á. Serrano, *Nat. Phys.* **14**, 583 (2018).
- [23] B. L. Chen, D. H. Hall, and D. B. Chklovskii, *Proc. Natl. Acad. Sci. USA* **103**, 4723 (2006).
- [24] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, *Nucleic Acids Res.* **34**, D535 (2006).
- [25] M. De Domenico, V. Nicosia, A. Arenas, and V. Latora, *Nat. Commun.* **6**, 6864 (2015).
- [26] M. De Domenico, A. Lancichinetti, A. Arenas, and M. Rosvall, *Phys. Rev. X* **5**, 011027 (2015).
- [27] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, *Internet Math.* **6**, 29 (2009).
- [28] J. Serrà, Á. Corral, M. Boguná, M. Haro, and J. L. Arcos, *Sci. Rep.* **2**, 521 (2012).
- [29] J. Kunegis, in *Proceedings of the 22nd International Conference on World Wide Web* (ACM, New York, 2013) pp. 1343–1350.
- [30] T. Rolland, M. Tasan, B. Charlotteaux, S. J. Pevzner, Q. Zhong, N. Sahni, S. Yi, I. Lemmens, C. Fontanillo, R. Mosca et al., *Cell* **159**, 1212 (2014).
- [31] K. Claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, in *2009 Cybersecurity Applications Technology Conference for Homeland Security* (IEEE, New York, 2009) pp. 205–211.
- [32] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, *J. Stat. Mech.* (2005) P09008.
- [33] F. Radicchi, *Phys. Rev. E* **97**, 022316 (2018).
- [34] F. Radicchi, *Nat. Phys.* **11**, 597 (2015).
- [35] S. Osat, A. Faqeesh, and F. Radicchi, *Nat. Commun.* **8**, 1540 (2017).
- [36] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin, *Nature (London)* **464**, 1025 (2010).
- [37] D. B. Johnson, *J. ACM* **24**, 1 (1977).
- [38] F. Papadopoulos, C. Psomas, and D. Krioukov, *IEEE/ACM Trans. Netw.* **23**, 198 (2015).
- [39] G. Alanis-Lobato, P. Mier, and M. A. Andrade-Navarro, *Sci. Rep.* **6**, 30108 (2016).
- [40] A. Muscoloni, J. M. Thomas, S. Ciucci, G. Bianconi, and C. V. Cannistraci, *Nat. Commun.* **8**, 1615 (2017).
- [41] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, *J. Stat. Mech.* (2008) P10008.
- [42] M. Rosvall and C. T. Bergstrom, *Proc. Natl. Acad. Sci. USA* **105**, 1118 (2008).
- [43] P. Ronhovde and Z. Nussinov, *Phys. Rev. E* **81**, 046114 (2010).
- [44] M. E. J. Newman and M. Girvan, *Phys. Rev. E* **69**, 026113 (2004).

- [45] Y. Kuramoto, *Chemical Oscillations, Waves, and Turbulence* (Dover Publications, New York, 1984).
- [46] A. Lancichinetti, S. Fortunato, and F. Radicchi, *Phys. Rev. E* **78**, 046110 (2008).
- [47] J. Leskovec, J. Kleinberg, and C. Faloutsos, in *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining* (ACM, New York, 2005) pp. 177–187.
- [48] R. Guimerà, S. Mossa, A. Turtschi, and L. A. N. Amaral, *Proc. Natl. Acad. Sci. USA* **102**, 7794 (2005).
- [49] L. Šubelj and M. Bajec, *Eur. Phys. J. B* **81**, 353 (2011).
- [50] M. Ripeanu and I. Foster, in *Peer-to-Peer Systems*, edited by P. Druschel, F. Kaashoek, and A. Rowstron (Springer, Berlin, Heidelberg, 2002) pp. 85–93.
- [51] M. De Domenico, M. A. Porter, and A. Arenas, *J. Complex Netw.* **3**, 159 (2015).
- [52] L. Lovász, *Large Networks and Graph Limits* (American Mathematical Society, Rhode Island, 2012), Vol. 60.

Supplemental Material : Characterizing the analogy between hyperbolic embedding and community structure of complex networks

Ali Faqeeh,^{1,2} Saeed Osat,³ and Filippo Radicchi²

¹*MACSI, Department of Mathematics and Statistics, University of Limerick, Limerick, Ireland*

²*Center for Complex Networks and Systems Research, School of Informatics,*

Computing, and Engineering, Indiana University, Bloomington, Indiana 47408, USA

³*Quantum Complexity Science Initiative, Skolkovo Institute of Science and Technology, Skoltech Building 3, Moscow, 143026, Russia*

Hyperbolic embedding and community detection

In table S1, we provide a list of all networks considered in our analysis.

We obtain hyperbolic coordinates of networks in the following way. For real networks, we either rely on embeddings publicly available [1, 2] or we apply publicly available algorithms to the graphs [2]. URLs of electronic resources for all networks are provided in table S1. In the hyperbolic embeddings that we performed, we made use of the algorithm provided in https://bitbucket.org/dk-lab/2015_code_hypermap. As prescribed in Ref. [2], the value of the temperature T used in the embedding corresponds to the one leading to maximal success rate in greedy routing [3, 4] (see section below). We further consider two instances of the popularity-similarity optimization model (PSOM) [5]. They are generated using different values of the model parameters. The code to generate instances of the PSOM has been taken from <https://www.cut.ac.cy/ecei/staff/f.papadopoulos>.

We use three distinct methods for detecting communities in networks: the Louvain algorithm [6], Infomap [7], and the algorithm by Ronhovde and Nussinov [8]. Louvain and Infomap are used in the analysis about the relation between hyperbolic embedding and community structure (see Table S1). The algorithm by Ronhovde and Nussinov is used in the analysis of greedy routing. For Louvain and Infomap we rely on the algorithms implemented in the library <http://igraph.org/python>. We consider always the “best” (i.e., the one with maximum modularity for Louvain, the one with minimum description length for Infomap) partitions found by the algorithms. The implementation of the algorithm by Ronhovde and Nussinov was taken from <http://www.elemartelot.org/index.php/programming/cd-code>. We chose this algorithm to study greedy routing as it allows for a finer tuning of the resolution of the community structure than the other two algorithms. After obtaining the modular structure from this algorithm, we perform an additional step to improve the quality of communities: If there is any community with size one we change the community label of the only member of that community to the label of its highest degree neighbor.

Community structure and robustness of real-world multiplex networks

We performed the same type of analysis as in Ref. [15] by studying the relation between system robustness and “geometric” correlations among the network layers in real multiplex networks. We just replaced hyperbolic embedding

with community structure. Specifically, given a multiplex network composed of two layers, we first detect communities in the largest connected component of both layers independently by using either Louvain or Infomap. Correlation between the community structure of the layers is measured using the normalized mutual information (NMI) defined in Ref. [26]. As the number of nodes in the layers may be different, in the computation of the NMI values, we considered only nodes appearing in both layers. We finally used the obtained NMI values in the scatter plots of Fig. S1. We find that the robustness of the various networks can be predicted equally well by looking at correlations among either hyperbolic coordinates or community memberships of the nodes in the two layers (see panels a–c). Further, we find that NMI values in the various representations are strongly correlated (panels d–e).

Multiplex networks with correlated community structure

The first step in the creation of a single instance of our multiplex model consists in generating a single instance of the Lancichinetti-Fortunato-Radicchi (LFR) model [27]. The LFR model is a variant of the degree-corrected stochastic block model. The model allows to generate single-layer networks with built-in community structure, where both the degree distribution $P(k)$ and community size distribution $P(S)$ are power-law functions, i.e., $P(k) \sim k^{-\gamma}$ and $P(S) \sim S^{-\beta}$. In addition to the exponents γ and β , in the generation of one instance of the LFR model, one needs to specify the value of several parameters, including: average degree $\langle k \rangle$, maximum degree k_{max} , minimum s_{min} and maximum s_{max} community size, size of the network N , and the mixing parameter μ . The mixing parameter $0 \leq \mu \leq 1$ specifies the fraction of edges that a single node shares with nodes outside its own community. This parameter plays a fundamental role to determine how strong the community structure is. Low values of μ correspond to a strong community structure. As μ increases, community structure becomes fuzzy. The maximal value of μ for which planted community structure is exactly recoverable is bounded by a quantity calculated in Ref. [28]. In our simulations, we use $\mu = 0.1$ to represent a regime of strong community structure, and $\mu = 0.3$ for regime of loose community structure. These values have been chosen arbitrarily, thinking to the application of the model here. For example, we didn’t use μ values too close to zero to avoid the presence of disconnected components.

Once a single instance of the LFR model is generated, we use that instance to define the topology of both layers of the multiplex. Node labels of the two layers are initially iden-

Table S1. Relation between community structure and hyperbolic embedding in real and synthetic networks. From left to right, we report: name of the network, size of the giant component N , number of communities C identified by the Louvain algorithm, value of the modularity Q corresponding to the Louvain partition, angular coherence $\bar{\xi}$ of the Louvain partition, number of communities C identified by Infomap, value of the modularity Q corresponding to the Infomap partition, angular coherence $\bar{\xi}$ of the Infomap partition, reference(s) of the papers where the dataset was reported and/or where hyperbolic coordinates of the network were obtained, urls of the websites where the corresponding data can be downloaded. If the url is denoted by *, this means that data were obtained from a private communication and they are available upon request from the authors of Ref. [9].

network	Louvain				Infomap				Refs.	url
	N	C	Q	$\bar{\xi}$	C	Q	$\bar{\xi}$			
IPv4 Internet	37,542	31	0.61	0.72	1,625	0.47	0.94	[1]	http://koljakleineberg.wordpress.com/materials	
IPv6 Internet	5,143	19	0.48	0.53	418	0.41	0.86	[1]	http://koljakleineberg.wordpress.com/materials	
C. Elegans, layer 1	248	9	0.65	0.70	29	0.61	0.83	[1, 10, 11]	http://koljakleineberg.wordpress.com/materials	
C. Elegans, layer 2	258	9	0.50	0.82	23	0.46	0.84	[1, 10, 11]	http://koljakleineberg.wordpress.com/materials	
C. Elegans, layer 3	278	7	0.44	0.87	11	0.42	0.86	[1, 10, 11]	http://koljakleineberg.wordpress.com/materials	
D. Melanogaster, layer 1	752	17	0.64	0.82	70	0.59	0.91	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
D. Melanogaster, layer 2	633	17	0.64	0.72	68	0.60	0.89	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
arXiv, layer 1	1,537	32	0.87	0.78	130	0.81	0.94	[1, 14]	http://koljakleineberg.wordpress.com/materials	
arXiv, layer 2	2,121	35	0.86	0.74	190	0.79	0.96	[1, 14]	http://koljakleineberg.wordpress.com/materials	
arXiv, layer 3	129	10	0.81	0.88	17	0.78	0.93	[1, 14]	http://koljakleineberg.wordpress.com/materials	
arXiv, layer 4	3,669	46	0.82	0.69	290	0.74	0.91	[1, 14]	http://koljakleineberg.wordpress.com/materials	
arXiv, layer 5	608	23	0.85	0.86	61	0.79	0.96	[1, 14]	http://koljakleineberg.wordpress.com/materials	
arXiv, layer 6	336	17	0.84	0.96	38	0.80	0.98	[1, 14]	http://koljakleineberg.wordpress.com/materials	
Physician, layer 1	106	8	0.51	0.78	13	0.52	0.80	[15]	http://koljakleineberg.wordpress.com/materials	
Physician, layer 2	113	10	0.56	0.79	14	0.55	0.77	[15]	http://koljakleineberg.wordpress.com/materials	
Physician, layer 3	110	9	0.60	0.53	18	0.59	0.72	[15]	http://koljakleineberg.wordpress.com/materials	
SacchPomb, layer 1	751	21	0.79	0.53	86	0.73	0.83	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
SacchPomb, layer 2	182	13	0.82	0.79	28	0.78	0.91	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
SacchPomb, layer 3	2,340	25	0.52	0.78	119	0.47	0.88	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
SacchPomb, layer 4	819	11	0.60	0.69	67	0.56	0.88	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
Human brain, layer 1	85	5	0.62	0.87	8	0.62	0.92	[15]	http://koljakleineberg.wordpress.com/materials	
Human brain, layer 2	78	6	0.55	0.85	8	0.56	0.88	[15]	http://koljakleineberg.wordpress.com/materials	
Rattus, layer 1	1,866	32	0.69	0.71	129	0.65	0.87	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
Rattus, layer 2	529	20	0.85	0.75	61	0.80	0.93	[1, 12, 13]	http://koljakleineberg.wordpress.com/materials	
Air/Train, layer 1	69	5	0.34	0.68	6	0.19	0.62	[15]	http://koljakleineberg.wordpress.com/materials	
Air/Train, layer 2	67	6	0.26	0.73	5	0.04	0.41	[15]	http://koljakleineberg.wordpress.com/materials	
ARK200909	24,091	29	0.62	0.77	980	0.53	0.94	[2]	http://bitbucket.org/dk-lab/2015_code_hypermap	
ARK201003	26,307	29	0.62	0.71	1,070	0.52	0.94	[2]	http://bitbucket.org/dk-lab/2015_code_hypermap	
ARK201012	29,333	28	0.60	0.80	1,171	0.49	0.94	[2]	http://bitbucket.org/dk-lab/2015_code_hypermap	
Enron emails	33,696	291	0.58	0.66	1,546	0.52	0.82	[9, 16]	*	
Music chords	2,476	8	0.29	0.57	6	0.00	0.16	[9, 17]	*	
OpenFights Air Transp.	3,397	26	0.65	0.89	167	0.61	0.95	[9, 18]	*	
Human Metabolites	1,436	18	0.67	0.78	101	0.62	0.90	[9, 19]	*	
Human HI-II-14 proteome	4,100	42	0.47	0.54	334	0.43	0.80	[9, 20]	*	
AS Internet	23,748	24	0.60	0.78	994	0.52	0.94	[9, 21]	*	
AS Oregon Interent, $T = 0.58$	6,474	31	0.63	0.66	412	0.54	0.88	[22]	http://snap.stanford.edu/data/as.html	
Air Transportation, $T = 0.14$	3,618	36	0.69	0.93	246	0.64	0.97	[23]	http://seeslab.info/downloads	
P2P, $T = 0.92$	6,299	19	0.47	0.77	598	0.41	0.85	[24]	http://snap.stanford.edu/data/p2p-Gnutella08.html	
Euro Roads, $T = 0.28$	1,039	25	0.86	0.36	134	0.77	0.71	[25]	http://konect.uni-koblenz.de/networks/subelj_euroroad	
PSOM, $\langle k \rangle = 5, \gamma = 2.1, T = 0.1$	4,114	40	0.85	0.99	248	0.77	1.00	[5]	http://www.cut.ac.cy/ececi/staff/f.papadopoulos	
PSOM, $\langle k \rangle = 5, \gamma = 2.1, T = 0.9$	4,180	30	0.70	0.75	461	0.58	0.85	[5]	http://www.cut.ac.cy/ececi/staff/f.papadopoulos	

tical, so that the adjacency matrices of the two layers are identical. We then start relabeling nodes of one layer only. As already mentioned in the main text, we use two different strategies for relabeling. In the first strategy, we make use of the known community structure. In essence, in the relabeling procedure, the label of every node is exchanged

with the one of another node randomly chosen from the same community. In the other procedure instead, the constraint on the group memberships is not used. This second variant corresponds to the same model already considered in Refs. [29, 30]. In this second variant, we perform a number of label swaps such that the value of the edge overlap

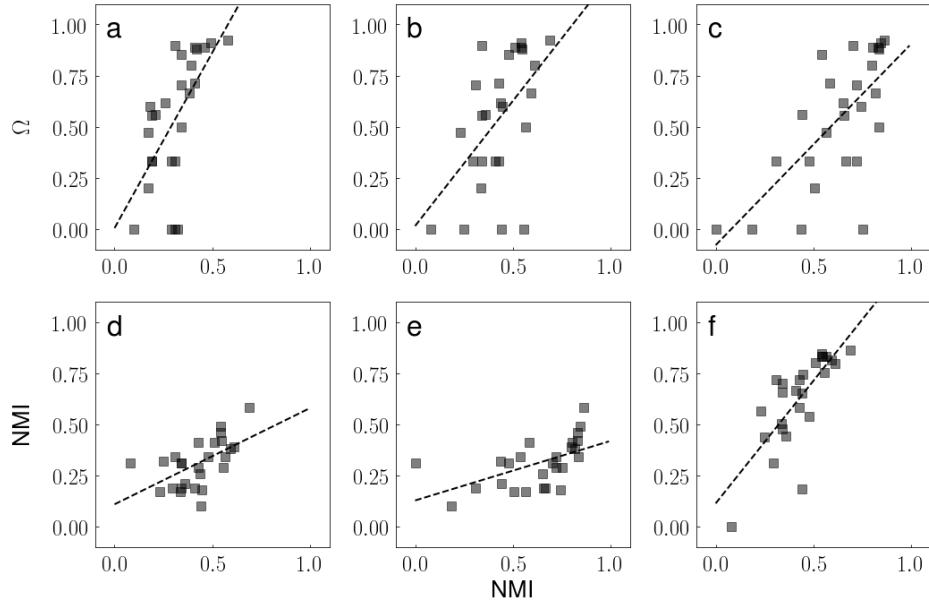


Figure S1. Community structure and robustness of real multiplex networks. We consider the 26 multiplex networks analyzed in Ref. [15]. As in Ref. [15], we rely on the quantity Ω as a proxy to evaluate the robustness of a given multiplex network. $\Omega = (\Delta N - \Delta N_{rs}) / (\Delta N + \Delta N_{rs})$, where ΔN and ΔN_{rs} are respectively the widow sizes of the transitions in targeted and random percolation processes on the network. Ω values for the various networks have been taken from the supplemental material of Ref. [15]. The normalized mutual information (NMI) serves to quantify similarity between the embedding of the nodes in the two layers. Values of the NMI for hyperbolic embedding have also been taken from the supplemental material of Ref. [15]. We calculated instead NMI values among the community structures found for the layers of a multiplex using the definition provided in Ref. [26]. Communities in each layer are found using either Louvain or Infomap. (a) As a reference, we reproduced the same plot as in Fig. 4 of Ref. [15], where each network represents a point in the (NMI, Ω)-plane. The dashed line is obtained with simple linear regression. The correlation coefficient calculated from the data points is $r = 0.63$. (b) Same as in panel a, but for NMI values calculated using the community structures found by the Louvain algorithm. Here $r = 0.54$. (c) Same as in panel b, but for NMI values calculated using the community structures found by Infomap. We measured $r = 0.68$ in this case. (d, e, and f) We compare NMI values obtained using the various embedding methods. The various panels represent: (d) Louvain vs. hyperbolic ($r = 0.56$); (e) Infomap vs. hyperbolic ($r = 0.55$); (f) Infomap vs. Louvain ($r = 0.76$).

among the two layers is comparable with the one obtained in the first variant of the model. Both variants of the multiplex model essentially lead to very small values of edge overlap and degree-degree correlation between layers. The first variant, however, preserves perfect correlation between the community structure of the two layers, while the second variant destroys it completely.

The robustness of single instances of the multiplex model described above are then studied as in Ref. [15]. Every node i in the network has associated the score $K_i = \max(k_i^{(1)}, k_i^{(2)})$, with $k_i^{(x)}$ the degree of node i in layer x . Nodes are then ranked in descending order according to this score, with ties randomly broken. The top node is removed from the network. After every removal, the score K_i of every node i still in the system is recomputed. Further, the relative size of the mutually connected giant component is evaluated to construct a percolation phase diagram [31].

We considered various sets of parameters for the generation of the LFR model. All of them provide the same type of message. When the community structure is strong (i.e., small μ values), the model with correlated community structure undergoes a smooth percolation transition. If correlation in community structure is destroyed, the transition becomes abrupt. If the community structure is not strong

(i.e., large μ values), then both relabeling schemes lead to an abrupt transition. The result is valid also for LFR models with homogenous degree distribution (see Figure S2).

Greedy routing

As already considered in Refs. [3, 4], we imagine that a packet is traveling from the source node s to the target node t in a network with N nodes and adjacency matrix A . The packet moves on edges of the network, performing a single hop at each stage of the dynamics. Greedy routing relies on a definition of “distance” between pairs of nodes in the network. At every stage r of the dynamics towards the target node t , a packet sitting on node $p_r = i$ choose to move to the node $j_{(best)}^{(i)}$ defined in Eq. (4) of the main text. In essence, $j_{(best)}^{(i)}$ is the neighbor of node i that has minimal distance to the target node t . In our numerical simulations, we avoid immediate backtracking walks of the packet, therefore node $j_{(best)}^{(i)} = p_{r+1} \neq p_{r-1}$, i.e., cannot be equal to the node visited before node i ; this condition improves significantly (not shown) the performance of both methods considered in this paper. The packet continues to travel until one of these two conditions is met: (i) the packet arrives at destination after

R steps, i.e., $p_R = t$; (ii) the packet visits twice the same node, i.e., $p_r = p_v$, with $v < r$. Condition (i) corresponds to success. Condition (ii) represents failure and the packet is discarded. To evaluate performance of the routing protocol, we use at least $B = 10,000$ numerical simulations. In each simulation, source s and target t nodes are randomly chosen among the nodes in the giant connected component of the network. We quantify three different metrics of performance:

- 1) The success rate z , i.e., the fraction of packets correctly delivered. This is a metric of performance introduced in Ref. [3]. Results for this metric are presented in Figure 3 of the main text.
- 2) The average value of $\langle R \rangle$, i.e., the average length of the paths for successfully delivered packets. This metric of performance was also introduced in Ref. [3]. Results for this metric are presented in Figures S3c and d.
- 3) Efficiency $\eta = z \langle 1/R \rangle$, where $\langle 1/R \rangle$ represents the mean value of the inverse of the path length obtained for each of the successfully delivered packets. This

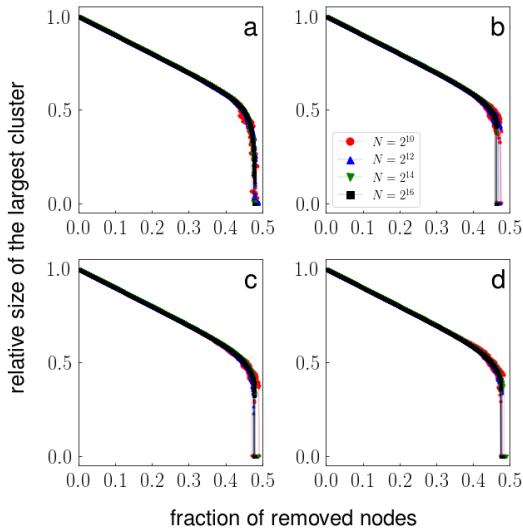


Figure S2. Robustness of multiplex networks with correlated community structure. We measure the relative size of the largest mutually connected cluster as a function of the fraction of nodes removed from the system. The synthetic multiplex graphs were obtained using the recipe described in the text, where two Lancichinetti-Fortunato-Radicchi (LFR) models with size N are coupled together. The LFR models are such that: the average degree is $\langle k \rangle = 6$ and the maximum degree is $k_{max} = 6$, so that degree of all nodes is $k = 6$; communities have identical size $S = 64$. For every value of N we show the results for five distinct instances of the model. (a) LFR graphs are generated with $\mu = 0.1$. Labels are exchanged only among nodes within the same clusters. (b) Same as in panel a. However, relabeling of nodes is allowed among all nodes in the network. Probabilities of relabeling in panels a and b are such that the edge overlap among layers is the same for both models. (c and d) Same as in panels a and b, respectively, but for LFR graphs constructed using $\mu = 0.3$.

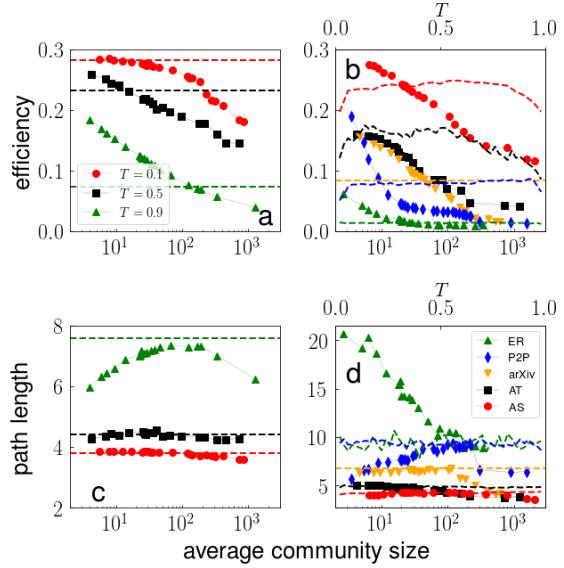


Figure S3. Same analysis as in Figure 3 of the main text. Performance is measured in terms of efficiency (panels a and b), and the average path length of successfully delivered packets (panels c and d).

definition of η is based on a metric of performance introduced in Ref. [32]. Results for this metric are presented in Figures S3a and b.

It is worth noting that the efficiency measure (which is a balance between success rate and path length) shows similar results as those of the success rate (Figure S3a and b); this is because for almost all the networks of Figure S3, the path length does not change remarkably as the mean community size or the temperature is altered (Figures S3c and d). Thus, the success rate results (investigated in Figure 3 of the main text) are sufficient to assess the performance of the two routing methods investigated in this paper.

In the standard application of network hyperbolic embedding, the distance between pairs of nodes is given by their distance in the hyperbolic space [3, 4]. In our community-based routing protocol, we substituted the distance in the hyperbolic space with the analogous quantity based on the a

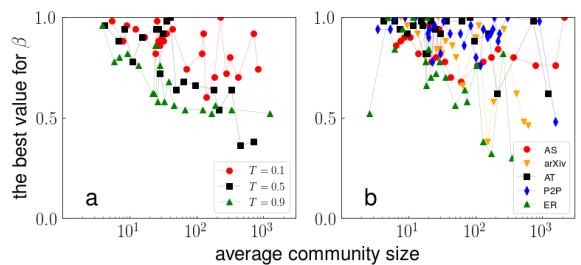


Figure S4. Same analysis as in Figure 3 of the main text. For each modular structure the β value for which we obtained the highest success rate is reported.

a priori given community structure of the graph. Specifically, we define the weight between the connected modules g and q as

$$w_{g,q} = 1 - \ln \rho_{g,q} \quad , \text{ if } \rho_{g,q} > 0 , \quad (\text{S1})$$

where

$$\rho_{g,q} = \frac{\sum_{i>j} A_{i,j} \delta_{\sigma_i,g} \delta_{\sigma_j,q}}{\sum_{i>j} A_{i,j} \delta_{\sigma_i,g}} . \quad (\text{S2})$$

In the above equation, $\delta_{x,y} = 1$, if $x = y$, while $\delta_{x,y} = 0$, otherwise; $A_{i,j} = A_{j,i} = 1$ if nodes i and j are connected, while $A_{i,j} = A_{j,i} = 0$, otherwise; σ_i is the group membership of node i according to the given community structure. Eq. (S2) is the ratio between the total number of edges shared between nodes within communities g and q , and the total degree of nodes in community g . $\rho_{g,q}$ can be also interpreted as the probability that following a random edge of a random node in module g we reach a node in module q . We consider each community as a supernode, and the network as a super-network composed of supernodes connected with weighted superedges. The weight of the superedge between supernodes g and q is defined in Eq. (S1). Then, we find the length of the shortest paths between every pair of supernodes. This operation relies on the algorithm by Johnson [33]. The output is a full matrix D that includes the distances between every pair of modules. The generic element $D_{g,q}$ of this matrix contains a sum of weights defined in Eq. (S1), which is basically equivalent to a sort of expected path length between communities g and q , under the hypothesis that connections were generated according to the stochastic block model [34]. Given that we are at node i at stage r of the trajectory of the packet, the “distance” between a neighbor j of node i and the target t is finally defined as

$$d_{j,t} = \beta D_{\sigma_j, \sigma_t} + (1 - \beta) \left\{ \left[1 - \log(k_j \rho_{\sigma_i, \sigma_j}) \right] - \left[1 - \log(\rho_{\sigma_i, \sigma_j}) \right] \right\} \quad (\text{S3})$$

$$= \beta D_{\sigma_j, \sigma_t} - (1 - \beta) \ln k_j \quad (\text{S4})$$

where k_j is the degree of node j , and $0 \leq \beta \leq 1$. The previous expression defines a measure of “distance” between node j and module σ_t . This is computed as a distance between modules σ_j and σ_t , but corrected for the fact that we are aware of the degree of node j . This definition of distance is motivated by the degree-corrected stochastic block model in which the probability that following a randomly chosen edge from community σ_j we reach a node in community $\sigma_{j'}$ is proportional to $k_j \rho_{\sigma_j, \sigma_{j'}}$. Note that we are aware also of the degrees of nodes i and t , but this information is not helpful in the protocol. The factor β serves to weight the importance of the community structure vs. the degree of the individual nodes in the definition of distance. This factor can be tuned appropriately to optimize the success rate of the greedy routing protocol. Optimal values used in our simulations are displayed in Figure S4. As Figure S4 illustrates, the most optimum value of β depends on the network structure and also on the considered modular structure; more specifically, β is more likely to be close to 1 for networks with lower temperatures (or effectively those with higher clustering coefficients) and for modular structures with smaller mean community sizes.

-
- [1] K.-K. Kleineberg, M. Boguná, M. Á. Serrano, and F. Papadopoulos, *Nature Physics* **12**, 1076 (2016).
 - [2] F. Papadopoulos, R. Aldecoa, and D. Krioukov, *Physical Review E* **92**, 022807 (2015).
 - [3] M. Boguna, D. Krioukov, and K. C. Claffy, *Nature Physics* **5**, 74 (2009).
 - [4] M. Boguná, F. Papadopoulos, and D. Krioukov, *Nature Communications* **1**, 62 (2010).
 - [5] F. Papadopoulos, M. Kitsak, M. Á. Serrano, M. Boguná, and D. Krioukov, *Nature* **489**, 537 (2012).
 - [6] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, *Journal of statistical mechanics: theory and experiment* **2008**, P10008 (2008).
 - [7] M. Rosvall and C. T. Bergstrom, *Proceedings of the National Academy of Sciences* **105**, 1118 (2008).
 - [8] P. Ronhovde and Z. Nussinov, *Phys. Rev. E* **81**, 046114 (2010).
 - [9] G. García-Pérez, M. Boguna, and S. M. Ángeles, *Nature Physics* **14**, 583 (2018).
 - [10] B. L. Chen, D. H. Hall, and D. B. Chklovskii, *Proceedings of the National Academy of Sciences of the United States of America* **103**, 4723 (2006).
 - [11] M. De Domenico, M. A. Porter, and A. Arenas, *Journal of Complex Networks* **3**, 159 (2015).
 - [12] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, *Nucleic acids research* **34**, D535 (2006).
 - [13] M. De Domenico, V. Nicosia, A. Arenas, and V. Latora, *Nature communications* **6**, 6864 (2015).
 - [14] M. De Domenico, A. Lancichinetti, A. Arenas, and M. Rosvall, *Physical Review X* **5**, 011027 (2015).
 - [15] K.-K. Kleineberg, L. Buzna, F. Papadopoulos, M. Boguná, and M. Á. Serrano, *Physical Review Letters* **118**, 218301 (2017).
 - [16] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, *Internet Mathematics* **6**, 29 (2009).
 - [17] J. Serrà, Á. Corral, M. Boguná, M. Haro, and J. L. Arcos, *Scientific reports* **2**, 521 (2012).
 - [18] J. Kunegis, in *Proceedings of the 22nd International Conference on World Wide Web* (ACM, 2013) pp. 1343–1350.
 - [19] M. Á. Serrano, M. Boguná, and F. Sagués, *Molecular biosystems* **8**, 843 (2012).
 - [20] T. Rolland, M. Taşan, B. Charlotteaux, S. J. Pevzner, Q. Zhong, N. Sahni, S. Yi, I. Lemmens, C. Fontanillo, R. Mosca, et al., *Cell* **159**, 1212 (2014).
 - [21] K. Claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, in *2009 Cybersecurity Applications Technology Conference for Homeland Security* (2009) pp. 205–211.
 - [22] J. Leskovec, J. Kleinberg, and C. Faloutsos, in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining* (ACM, 2005) pp. 177–187.
 - [23] R. Guimerà, S. Mossa, A. Turtschi, and L. A. N. Amaral, *Proceedings of the National Academy of Sciences* **102**, 7794 (2005).
 - [24] M. Ripeanu and I. Foster, in *Peer-to-Peer Systems*, edited by P. Druschel, F. Kaashoek, and A. Rowstron (Springer Berlin Heidelberg, Berlin, Heidelberg, 2002) pp. 85–93.
 - [25] L. Šubelj and M. Bajec, *The European Physical Journal B* **81**, 353 (2011).
 - [26] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, *Journal of Statistical Mechanics: Theory and Experiment* **2005**, P09008 (2005).
 - [27] A. Lancichinetti, S. Fortunato, and F. Radicchi, *Physical review E* **78**, 046110 (2008).
 - [28] F. Radicchi, *Phys. Rev. E* **97**, 022316 (2018).

- [29] F. Radicchi, *Nature Physics* **11**, 597 (2015).
- [30] S. Osat, A. Faqeesh, and F. Radicchi, *Nature Communications* **8**, 1540 (2017).
- [31] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin, *Nature* **464**, 1025 (2010).
- [32] A. Muscoloni, J. M. Thomas, S. Ciucci, G. Bianconi, and C. V. Cannistraci, *Nature Communications* **8**, 1615 (2017).
- [33] D. B. Johnson, *Journal of the ACM (JACM)* **24**, 1 (1977).
- [34] B. Karrer and M. E. Newman, *Physical Review E* **83**, 016107 (2011).

Chapter Six

Optimal percolation on multiplex networks

Contribution

As indicated in the contribution part of the paper this research is done equally by all the authors. As the main author I carried out the main part which is the simulation part i.e., designing and coding all the algorithms used in this paper which includes generalization of the optimal percolation algorithms from single graphs to multiplex networks.

ARTICLE

DOI: 10.1038/s41467-017-01442-2 **OPEN**

Optimal percolation on multiplex networks

Saeed Osat^{1,3}, Ali Fafeeh² & Filippo Radicchi²

Optimal percolation is the problem of finding the minimal set of nodes whose removal from a network fragments the system into non-extensive disconnected clusters. The solution to this problem is important for strategies of immunization in disease spreading, and influence maximization in opinion dynamics. Optimal percolation has received considerable attention in the context of isolated networks. However, its generalization to multiplex networks has not yet been considered. Here we show that approximating the solution of the optimal percolation problem on a multiplex network with solutions valid for single-layer networks extracted from the multiplex may have serious consequences in the characterization of the true robustness of the system. We reach this conclusion by extending many of the methods for finding approximate solutions of the optimal percolation problem from single-layer to multiplex networks, and performing a systematic analysis on synthetic and real-world multiplex networks.

¹Molecular Simulation Laboratory, Department of Physics, Faculty of Basic Sciences, Azarbaijan Shahid Madani University, Tabriz 53714-161, Iran. ²Center for Complex Networks and Systems Research, School of Informatics and Computing, Indiana University, Bloomington, IN 47408, USA. ³Quantum Complexity Science Initiative, Skolkovo Institute of Science and Technology, Skoltech Building 3, Moscow 143026, Russia. Correspondence and requests for materials should be addressed to F.R. (email: filiradi@indiana.edu)

A multiplex is a network in which nodes are connected through different types or flavors of pairwise edges^{1–3}. A convenient way to think of a multiplex is as a collection of network layers, each representing a specific type of edges. Multiplex networks are genuine representations for several real-world systems, including social^{4,5}, and technological systems^{6,7}. From a theoretical point of view, a common strategy to understand the role played by the co-existence of multiple network layers is based on a rather simple approach. Given a process and a multiplex network, one studies the process on the multiplex and on the single-layer projections of the multiplex (e.g., each of the individual layers, or the network obtained from aggregation of the layers). Recent research has demonstrated that ignoring the effective co-existence of different types of interactions in the study of a multiplex network may have dramatic consequences in the ability to model and predict properties of the system. Examples include dynamical processes, such as diffusion^{8,9}, epidemic spreading^{10–13}, synchronization¹⁴, and controllability¹⁵, as well as structural processes such as those typically framed in terms of percolation models^{16–29}.

The vast majority of the work on structural processes on multiplex networks have focused on ordinary percolation models where nodes (or edges) are considered either in a functional or in a non-functional state with homogenous probability³⁰. In this paper, we shift the focus on the optimal version of the percolation process: we study the problem of identifying the smallest set of nodes in a multiplex network such that, if these nodes are removed, the network is fragmented into many disconnected clusters with non-extensive sizes. We refer to the nodes belonging to this minimal set as structural nodes (SNs) of the multiplex network. The solution of the optimal percolation problem has

direct applicability in the context of robustness, representing the cheapest way to dismantle a network^{31–33}. The solution of the problem of optimal percolation is, however, important in other contexts, being equivalent to the best strategy of immunization to a spreading process, and also to the best strategy of seeding a network for some class of opinion dynamical models^{34–37}. Despite its importance, optimal percolation has been introduced and considered in the framework of single-layer networks only recently^{35,36}. Optimal percolation is a NP-complete problem³². Hence, on large networks, we can only use heuristic methods to find approximate solutions. Most of the research activity on this topic has indeed focused on the development of greedy algorithms^{31–33,35}.

Here we consider the generalization of optimal percolation to multiplex networks. Our generalization consists in the redefinition of the problem in terms of mutual connectedness¹⁶. To this end, we reframe several algorithms for optimal percolation in single-layer networks to obtain methods that consider the multiplex structure of networks as well. Basically all the algorithms we use provide coherent solutions to the problem, finding sets of SNs that are almost identical. Our main focus, however, is not on the development of new algorithms, but on understanding the consequences that arise from neglecting the multiplex nature of a network under an optimal percolation process. We compare the actual solution of the optimal percolation problem in a multiplex network with the solutions to the same problem for single-layer networks extracted from the multiplex system. We show that “forgetting” about the presence of multiple layers can be potentially dangerous, leading to the overestimation of the true robustness of the system mostly due to the identification of a very high number of false SNs. We reach this conclusion with a systematic analysis of both synthetic and real-world multiplex networks.

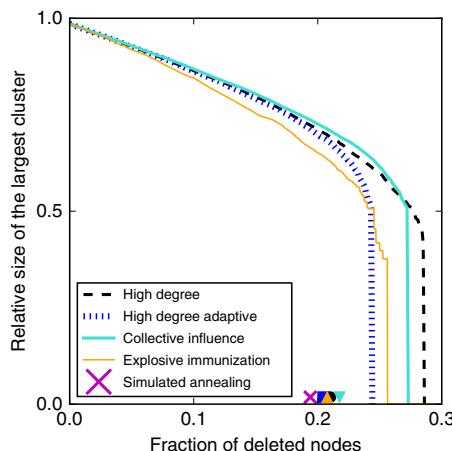


Fig. 1 Performance of different algorithms aimed at solving the optimal percolation problem. We consider a multiplex network with $N=10,000$ nodes. The multiplex is composed of two network layers generated independently according to the Erdős–Rényi model with average degree $\langle k \rangle = 5$. Each curve represents the relative size of the GMCC as a function of the relative number of nodes inserted in the set of SNs, thus removed from the multiplex. Colored markers indicate the effective fraction of nodes left in the set of SNs after a greedy post-processing technique is applied to the set found by the corresponding algorithm. The purple cross identifies instead the size of the set of SNs found through Simulated Annealing optimization. Please note that the ordinate value of the markers has no meaning; in all cases, the relative size of the largest cluster is smaller than $N^{1/2}$. Details on the implementation of the various algorithms are provided in Supplementary Notes 1, 2

Results

Identifying structural nodes in multiplex networks. We consider a multiplex network composed of N nodes arranged in two layers. Each layer is an undirected and unweighted network. Connections of the two layers are encoded in the adjacency matrices **A** and **B**. The generic element $A_{ij} = A_{ji} = 1$ if nodes i and j are connected in the first layer, whereas $A_{ij} = A_{ji} = 0$, otherwise. The same definition applies to the second layer, and thus to the matrix **B**. The aggregated network obtained from the superposition of the two layers is characterized by the adjacency matrix **C**, with generic elements $C_{ij} = A_{ij} + B_{ij} - A_{ij}B_{ij}$. We focus our attention on clusters of mutually connected nodes¹⁶: two nodes in a multiplex network are mutually connected, and thus part of the same cluster of mutually connected nodes, only if they are connected by at least a path, composed of nodes within the same cluster, in every layer of the system. In particular, we focus our attention on the largest among these cluster, usually referred to as the giant mutually connected cluster (GMCC). Our goal is to find the minimal set of nodes such that, if removed from the multiplex, no mutual cluster with a size greater than $N^{1/2}$ is found in the network. This is a common prescription, yet not the only one possible, to ensure that all clusters have non-extensive sizes in systems with a finite number of elements³⁵. Whenever we consider single-layer networks, the above prescription applies to the single-layer clusters in the same exact way.

We generalize most of the algorithms devised to find approximate solutions to the optimal percolation problem in single-layer networks to multiplex networks^{31–33,35,36}. Details on the implementation of the various methods are provided in the Supplementary Note 1. We stress that the generalization of these methods is not trivial at all. For instance, most of the greedy

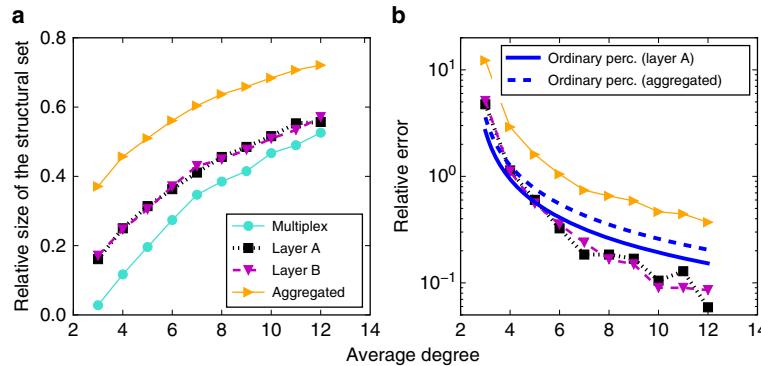


Fig. 2 Optimal percolation problem on synthetic multiplex networks. **a** We consider multiplex networks with $N=1,000$ and layers generated independently according to the Erdős–Rényi model with average degree $\langle k \rangle$. We estimate the relative size of the set of SNs on the multiplex as a function of $\langle k \rangle$ (turquoise circles), and compare it with the same quantity but estimated on the individual layers (black squares and purple triangles) or the aggregated (orange triangles). **b** The relative errors of single-layer estimates of the size of the structural set with respect to the ground-truth value provided by the multiplex estimate. Colors and symbols are the same as those used in **a**. The blue curves with no markers represent instead the theoretically expected behavior for an ordinary percolation process¹⁶

methods use node degrees as crucial ingredients to calculate and assign scores to each of the nodes, and then remove nodes with respect to their scores. In a multiplex network, however, a node has multiple degree values, one for every layer. In this respect, it is not clear what is the most effective way of combining these numbers to assign a single score to a node: they may be summed, thus obtaining a number approximately equal to the degree of the node in the aggregated network derived from the multiplex, but also multiplied, or combined in more complicated ways. We find that the results of the various algorithms are not particularly sensitive to this choice, provided that the simple but effective post-processing technique considered in refs. ^{31–33} is applied to the set of SNs found by a given method. In Fig. 1, for example, we show the performance of several greedy algorithms when applied to a multiplex network composed of two layers generated independently according to the Erdős–Rényi (ER) model. Although the mere application of an algorithm may lead to different estimates of the size of the set of SNs, if we greedily remove from these sets the nodes that do not increase the size of the GMCC to the predefined sub-linear threshold ($N^{1/2}$)^{31–33} (Supplementary Note 2), the sets obtained after this post-processing technique have almost identical sizes (Supplementary Figs. 1–4).

As Fig. 1 clearly shows, the best results, in the sense that the size of the set of SNs is minimal, is found with a simulated annealing (SA) optimization strategy³² (see details in the Supplementary Note 1). The fact that the SA method is outperforming score-based algorithms is not surprising. SA actually represents one of the best strategies that one can apply in hard-optimization tasks. In our case, it provides us with a reasonable upper bound on the size of the set of SNs that can be identified in a multiplex network. The second advantage of SA in our context is that it does not rely on ambiguous definitions of ingredients (e.g., node degree). Despite its better performance, SA has a serious drawback in terms of computational speed. As a matter of fact, the algorithm can be applied only to multiplex networks with moderate sizes. As here we are interested in understanding properties of the optimal percolation problem in multiplex networks, the analysis presented in the main text of the paper is entirely based on results obtained through SA optimization. This provides us with a solid ground to support our statements. Extending the analysis of score-based algorithms to larger multiplex networks leads to qualitatively similar results (Supplementary Note 3, Supplementary Figs. 5–8).

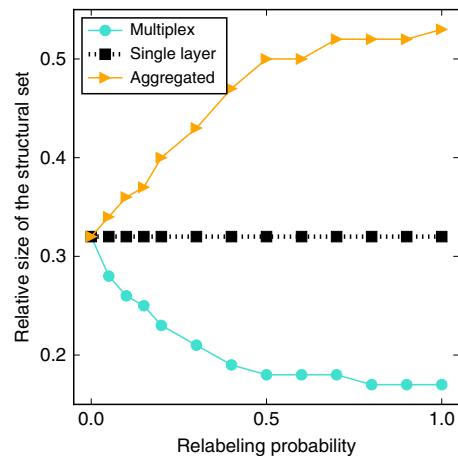


Fig. 3 Dependence of the size of the structural set on edge overlap and interlayer degree-degree correlation. We consider multiplex networks, in which, initially, both layers are a copy of a random network generated according to the Erdős–Rényi model with $N=1,000$ nodes and average degree $\langle k \rangle = 5$. Then, in one of the layers, each node is selected to switch its label with another randomly chosen node with a certain probability α . We determine, as a function of α , the mean value of the relative size of the set of SNs over 100 realizations of the SA algorithm on the multiplex network

The size of the set of structural nodes. We consider the relative size of the set of SNs, denoted by q_s , for a multiplex composed of two independently fabricated ER network layers as a function of their average degree $\langle k \rangle$. We compare the results obtained applying the SA algorithm to the multiplex, namely q_M , with those obtained using SA on the individual layers, i.e., q_A and q_B , or the aggregated network generated from the superposition of the two layers, i.e., q_S . By definition, we expect that $q_M \leq q_A \simeq q_B \leq q_S$. What we do not know, however, is how bad/good are the measures q_A , q_B and q_S in the prediction of the effective robustness of the multiplex q_M . For ordinary random percolation on ER multiplex networks with negligible overlap, we know that $q_M \simeq 1 - 2.4554/\langle k \rangle$ ¹⁶,

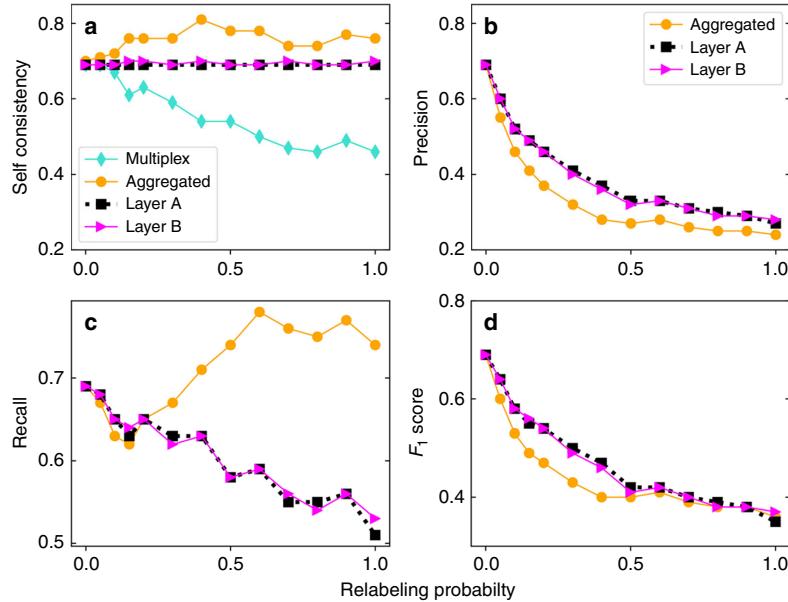


Fig. 4 Comparison between structural sets obtained for different network representations. We consider the multiplex networks described in Fig. 3 and the sets of SNs found for the multiplex and single-layer based representations of these networks. **a** As the set of SNs found in different instances of the optimization algorithm are different from each other, we first quantify the self-consistency of those solutions across 100 independent runs of the SA algorithm. We then assume that the multiplex representation provides the ground-truth classification of the nodes. We compare the results of the other representations with the ground truth by measuring their precision **b**, their sensitivity or recall **c**, and their F_1 score **d**.

$q_A \simeq q_B \simeq 1 - 1/\langle k \rangle^{38}$, and $q_S \simeq 1 - 1/(2\langle k \rangle)^{38}$. Relative errors are therefore $\epsilon_A \simeq \epsilon_B \simeq (2.4554 - 1)/(\langle k \rangle - 2.4554)$, and $\epsilon_S \simeq (2.4554 - 1/2)/(\langle k \rangle - 2.4554)$. We find that the relative error for optimal percolation behaves more or less in the same way as that of ordinary percolation (Fig. 2b), noting that, as $\langle k \rangle$ is increased, the decrease in the relative error associated with the individual layers is slightly faster than what expected for ordinary percolation. The relative error associated with the aggregated network is larger than the one expected from the theory of ordinary percolation. As shown in Fig. 2a, for sufficiently large $\langle k \rangle$, dismantling an ER multiplex network is almost as hard as dismantling any of its constituent layers.

Edge overlap and degree correlations. Next, we test the role played by edge overlap and layer-to-layer degree correlation in the optimal percolation problem. These are the ingredients that dramatically change the nature of the ordinary percolation transition in multiplex networks^{26,39–43}. In Fig. 3, we report the results of a simple analysis. We take advantage of the model introduced in ref. ⁴⁴. This is one of the simplest models able to tune a system from a multiplex to a simplex topology. The system is composed of two identical network layers. Nodes in one of the two layers are relabeled with a certain probability α . For $\alpha = 0$, multiplex, aggregated network and single-layer graphs are all identical. For $\alpha = 1$, the networks are analogous to those considered in the previous section. We note that this model does not allow to disentangle the role played by edge overlap among layers and the one played by the correlation of node degrees. For $\alpha = 0$, edge overlap amounts to 100%, and there is a one-to-one match between the degree of a node in one layer and its degree in the other layer. As α increases, both edge overlap and degree correlation decrease simultaneously. As it is apparent from the results of Fig. 3, the system reaches the multiplex regime for very small values of α , in the sense that the relative size of the set of SNs

deviates instantly from its value for $\alpha = 0$. This is in line with what already found in the context of ordinary percolation processes in multiplex networks: as soon as there is a finite fraction of edges that are not shared by the two layers, the system behaves exactly as a multiplex^{26,39–43}.

Accuracy and sensitivity. So far, we focused our attention only on the size of the set of SNs. We neglected, however, any analysis regarding the identity of the nodes that actually compose this set. To proceed with such an analysis, we note that different runs of the SA algorithm (or any algorithm with stochastic features) generally produce slightly different sets of SNs, even if they all have almost identical sizes. The issue is not related to the optimization technique, rather to the existence of degenerate solutions to the problem. In this respect, we work with the quantities p_i , each of which describes the probability that a node i appears in the set of SNs in a realization of the detection method (here, the SA algorithm). This treatment takes into account the fact that a node may belong to the set of SNs in a number of realizations of the detection method and may be absent from this set in some other realizations.

We define self-consistency of a SN-detection method as $S = [\sum_i p_i^2]/[\sum_i p_i]$, which describes the ratio of the expected overlap between two SNs obtained from two independent realizations of the detection method to the expected size of the SN. If the set of SNs is identical across different runs, then $S = 1$. The minimal value we can observe is $S = Q/N$, assuming that the size of the structural set is equal to Q in all runs, but nodes belonging to this set are changing all the times, so that for every node i we have $p_i = Q/N$. As reported in Fig. 4a, self-consistency S assumes high values for single-layer representations of the network, even for synthetic multiplex networks. On the other hand, S decreases significantly as the overlap and interlayer degree correlations decrease (Fig. 4a). Low S values for

Table 1 Optimal percolation on real multiplex networks

Network	Layers	N	Multiplex		Single layers						Aggregate					
			q_M	S	q_A	P_A	R_A	$F_1^{(A)}$	q_B	P_B	R_B	$F_1^{(B)}$	q_S	P_S	R_S	$F_1^{(S)}$
Air transportation ²⁶	American Air.—Delta	84	0.12	0.85	0.14	0.58	0.70	0.63	0.32	0.29	0.79	0.42	0.35	0.32	0.92	0.47
	American Air.—United	73	0.10	0.99	0.16	0.32	0.52	0.40	0.14	0.68	1.00	0.81	0.25	0.39	1.00	0.56
C. Elegance ^{47,48}	United—Delta	82	0.10	1.00	0.27	0.23	0.62	0.34	0.12	0.80	1.00	0.89	0.33	0.30	1.00	0.46
	Electric—Chem. Mon.	238	0.09	0.69	0.16	0.41	0.71	0.52	0.26	0.22	0.60	0.32	0.35	0.21	0.79	0.33
	Electric—Chem. Pol.	252	0.12	0.79	0.15	0.50	0.63	0.56	0.39	0.24	0.78	0.37	0.45	0.22	0.82	0.35
Arxiv ⁴⁹	Chem. Mon.—Chem. Pol.	259	0.25	0.82	0.28	0.69	0.77	0.73	0.39	0.51	0.79	0.62	0.42	0.48	0.80	0.60
	physics.data-an—cond-mat.dis-nn	1400	0.05	0.78	0.10	0.38	0.77	0.51	0.07	0.55	0.75	0.63	0.13	0.31	0.81	0.45
	physics.data-an—cond-mat.stat-mech	709	0.03	0.73	0.08	0.23	0.67	0.34	0.03	0.64	0.72	0.68	0.09	0.22	0.74	0.34
Drosophila M. ^{50,51}	cond-mat.dis-nn—cond-mat.stat-mech	499	0.02	0.50	0.06	0.13	0.46	0.20	0.04	0.23	0.51	0.32	0.09	0.13	0.65	0.22
	Direct—Supp. Gen.	676	0.01	0.62	0.07	0.12	0.60	0.20	0.11	0.09	0.64	0.16	0.19	0.07	0.87	0.13
	Direct—Add. Gen.	626	0.01	0.81	0.07	0.06	0.64	0.11	0.09	0.05	0.59	0.09	0.16	0.04	0.87	0.08
Homo S. ^{48,50}	Supp. Gen.—Add. Gen.	557	0.09	0.82	0.14	0.44	0.74	0.55	0.12	0.50	0.70	0.58	0.20	0.35	0.80	0.49
	Direct—Supp. Gen.	4465	0.05	0.72	0.16	0.20	0.73	0.31	0.13	0.23	0.64	0.34	0.27	0.15	0.89	0.26
	Physical—Supp. Gen.	5202	0.05	0.75	0.15	0.23	0.77	0.35	0.13	0.22	0.63	0.33	0.26	0.16	0.90	0.27

From left to right we report the following information. The first three columns contain the name of the system, the identity of the layers, and the number of nodes of the network. The fourth and fifth columns are results obtained from the optimal percolation problem studied on the multiplex network, and contain information about the relative size q_M and self-consistency metric S of the set of SNs. Then, we report results obtained for the first single-layer network of the multiplex, namely the fraction q_A of nodes in the structural set, the precision P_A , the recall R_A , and the F_1 score of the set of SNs of the first layer. The next four columns are identical to those, but refer to the second layer. Finally, the four rightmost columns contain information about the fraction q_S of nodes in the structural set, P_S precision, R_S recall, and the F_1 score of the set of SNs for the aggregated network obtained from the superposition of the two layers. All results have been obtained with 100 independent instances of the SA optimization algorithm

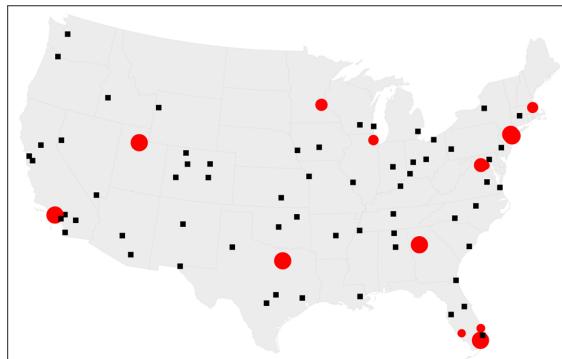


Fig. 5 Optimal percolation on multiplex transportation networks. We consider the multiplex network of US domestic flights operated in January 2014 by American Airlines and Delta. Red circles represent nodes that were identified as members of the set of structural nodes in different realizations of the optimal percolation on the multiplex representation of the network. The size of each circle is proportional to the probability of finding that node in the set of SNs. All other airports in the multiplex are represented as black squares. Interestingly, not all the 14 structural nodes match the top 14 busiest “hubs” (https://en.wikipedia.org/wiki/List_of_the_busiest_airports_in_the_United_States), nor the probabilities follow the same order as the flight traffic of these airports. The results have been obtained with 100 independent instances of the SA optimization algorithm

multiplexes with small overlap and correlation together with the small sizes of their set of SNs (Fig. 2) suggest that in such networks many slightly different SN sets may exist.

Next, we turn our attention on quantifying how the sets of SNs identified in single-layer or aggregated networks are

representative of the ground-truth sets found on multiplex networks. We denote by p_i and w_i the probability that node i is found within the set of SNs of, respectively, a multiplex network (ground truth) and a specific single-layer representation of that multiplex. To compare the sets represented by w_i with the ground-truth sets, we adopt three standard metrics in information retrieval^{45,46}, namely precision, recall and the Van Rijsbergen’s F_1 score. Precision is defined as $P = [\sum_i p_i w_i]/[\sum_i w_i]$, i.e., the ratio of the expected number of correctly detected SNs to the expected total number of detected SNs. Recall is defined as $R = [\sum_i p_i w_i]/[\sum_i p_i]$, i.e., the ratio of the expected number of correctly detected SNs to the expected number of actual SNs of the multiplex. We note that the self-consistency we previously defined corresponds to precision and recall of the ground-truth set with respect to itself, thus providing a base line for the interpretation of the results. The F_1 score defined as $F_1 = (2)/(1/P + 1/R)$ provides a balanced measure in terms of P and R . As Fig. 4b shows, P deteriorates as the edge overlap and interlayer degree correlation decrease. In particular, when overlap and correlation between the layers of the multiplex network are not large, precision values for the sets of SNs identified in single layers or in the superposition of the layers are quite small ($P \approx 0.3$), even smaller than the ratio of the q_M of the multiplex to the q of any of these sets (Fig. 3). This means that, when the multiplex nature of the system is neglected, two systematic errors are committed. First, the number of SNs is greatly overestimated; second, a significant number of the true SNs of the multiplex are not identified. The quantity R , on the other hand, behaves differently for single-layer and aggregated networks (Fig. 4c). In single layers, we see that R systematically decreases as the relabeling probability increases. The structural set of nodes obtained on the superposition of the layers instead provides large values of R . This is not due to a good performance rather to the fact that the set of SNs identified on the aggregated network is

very large (Fig. 3), and it is further supported by the results of Fig. 4c, d, where large R values do not correspond to high F_1 scores.

Real-world multiplex networks. In Table 1, we present summary statistics of the solution of the optimal percolation problem studied on several real-world multiplex networks generated from empirical data. For most of these networks, we find high values of self-consistency among solutions. This implies that there is a certain small group of nodes that have a major importance in the robustness of such real-world networks to the optimal percolation process. For most of the networks, the F_1 scores are low, indicating that on real-world networks we loose essential information about the optimal percolation problem if the multiplex structure is not taken into account.

To provide a practical case study with an intuitive interpretation, we depict in Fig. 5 the solution of the optimal percolation problem on a multiplex network describing air transportation in the United States. SA identifies always 10 airports in the set of SNs of this network. There is a slight variability among different instances of the SA optimization, with a total of 14 distinct airports appearing in the structural set at least once over 100 SA instances. However, changes in the SN set from run to run mostly regard airports in the same geographical region. Overall, airports in the structural set are scattered homogeneously across the country, suggesting that the GMCC of the network mostly relies on hubs serving specific geographical regions, rather than global hubs in the entire transportation system. For instance, the probabilities that describe the membership of the airports to the set of SNs do not strictly follow the same order as that of the recorded flight traffics; nor merely the number of connections of the airports (not shown) is sufficient to determine the SNs.

Discussion

In this paper, we studied the optimal percolation problem on multiplex networks. The problem regards the detection of the minimal set of nodes (i.e., the set of structural nodes, SNs) such that, if its members are removed from the network, the network is dismantled. The solution to the problem provides important information on the microscopic parts that should be maintained in a functional state to keep the overall system functioning, in a scenario of maximal stress. Our study focused mostly on the characterization of the SN sets of a given multiplex network in comparison with those found on the single-layer projections of the same multiplex, i.e., in a scenario where one “forgets” about the multiplex nature of the system. Our results demonstrate that, generally, multiplex networks have considerably smaller sets of SNs compared to the SN sets of their single-layer based network representations. The error committed when relying on single-layer representations of the multiplex does not regard only the size of the SN sets, but also the identity of the SNs. Both issues emerge in the analysis of synthetic network models, where edge overlap and/or interlayer degree-degree correlations seem to fully explain the amount of discrepancy between the SN set of a multiplex and the SN sets of its single-layer based representations. These issues are apparent also in many of the real-world multiplex networks we analyzed. Overall, we conclude that neglecting the multiplex structure of a network system subjected to maximal structural stress may result in significant inaccuracies about its robustness.

Data availability. Real multiplex networks analyzed in the paper have been constructed using data publicly available on the Web (see references in Table 1). The source code of the

implementation of the various algorithms used in the paper is available from the authors upon request.

Received: 29 May 2017 Accepted: 19 September 2017

Published online: 16 November 2017

References

- Boccaletti, S. et al. The structure and dynamics of multilayer networks. *Phys. Rep.* **544**, 1–122 (2014).
- Kivelä, M. et al. Multilayer networks. *J. Complex Networks* **2**, 203–271 (2014).
- Lee, K.-M., Min, B. & Goh, K.-I. Towards real-world complexity: an introduction to multiplex networks. *EPJ B* **88**, 1–20 (2015).
- Szell, M., Lambiotte, R. & Thurner, S. Multirelational organization of large-scale social networks in an online world. *Proc. Natl Acad. Sci. USA* **107**, 13636–13641 (2010).
- Mucha, P. J., Richardson, T., Macon, K., Porter, M. A. & Onnela, J.-P. Community structure in time-dependent, multiscale, and multiplex networks. *Science* **328**, 876–878 (2010).
- Barthélemy, M. Spatial networks. *Phys. Rep.* **499**, 1–101 (2011).
- Cardillo, A. et al. Emergence of network features from multiplexity. *Sci. Rep.* **3**, 1344 (2013).
- Gómez, S. et al. Diffusion dynamics on multiplex networks. *Phys. Rev. Lett.* **110**, 028701 (2013).
- De Domenico, M., Solé-Ribalta, A., Gómez, S. & Arenas, A. Navigability of interconnected networks under random failures. *Proc. Natl Acad. Sci. USA* **111**, 8351–8356 (2014).
- Dickison, M., Havlin, S. & Stanley, H. E. Epidemics on interconnected networks. *Phys. Rev. E* **85**, 066109 (2012).
- Saumell-Mendiola, A., Serrano, M. A. & Boguñá, M. Epidemic spreading on interconnected networks. *Phys. Rev. E* **86**, 026106 (2012).
- Granell, C., Gómez, S. & Arenas, A. Dynamical interplay between awareness and epidemic spreading in multiplex networks. *Phys. Rev. Lett.* **111**, 128701 (2013).
- De Domenico, M., Granell, C., Porter, M. A. & Arenas, A. The physics of spreading processes in multilayer networks. *Nat. Phys.* **12**, 901–906 (2016).
- del Genio, C. I., Gómez-Gardeñes, J., Bonamassa, I. & Boccaletti, S. Synchronization in networks with multiple interaction layers. *Sci. Adv.* **2**, 11 (2016).
- Pósfai, M., Gao, J., Cornelius, S. P., Barabási, A.-L. & D’Souza, R. M. Controllability of multiplex, multi-time-scale networks. *Phys. Rev. E* **94**, 032316 (2016).
- Buldyrev, S. V., Parshani, R., Paul, G., Stanley, H. E. & Havlin, S. Catastrophic cascade of failures in interdependent networks. *Nature* **464**, 1025–1028 (2010).
- Parshani, R., Buldyrev, S. V. & Havlin, S. Interdependent networks: reducing the coupling strength leads to a change from a first to second order percolation transition. *Phys. Rev. Lett.* **105**, 048701 (2010).
- Parshani, R., Rozenblat, C., Ietri, D., Ducruet, C. & Havlin, S. Inter-similarity between coupled networks. *EPL* **92**, 68002 (2011).
- Baxter, G., Dorogovtsev, S., Goltsev, A. & Mendes, J. Avalanche collapse of interdependent networks. *Phys. Rev. Lett.* **109**, 248701 (2012).
- Watanabe, S. & Kabashima, Y. Cavity-based robustness analysis of interdependent networks: Influences of intranetwork and internetwork degree-degree correlations. *Phys. Rev. E* **89**, 012808 (2014).
- Son, S.-W., Bizhani, G., Christensen, C., Grassberger, P. & Paczuski, M. Percolation theory on interdependent networks based on epidemic spreading. *EPL* **97**, 16006 (2012).
- Min, B., Do Yi, S., Lee, K.-M. & Goh, K.-I. Network robustness of multiplex networks with interlayer degree correlations. *Phys. Rev. E* **89**, 042811 (2014).
- Bianconi, G., Dorogovtsev, S. N. & Mendes, J. F. F. Mutually connected component of networks of networks with replica nodes. *Phys. Rev. E* **91**, 012804 (2015).
- Bianconi, G. & Dorogovtsev, S. N. Multiple percolation transitions in a configuration model of a network of networks. *Phys. Rev. E* **89**, 062814 (2014).
- Radicchi, F. & Arenas, A. Abrupt transition in the structural formation of interconnected networks. *Nat. Phys.* **9**, 717–720 (2013).
- Radicchi, F. Percolation in real interdependent networks. *Nat. Phys.* **11**, 597–602 (2015).
- Cellai, D. & Bianconi, G. Multiplex networks with heterogeneous activities of the nodes. *Phys. Rev. E* **93**, 032302 (2016).
- Bianconi, G. & Radicchi, F. Percolation in real multiplex networks. *Phys. Rev. E* **94**, 060301 (2016).
- Radicchi, F. & Bianconi, G. Redundant interdependencies boost the robustness of multiplex networks. *Phys. Rev. X* **7**, 011013 (2017).

30. Stauffer, D. & Aharony, A. *Introduction to Percolation Theory* (Taylor and Francis, London, 1994).
31. Mugisha, S. & Zhou, H.-J. Identifying optimal targets of network attack by belief propagation. *Phys. Rev. E* **94**, 012305 (2016).
32. Braustein, A., Dall'Asta, L., Semerjian, G. & Zdeborov, L. Network dismantling. *Proc. Natl Acad. Sci. USA* **113**, 12368–12373 (2016).
33. Zdeborová, L., Zhang, P. & Zhou, H.-J. Fast and simple decycling and dismantling of networks. *Sci. Rep.* **6**, 37954 (2016).
34. Altarelli, F., Braustein, A., Dall'Asta, L. & Zecchina, R. Large deviations of cascade processes on graphs. *Phys. Rev. E* **87**, 062115 (2013).
35. Clusella, P., Grassberger, P., Pérez-Reche, F. J. & Politi, A. Immunization and targeted destruction of networks using explosive percolation. *Phys. Rev. Lett.* **117**, 208301 (2016).
36. Morone, F. & Makse, H. A. Influence maximization in complex networks through optimal percolation. *Nature* **524**, 65–68 (2015).
37. Pei, S., Teng, X., Shaman, J., Morone, F. & Makse, H. A. Efficient collective influence maximization in cascading processes with first-order transitions. *Sci. Rep.* **7**, 45240 (2017).
38. Molloy, M. & Reed, B. A critical point for random graphs with a given degree sequence. *Random Struct. Algor.* **6**, 161–180 (1995).
39. Cellai, D., López, E., Zhou, J., Gleeson, J. P. & Bianconi, G. Percolation in multiplex networks with overlap. *Phys. Rev. E* **88**, 052811 (2013).
40. Bianconi, G. Statistical mechanics of multiplex networks: Entropy and overlap. *Phys. Rev. E* **87**, 062806 (2013).
41. Min, B., Lee, S., Lee, K.-M. & Goh, K.-I. Link overlap, viability, and mutual percolation in multiplex networks. *Chaos Soliton. Fract.* **72**, 49–58 (2015).
42. Baxter, G. J., Bianconi, G., da Costa, R. A., Dorogovtsev, S. N. & Mendes, J. F. Correlated edge overlaps in multiplex networks. *Phys. Rev. E* **94**, 012303 (2016).
43. Cellai, D., Dorogovtsev, S. N. & Bianconi, G. Message passing theory for percolation models on multiplex networks with link overlap. *Phys. Rev. E* **94**, 032301 (2016).
44. Bianconi, G. & Dorogovtsev, S. N. Percolation in networks of networks with random matching of nodes in different layers. Preprint at <https://arxiv.org/abs/1411.4160> (2014).
45. Chu, W. & Lin, T. Y. *Foundations and Advances in Data Mining* Vol. 180 (Springer-Verlag, Berlin Heidelberg, 2005).
46. Maino, G. & Foresti, G. L. *Image Analysis and Processing -- ICIAP 2011*, Vol. 6979 (Springer-Verlag, Berlin Heidelberg, 2011).
47. Chen, B. L., Hall, D. H. & Chklovskii, D. B. Wiring optimization can relate neuronal structure and function. *Proc. Natl Acad. Sci. USA* **103**, 4723–4728 (2006).
48. De Domenico, M., Porter, M. A. & Arenas, A. Muxviz: a tool for multilayer analysis and visualization of networks. *J. Complex Netw.* **3**, cnu038 (2014).
49. De Domenico, M., Lancichinetti, A., Arenas, A. & Rosvall, M. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys. Rev. X* **5**, 011027 (2015).
50. Stark, C. et al. Biogrid: a general repository for interaction datasets. *Nucleic Acids Res.* **34**, D535–D539 (2006).
51. De Domenico, M., Nicosia, V., Arenas, A. & Latora, V. Structural reducibility of multilayer networks. *Nat. Commun.* **6**, 6864 (2015).

Acknowledgements

A.F. and F.R. acknowledge support from the US Army Research Office (W911NF-16-1-0104). F.R. acknowledges support from the National Science Foundation (Grant CMMI-1552487).

Author contributions

All authors contributed to all aspects of this work.

Additional information

Supplementary Information accompanies this paper at doi:[10.1038/s41467-017-01442-2](https://doi.org/10.1038/s41467-017-01442-2).

Competing interests: The authors declare no competing financial interests.

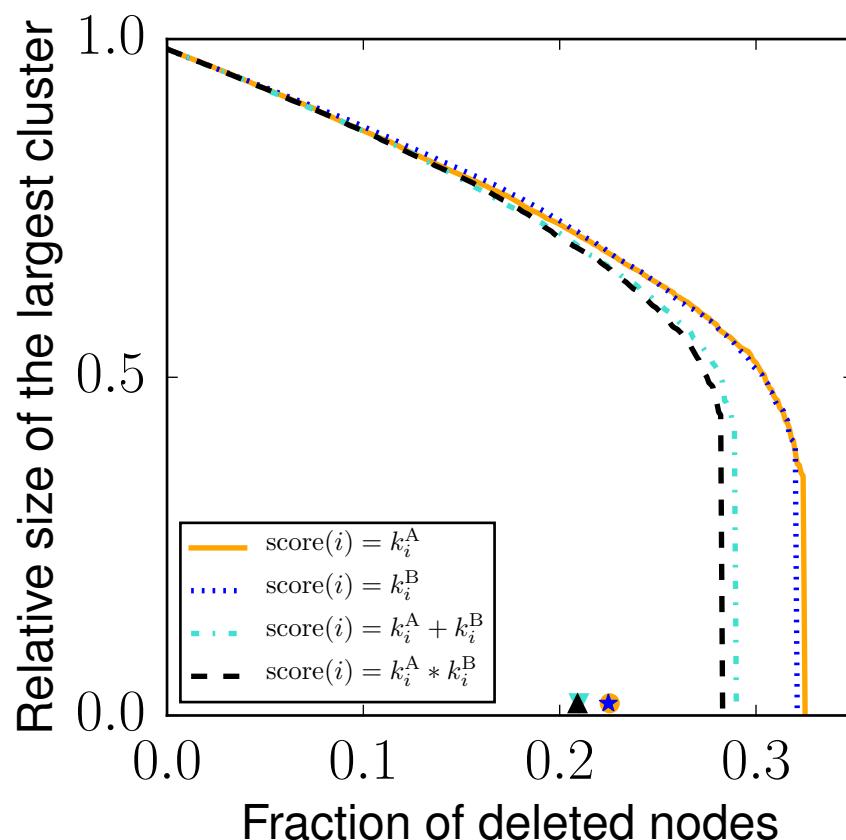
Reprints and permission information is available online at <http://npg.nature.com/reprintsandpermissions/>

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

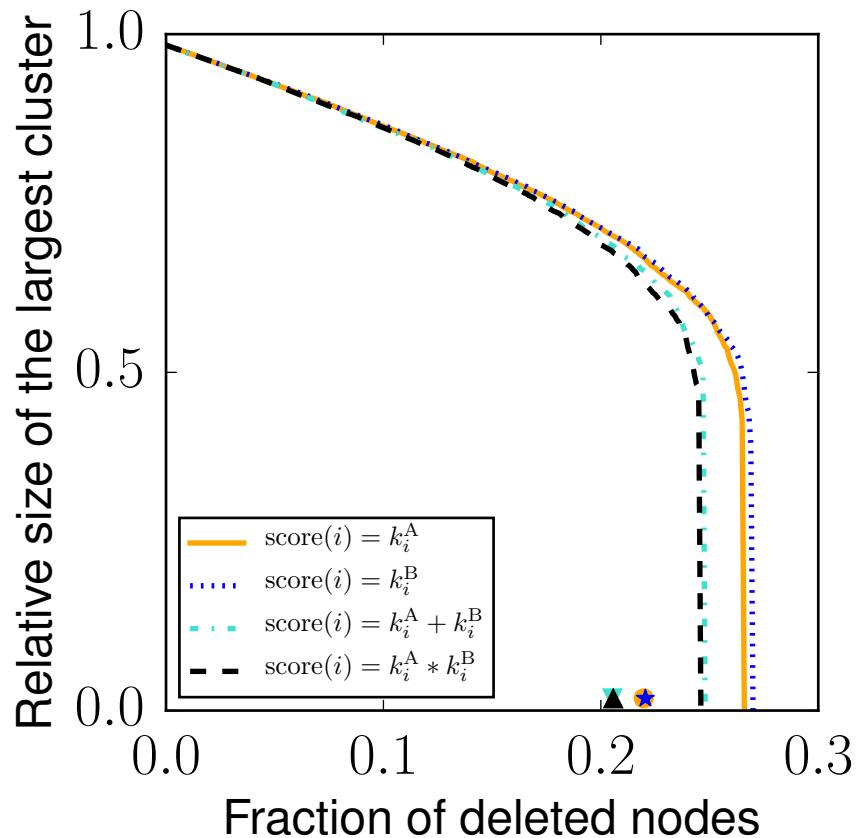


Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

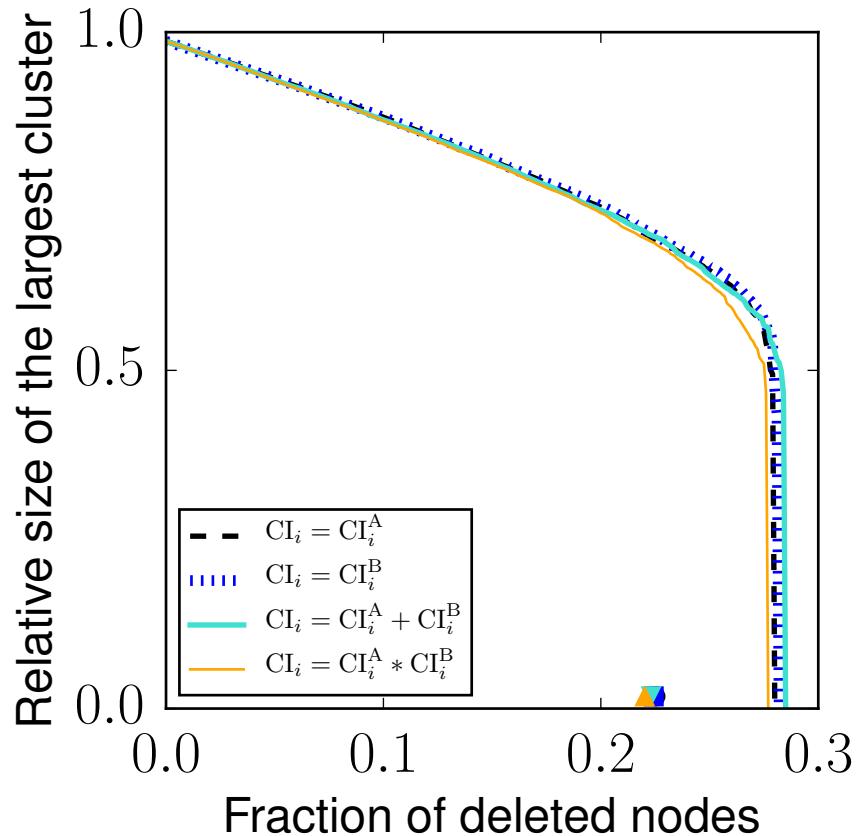
© The Author(s) 2017



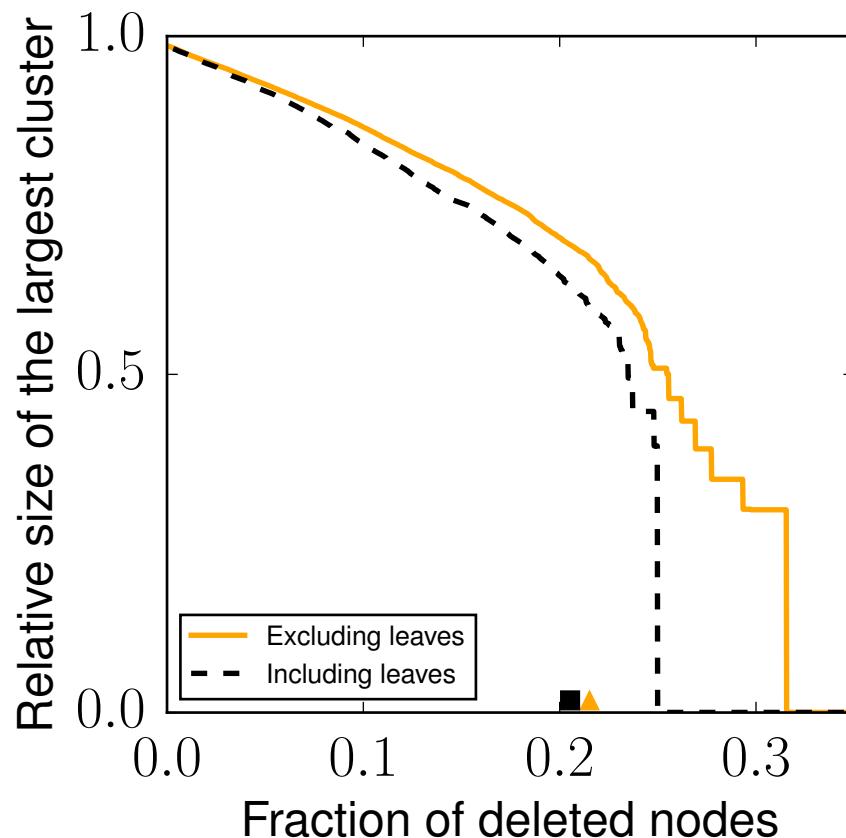
Supplementary Figure 1. Optimal percolation for a multiplex network using the High Degree (HD) algorithm. We consider a multiplex network composed of two identical layers with $N = 10000$ nodes generated according to the Erdős–Rényi model with average degree $\langle k \rangle = 5.0$. Different line styles correspond to four different methods of defining node scores in multiplex networks. Markers are the dismantling fraction obtained with these four methods when combined with the Greedy Reinserting (GR) procedure (see [Supplementary Note 2](#)).



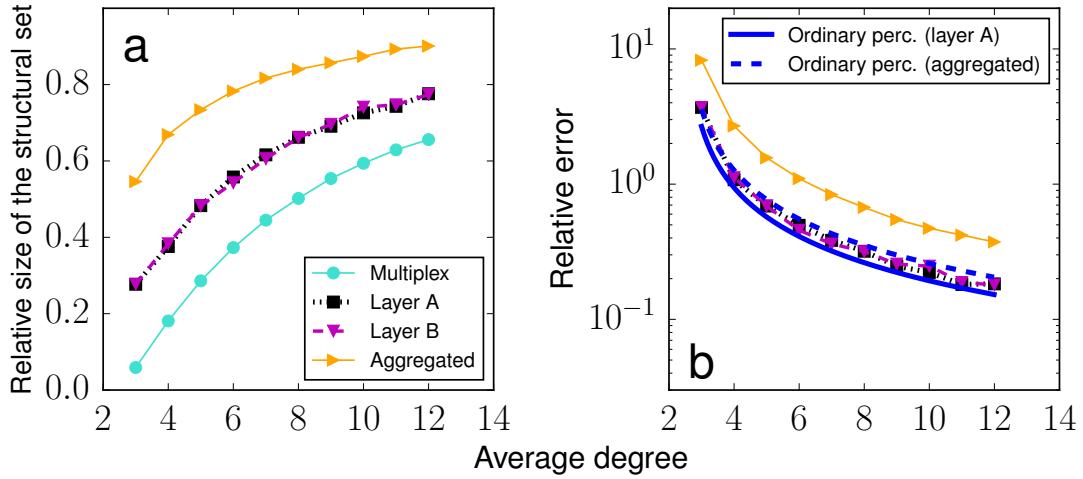
Supplementary Figure 2. Optimal percolation for a multiplex network using the High Degree Adaptive (HDA) algorithm. We consider a multiplex network composed of two identical layers with $N = 10000$ nodes generated according to the Erdős–Rényi model with average degree $\langle k \rangle = 5.0$. Different line styles correspond to four different methods of defining node scores in multiplex networks. Markers are the dismantling fraction obtained with these four methods when combined with the Greedy Reinserting (GR) procedure (see [Supplementary Note 2](#)).



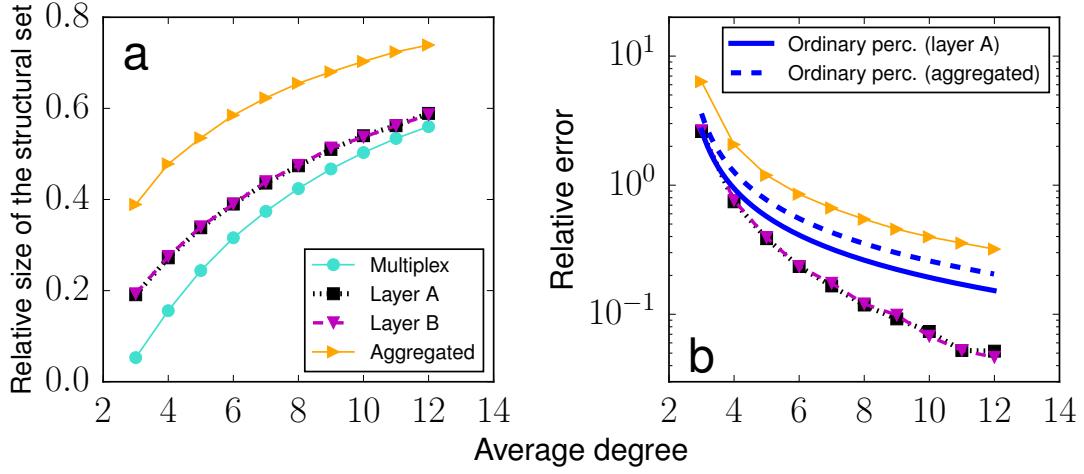
Supplementary Figure 3. Optimal percolation for a multiplex network using the Collective Influence (CI) algorithm. We consider a multiplex network composed of two identical layers with $N = 10000$ nodes generated according to the Erdős–Rényi model with average degree $\langle k \rangle = 5.0$. Different line styles correspond to four different methods of defining CI scores in multiplex networks. Markers are the dismantling fraction obtained with these four methods when combined with the Greedy Reinserting (GR) procedure (see [Supplementary Note 2](#)).



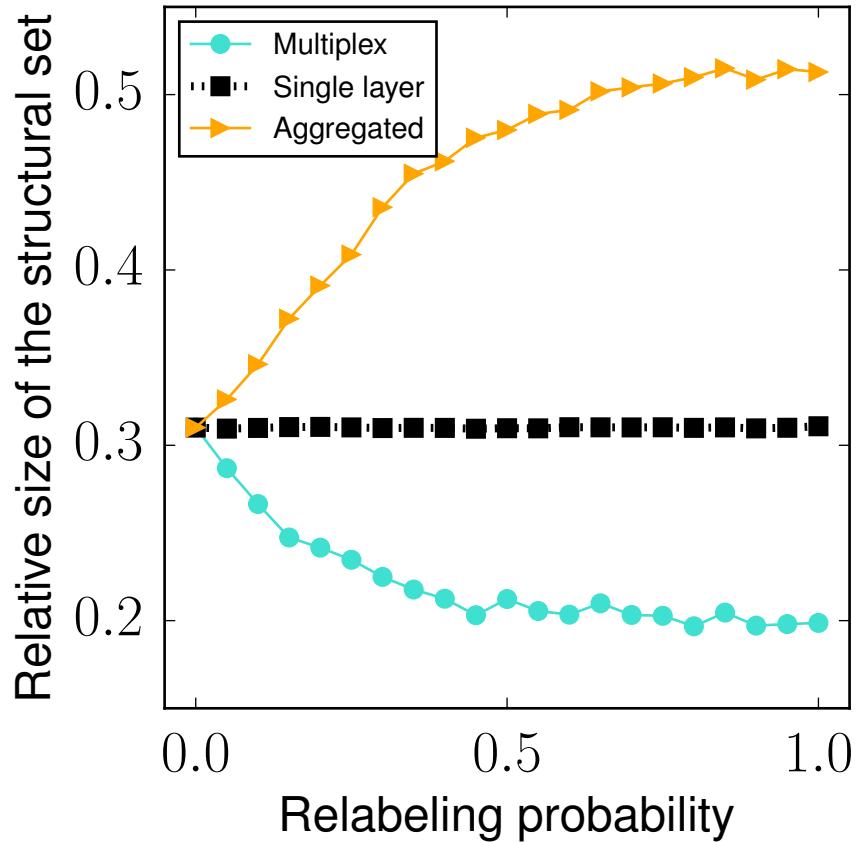
Supplementary Figure 4. Optimal percolation for a multiplex network using the Explosive Immunization (EI) algorithm. We consider a multiplex network composed of two identical layers with $N = 10000$ nodes generated according to the Erdős–Rényi model with average degree $\langle k \rangle = 5.0$. Different line styles correspond to two different kernels: Eq. 3 that does not exclude the leaves and Eq. 3 which excludes the effect of leaves by replacing $\sqrt{|M|}$ with $\sqrt{|M|} - 1$. Markers are the dismantling fraction obtained with these two methods when combined with the Greedy Reinserting (GR) procedure (see [Supplementary Note 2](#)).



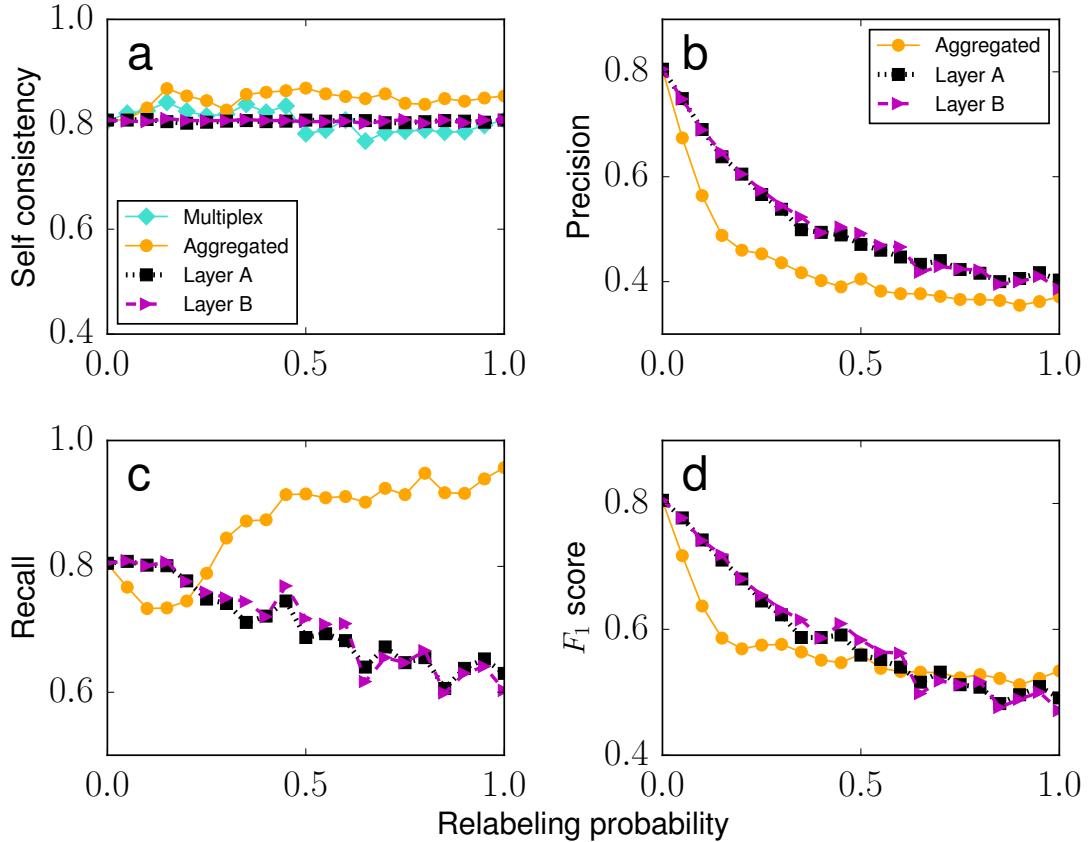
Supplementary Figure 5. Optimal percolation problem in synthetic multiplex networks using HD algorithm. a) We consider multiplex networks with $N = 10^5$ and layers generated independently according to the Erdős–Rényi model with average degree $\langle k \rangle$. We estimate the relative size of the set of SNs on the multiplex as a function of $\langle k \rangle$ (turquoise circles), and compare it with the same quantity but estimated on the individual layers (black squares and purple triangles) or the aggregated (orange triangles). b) The relative errors of single-layer estimates of the size of the structural set with respect to the ground-truth value provided by the multiplex estimate. Colors and symbols are the same as those used in panel (a). The blue curves with no markers represent instead the theoretically expected behaviour for an ordinary percolation process.



Supplementary Figure 6. Optimal percolation problem in synthetic multiplex networks using HDA algorithm. a) We consider multiplex networks with $N = 10^5$ and layers generated independently according to the Erdős–Rényi model with average degree $\langle k \rangle$. We estimate the relative size of the set of SNs on the multiplex as a function of $\langle k \rangle$ (turquoise circles), and compare it with the same quantity but estimated on the individual layers (black squares and purple triangles) or the aggregated (orange triangles). b) The relative errors of single-layer estimates of the size of the structural set with respect to the ground-truth value provided by the multiplex estimate. Colors and symbols are the same as those used in panel (a). The blue curves with no markers represent instead the theoretically expected behaviour for an ordinary percolation process.



Supplementary Figure 7. The effect of reducing edge overlaps and interlayer degree-degree correlation by partially relabeling nodes in multiplex networks with initially identical layers. Initially, both layers are a copy of a random network generated by an Erdős-Rényi model with $N = 1,000$ nodes and average degree $\langle k \rangle = 5$. Then, in one of the layers, each node is selected to switch its label with another randomly chosen node with a certain probability α . For each α , we determine the mean of the relative size of the set of SNs over 100 realizations of the HDA+GR algorithm on the multiplex network.



Supplementary Figure 8. The effect of reducing edge overlaps and interlayer degree-degree correlation by partially relabeling nodes in multiplex networks with initially identical layers. We consider the multiplex networks described in Supplementary Figure 7 and the sets of SNs found for the multiplex and single layer based representations of these networks. a) As the set of SNs found in different instances of the optimization algorithm are different from each other, we first quantify the self-consistency of those solutions across 100 independent runs of the HDA+GR algorithm. We then assume that the multiplex representation provides the ground-truth classification of the nodes. We compare the results of the other representation with the ground truth by measuring their precision (panel b), their sensitivity or recall (panel c), and their F_1 score (panel d).

Supplementary Note 1. DISMANTLING ALGORITHMS

In this section, we briefly discuss some of the most effective dismantling algorithms on monoplex networks and their generalization to multiplex networks with two layers. The aim of these algorithms is to approximate the set of structural nodes which is the minimal set of nodes that their removal dismantles the network into vanishingly small (non-extensive) clusters. We first introduce some of the score-based algorithms. In such algorithms, at each step, a score for each node is calculated and the node with the highest score is removed (when several nodes have the same score, one of them is removed at random). We discuss four different types of such algorithms: High Degree (HD), High Degree Adaptive (HDA), Collective Influence (CI) and Explosive Immunization (EI). These methods are partially deterministic in nature, i.e., nearly the same set of structural nodes are discovered at each realizations of the algorithm. Besides score-based algorithms, we present Simulated Annealing (SA) algorithm which is a greedy algorithm that searches the solution space of the dismantling problem to find the best approximation to the structural sets. The SA method takes into account the collective behavior of the dismantling problem and provides several dismantling sets for different realizations of the algorithm on the same network structure.

High Degree (HD)

In a monoplex network, the easiest way to dismantle a network, is a degree-based attack. After sorting the nodes with respect to their degrees, the nodes with the highest degree are removed one by one (in cases where there are more than one node with a certain degree, one of them is removed at random), until the network is dismantled. As this algorithm is deterministic (except from the randomness in choosing a node from those with the same degree), the set of nodes that are removed to dismantle the network is almost unique.

In multiplex networks, the degree of a node can be defined in various ways. We consider four different cases: the score of a node is defined as (i) its degree in layer A, (ii) its degree in layer B, (iii) the sum of its degrees across all the layers, and (iv) the product of its degrees across all the layers. It is worth mentioning that, when using HD, HDA and CI methods, at each step we remove $0.001 \times N$ of the nodes, where N is the total number of nodes (in each layer) of the multiplex.

As Supplementary Figure 1 illustrates, to destruct a multiplex, the two scores defined as a combination of degrees in different layers are more effective than those based on the degrees in only one of the layers. In the main script and in the rest of the Supplemental Material (SM) when we refer to HD method, we mean the one in which the score of a node is the product of its degrees across all the layers.

High Degree Adaptive (HDA)

In the HD algorithm if we take into account the history of the process and recalculate, at each step, the degrees of the nodes, it is referred to as an HDA algorithm. Since the HDA algorithm is adaptive it is expected to work better than the HD method. In each step of the monoplex version of the HDA, we remove a fraction $0.001 \times N$ of the nodes that had the highest degrees; then we recalculate the degree of the nodes present in the Giant Connected Component (GCC) of the network. We repeat this process until the size of the GCC reduces

to \sqrt{N} or smaller; this threshold satisfies the condition that in the dismantled network the size of all the clusters is a sub-linear function of N .

Like the HD case, we can define at least four methods to define the degree of a node. Please notice that in the multiplex version, when we update the degrees, we exclude those neighbors that are not in the Giant Mutually Connected Component (GMCC) of the network. Supplementary Figure 2 shows the effectiveness of the HDA algorithm for the different definitions of nodes' degrees. Similar to the results for HD, it is more effective to combine the scores of different layers, than considering layers as isolated networks. In all the subsequent sections and in the main script, when we refer to HDA, we mean the one in which the score of a node is defined as the multiplication of its degrees across all the layers.

Collective Influence (CI)

In the monoplex version of the CI algorithm [1], the score $\text{CI}_i(l)$ of node i is equal to the excess degree of i multiplied by the sum of the excess degrees of its neighbours at a specific distance l from i :

$$\text{CI}_i(l) = (k_i - 1) \sum_{j \in \partial\text{Ball}(i,l)} (k_j - 1), \quad (1)$$

where $\partial\text{Ball}(i,l)$ denotes the neighbors of i at the distance l (i.e., the nodes that have a geodesic distance l from i). At each step, the CI score is adaptively calculated for all the nodes; then nodes with the highest score are removed from the network, until the network is dismantled. It was shown [1] that the performance of the CI method increases with l up to $l = 4$; for $l > 4$ the performance is not improved appreciably as l is increased.

To adapt the CI algorithm to multiplex networks with two layers, we considered several possible definitions of the CI score in the multiplex: (i) using the CI obtained based only on the structure of layer A, (ii) based only on the structure of layer B, (iii) the sum of the CIs of a node in layer A and layer B, and (iv) the product of these two CI scores. Supplementary Figure 3 illustrates that, the generalizations of the CI method we considered here are not as effective as those derived based on the HD (Supplementary Figure 1) and HDA (Supplementary Figure 2) methods. Thus, methods based on the CI measures of the layers do not provide an effective algorithm for the optimal percolation problem.

Explosive Immunization (EI)

The EI algorithm is based on a method referred to as explosive percolation. The original explosive percolation method was introduced by Achlioptas *et al.* [2]. In this method at first all the edges are removed; then they are gradually reintroduced to the network, but in a specific order that prevents the formation of the GCC, until a point where the formation of the GCC is inevitable. To add a new edge, first several random edges are selected. Then a score is calculated for each of the selected edges using a predefined kernel (a possible kernel, for example, defines the score as the sum of the sizes of the two clusters connected by the corresponding edge). Then the edge with the minimum score is added back the network. The scores represent the contribution of each edge in the formation of the giant cluster. When the network reaches the point where the formation of the giant cluster is inevitable, the rest of the edges are added back using the same kernel.

A problem related to explosive percolation is the optimal immunization [3] in which the goal is to find the blocker nodes, which if get vaccinated, the giant connected component of the susceptible nodes breaks down; this break down eliminates a large scale epidemic spread. Clusella *et al.* [3] proposed a reverse approach to find the blockers. They introduced an algorithm that locates instead all the nodes that are irrelevant to the formation of the giant susceptible cluster. In this respect, their algorithm is a modified version of the explosive percolation. Their algorithm considers the site percolation version of the explosive percolation, in which all the links are present but, in the beginning, all the nodes are absent. Then, all the non-blocker nodes (that have no contribution to the formation of the giant susceptible cluster) are added gradually. The remaining nodes are the blocker nodes which should be vaccinated. We refer to this method as the explosive immunization (EI) algorithm.

For a monoplex network we implement the EI algorithm as follows. At each step, we select $N^{(C)} = 1000$ candidate nodes from the set of absent nodes, and calculate the score σ_i of each of them using the following kernel:

$$\sigma_i = \sum_{j \in N_i} (\sqrt{|C_j|} - 1) + k_i^{(\text{eff})}, \quad (2)$$

where, N_i represents the set of all connected components (CCs) linked to node i , each of which has a size C_j , and $k_i^{(\text{eff})}$ is an effective degree attributed to each node (please see Ref. [3] for the details). Then the nodes with the lowest scores are added to the network. This procedure is continued until the size of the GCC exceeds a predefined threshold g^* (For the simulations of this paper we used $g^* = \sqrt{N}$). The minus one term in Eq. 2 is excluding any leaves connected to node i , since they do not contribute to the formation of the GCC and should be ignored in the score of a node.

In our extension of the EI method to multiplex networks, we consider the different kernel but otherwise perform the exact same procedure as the one described above. The new kernel (Eq. 3) we use is based on the sizes of the mutually connected components (MCCs) rather than on the sizes of CCs:

$$\sigma_i = 1/2 \left[\sum_{j \in N_i^{[A]}} (\sqrt{|M_j|}) + \sum_{j \in N_i^{[B]}} (\sqrt{|M_j|}) \right] + \sqrt{k_i^{[A](\text{eff})} k_i^{[B](\text{eff})}}, \quad (3)$$

where $N_i^{[A]}$ is the set of neighbors of node i in layer A, M_j is the size of the MCC to which node j belongs, and $k_i^{[A](\text{eff})}$ is the effective degree of i in layer A obtained using the same definition proposed [3] for the monoplex version of the EI method.

In Eq. 3 we do not add a minus 1 term to exclude the leaves; the reason is that while in monoplex networks leaves do not have a significant contribution in the formation of the GCC, in multiplex networks even a leaf node is important in the formation of the GMCC. This is because at the sub-critical regime of multiplex networks usually most of the MCCs are isolated nodes or have very small sizes. Supplementary Figure 4 certifies that if the leaves were excluded instead, the performance of the algorithm would decrease. It is worth to mention that in the simulations of Figure 1 of the main text, we used a $N^{(C)} = 1000$.

Simulated Annealing (SA)

The simulated annealing (SA) method has been used for the dismantling problem in monoplex networks [4]. Generally an SA algorithm defines an energy function that attributes

energy values to each configuration of the system. The phase space of the system is searched for the optimal configuration (the one with the minimum energy) by Markov Chain Monte Carlo moves that switch the system from one configuration to another. In dismantling of multiplex networks, the algorithm should find the minimal set of nodes which if deleted the size of the GMCC becomes non-extensive. Each configuration of the multiplex network is represented by $\{R, g\}$, where R and g are, respectively, the number of removed nodes (each node and all its corresponding replica nodes are counted as one node), and the relative size of the GMCC. The energy of a configuration is defined as follows:

$$\varepsilon = Rv + g, \quad (4)$$

where v is the cost of removing a node from the multiplex network and in the simulations presented in this paper it is set $v = 0.6$. At each step t of the algorithm, one node, present or removed, is selected at random; then one of the following sets of operations are performed:

- If the node is present and it belongs to the GMCC, it is removed (thus $R_t = R_{t-1} + 1$) and the new size of the GMCC (g_t) is calculated.
- If the node is present but it does not belong to the GMCC, then it is removed (thus $R_t = R_{t-1} + 1$); but since it did not belong to the GMCC, $g_t = g_{t-1}$.
- If the node is in the set of removed nodes, it is added back to the network and $R_t = R_{t-1} - 1$. Then M_i (the size of the MCC formed after inserting i) is calculated and $g_t = \max(M_i, g_{t-1})$.

Afterwards the energy of the new configuration ε_t is calculated and the set of operations is accepted with a probability equal to $\min(1, e^{-\beta(\varepsilon_{\text{new}} - \varepsilon)})$. If it is accepted, the new configuration ($\{R_t, g_t\}$) is retained, otherwise, the operations are omitted and the old configuration ($\{R_{t-1}, g_{t-1}\}$) is preserved.

Here, β is interpreted as the inverse of the temperature of the annealing process. The SA algorithm starts with a β_{\min} and, at each step, β is slightly increased by $\delta\beta$. A smaller $\delta\beta$ means a slower decrease in the temperature which allows the SA method to better search for the optimal configurations, at the expense of increasing the running time of the algorithm. In this paper we change the values of β from 0.5 to 20.0 with $\delta\beta = 10^{-6}$.

In Figure 1 of the main text, we show that the SA method outperforms the four score-based algorithms; thus, for the analysis of the optimal percolation problem, we mostly use the SA method (see the main text). In [Supplementary Note 3](#), we also provide results for the second best algorithm, i.e., the HDA method and show that the results are qualitatively similar to those of the SA method.

Supplementary Note 2. GREEDY REINSERTING (GR)

After a network (either isolated or multiplex) is dismantled using a greedy or score-based algorithm, there are some removed nodes that if added back to the network, the size of the GMCC is not increased substantially, i.e., no cluster with an extensive size is created if they are reinserted. Such nodes may have been removed because the greedy or score-based algorithms are not exact in the sense that they do not take into account the collective nature of the dismantling problem. An approach that addresses this issue is referred to as the greedy reinserting (GR) method [4, 5]. In the GR method, after a set of structural nodes is detected using another algorithm, at each step a randomly chosen node from the set is reinserted to the network, and unless its reinsertion does not increase the size of the GMCC to a threshold \sqrt{N} , it is removed again. This process is continued until practically none of the nodes remained in the set can be added to the network without keeping the size of the GMCC non-extensive.

As shown in Supplementary Figures 1-4 and Figure 1 of the main text, the GR method boosts effectively the performance of every one of the score-based dismantling algorithms and returns sets of structural nodes with almost identical sizes irrespective of the initial algorithm used. Moreover, the result of each of the score-based algorithms combined with the GR method is nearly as good as that of the SA method (the SA method itself is not improved appreciably by applying a GR method afterwards). These results suggest that probably the sets obtained by the SA method and any one of the score-based algorithms combined with GR are to a considerable extent similar to each other.

Supplementary Note 3. COMPLEMENTARY RESULTS FOR THE DEGREE-BASED METHODS

In this section, we provide further results for the HD (Supplementary Figure 5) and the HDA (Supplementary Figure 6) methods, and also for the combination of the GR method with HDA (Supplementary Figures 7–8); we compare these results with some of the results of the SA algorithm presented in the main text. In contrast to the SA algorithm, the degree-based algorithms are much more efficient in terms of the running time; hence, we were able to produce some of the results (see Supplementary Figures 5–6) for larger ER networks.

Supplementary Figures 5 and 6 show that, for both HD and HDA performed on the aggregated representation, the behavior of q_c (the relative size of the set of structural nodes) with respect to the network average degree resembles the results of the SA algorithm. On the other hand, HDA matches better to the result of SA for optimal percolation on each of the single layers of the multiplex network. In particular, for networks with sufficiently large degree, HDA on each of the layers can find a q_c very close to the q_c it obtains for the multiplex representation.

As shown in Supplementary Figure 7, the behavior of q_c with respect to the relabeling probability (higher relabeling probability indicates lower density of overlapping edges and lower interlayer degree-degree correlation) obtained with the HDA+GR method is qualitatively similar to the results of the SA algorithm (Figure 3 of the main text). Moreover, Supplementary Figure 8a shows that, as expected, HDA+GR has a relatively higher self-consistency compared to that of the SA algorithm reported in Figure 4 of the main text. It is worth noting that, in contrast to SA, the self-consistency of HDA+GR does not decrease with the relabeling probability in the multiplex representation (Supplementary Figure 8a).

Interestingly, despite the qualitative similarity of the results, HDA+GR returns higher values of precision (Supplementary Figure 8b), recall (Supplementary Figure 8c), and F_1 -score (Supplementary Figure 8d) than those of SA (see Figure 4 of the main text). As there is not much randomness in HDA+GR, the structural nodes are dominantly determined by the sequence of (adaptive) degrees of the nodes and the sets from different network representations have a higher overlap compared to those find by the SA algorithm.

SUPPLEMENTARY REFERENCES

- [1] Morone, F. & Makse, H. A. Influence maximization in complex networks through optimal percolation. *Nature* **524**, 65–68 (2015).
- [2] Achlioptas, D., D’souza, R. M. & Spencer, J. Explosive percolation in random networks. *Science* **323**, 1453–1455 (2009).
- [3] Clusella, P., Grassberger, P., Pérez-Reche, F. J. & Politi, A. Immunization and targeted destruction of networks using explosive percolation. *Phys. Rev. Lett.* **117**, 208301 (2016).
- [4] Braunstein, A., DallAsta, L., Semerjian, G. & Zdeborov, L. Network dismantling. *Proc. Natl. Acad. Sci. USA* **113**, 12368–12373 (2016).
- [5] Zdeborová, L., Zhang, P. & Zhou, H.-J. Fast and simple decycling and dismantling of networks. *Sci. Rep.* **6**, 37954 (2016).

Chapter Seven

k-core structure of real multiplex networks

Contribution

I initially designed the core idea of the research and then through discussion with my co-authors we conducted the research. All the simulations and analysis were done by me. The co-authors helped me in the design part of the research. Initial draft of the manuscript was written by me and Prof. Papadopoulos. All the authors contributed equally to the writing and editing of the paper.

***k*-core structure of real multiplex networks**Saeed Osat^{1,*}, Filippo Radicchi,² and Fragkiskos Papadopoulos³¹Deep Quantum Labs, Skolkovo Institute of Science and Technology, Moscow 143026, Russia²Center for Complex Networks and Systems Research, Luddy School of Informatics, Computing, and Engineering, Indiana University, Bloomington, Indiana 47408, USA³Department of Electrical Engineering, Computer Engineering and Informatics, Cyprus University of Technology, 33 Saripolou Street, 3036 Limassol, Cyprus

(Received 25 November 2019; accepted 22 April 2020; published 14 May 2020)

Multiplex networks are convenient mathematical representations for many real-world—biological, social, and technological—systems of interacting elements, where pairwise interactions among elements have different flavors. Previous studies pointed out that real-world multiplex networks display significant interlayer correlations—degree-degree correlation, edge overlap, node similarities—able to make them robust against random and targeted failures of their individual components. Here, we show that interlayer correlations are important also in the characterization of their k -core structure, namely, the organization in shells of nodes with an increasingly high degree. Understanding of k -core structures is important in the study of spreading processes taking place on networks, as for example in the identification of influential spreaders and the emergence of localization phenomena. We find that, if the degree distribution of the network is heterogeneous, then a strong k -core structure is well predicted by significantly positive degree-degree correlations. However, if the network degree distribution is homogeneous, then strong k -core structure is due to positive correlations at the level of node similarities. We reach our conclusions by analyzing different real-world multiplex networks, introducing novel techniques for controlling interlayer correlations of networks without changing their structure, and taking advantage of synthetic network models with tunable levels of interlayer correlations.

DOI: [10.1103/PhysRevResearch.2.023176](https://doi.org/10.1103/PhysRevResearch.2.023176)**I. INTRODUCTION**

A multiplex network is a collection of single-layer networks sharing common nodes, where each layer captures a different type of pairwise interaction among nodes [1–5]. This is a convenient and meaningful representation for many real-world networked systems, including social [6,7], technological [8], and biological systems [9–11]. The simultaneous presence of different types of interactions is at the root of the observation of collective phenomena generally not possible in single-layer networks. A paradigmatic example is provided in the seminal study by Buldyrev *et al.* [12] where it was shown that, if multiplexity is interpreted as a one-to-one interdependence among corresponding nodes in the various layers, then the mutual connectedness of a multiplex network displays an abrupt breakdown under random failures of its nodes. Other examples of anomalous behavior of multiplex networks regard both dynamical and structural processes [13–20]. Although multiplexity seems a necessary condition for the emergence of nontrivial collective behavior, the magnitude of the anomalous behavior in real-world multiplex networks is often suppressed

by the presence of strong interlayer correlations, such as link overlap, degree-degree correlations, geometric correlations, and correlated community structure [16,21–25].

An important feature characterizing structural and dynamical properties of single-layer networks is the so-called k -core structure [26,27]. The k -core of a network is the maximal subgraph of the network in which all vertices have degree at least equal to k (see Appendix A 1). The notion of k -core is used to define so-called k -shells of nodes, and further to define the node centrality metric k_s named k -shell index or coreness (Appendix A 1). k -cores, and k -shells, are particularly important for the understanding of spreading processes on networks [28]. For instance, the coreness of a node is a good indicator of its spreading power [29]. Also, in many real-world networks, the notion of maximal k -core, i.e., the core with the largest k , represents a good structural proxy for the understanding of dynamical localization phenomena in spreading processes [30]. Finally, the extinction of species located in the maximal k -core well predicts the collapse of networks describing mutualistic ecosystems [31].

The notion of k -core can be generalized to the case of multiplex networks [32]. In a multiplex of L layers, the \mathbf{k} -core is defined for a vector of degree threshold values $\mathbf{k} = (k_1, \dots, k_\ell, \dots, k_L)$. Specifically, it is the maximal set of nodes such that each node complies with the corresponding degree threshold condition in each layer of the multiplex (Appendix A 1). In Ref. [32], Azimi-Tafreshi and collaborators studied the emergence of \mathbf{k} -cores in random uncorrelated multiplex network models with arbitrary degree distributions.

*saeedosat13@gmail.com

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

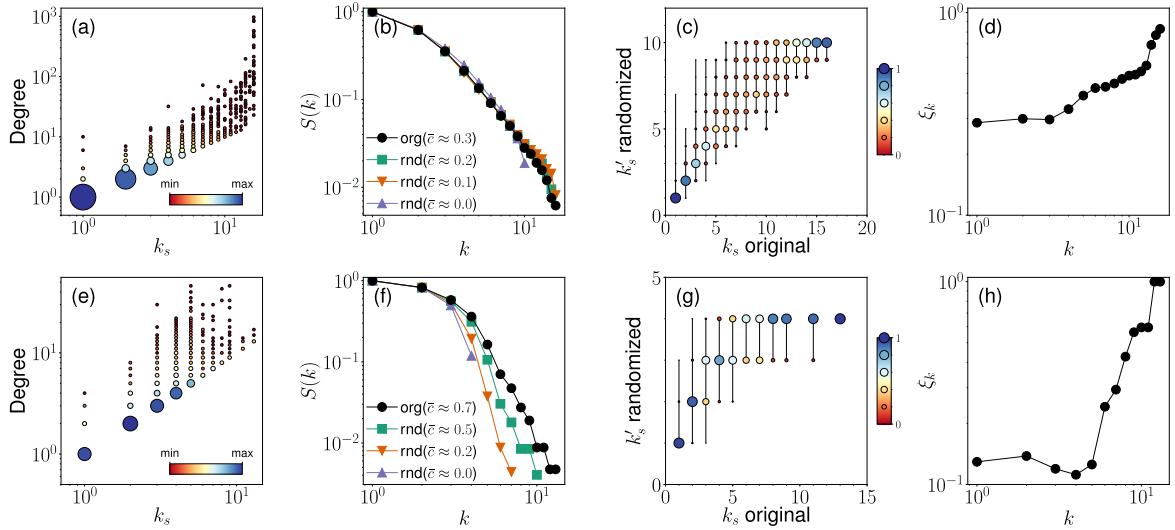


FIG. 1. k -core structure of real-world networks. We analyze single-layer networks. The top row refers to results valid for the IPv6 Internet; the bottom row refers to results valid for the arXiv coauthorship network. [(a) and (e)] Scatter plot of node degrees vs coreness. The size of the symbols is proportional to the number of nodes having each specific degree and k -shell index values. [(b) and (f)] Relative size $S(k)$ of the k -core (see Appendix A 1) in the real networks (labeled as “org”) and their randomized counterparts (labeled as “rnd”). Randomized networks are obtained by shuffling random pairs of edges while controlling for the average value of the clustering coefficient \bar{c} (Appendix A 2). $\bar{c} \approx 0$ is obtained after 10,000 and 2000 link rewirings in the Internet and arXiv, respectively. [(c) and (g)] k -shell index of nodes before and after network randomization (obtained for $\bar{c} \approx 0$). The size of the symbols is proportional to the percentage of nodes whose coreness changed from k_s in the original network to k'_s in the reshuffled network. [(d) and (h)] Angular coherence ξ_k of the nodes belonging to each k -core.

They showed that \mathbf{k} -cores in multiplex networks are characterized by abrupt transitions, but their properties cannot be easily deduced from those of the k -cores of the individual network layers. They further studied the \mathbf{k} -core structure of a few real-world networks. They noted that these systems display significant differences from the theoretical predictions that can be obtained in the framework developed for uncorrelated networks, thus indicating the necessity of a better understanding of the role of structural correlations in the characterization of the \mathbf{k} -core structure of real-world multiplex networks.

In this paper, we build on the work of Azimi-Tafreshi *et al.* [32] and perform a systematic characterization of the \mathbf{k} -core structure of real-world multiplex networks. We consider a large variety of systems, and study how the size of the \mathbf{k} -core depends on the choice of the vector \mathbf{k} . Specifically, we compare the \mathbf{k} -core of real-world networks with the core observed for the same choice of the vector \mathbf{k} on randomized versions of the networks where interlayer correlations are destroyed. We find that real-world multiplex networks possess non-null \mathbf{k} -cores while their reshuffled versions do not. We interpret this fact as a sign of the strength of the \mathbf{k} -core structure of real-world multiplex networks. To provide an intuitive explanation of this finding, we take advantage of the geometric interpretation of interlayer correlations in terms of network hyperbolic embedding [33,34]. Our choice is motivated by a series of recent studies where it has been shown that not only real-world multiplex networks display significant geometric correlations [23], but also that the amount of these correlations is a good predictor of the robustness of the system under targeted attacks [24,25]. In network hyperbolic

embedding, nodes of a network are mapped to points of the two-dimensional hyperbolic disk [35]. The radial coordinate of a node in the disk quantifies the popularity of the node; the difference between angular coordinates is related instead to the level of similarity between pairs of nodes. Geometric correlations in a multiplex network are quantified by looking at the coordinates of the same node in different layers, provided that the layers are embedded independently in the hyperbolic space. Geometric correlations can be quantified either for radial or angular coordinates of the nodes. Both types of correlations are able to provide insights about the \mathbf{k} -core structure of a multiplex. Specifically, we show that the more heterogeneous are the degree distributions of the layers, the more pivotal is the role of popularity correlations in the emergence of strong \mathbf{k} -core structure. On the other hand, the less heterogeneous are the degree distributions, the more crucial is the role of similarity correlations. These observations are in remarkable agreement with the behavior observed in synthetic multiplex networks where we can control the level of geometric correlations across the layers [23].

II. RESULTS

A. Single-layer networks

We start by studying the k -core structure of single-layer networks. Most of our results for single-layer networks are not novel as the problem was already studied in Ref. [36]. We replicate and expand the analysis of Ref. [36] here for two main reasons. First, the repetition of the analysis of Ref. [36] allows us to have a self-contained paper. Second and more

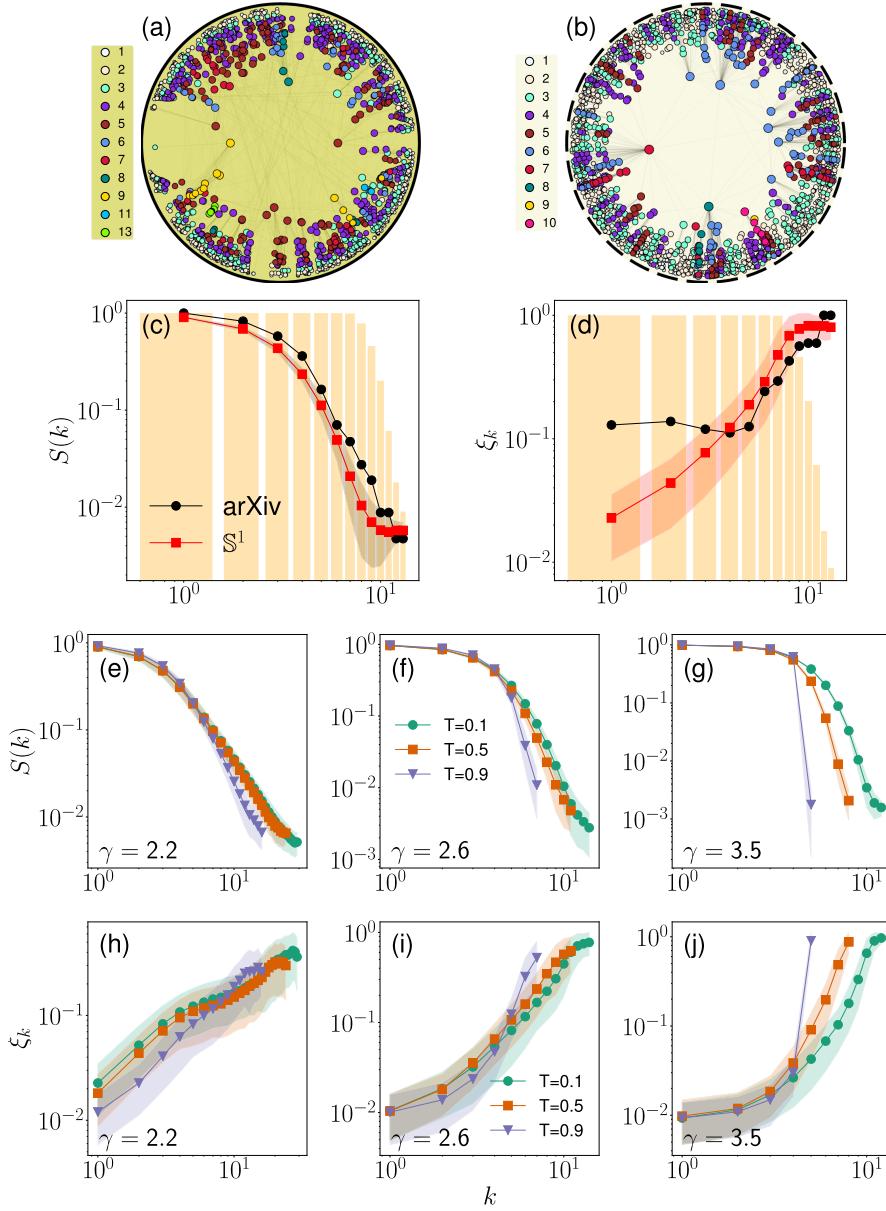


FIG. 2. k -core structure of single-layer networks. (a) Hyperbolic embedding of the arXiv network. The position of the nodes in the disk is determined by their hyperbolic coordinates; different colors serve to differentiate nodes depending on their k -shell index value. (b) Same as in (a) but for an instance of the S^1 model built using similar characteristics as in the arXiv network (i.e., same network size N , and same values of the degree exponent γ , average degree \bar{k} , and average clustering coefficient \bar{c}). (c) Relative size $S(k)$ of the k -core as a function of the threshold value k for the arXiv network and the S^1 model. The results for the modeled network are average values over 1000 network instances. The shaded area identifies the region corresponding to one standard deviation away from the average. The average value is computed only over non-null k -cores, and the bars in the background of the figure display the fraction of instances where such nonempty cores were indeed present. (d) We consider the same data as in (c) but monitor the angular coherence ξ_k as a function of k . (e) $S(k)$ vs k for the S^1 . We set here the size of the network $N = 10\,000$, degree exponent $\gamma = 2.2$, and average degree $\bar{k} = 6$. We consider three different values of the temperature parameter T . This serves to tune the average value of the clustering coefficient \bar{c} of the model, as T is inversely proportional to \bar{c} . Results are averaged over 200 instances of the model. Shaded areas stand for one standard deviation away from the average. (f) Same as in (e) but for $\gamma = 2.6$. (g) Same as in (e) but for $\gamma = 3.5$. (h) We consider the same networks as in (e) but we monitor angular coherence ξ_k vs k . (i) Same as in (h) but for $\gamma = 2.6$. (j) Same as in (h) but for $\gamma = 3.5$.

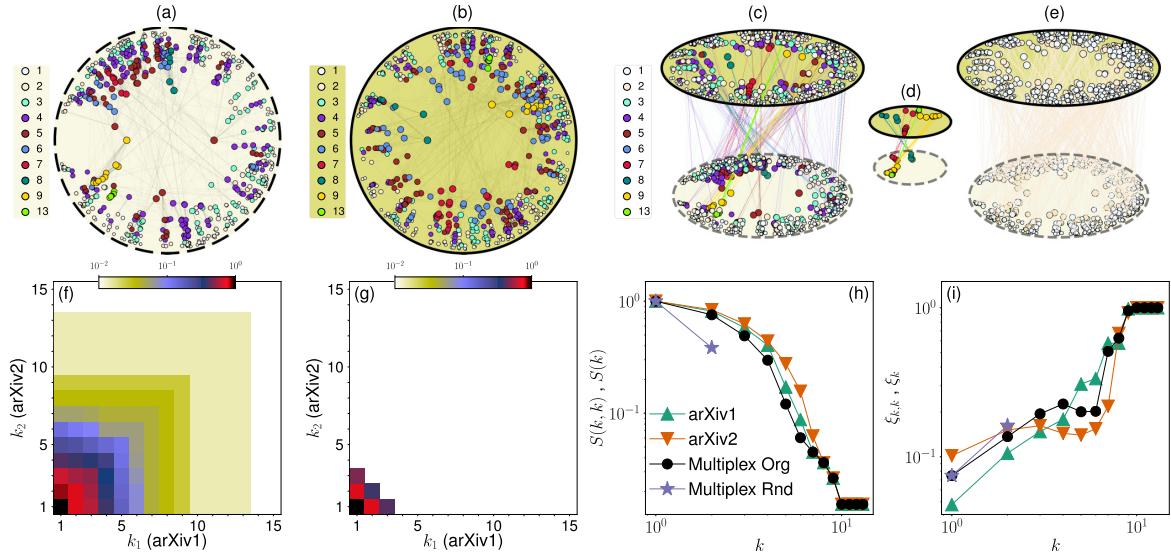


FIG. 3. **k**-core of real-world multiplex networks. [(a) and (b)] Hyperbolic embedding of the arXiv multiplex network. (a) refers to the layer arXiv1, while (b) to the layer arXiv2. The position of the nodes in the disk is determined by their hyperbolic coordinates, and only nodes that exist in both layers are shown (911 nodes); different colors serve to differentiate nodes depending on their k -shell index value. (c) Correspondence among nodes belonging to the (k, k) -shells (see Appendix A 1) of the arXiv multiplex network. (d) Same as in (c) but for $k \geq 7$. (e) Same as in (c) but for the randomized version of the multiplex where the node labels of one of the two layers are randomly reshuffled. (f) Relative size $S(k_1, k_2)$ of the (k_1, k_2) -core for the arXiv multiplex network. (g) Same as in (f) but for the randomized version of the multiplex network. (h) Relative size $S(k, k)$ of the (k, k) -core for the arXiv multiplex network, and its randomized version. These curves are compared with those of the relative size $S(k)$ of the k -core of the individual layers. (i) Same as in (h) but for the metrics of angular coherence $\xi_{k,k}$ and ξ_k .

important, the analysis serves to properly calibrate our framework before extending it to the study of the **k**-core structure of multiplex networks. Such a calibration is of fundamental importance as findings on single-layer networks provide us with proper baselines for the interpretation of results valid for multiplex **k**-core structures, including testable hypotheses on their expected behavior.

In Fig. 1, we report results obtained by analyzing two single-layer networks: a snapshot of the Internet at the IPv6 level [37] and the co-authorship network formed by the authors of papers in the “Biological Physics” category of arXiv [38]. Details on the data and results for other networks can be found in Ref. [39], Secs. I and II. The k -shell index of the nodes is strongly correlated with their degree (Figs. 1(a) and 1(e) and Ref. [39], Fig. 2(a)). However, as previously noted in Ref. [29], nodes with the same value of the k -shell index may correspond to very different degree values. Further, we note that the degree distribution of the Internet is much broader than the one of the arXiv (see Figs. 1(a) and 1(e) and Ref. [39], Fig. 1). Specifically, the degree distributions of both networks can be modeled quite well in terms of power laws, i.e., $P(k) \sim k^{-\gamma}$, with degree exponent $\gamma = 2.1$ for the Internet and $\gamma = 2.6$ for the arXiv, thus indicating that the degree distribution of the Internet is more heterogeneous than the one of the arXiv. The correlation between k -shell index and node degree weakens significantly as we move into inner k -shells in the arXiv but not in the Internet (Ref. [39], Fig. 2(a)). We have verified that the less heterogeneous is the degree distribution

the weaker is the correlation between k -shell index and degree (Supplemental Material [39], Fig. 2(b)).

To quantify the quality of the k -core structure we consider the relative size $S(k)$ of the k -core as a function of the value of the threshold k . If there is a rich collection of k -cores with a wide spectrum of k 's, then the k -core structure is strong; it is weak, otherwise. Figures 1(b) and 1(f) show that the k -core structures of the Internet and arXiv are strong. In particular, we see that $S(k)$ decreases smoothly as k increases, while $S(k) > 0$ up to $k = 16$ for the Internet, and up to $k = 13$ for the arXiv.

Reference [36] showed in experiments with synthetic networks that both degree heterogeneity and clustering improve the quality of the k -core structure. To study how these properties affect the quality of the k -core structure of real networks, we study the behavior of $S(k)$ on degree-preserving randomized versions of the networks. The randomization is performed by rewiring randomly chosen links till the value of the average clustering in the network is reduced to a predefined value (see Appendix A 2). We see in Figs. 1(b) and 1(f) that the randomization affects the k -core structure of the Internet to a much lesser extent than the k -core structure of the arXiv, while the effect is stronger the more we destroy clustering. As Figs. 1(c) and 1(g) clearly show, the effect of the randomization consists in redistributing nodes to lower k -shell values. Specifically, these figures show the percentage of nodes, indicated by the circles, whose k -shell index changes from k_s in the original network to k'_s in the

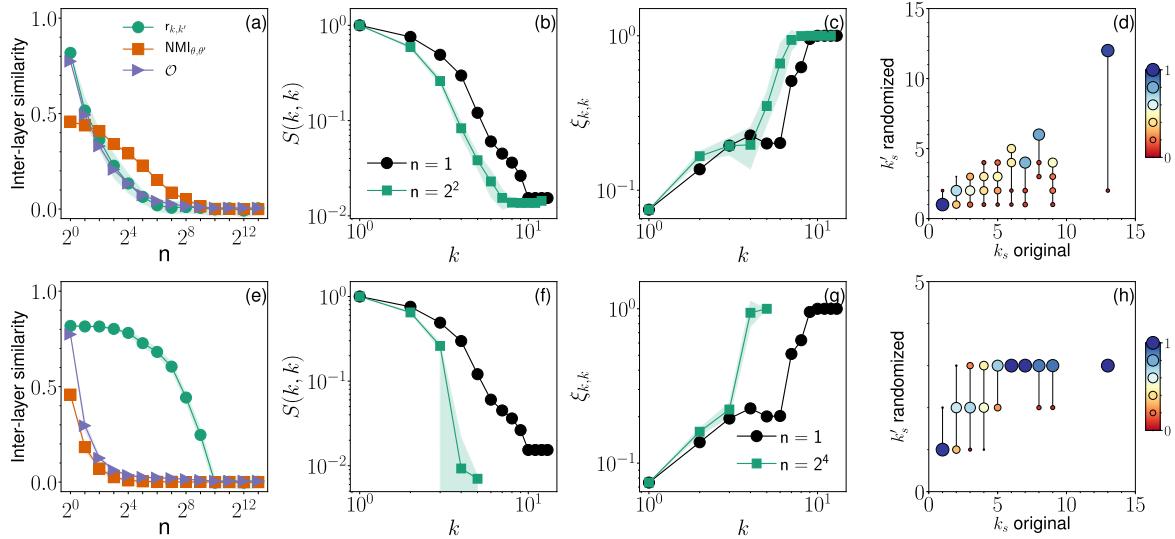


FIG. 4. Interlayer correlations and the k -core structure of the arXiv multiplex network. We analyze the arXiv multiplex network. (a) Different metrics of interlayer similarity as a function of the group size n used to randomize node labels, thus breaking interlayer degree correlations. For $n = 1$, node labels of the network are not randomized; full shuffle of node labels is obtained for large n values. We focus here on the case where interlayer degree correlation is broken but we preserve interlayer correlation among node angular coordinates (see main text for details). Metrics of similarities considered here are the Pearson correlation coefficient $r_{k,k'}$ among the degrees of nodes in the two layers; normalized mutual information $NMI_{\theta,\theta'}$ of the angular coordinates of the nodes in the two layers; and edge overlap \mathcal{O} among the two layers (Appendix A 6). (b) Relative size $S(k,k)$ of the (k,k) -core. The results of the original multiplex network ($n = 1$) are compared with those valid for $n = 4$. At this level of randomization, we find that $r_{k,k'} = 0.36$ and $NMI_{\theta,\theta'} = 0.41$. These numbers should be compared respectively with $r_{k,k'} = 0.82$ and $NMI_{\theta,\theta'} = 0.46$ of the original network. The results for $n = 4$ are average values obtained on 100 independent randomizations. Shaded areas identify the region corresponding to one standard deviation away from the average. (c) Same as in (b) but for the angular coherence $\xi_{k,k}$. (d) Scatter plot of the (k,k) -shell index of nodes in the original vs the randomized multiplex network. The size of the symbols is proportional to the percentage of points in the scatter plot. [(e)–(h)] Same as in (a)–(d), respectively. We consider here the case where interlayer correlation among nodes' angular coordinates is destroyed but interlayer correlation among node degrees is preserved (see main text for details). The results of the original network are compared with those obtained for $n = 16$, when $r_{k,k'} = 0.78$ and $NMI_{\theta,\theta'} = 0.01$.

randomized network. We see that changes of the k -shell values induced by the randomization are much more apparent for the arXiv than in the Internet—nodes in the arXiv are redistributed to significantly lower shells. For instance, we see in Fig. 1(g) that nodes belonging to $k_s = 11$ in the original network move to $k'_s = 4$ and $k'_s = 3$ in the randomized network. These results indicate that networks with more heterogeneous degree distributions can have strong k -core structures even if their clustering is weak. On the other hand, if the degree distribution is less heterogeneous, clustering becomes more important for having a strong k -core structure. In the next section, we explicitly verify these observations in controlled experiments with synthetic networks [Figs. 2(e)–2(g)].

B. Hyperbolic embedding

To better capture the role of correlations for the characterization of the k -core structure of networks, we decided to take advantage of the vectorial representation of nodes in the hyperbolic space [33,35,40]. According to this mapping, every node i of a network becomes a point, identified by the coordinates (r_i, θ_i) , in the two-dimensional hyperbolic disk (see Appendixes A 3 and A 4). The radial coordinate r_i quanti-

fies the popularity of node i in the network, and basically corresponds to the degree k_i of the node (Appendix A 4). The angular coordinate θ_i serves to quantify pairwise similarities, in the sense that the angular distance between pairs of nodes is inversely proportional to their similarity. Whereas radial coordinates do not convey more explicative information than node degrees, angular coordinates offer the opportunity to deal with node similarities in continuous space, thus allowing for smooth and easily quantifiable metrics of similarities of arbitrary sets of nodes, including k -cores. Specifically, we use a measure of coherence among angular coordinates of nodes within the k -core, namely, ξ_k , to measure the average level of similarity among the nodes within the k -core [25] (see Appendix A 5). By definition $\xi_k \in [0, 1]$, with $\xi_k = 0$ meaning that the angular coordinates of the k -core are uniformly scattered around the disk, and $\xi_k = 1$ meaning that all nodes within the k -core have identical value for their angular coordinates. Figures 1(d) and 1(h) show ξ_k as a function of k for the Internet and arXiv networks, respectively. We see that ξ_k increases with k , meaning that as we move to inner k -cores, angular coordinates of the nodes tend to be more localized. Similar results hold if one analyzes other real networks and if one measures angular coherence in the k -shells instead of the k -cores (see Ref. [39], Sec. II).

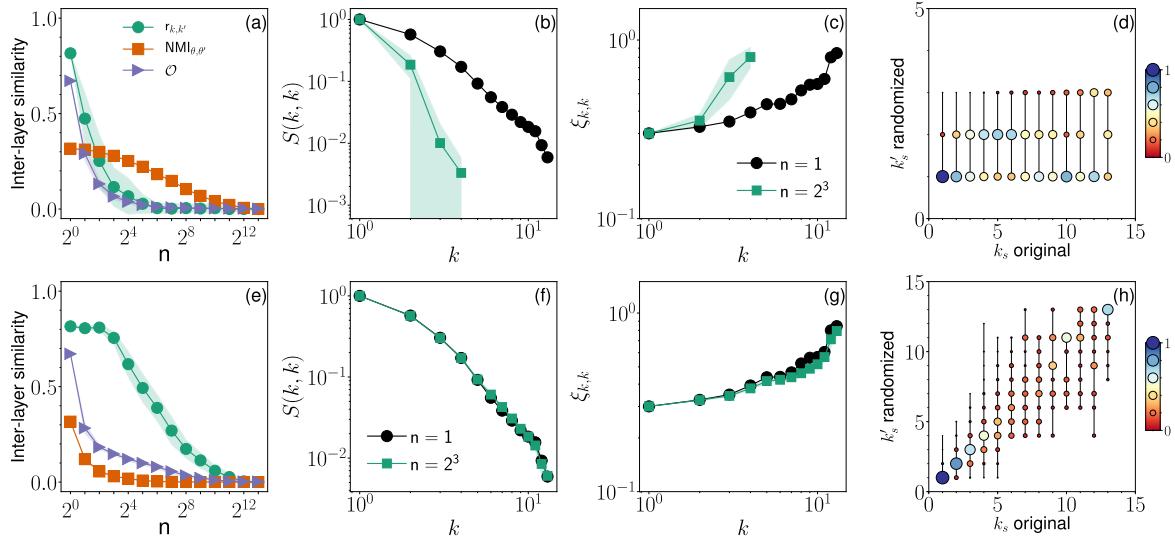


FIG. 5. Interlayer correlations and the \mathbf{k} -core structure of the Internet multiplex network. Same analysis as in Fig. 4 but for the IPv4/IPv6 Internet multiplex network. Correlations of the original network are such that $r_{k,k'} = 0.82$ and $NMI_{\theta,\theta'} = 0.32$. Results of the real-world system are compared with those obtained after destroying interlayer degree correlations such that $r_{k,k'} = 0.12$ and $NMI_{\theta,\theta'} = 0.28$ in the top-row panels, and after destroying angular correlations such that $r_{k,k'} = 0.76$ and $NMI_{\theta,\theta'} = 0.03$ in the bottom-row panels.

We take advantage of network hyperbolic embedding not only for descriptive purposes but also to perform controlled experiments. We leverage models introduced in the literature on network hyperbolic embedding to better understand the role played by clustering and node similarities in predicting the strength of network k -core structure. Specifically, we rely on network instances generated according to the S^1 model [33,41], which is isomorphic to hyperbolic geometric graphs (see Appendix A 3). The model generates synthetic networks with arbitrary degree distribution and clustering strength.

In Fig. 2, we perform a direct comparison between the relative size $S(k)$ and angular coherence ξ_k of the k -core structure of the arXiv collaboration network and of a synthetic graph generated according to the S^1 model with similar values of number of nodes, average degree, and average clustering coefficient as of the arXiv collaboration network. The synthetic network has a power-law degree distribution $P(k) \sim k^{-\gamma}$ with exponent $\gamma = 2.6$, compatible with the one of the real-world network (Ref. [39], Sec. I). We see that the two graphs display a qualitatively similar behavior with respect to $S(k)$ [Fig. 2(c)] and ξ_k [Fig. 2(d)] as functions of the threshold value k .

Synthetic networks allow us to play with the ingredients that we believe are important in the characterization of network k -core structure. We see that the range of k values for which we have non-null k -cores widen not only when the degree distribution becomes more heterogeneous (lower γ values) but also when the clustering coefficient increases [Figs. 2(e)-2(g)]. In all these cases, nodes belonging to inner k -cores always have more similar angular coordinates in the hyperbolic embedding [Figs. 2(h)-2(j)].

C. Multiplex networks

We now turn our attention to the study of the \mathbf{k} -core structure of real-world multiplex networks. For simplicity, we limit our attention to two-layer multiplex networks only, so that $\mathbf{k} = (k_1, k_2)$. We note that a necessary condition for having a non-null (k_1, k_2) -core is that the k_1 -core of layer $\ell = 1$ and the k_2 -core of layer $\ell = 2$ are simultaneously non null. The condition is clearly not sufficient, as there could be combinations (k_1, k_2) associated to empty cores in the multiplex but still showing nonempty cores at the level of the individual layers. As a consequence, we expect that multiplex networks displaying low interlayer correlation at the node level will be weak in terms of \mathbf{k} -core structure, in the sense that nonempty cores will exist only for limited choices of the thresholds (k_1, k_2) . Based on our knowledge of the relation between k -core strength and hyperbolic network embedding, we further expect that interlayer correlations that are important in the prediction of the strength of the \mathbf{k} -core structure of a multiplex are not only those relative to the degree of the nodes but also those concerning the similarity among pairs of nodes.

In Fig. 3, we consider a multiplex version of the arXiv collaboration network, where one layer is obtained by considering manuscripts of the section ‘‘Biological Physics’’ (i.e., the one considered already in Figs. 1 and 2), and the other based on manuscripts of the section ‘‘Data Analysis, Statistics and Probability.’’ For sake of brevity, we will refer to them as arXiv1 and arXiv2, respectively. We observe that the \mathbf{k} -core structure of the multiplex network is quite robust, in the sense that the relative size $S(k_1, k_2)$ of the (k_1, k_2) -core is strictly larger than zero for a wide range of choices of the threshold

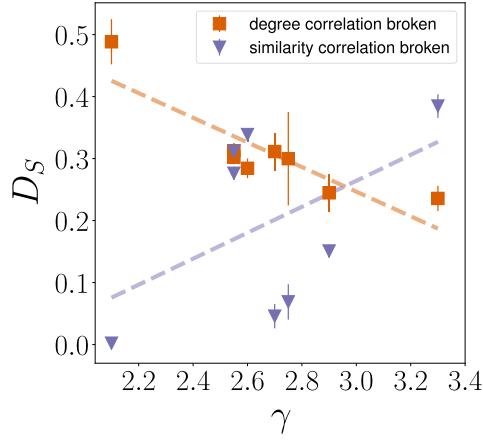


FIG. 6. Quantifying the effect of interlayer degree and similarity correlations in the k -core structure of real-world multiplex networks. Relative difference $D_S = [\sum_k S(k, k)^{\text{org}} - \sum_k S(k, k)^{\text{rnd}}] / \sum_k S(k, k)^{\text{org}}$ between the relative (k, k) -core sizes $S(k, k)^{\text{org}}$ and $S(k, k)^{\text{rnd}}$ of real-world multiplex networks and their randomized counterparts. In the randomized counterparts, we either destroy interlayer degree correlation or correlation among the nodes' angular coordinates. Each point in the figure corresponds to one of the real-world multiplex networks considered in this study. The points from left to right correspond to the following multiplex networks: IPv4/IPv6 Internet, arXiv1-arXiv4, arXiv2-arXiv4, arXiv1-arXiv2, Drosophila1-Drosophila2, Air-Train, C.Elegans2-C.Elegans3, and arXiv1-arXiv5 (Ref. [39], Sec. I). The x axis shows the average degree exponent γ across the two layers of each multiplex. Results in each case are obtained by taking the average value of D_S over 100 randomized counterparts. Error bars correspond to one standard deviation away from the average. The Pearson correlation coefficient between D_S and γ is $r_{D_S, \gamma} = -0.87$ when degree correlation is broken and $r_{D_S, \gamma} = 0.48$ when similarity correlation is broken. The dashed lines represent least squares regression lines.

values (k_1, k_2) [Fig. 3(f)]. This fact becomes apparent when the results valid for the real network are contrasted with those valid for a randomized version of the network [Fig. 3(g)]. The randomization here consists of randomly shuffling the labels of the nodes of one of the two layers, so that the topology of both layers remains unchanged but interlayer correlations are completely destroyed (Appendix A 2). As a visual inspection of Figs. 3(f) and 3(g) reveals, the real network displays nonempty cores in a much wider region of the (k_1, k_2) plane than the randomized version of the network. The result is highlighted in Fig. 3(h) for the special case $k_1 = k_2 = k$, where we see that the $S(k, k)$ of the real-multiplex network behaves almost identically to the $S(k)$ of the individual layers. On the contrary, the randomized version of the multiplex network displays an empty core already for $k > 2$. We can interpret the robustness of the k -core of the real multiplex network in terms of interlayer correlations. Indeed in Fig. 3(i), we see that nodes belonging to inner cores have simultaneously high angular coherence $\xi_{k,k}$ (Appendix A 5) in both layers of the real multiplex, a situation visualized in Figs 3(c) and 3(d) versus Fig. 3(e) for the randomized version of the network.

Similar results hold for other real-world multiplex networks (Ref. [39], Sec. III).

Next, we investigate the extent to which degree and similarity correlations affect the k -core structure, separately. To this end, we take advantage of network hyperbolic embedding, where layers are embedded independently, thus each node has radial and angular coordinates for each layer of the multiplex. Also in this case, we consider the degree of the nodes instead of their radial coordinate, being the two quantities clearly related one to the other. We break each type of correlation while preserving the other type of correlation. To break degree correlations, we consider the common nodes in the two layers of the multiplex, i.e., the nodes that are simultaneously present in both layers. Then, we select one of the layers and sort the common nodes with respect to their angular coordinates. We group the nodes in consecutive groups of size n , and in each group we reshuffle node labels. If n is sufficiently small, correlations among angular coordinates are approximately preserved since the angular coordinates of nodes do not change significantly within the group. Clearly, for $n = 1$, no reshuffling is performed, while if $n = N$, where N is the number of common nodes, then all types of interlayer correlations are broken. To break correlations among angular coordinates while preserving degree correlations we follow a similar procedure. Specifically, we select one of the layers, sort the common nodes with respect to their degrees, group nodes in consecutive groups of size n , and reshuffle node labels in each group.

The top row of Fig. 4 shows the results valid for the arXiv multiplex network when degree correlations are broken while correlations among angular coordinates are preserved; the bottom row of Fig. 4 reports results valid when degree correlations are preserved but correlations among angular coordinates are destroyed. As expected, interlayer degree correlation, measured in terms of Pearson correlation coefficient $r_{k,k'}$ (see Appendix A 6), decreases with the size n of the groups used in the randomization procedure [Fig. 4(a)]. Similarly, correlation among angular coordinates of the nodes, measured in terms of the normalized mutual information $\text{NMI}_{\theta, \theta'}$ (Appendix A 6), decreases as n increases. There is, however, a range of n values where $r_{k,k'}$ is low and $\text{NMI}_{\theta, \theta'}$ high, indicating that correlation at the level of angular coordinates is preserved but degree correlation is destroyed. We consider the randomized version of the network obtained for $n = 4$, thus belonging to the aforementioned range of suitable n values, and study differences between its (k, k) -core structure and the one of the real multiplex network [Figs. 4(b) and 4(c)]. The (k, k) -core of the real network is only slightly more robust than the one of the randomized network [Fig. 4(b)]. Angular coordinates of the nodes in the inner cores are still strongly correlated [Fig. 4(c)]. The same analysis gives a completely different result in the case of the Internet multiplex network, where the two layers are given by the IPv4 and IPv6 topologies, respectively (see Ref. [39], Sec. I for details on the data). Reducing degree correlation in this case destroys the (k, k) -core structure [Figs. 5(b)–5(d)].

If we repeat the same exercise but now destroying correlations among angular coordinates while preserving correlations between degrees, we see a completely different picture. For the arXiv multiplex network, the randomization procedure

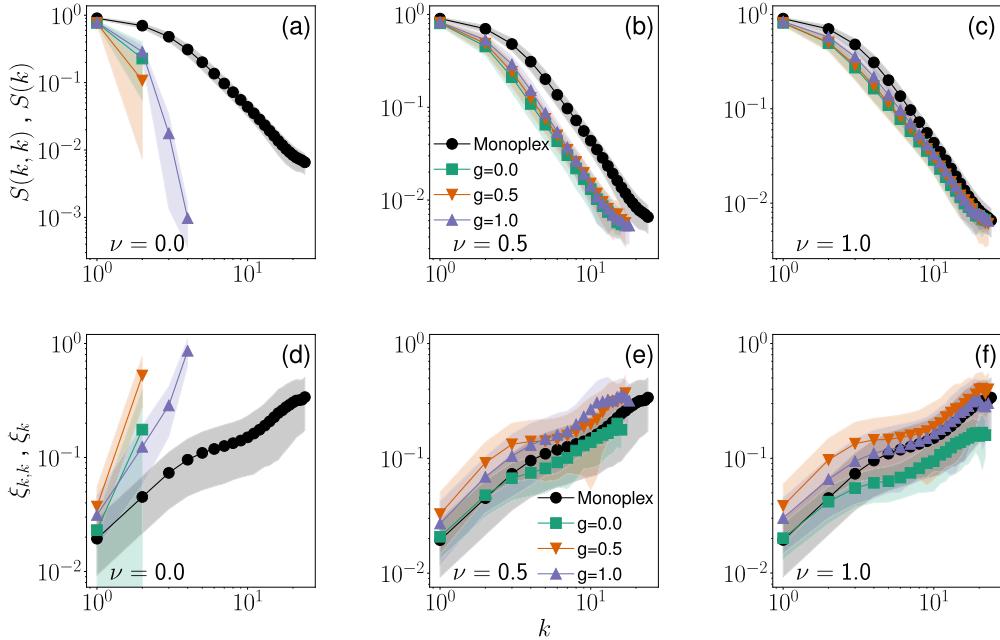


FIG. 7. **k**-core structure of synthetic multiplex networks. We study here the effect of degree and angular correlations on the size of the (k, k) -core $S(k, k)$ and its coherence $\xi_{k,k}$, in two-layer synthetic multiplex networks constructed according to the geometric multiplex model (Appendix A 7). Interlayer degree correlation can be tuned using the parameter $\nu \in [0, 1]$, with $\nu = 0$ corresponding to the uncorrelated case, and $\nu = 1$ to the case where degrees are maximally correlated. Interlayer correlation among angular coordinates of nodes is tuned using the model parameter $g \in [0, 1]$. When generating network instances according to the GMM, we imposed that each layer of the multiplex has $N = 10\,000$ nodes, power-law degree distribution with exponent $\gamma = 2.2$, average degree $\bar{k} \approx 6$, and temperature $T = 0.5$ (i.e., average clustering coefficient $\bar{c} = 0.45$). We consider various combinations of the model parameters ν and g . Results in each case are obtained by taking the average value over 100 realizations. Shaded areas denote regions corresponding to one standard deviation away from the average. (a) Relative size $S(k, k)$ of the (k, k) -core as a function of the threshold k . The curve corresponding to the monoplex is obtained by measuring $S(k)$ for the k -core of the individual layers, and then taking the average value. [(b) and (c)] Same as in (a) but for different choices of the model parameters. [(d)–(f)] We consider the same data as in (a)–(c), respectively but we monitor the metrics of angular coherence $\xi_{k,k}$ and ξ_k as functions of the threshold value k .

leads to the destruction of the **k**-core structure [Figs. 4(f)–4(h)]. Instead, for the Internet multiplex network, we see that the randomization procedure has virtually no effect on the strength of the **k**-core structure, keeping it unchanged with respect to the one of the original network [Figs. 5(f)–5(h)].

On the basis of our results, we hypothesize that both degree and similarity correlations matter for the emergence of strong **k**-core structures. In particular, when the degree distributions of the layers are less heterogeneous, like for the arXiv multiplex network, similarity correlations play a crucial role. On the other hand, when degree distributions are strongly heterogeneous, like in the case of the Internet multiplex network, degree correlations play a crucial role, and the effect of similarities is strongly attenuated (see Ref. [39], Sec. IV for results from other multiplex network data). This observation is also supported by Fig. 6, which quantifies the difference D_S between the curves of the original and randomized networks of Figs 4(b), 4(f) and 5(b), 5(f). The figure also shows D_S for other multiplex systems (considered in Ref. [39], Sec. IV). We see in Fig. 6 that when degree correlation is broken the difference D_S increases as the degree exponent γ decreases. On the other hand, when similarity correlation is broken D_S

tends to increase with γ . Figure 6 shows results from different systems that have different parameters (different layer sizes, average degrees, etc.). Therefore the fact that D_S in Fig. 6 is not strictly increasing or decreasing is expected.

To test our hypotheses, we rely on synthetic multiplex networks built according to the geometric multiplex model (GMM) [23]. This model allows to generate single-layer topologies using the S^1 model, and control for interlayer correlation between node degrees and angular coordinates (see Appendix A 7). In Figs. 7 and 8, we study the behavior of the **k**-core in two-layer synthetic multiplex networks constructed according to the model for different choices of the model parameters (more results can be found in Ref. [39], Sec. V). We confirm the validity of our claims. Both types of correlations are important for the characterization of the **k**-core of a multiplex network. Interlayer degree correlations (measured with ν) are more important than correlations between angular coordinates (measured with g) when the degrees of the nodes are broadly distributed. In this case, the role of pairwise similarities is much attenuated (see the difference between curves with different ν versus different g in Fig. 7). If instead, the network layers are characterized by homogeneous degree

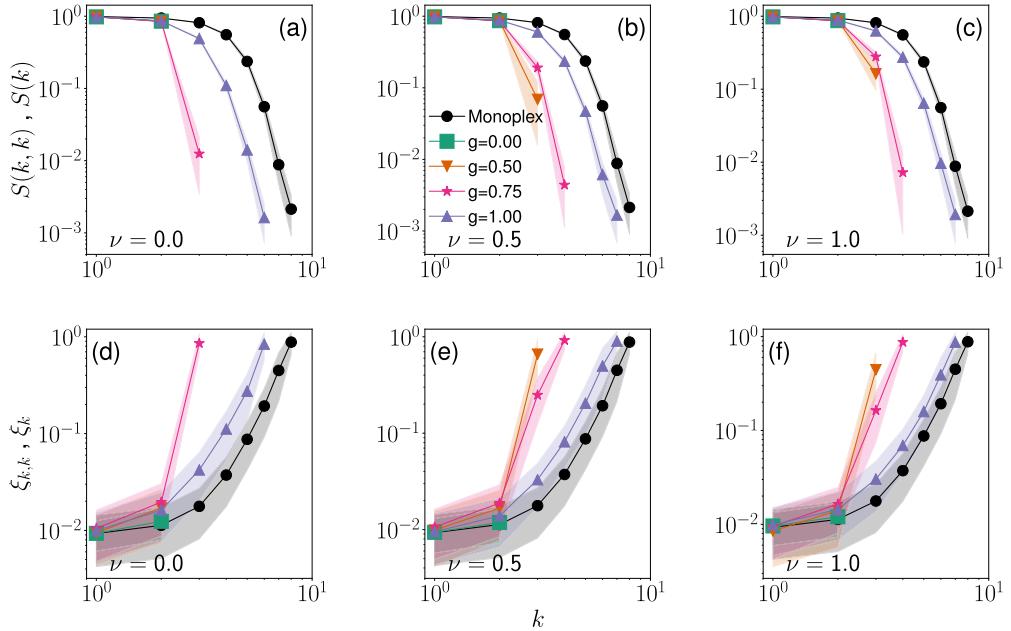


FIG. 8. **k**-core structure of synthetic multiplex networks. Same as in Fig. 7 but for a different value of the degree exponent $\gamma = 3.5$. Results for model parameter $g = 0.75$ are also shown in this figure. All other model parameters are identical to those used in Fig. 7.

distributions, similarity correlations are more important than degree correlations whose role is attenuated (Fig. 8). This effect is also illustrated in Figs. 9(a) and 9(b), which quantify the differences between the curves of the monoplex and multiplex networks of Figs. 7 and 8, as well as in Fig. 9(c), which illustrates a qualitatively similar behavior as the one observed for real networks in Fig. 6.

The above findings agree with intuition. When the degree distribution of a layer is more heterogeneous there is stronger correlation between higher k -shell index values and node degrees (Ref. [39], Fig. 2). In other words, the position of similarity of nodes matters less. Thus interlayer degree correlations are more important for having a wide **k**-core structure when the degree distributions of the layers are more heterogeneous. On the other hand, the less heterogeneous is the degree distribution the weaker is the correlation between higher k -shell index values and node degrees (Ref. [39], Fig. 2). In this case, the position of nodes in the similarity space matters more. Indeed, we have seen that nodes in inner cores have high angular coherence [cf. Fig. 1(h)]. Therefore interlayer similarity correlations become more important for having a strong **k**-core structure when the degree distributions of the layers are less heterogeneous.

III. DISCUSSION AND CONCLUSION

Understanding the principles behind the organization of real-world networks into cores or shells of nodes with increasing high degree is crucial for better understanding and predicting their structural and dynamical properties, their robustness, and the performance of spreading processes running on top of them. Yet, while the core organization of single-layer

networks has been extensively studied in the past, little is known about the core organization of real multiplex networks. In this paper, we performed a systematic characterization of the **k**-core structure of real-world multiplex networks, and shown that real multiplex networks possess a strong **k**-core structure that is due to interlayer correlations. Specifically, we showed that both degree and similarity correlations between nodes across layers are responsible for the observed strong **k**-core structures. The more heterogeneous are the degree distributions of the layers, the more pivotal is the role of degree correlations. On the other hand, the more homogeneous are the degree distributions, the more crucial is the role of similarity correlations. We reached our conclusions by taking advantage of network hyperbolic embedding, and showed that such a geometric description of networks provides a simple framework to naturally understand and characterize the **k**-core structure of real-world multiplex networks. As the core organization of a network is intimately related to the behavior of spreading phenomena [29], our results open the door for a geometric perspective in understanding and predicting the efficiency of spreading processes and the location of influential spreaders in real multiplex networks. Indeed, the wide **k**-core structure found in real multiplex systems, explained by interlayer geometric correlations, suggests that there are nodes, located into inner **k**-cores, which could potentially act as efficient spreaders in all layers of the multiplex simultaneously. For instance, we see in Fig. 10 that in the Internet and arXiv multiplexes nodes with high (k, k) -shell index in the multiplex have also high k -shell index in the individual layers. Further, in contrast to arXiv, where the nodes in the most inner k -shells of the individual layers belong also to the most inner (k, k) -shells of the

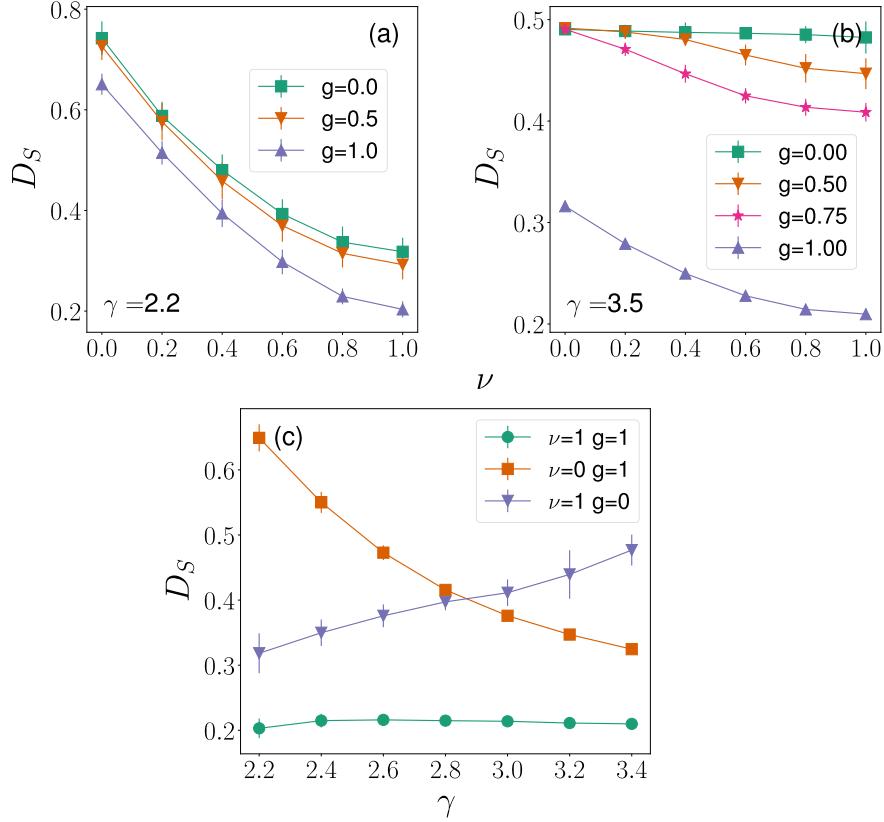


FIG. 9. Quantifying the effect of interlayer degree and similarity correlations in the k -core structure of synthetic multiplex networks. [(a) and (b)] Relative difference $D_S = [\sum_k S(k) - \sum_k S(k, k)] / \sum_k S(k)$ between the monoplex and multiplex relative sizes, $S(k)$ and $S(k, k)$, in two-layer synthetic multiplexes constructed as in Figs. 7 and 8. We consider various combinations of the model parameters ν and g . Results in each case are obtained by taking the average value over 100 realizations. Error bars correspond to one standard deviation away from the average. Reference [39], Fig. 30 shows also the relative difference $D_\xi = [\sum_k \xi_k - \sum_k \xi_{k,k}] / \sum_k \xi_k$ between the angular coherences ξ_k and $\xi_{k,k}$ of the networks of (a) and (b). (c) is the same as (a) and (b) but for different values of the degree exponent γ and parameters ν and g as shown in the legend.

multiplex, in the IPv4/IPv6 Internet there are nodes with high k -shell index values in the individual layers but not in the multiplex. This suggests that there are also nodes that could potentially be efficient spreaders in the individual layers but not in the multiplex. We leave such investigations for future work.

ACKNOWLEDGMENTS

F.R. acknowledges support from the National Science Foundation (CMMI-1552487) and the U.S. Army Research Office (W911NF-16-1-0104).

APPENDIX A: METHODS

1. Cores and shells

The k -core of a single-layer network is the maximal subgraph of the network in which all vertices have degree at least k . The k -core is identified by iteratively removing all nodes with degree less than k , recalculating the degrees of all the

remaining nodes, and continuing with the iterative scheme till there are no nodes with degree less than k . By definition, all nodes in the $(k+n)$ -core, with $n \geq 0$, are necessarily part of the k -core. The nodes that belong to the k -core but not to the $(k+1)$ -core form the k -shell of the network, and they are said to have k -shell index, or coreness, $k_s = k$. The relative size $S(k)$ of the k -core is

$$S(k) = \frac{N_k}{N}, \quad (A1)$$

where N_k is the number of nodes that belong to the k -core, and N is the total number of nodes in the network.

In a multiplex system of L layers, the k -core, with $\mathbf{k} = (k_1, \dots, k_\ell, \dots, k_L)$, is the set of the subgraphs, one for each layer, remaining after the following pruning procedure is performed [32]: all nodes whose degree in at least one layer ℓ is less than k_ℓ are removed from the system; the degree of all nodes in all layers is recomputed; the pruning continues iteratively until no node remains such that its degree in layer ℓ is less than the threshold k_ℓ . By definition, the subgraphs

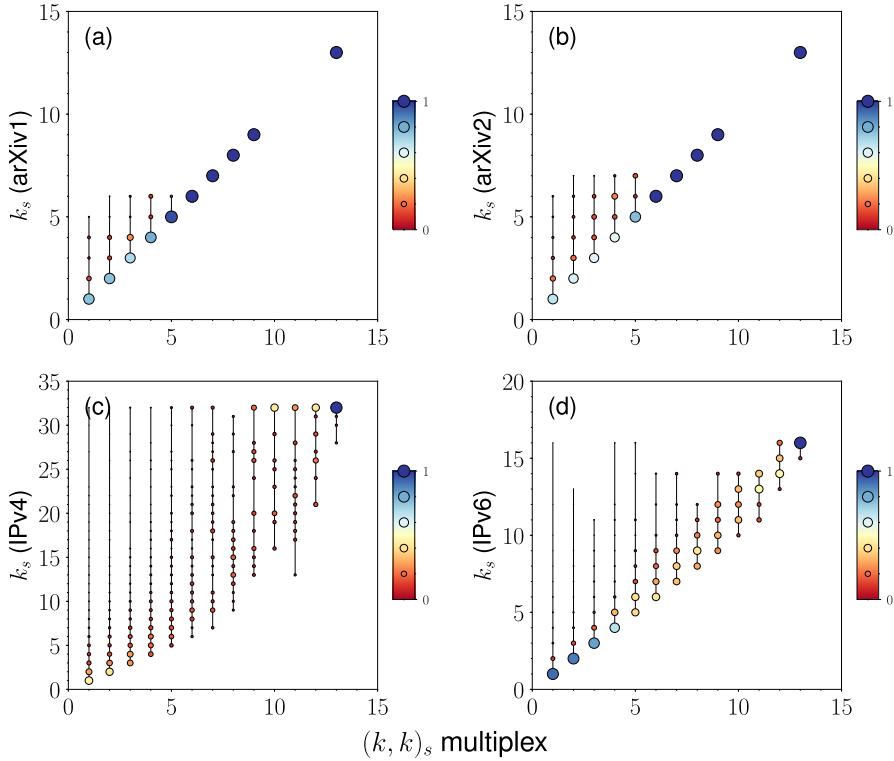


FIG. 10. Coreness in single layers vs coreness in the multiplex. (a) Coreness k_s in arXiv1 of the nodes that have coreness $(k, k)_s$ in the multiplex network consisting of arXiv1 and arXiv2. The size of the symbols is proportional to the percentage of nodes with coreness $(k, k)_s$ in the multiplex that have coreness k_s in arXiv1. (b) Same as (a) but for arXiv2. [(c) and (d)] Same as (a) and (b) but for the IPv4/IPv6 Internet.

belonging to the \mathbf{k} -core share the same set of nodes. Further, the $(\mathbf{k} + \mathbf{n})$ -core of a multiplex, with $\mathbf{n} = (n_1, \dots, n_\ell, \dots, n_L)$ where $n_\ell \geq 0$ for all $\ell = 1, \dots, L$, is necessarily a subset of the \mathbf{k} -core of the multiplex. Similar to single-layer networks one can also define \mathbf{k} -shells. Figure 3(c) in the main text illustrates the (k, k) -shells in the considered arXiv multiplex, i.e., the sets of nodes that belong to the (k, k) -core but not to the $(k+1, k+1)$ -core of the system, $k = 1, 2, \dots, 13$. The relative size $S(\mathbf{k})$ of the \mathbf{k} -core is

$$S(\mathbf{k}) = \frac{N_{\mathbf{k}}}{N}, \quad (\text{A2})$$

where $N_{\mathbf{k}}$ is the number of nodes belonging to the \mathbf{k} -core, and N is the number of common nodes between the layers of the multiplex.

2. Network randomization

a. Single-layer randomization

In Fig. 1, we employed a degree-preserving clustering-decreasing randomization procedure that works as follows. We select a random pair of links (i, j) and (s, t) in the network, and rewire them to (i, t) and (s, j) , provided that none of these links already exist in the network and that the rewiring decreases the average clustering coefficient \bar{c} [42] in

the network. If these two conditions are met, then the rewiring is accepted, otherwise it is not accepted, and a new pair of links is selected. This way each accepted rewiring step preserves the degree distribution in the network, and decreases its average clustering. We repeat the rewiring steps till we reach desired predefined values of the average clustering coefficient \bar{c} , as shown in the legends of Figs. 1(b) and 1(f).

b. Multiplex randomization

In Fig. 3, we employed a node label reshuffling procedure that destroys all correlations between two layers of a multiplex. Specifically, we randomly reshuffled the labels of the nodes of one layer, i.e., we interchanged the label of each node in that layer with the label of a randomly selected node from the same layer. This process randomly reshuffles the trans-layer node-to-node mappings without altering the layer topology.

3. \mathbb{S}^1 model

Each node i in the \mathbb{S}^1 model has hidden variables κ_i, θ_i . The hidden variable κ_i is the node expected degree in the resulting network, while θ_i is the angular (similarity) coordinate of the node on a circle of radius $R = N/(2\pi)$, where N is the total number of nodes. To construct a network with the \mathbb{S}^1 model

that has size N , average node degree \bar{k} , power law degree distribution with exponent $\gamma > 2$, and temperature $T \in [0, 1]$, we perform the following steps.

(i) Sample the angular coordinates of nodes θ_i , $i = 1, 2, \dots, N$, uniformly at random from $[0, 2\pi]$, and their hidden variables κ_i , $i = 1, 2, \dots, N$, from the probability density function

$$\rho(\kappa) = (\gamma - 1)\kappa_0^{\gamma-1}\kappa^{-\gamma}, \quad (\text{A3})$$

where $\kappa_0 = \bar{k}(\gamma - 2)/(\gamma - 1)$ is the expected minimum node degree.

(ii) Connect every pair of nodes i, j with probability

$$p(\chi_{ij}) = \frac{1}{1 + \chi_{ij}^{1/T}}, \quad (\text{A4})$$

where $\chi_{ij} = R\Delta\theta_{ij}/(\mu\kappa_i\kappa_j)$ is the effective distance between i and j , $\Delta\theta_{ij} = \pi - |\pi - |\theta_i - \theta_j||$ is the angular distance, and $\mu = \sin T\pi/(2\bar{k}T\pi)$ is derived from the condition that the expected degree in the network is indeed \bar{k} .

The S^1 model is isomorphic to hyperbolic geometric graphs (\mathbb{H}^2 model) after transforming the expected node degrees κ_i to radial coordinates r_i via

$$r_i = R_H - 2 \ln \frac{\kappa_i}{c}, \quad (\text{A5})$$

where R_H is the radius of the hyperbolic disk where all nodes reside,

$$R_H = 2 \ln \frac{N}{c}, \quad (\text{A6})$$

while $c = \bar{k} \frac{\sin T\pi}{2T} \left(\frac{\gamma-2}{\gamma-1} \right)^2$. After this change of variables the connection probability in (A4) becomes

$$p(x_{ij}) = \frac{1}{1 + e^{\frac{1}{2T}(x_{ij} - R_H)}}, \quad (\text{A7})$$

where $x_{ij} = r_i + r_j + 2 \ln(\Delta\theta_{ij}/2)$ is approximately the hyperbolic distance between nodes i, j [33].

4. Hyperbolic embedding

The hyperbolic embeddings of all considered real-world networks have been obtained in Ref. [23] using the HYPERMAP embedding method [34]. The method is based on maximum likelihood estimation. On its input it takes the network adjacency matrix A . The generic element of the matrix is $A_{ij} = A_{ji} = 1$ if there is a link between nodes i and j , and $A_{ij} = A_{ji} = 0$ otherwise. The embedding infers radial and angular coordinates, respectively indicated as r_i and θ_i , for all nodes $i \leq N$. The radial coordinate r_i is related to the observed node degree k_i as

$$r_i \sim \ln N - 2 \ln k_i. \quad (\text{A8})$$

The angular coordinates of nodes are found by maximizing the likelihood

$$\mathcal{L} = \prod_{1 \leq j < i \leq N} p(x_{ij})^{A_{ij}} [1 - p(x_{ij})]^{1-A_{ij}}. \quad (\text{A9})$$

The product in the above relation goes over all node pairs i, j in the network, x_{ij} is the hyperbolic distance between pair i, j [33] and $p(x_{ij})$ is the connection probability in Eq. (A7).

5. Angular coherence

a. Single-layer networks

To quantify how similar are the angular coordinates of nodes in the k -cores, we use angular coherence, a metric previously used to quantify the extent to which nodes within the same community have similar angular coordinates [25]. We define the angular coherence of a k -core as the module $0 \leq \xi_k \leq 1$, given by

$$\xi_k e^{i\phi_k} = \frac{1}{N_k} \sum_{j \in k\text{-core}} e^{i\theta_j}, \quad (\text{A10})$$

where the sum is taken over the set of nodes that belong to the k -core, N_k is the number of nodes that belong to the k -core, and θ_j is the angular coordinate of node j . The angular coherence resembles the order parameter of the Kuramoto model that captures the coherence of oscillators [43]. The higher is the $\xi_k \in [0, 1]$ the more localized in the similarity space are the nodes of the k -core. At $\xi_k = 1$ all nodes have the same angular coordinates, while at $\xi_k = 0$ nodes are uniformly distributed in $[0, 2\pi]$. ϕ_k in Eq. (A10) can be seen as the k -core “angular coordinate,” i.e., it is a measure of where the k -core is mostly concentrated along the angular similarity direction. We note that the angular coherence of a k -core is an average metric, taken over the nodes that belong to the k -core. Therefore the value of ξ_k does not depend on the number of nodes N_k that belong to the k -core.

b. Multiplex networks

For two-layer multiplex networks, we define the angular coherence of the nodes belonging to the (k, k) -core as the module $0 \leq \xi_{k,k} \leq 1$, given by averaging the angular coherences of the corresponding nodes in the individual layers,

$$\xi_{k,k} e^{i\phi_{k,k}} = \frac{1}{2} \sum_{\ell=1}^2 \left(\frac{1}{N_{k,k}} \sum_{j \in (k,k)\text{-core}} e^{i\theta_j^\ell} \right), \quad (\text{A11})$$

where $N_{k,k}$ is the number of nodes belonging to the (k, k) -core, and θ_j^ℓ is the angular coordinate of node j in layer $\ell = 1, 2$. Similar to ξ_k , $\xi_{k,k}$ does not depend on the number of nodes $N_{k,k}$ that belong to the (k, k) -core.

6. Interlayer similarity

a. Degree correlation

Degree correlation between two layers of a multiplex network is quantified using the Pearson correlation coefficient [23]

$$r_{k,k'} = \frac{\text{cov}(k, k')}{\sigma_k \sigma_{k'}}, \quad (\text{A12})$$

where $\text{cov}(X, X')$ denotes the covariance between two random variables X and X' and σ_x denotes the standard deviation of random variable X . $r_{k,k'}$ takes values in $[-1, 1]$ and is computed across the nodes that are common in the two layers. For $r_{k,k'} = 1$, the degrees of the nodes in the two layers are fully correlated, for $r_{k,k'} = 0$ they are uncorrelated, while for $r_{k,k'} = -1$ they are fully anticorrelated.

b. Angular correlation

Angular correlation between the two layers of a multiplex is quantified using the normalized mutual information [23]

$$\text{NMI}_{\theta,\theta'} = \frac{\text{MI}(\theta; \theta')}{\max\{\text{MI}(\theta; \theta), \text{MI}(\theta'; \theta')\}}, \quad (\text{A13})$$

where MI is the mutual information, computed using the method proposed in Ref. [44]. $\text{NMI}_{X,X'}$ takes values in $[0, 1]$ and is computed across the common nodes in the two layers. $\text{NMI}_{X,X'} = 0$ means no correlation between X and X' , while $\text{NMI}_{X,X'} = 1$ means perfect correlation.

c. Edge overlap

The edge overlap \mathcal{O} between two layers is given by

$$\mathcal{O} = \frac{\sum_{i>j} A_{ij} A'_{ij}}{\min\{\sum_{i>j} A_{ij}, \sum_{i>j} A'_{ij}\}}, \quad (\text{A14})$$

where A and A' are the adjacency matrices of the two layers. The numerator in (A14) is the number of overlapping links between the two layers, while the denominator is the maximum possible number of overlapping links.

7. Geometric multiplex model

The geometric multiplex model (GMM) generates single-layer topologies using the S^1 model (Appendix A 3), and allows for degree and angular coordinate correlations across the layers. Specifically, correlations can be tuned by varying the model parameters $v \in [0, 1]$ (degree correlations) and $g \in [0, 1]$ (angular correlations) [23]. Degree (angular) correlations are maximized at $v \rightarrow 1$ ($g \rightarrow 1$), while at $v \rightarrow 0$ ($g \rightarrow 0$) there are no degree (angular) correlations. The GMM implementation is available in Ref. [45].

- [1] Ginestra Bianconi, *Multilayer Networks: Structure and Function* (Oxford University Press, Oxford, 2018).
- [2] S. Boccaletti, G. Bianconi, R. Criado, C.I. del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendiña-Nadal, Z. Wang, and M. Zanin, The structure and dynamics of multilayer networks, *Phys. Rep.* **544**, 1 (2014).
- [3] K.-M. Lee, B. Min, and Kwang-II Goh, Towards real-world complexity: An introduction to multiplex networks, *Eur. Phys. J. B* **88**, 48 (2015).
- [4] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, Multilayer networks, *J. Complex Networks* **2**, 203 (2014).
- [5] M. De Domenico, A. Solé-Ribalta, E. Cozzo, M. Kivelä, Y. Moreno, M. A. Porter, S. Gómez, and A. Arenas, Mathematical Formulation of Multilayer Networks, *Phys. Rev. X* **3**, 041022 (2013).
- [6] M. Szell, R. Lambiotte, and S. Thurner, Multirelational organization of large-scale social networks in an online world, *Proc. Natl. Acad. Sci. USA* **107**, 13636 (2010).
- [7] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, Community structure in time-dependent, multi-scale, and multiplex networks, *Science* **328**, 876 (2010).
- [8] K. Claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, Internet Mapping: From Art to Science, in *Proceedings of the 2009 Cybersecurity Applications & Technology Conference for Homeland Security, CATCH '09* (IEEE, Washington, DC, 2009), pp. 205–211.
- [9] Ed Bullmore and O. Sporns, Complex brain networks: graph theoretical analysis of structural and functional systems, *Nat. Rev. Neurosci.* **10**, 186 (2009).
- [10] M. De Domenico, V. Nicosia, A. Arenas, and V. Latora, Structural reducibility of multilayer networks, *Nat. Commun.* **6**, 6864 (2015).
- [11] S. Lim, F. Radicchi, M. P van den Heuvel, and O. Sporns, Discordant attributes of structural and functional brain connectivity in a two-layer multiplex network, *Sci. Rep.* **9**, 2885 (2019).
- [12] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin, Catastrophic cascade of failures in interdependent networks, *Nature (London)* **464**, 1025 (2010).
- [13] M. De Domenico, C. Granell, M. A. Porter, and A. Arenas, The physics of spreading processes in multilayer networks, *Nat. Phys.* **12**, 901 (2016).
- [14] G. J. Baxter, S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, Avalanche Collapse of Interdependent Networks, *Phys. Rev. Lett.* **109**, 248701 (2012).
- [15] F. Radicchi and A. Arenas, Abrupt transition in the structural formation of interconnected networks, *Nat. Phys.* **9**, 717 (2013).
- [16] F. Radicchi, Percolation in real interdependent networks, *Nat. Phys.* **11**, 597 (2015).
- [17] F. Radicchi and G. Bianconi, Redundant Interdependencies Boost the Robustness of Multiplex Networks, *Phys. Rev. X* **7**, 011013 (2017).
- [18] S. Osat, A. Faqeeh, and F. Radicchi, Optimal percolation on multiplex networks, *Nat. Commun.* **8**, 1540 (2017).
- [19] S. Osat and F. Radicchi, Observability transition in multiplex networks, *Physica A* **503**, 745 (2018).
- [20] G. J. Baxter, G. Timár, and J. F. F. Mendes, Targeted damage to interdependent networks, *Phys. Rev. E* **98**, 032307 (2018).
- [21] S. D. S. Reis, Y. Hu, A. Babino, J. S. Andrade, Jr., S. Canals, M. Sigman, and H. A. Makse, Avoiding catastrophic failure in correlated networks of networks, *Nat. Phys.* **10**, 762 (2014).
- [22] V. Nicosia and V. Latora, Measuring and modeling correlations in multiplex networks, *Phys. Rev. E* **92**, 032805 (2015).
- [23] K.-K. Kleineberg, M. Boguñá, M. Ángeles Serrano, and F. Papadopoulos, Hidden geometric correlations in real multiplex networks, *Nat. Phys.* **12**, 1076 (2016).
- [24] K.-K. Kleineberg, L. Buzna, F. Papadopoulos, M. Boguñá, and M. Á. Serrano, Geometric Correlations Mitigate the Extreme Vulnerability of Multiplex Networks Against Targeted Attacks, *Phys. Rev. Lett.* **118**, 218301 (2017).
- [25] A. Faqeeh, S. Osat, and F. Radicchi, Characterizing the Analogy between Hyperbolic Embedding and Community Structure of Complex Networks, *Phys. Rev. Lett.* **121**, 098301 (2018).
- [26] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, k -core Organization of Complex Networks, *Phys. Rev. Lett.* **96**, 040601 (2006).

- [27] N. Azimi-Tafreshi, S. Osat, and S. N. Dorogovtsev, Generalization of core percolation on complex networks, *Phys. Rev. E* **99**, 022312 (2019).
- [28] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, Epidemic processes in complex networks, *Rev. Mod. Phys.* **87**, 925 (2015).
- [29] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, Identification of influential spreaders in complex networks, *Nat. Phys.* **6**, 888 (2010).
- [30] R. Pastor-Satorras and C. Castellano, Eigenvector localization in real networks and its implications for epidemic spreading, *J. Stat. Phys.* **173**, 1110 (2018).
- [31] F. Morone, G. Del Ferraro, and H. A. Makse, The k-core as a predictor of structural collapse in mutualistic ecosystems, *Nat. Phys.* **15**, 95 (2019).
- [32] N. Azimi-Tafreshi, J. Gómez-Gardeñes, and S. N. Dorogovtsev, k -core percolation on multiplex networks, *Phys. Rev. E* **90**, 032816 (2014).
- [33] D. Krioukov, F. Papadopoulos, M. Kitsak, A. Vahdat, and M. Boguñá, Hyperbolic geometry of complex networks, *Phys. Rev. E* **82**, 036106 (2010).
- [34] F. Papadopoulos, R. Aldecoa, and D. Krioukov, Network geometry inference using common neighbors, *Phys. Rev. E* **92**, 022807 (2015).
- [35] F. Papadopoulos, M. Kitsak, M. Á. Serrano, M. Boguñá, and D. Krioukov, Popularity versus similarity in growing networks, *Nature (London)* **489**, 537 (2012).
- [36] M. Boguñá, D. Krioukov, and K. C. Claffy, Navigability of complex networks, *Nat. Phys.* **5**, 74 (2009).
- [37] IPv6 topology data, <http://data.caida.org/datasets/topology/ark/ipv6/as-links/2015/01/>.
- [38] M. De Domenico, A. Lancichinetti, A. Arenas, and M. Rosvall, Identifying Modular Flows on Multilayer Networks Reveals Highly Overlapping Organization in Interconnected Systems, *Phys. Rev. X* **5**, 011027 (2015).
- [39] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevResearch.2.023176> for more information.
- [40] M. Boguñá, F. Papadopoulos, and D. Krioukov, Sustaining the Internet with hyperbolic mapping, *Nat. Commun.* **1**, 62 (2010).
- [41] M. Á. Serrano, D. Krioukov, and M. Boguñá, Self-Similarity of Complex Networks and Hidden Metric Spaces, *Phys. Rev. Lett.* **100**, 078701 (2008).
- [42] S. N. Dorogovtsev, *Lectures on Complex Networks* (Oxford University Press, Oxford, 2010).
- [43] J. A. Acebrón, L. L. Bonilla, C. J. Pérez Vicente, F. Ritort, and R. Spigler, The kuramoto model: A simple paradigm for synchronization phenomena, *Rev. Mod. Phys.* **77**, 137 (2005).
- [44] A. Kraskov, H. Stögbauer, and P. Grassberger, Estimating mutual information, *Phys. Rev. E* **69**, 066138 (2004).
- [45] Geometric multiplex model implementation, https://figshare.com/articles/Geometric_Multiplex_Model_Implementation/5513626.

Supplemental Material for k -core structure of real multiplex networks

Saeed Osat,^{1,*} Filippo Radicchi,² and Fragkiskos Papadopoulos³

¹*Deep Quantum Labs, Skolkovo Institute of Science and Technology, Moscow 143026, Russia*

²*Center for Complex Networks and Systems Research,
Luddy School of Informatics, Computing, and Engineering,
Indiana University, Bloomington, Indiana 47408, USA*

³*Department of Electrical Engineering, Computer Engineering and Informatics,
Cyprus University of Technology, 33 Saripolou Street, 3036 Limassol, Cyprus*

CONTENTS

I. Real-world network data	2
II. k -core structure of real-world networks	4
III. k -core structure of real-world multiplex networks	10
IV. Destroying inter-layer degree and similarity correlations	14
V. k -core structure of synthetic multiplex networks	17
References	22

* saeedosat13@gmail.com

I. REAL-WORLD NETWORK DATA

Table I gives an overview of the analyzed single-layer networks, while Table II gives an overview of the considered multiplex systems that consist of network layers from Table I.

Network	N	\bar{k}	γ	Ref.
Airport	3397	11.32	1.88	[1–3]
Drosophila	1770	10.01	1.91	[3, 4]
Metabolic	1436	6.57	2.6	[3, 5]
Proteome	4100	6.52	2.25	[3, 6]
Enron	33696	10.73	2.66	[3, 7, 8]
Internet	23748	4.92	2.17	[3, 9]
Music	2476	16.66	2.27	[3, 10, 11]
Words	7377	11.99	2.25	[3, 12]
arXiv1 (physics.bio-ph)	2956	4.13	2.6	[13, 14]
arXiv2 (physics.data-an)	3506	4.19	2.6	
arXiv3 (physics.soc-ph)	1594	3.79	6.0	
arXiv4 (cond-mat.dis-nn)	5465	5.30	2.5	
arXiv5 (math.OC)	1605	5.52	4.0	
arXiv6 (cond-mat.stat-mech)	1451	3.56	4.0	
arXiv7 (q-bio.MN)	1905	4.64	4.0	
arXiv8 (cs.SI)	4946	4.69	2.5	
C.Elegans1 (Electric)	253	4.06	2.9	[14–16]
C.Elegans2 (Chemical Monadic)	260	6.83	2.9	
C.Elegans3 (Chemical Polyadic)	278	12.25	2.9	
Drosophila1 (Suppressive)	838	4.43	2.6	[14, 17, 18]
Drosophila2 (Additive)	755	3.77	2.8	
Internet1 (IPv4)	37563	5.06	2.1	[9, 14, 19]
Internet2 (IPv6)	5163	5.21	2.1	
Air	69	5.22	2.6	[14, 20]
Train	69	9.33	2.9	

TABLE I. **Single-layer networks.** The first column indicates the name of the network, while N , \bar{k} and γ denote respectively the number of nodes in the network, the network's average degree, and the exponent γ of the power law that best approximates the network's degree distribution. More details on the data can be found in the references listed in the last column.

Multiplex network	N_{common}	$r_{k,k'}$	$\text{NMI}_{\theta,\theta'}$	Ref.
arXiv1 - arXiv2	911	0.82	0.46	[13, 14]
arXiv1 - arXiv4	1441	0.90	0.56	
arXiv1 - arXiv5	354	0.80	0.44	
arXiv2 - arXiv4	1323	0.86	0.48	
Internet1 - Internet2	4731	0.82	0.32	[9, 14]
C.Elegans2 - C.Elegans3	259	0.80	0.31	[14–16]
Drosophila1 - Drosophila2	500	0.83	0.26	[14, 17, 18]
Air - Train	69	0.80	0.08	[14, 20]

TABLE II. **Multiplex networks.** Each two-layer multiplex network is composed of layers from Table I. The first column indicates the two layers that constitute the multiplex. N_{common} is the number of common nodes between the two layers; $r_{k,k'}$ is the Pearson correlation coefficient among the degrees of the nodes in the two layers; and $\text{NMI}_{\theta,\theta'}$ is the normalized mutual information of the angular coordinates of the nodes in the two layers.

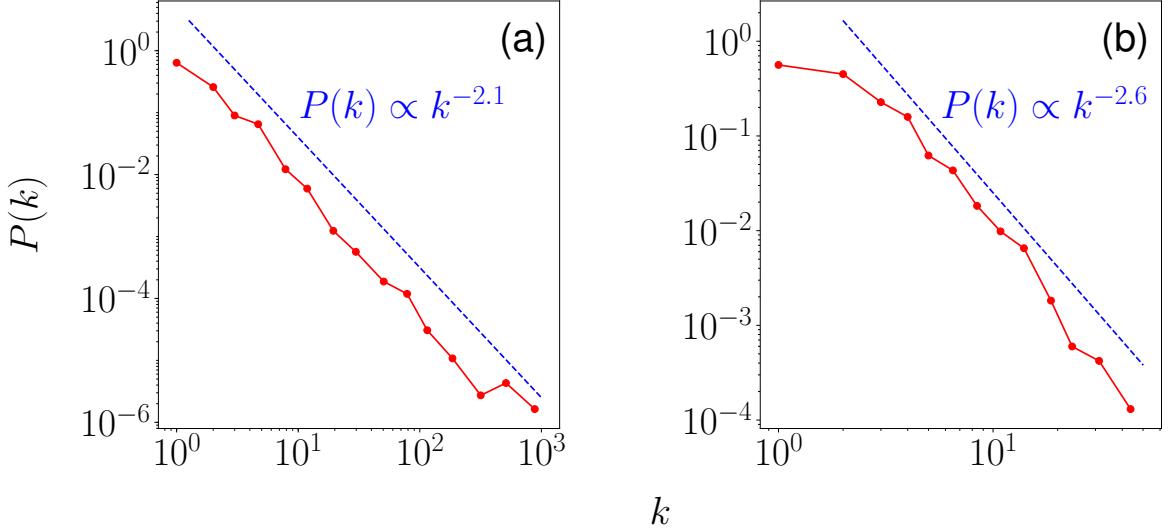


FIG. 1. Degree Distribution. Degree distribution of (a) the IPv6 Internet and (b) the arXiv co-authorship network (physics.bio-ph) considered in the main text. The red lines are the empirical distributions, while the dotted blue lines are power laws with exponents $\gamma = 2.1, 2.6$.

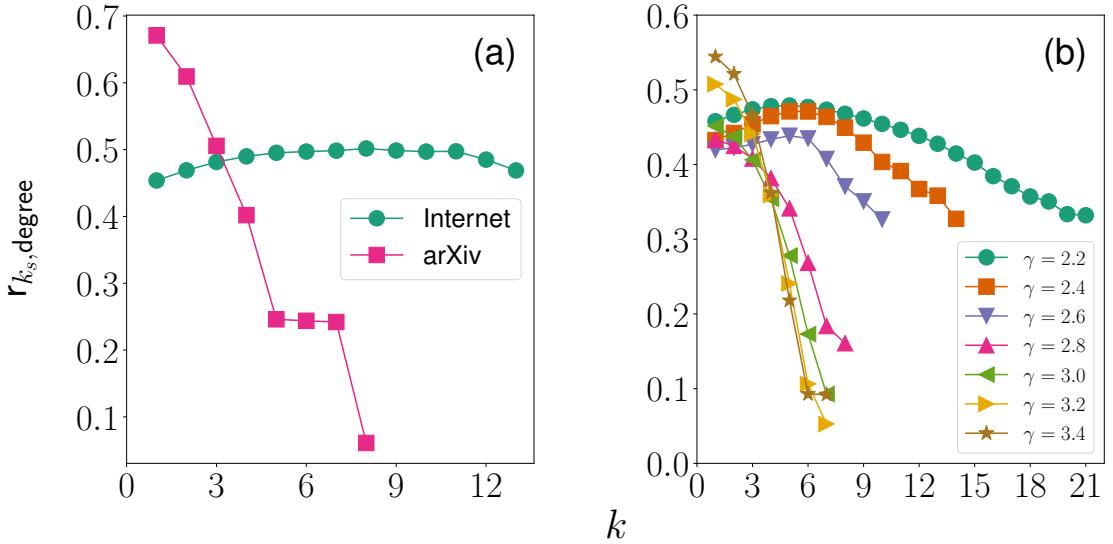


FIG. 2. Correlation between node degree and k -shell index. Pearson correlation coefficient between node degree and k -shell index, $r_{k_s, \text{degree}}$, for nodes with k -shell index $k_s \geq k$ where k as indicated in the x -axis. Panel (a) shows the results for the IPv6 Internet and the arXiv co-authorship network considered in the main text. Panel (b) shows the results for synthetic networks constructed with the S^1 -model (Methods section C in the main text) for various values of the degree exponent γ . The synthetic networks have $N = 10,000$ nodes, average degree $\bar{k} = 6$, and temperature parameter $T = 0.5$, while the results are averages over 100 network realizations. Results for a value of k are shown only if there were samples for that value in at least 20% of the realizations.

II. k -CORE STRUCTURE OF REAL-WORLD NETWORKS

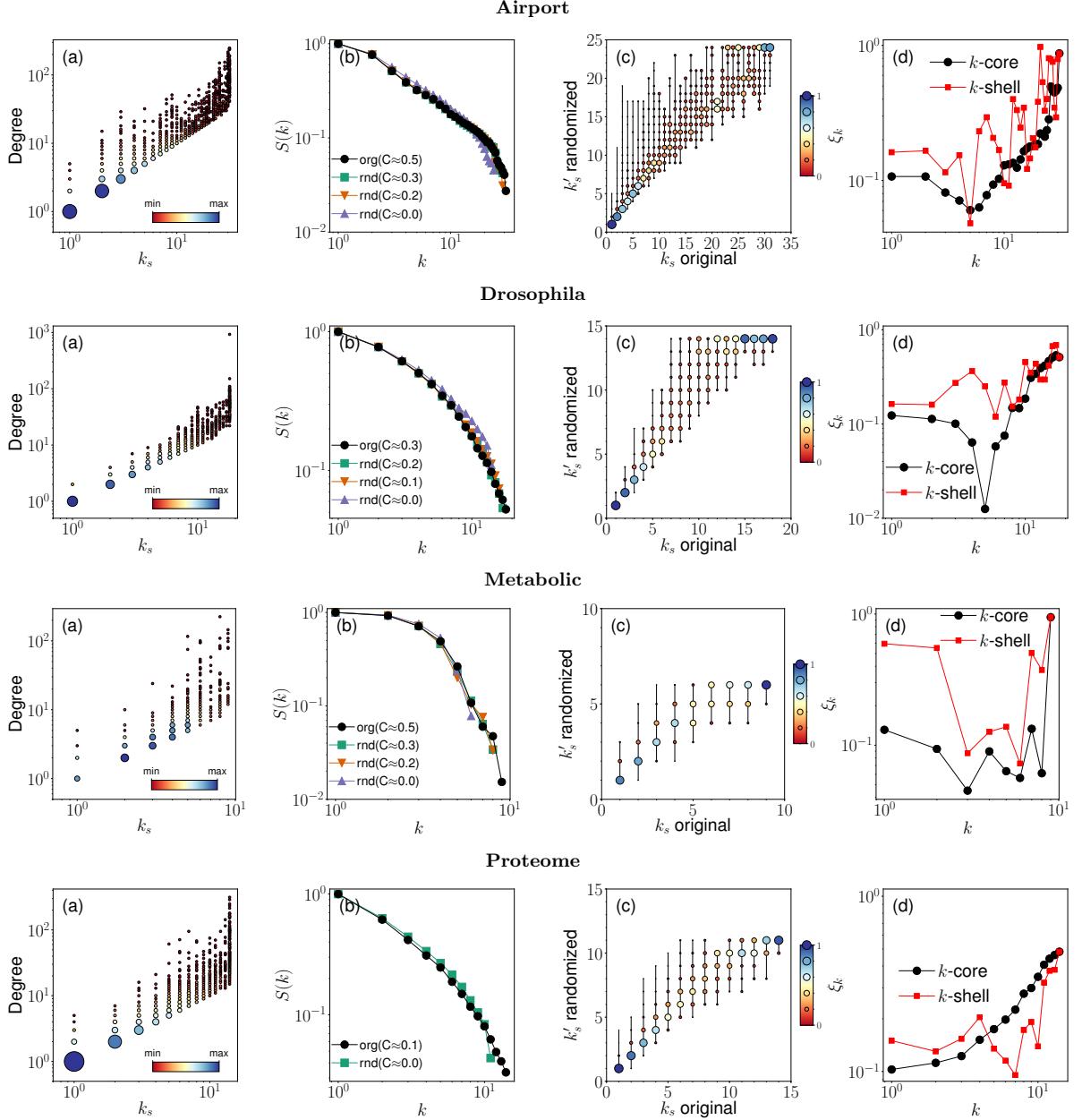


FIG. 3. **k -core structure of real-world networks.** Each row corresponds to a single-layer network from Table I. (a) Scatter plot of node degrees vs. coreness. The size of the symbols is proportional to the number of nodes having each specific degree and k -shell index values. (b) Relative size $S(k)$ of the k -core in the real networks (labeled as “org”) and their randomized counterparts (labeled as “rnd”). Randomized networks are obtained by shuffling random pairs of edges while controlling for the average value of the clustering coefficient C (≈ 0). (c) k -shell index of nodes before and after network randomization (obtained for $C \approx 0$). The size of the symbols is proportional to the percentage of nodes whose coreness changed from k_s in the original network to k'_s in the reshuffled network. (d) Angular coherence ξ_k of the nodes belonging to each k -core and k -shell.

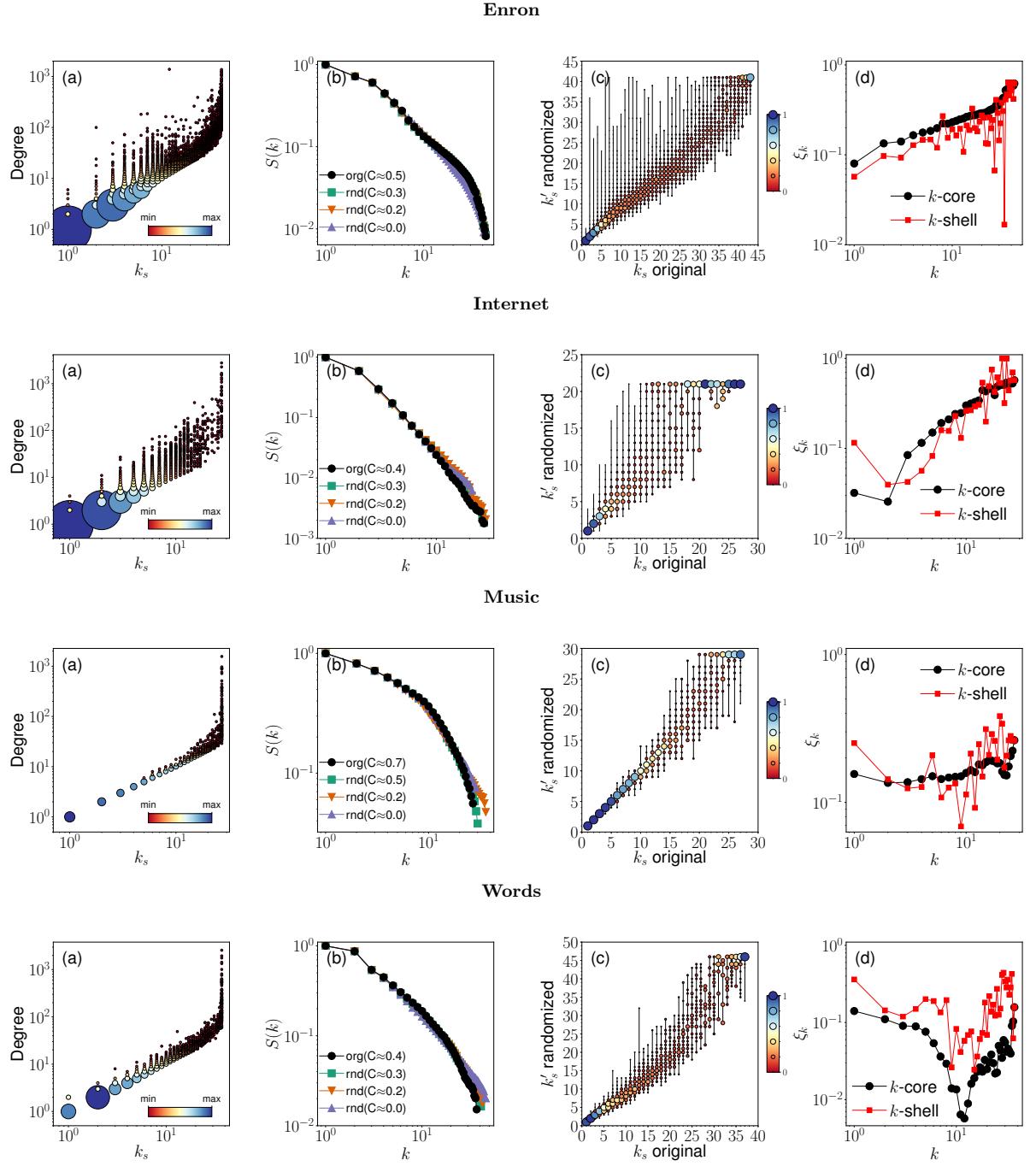
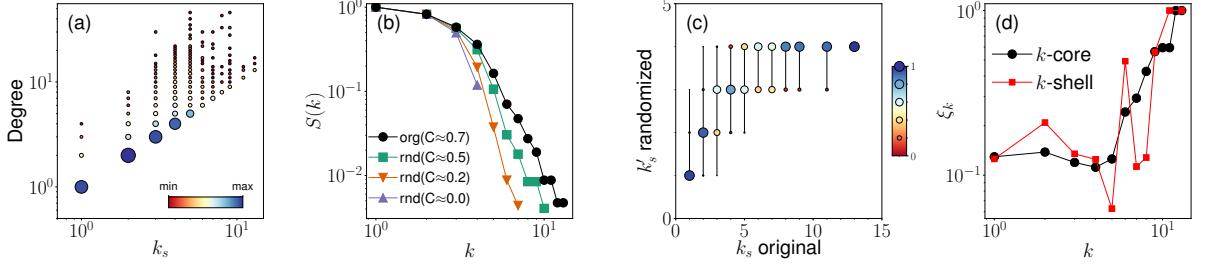
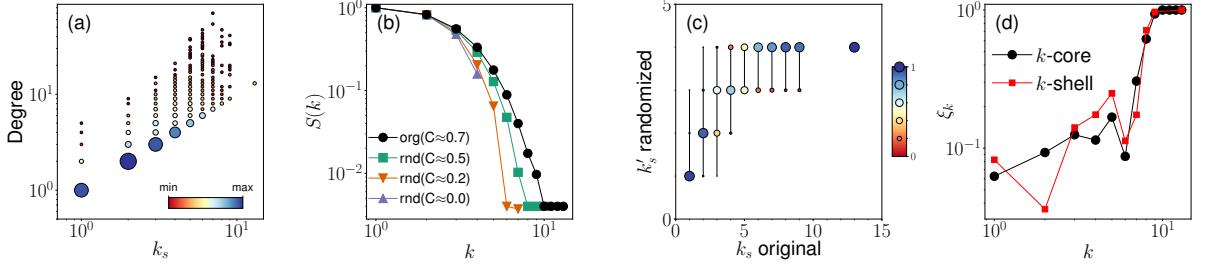


FIG. 4. *k*-core structure of real-world networks. Same as in Figure 3, but for a different set of networks from Table I.

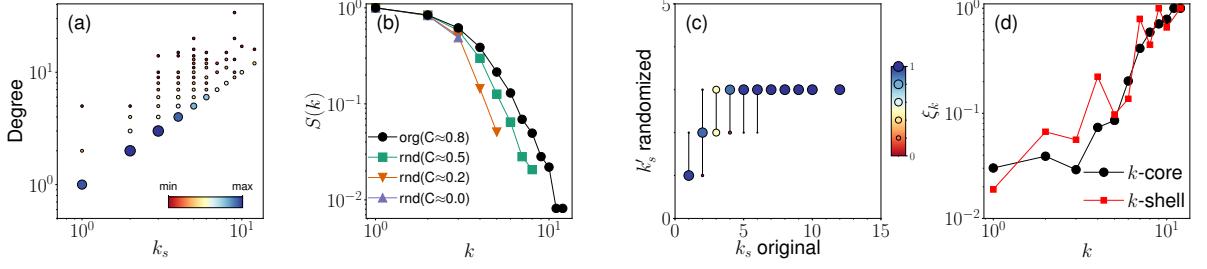
arXiv1



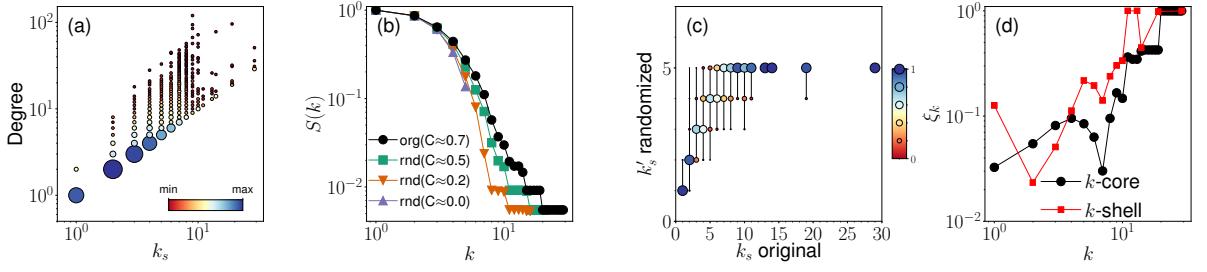
arXiv2



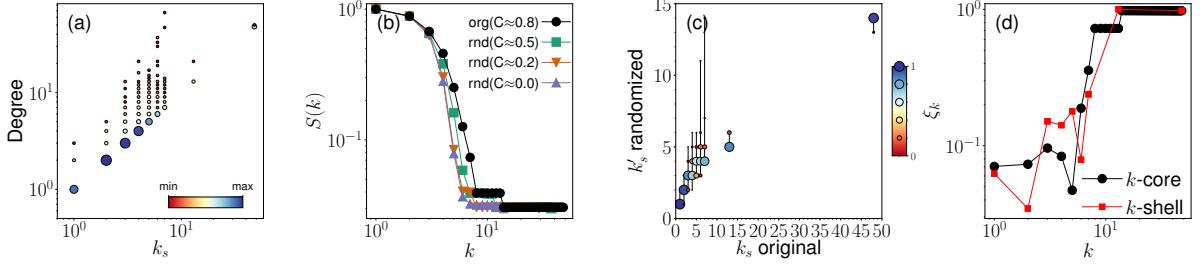
arXiv3



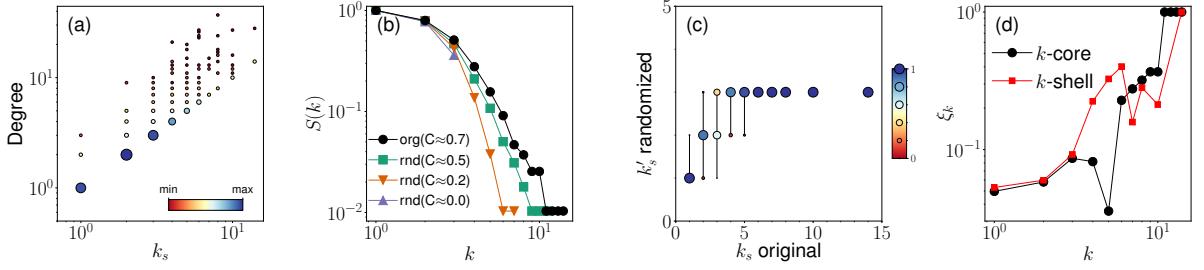
arXiv4

FIG. 5. *k*-core structure of real-world networks. Same as in Figure 3, but for a different set of networks from Table I.

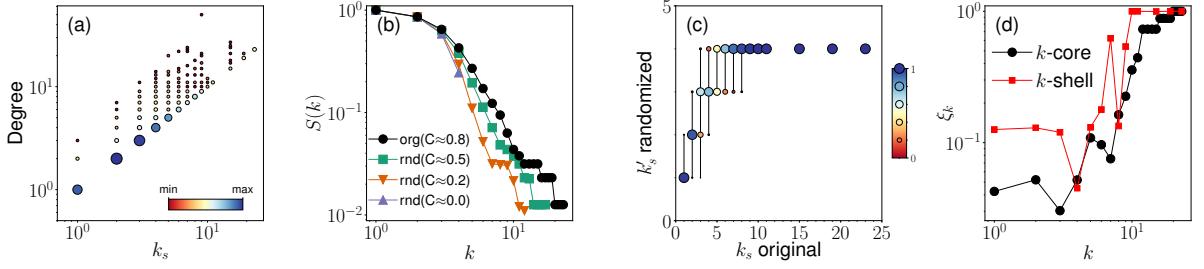
arXiv5



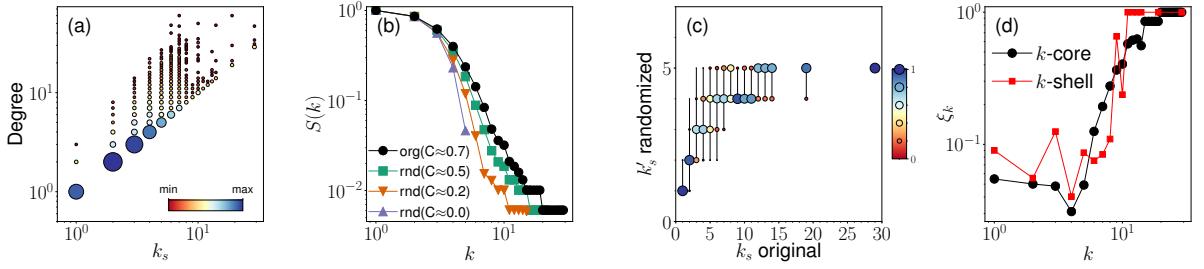
arXiv6



arXiv7



arXiv8

FIG. 6. *k*-core structure of real-world networks. Same as in Figure 3, but for a different set of networks from Table I.

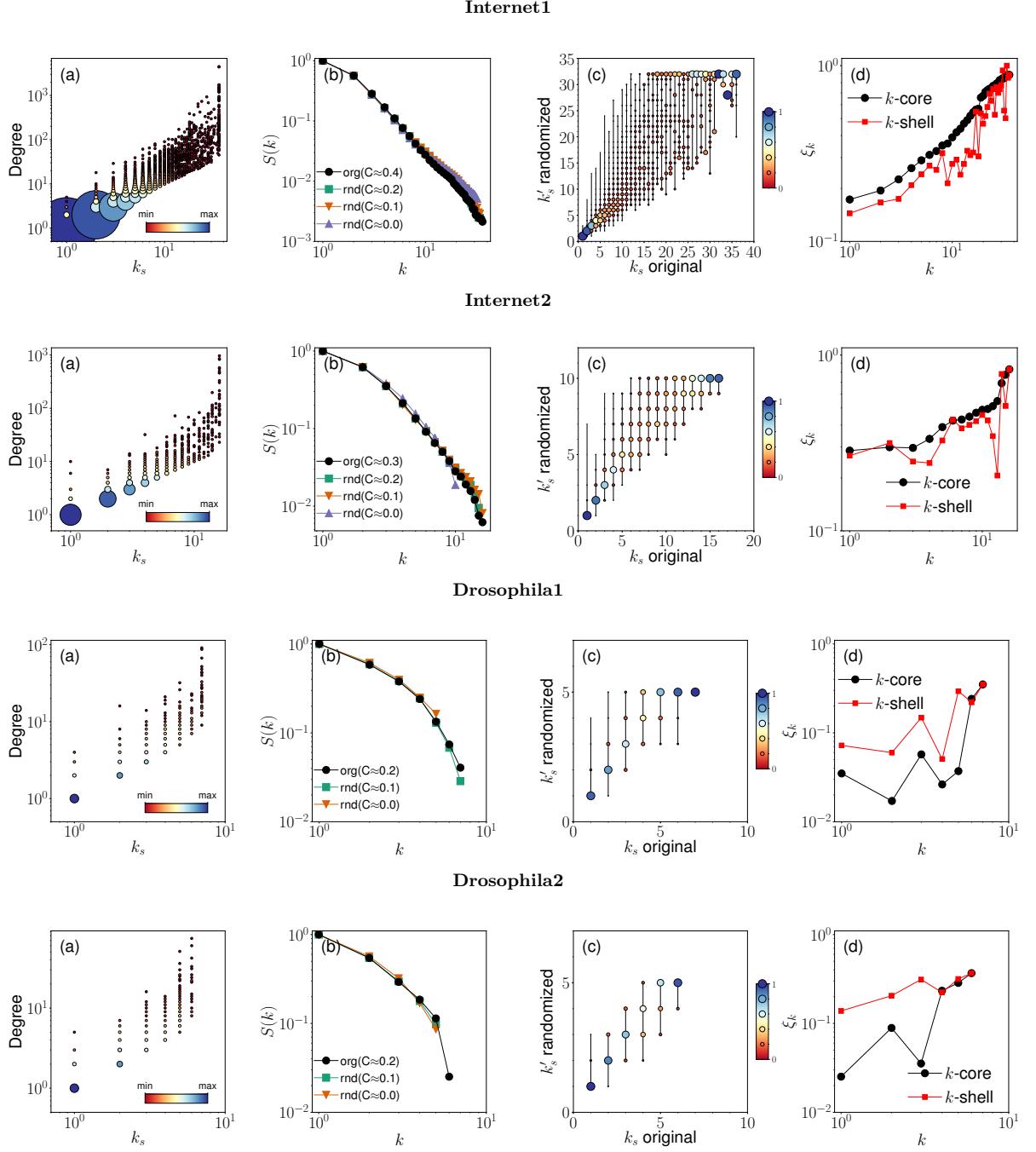
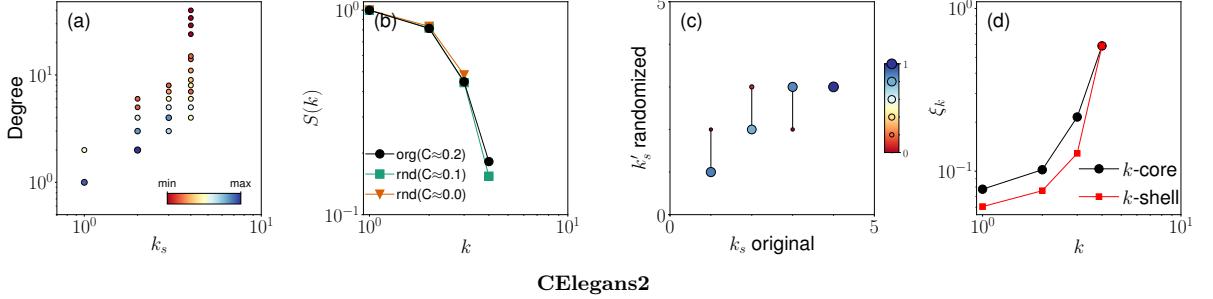
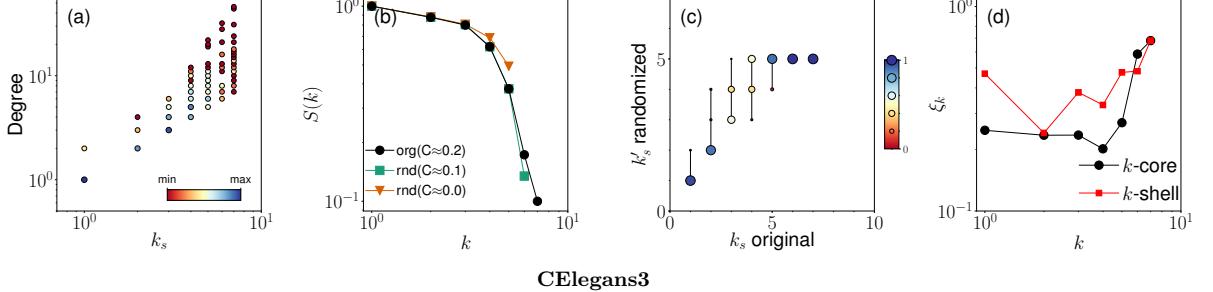


FIG. 7. ***k*-core structure of real-world networks.** Same as in Figure 3, but for a different set of networks from Table I.

CElegans1



CElegans2



CElegans3

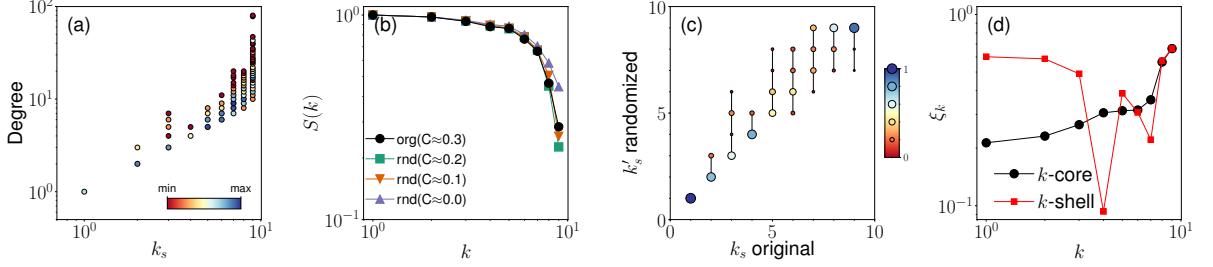


FIG. 8. *k*-core structure of real-world networks. Same as in Figure 3, but for a different set of networks from Table I.

III. k-CORE STRUCTURE OF REAL-WORLD MULTIPLEX NETWORKS

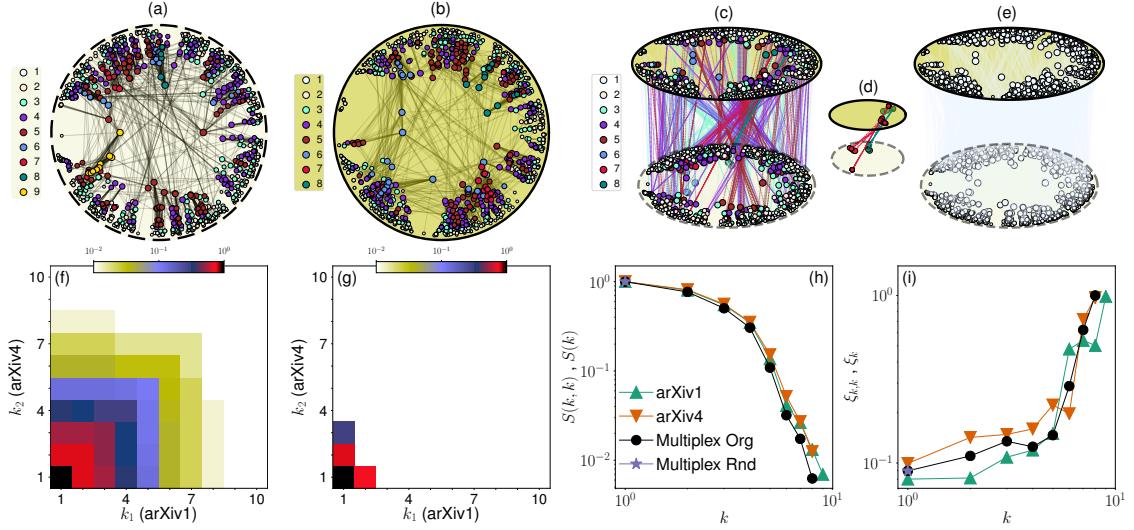


FIG. 9. k-core of real-world multiplex networks. (a and b) Hyperbolic embedding of the arXiv1–arXiv4 multiplex network. Panel a refers to the layer arXiv1, while panel b to the layer arXiv4. The position of the nodes in the disk is determined by their hyperbolic coordinates, and only nodes that exist in both layers are shown; different colors serve to differentiate nodes depending on their k -shell index value. (c) Correspondence among nodes belonging to the (k, k) -shells of the arXiv1–arXiv4 multiplex network. (d) Same as in panel c, but for $k \geq 7$. (e) Same as in panel c, but for the randomized version of the multiplex where the node labels of one of the two layers are randomly reshuffled. (f) Relative size $S(k_1, k_2)$ of the (k_1, k_2) -core for the arXiv1–arXiv4 multiplex network. (g) Same as in panel f, but for the randomized version of the multiplex network. (h) Relative size $S(k, k)$ of the (k, k) -core for the arXiv1–arXiv4 multiplex network, and its randomized version. These curves are compared with those of the relative size $S(k)$ of the k -core of the individual layers. (i) Same as in panel h, but for the metrics of angular coherence $\xi_{k,k}$ and ξ_k .

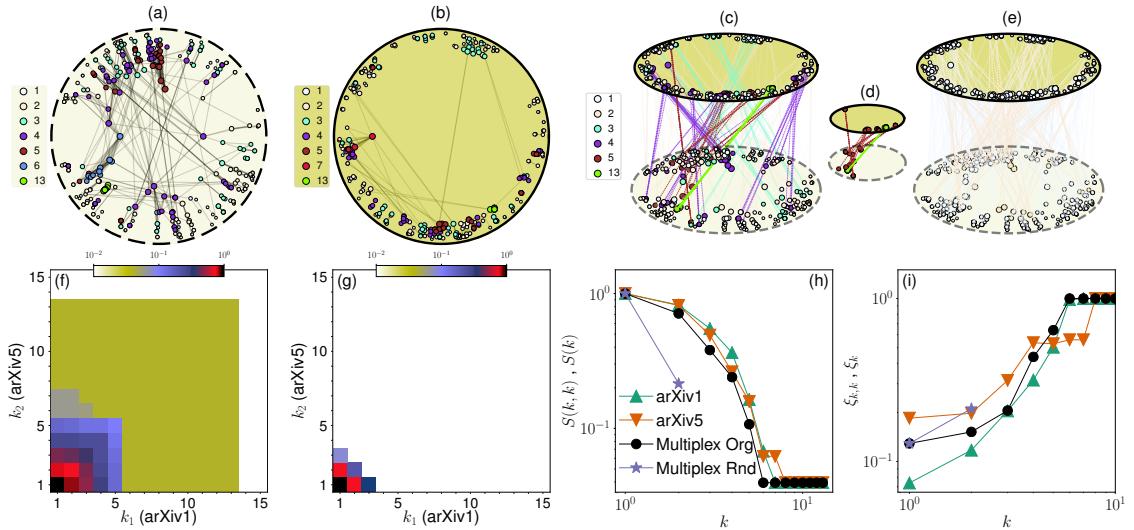


FIG. 10. k-core of real-world multiplex networks. Same as in Figure 9, but for the multiplex consisting of the layers arXiv1 (a) and arXiv5 (b). Panel (d) shows the correspondence among nodes belonging to the (k, k) -shells with $k \geq 5$.

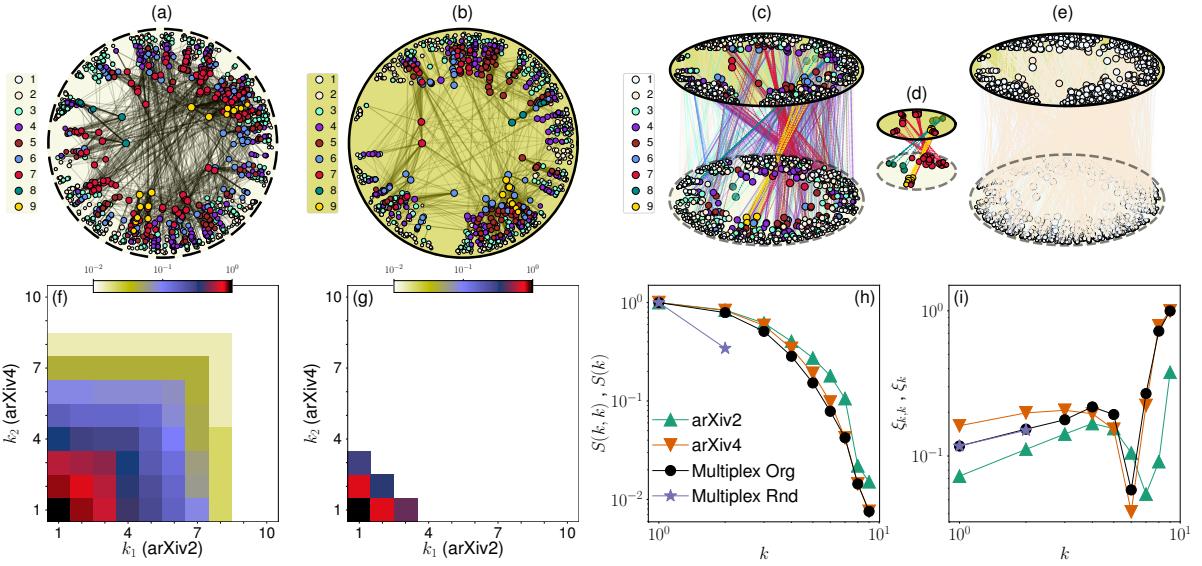


FIG. 11. **k**-core of real-world multiplex networks. Same as in Figure 9, but for the multiplex consisting of the layers arXiv2 (a) and arXiv4 (b). Panel (d) shows the correspondence among nodes belonging to the (k, k) -shells with $k \geq 7$.

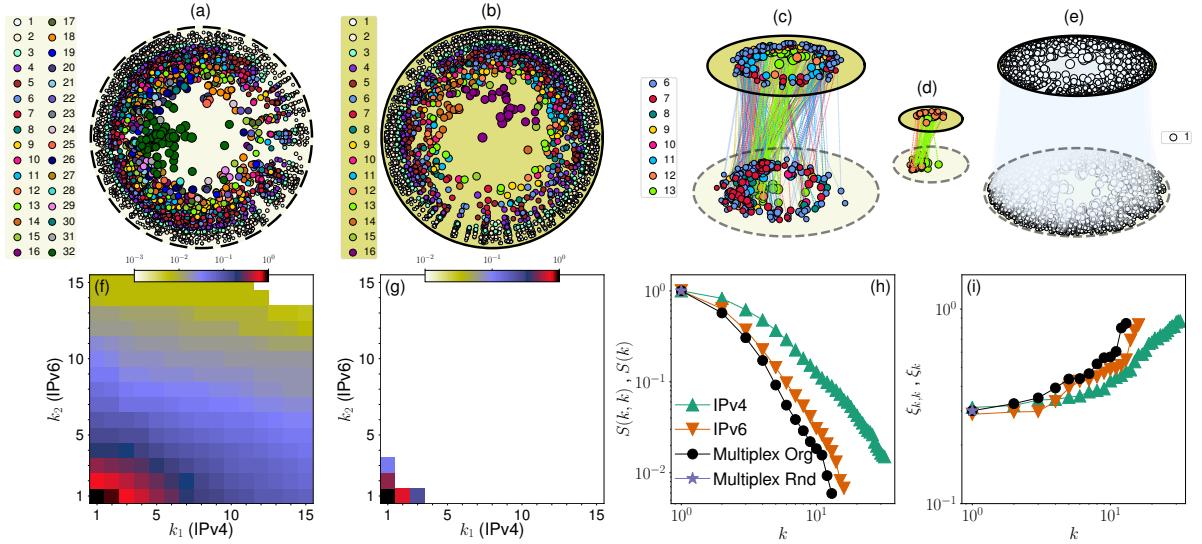


FIG. 12. **k**-core of real-world multiplex networks. Same as in Figure 9, but for the multiplex consisting of the layers IPv4 Internet (a) and the IPv6 Internet (b). Panel (c) shows the correspondence among nodes belonging to the (k, k) -shells with $k \geq 6$. (d) Same as in panel c, but for $k \geq 12$.

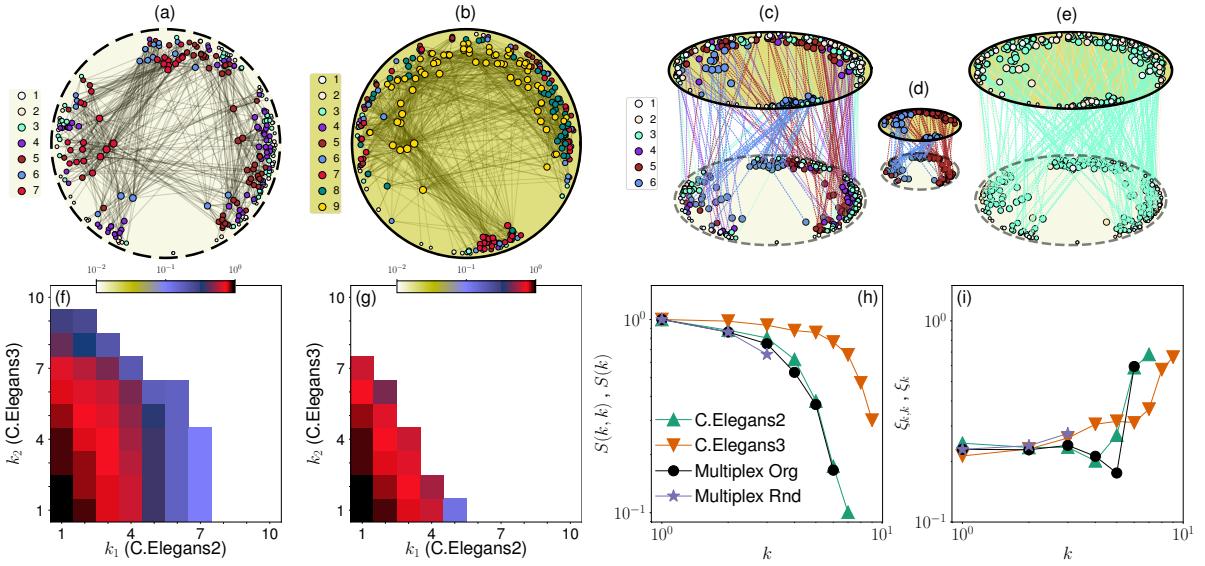


FIG. 13. **k**-core of real-world multiplex networks. Same as in Figure 9, but for the multiplex consisting of the layers C.Elegans2 (a) and C.Elegans3 (b). Panel (d) shows the correspondence among nodes belonging to the (k, k) -shells with $k \geq 5$.

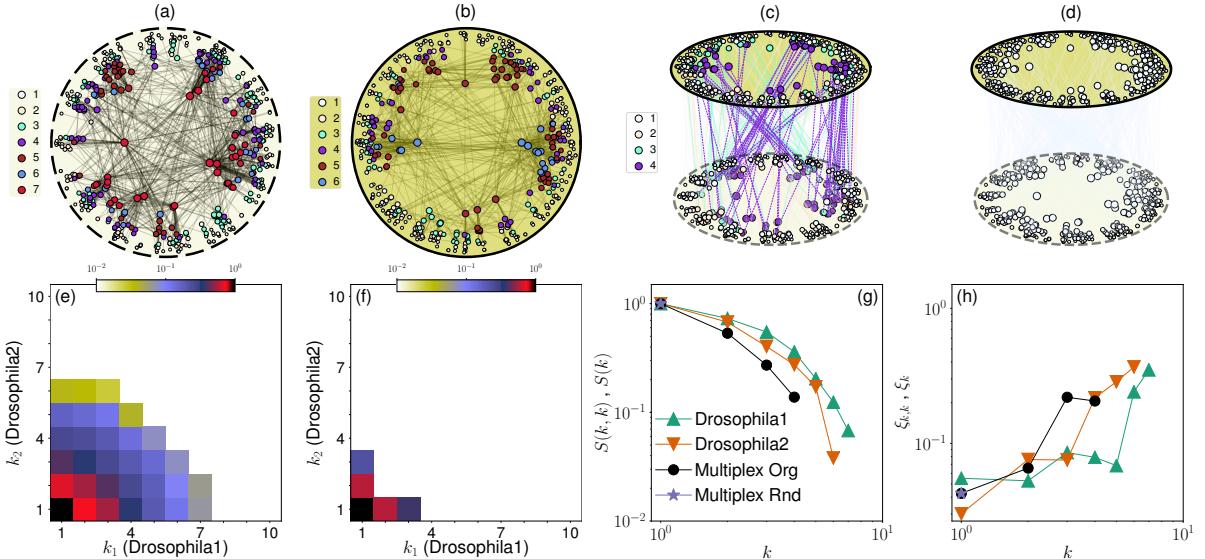


FIG. 14. **k**-core of real-world multiplex networks. Same as in Figure 9, but for the multiplex consisting of the layers Drosophila1 (a) and Drosophila2 (b). This figure does not contain a zoom-in of inner shells.

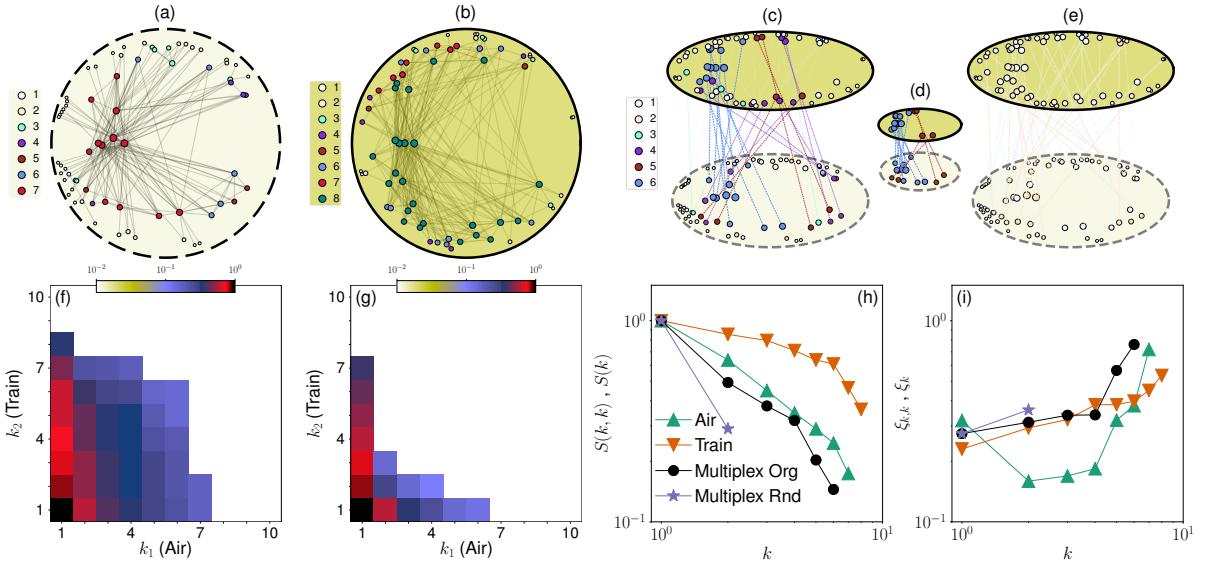


FIG. 15. **k**-core of real-world multiplex networks. Same as in Figure 9, but for the multiplex consisting of the layers Air (a) and Train (b). Panel (d) shows the correspondence among nodes belonging to the (k, k) -shells with $k \geq 5$.

IV. DESTROYING INTER-LAYER DEGREE AND SIMILARITY CORRELATIONS

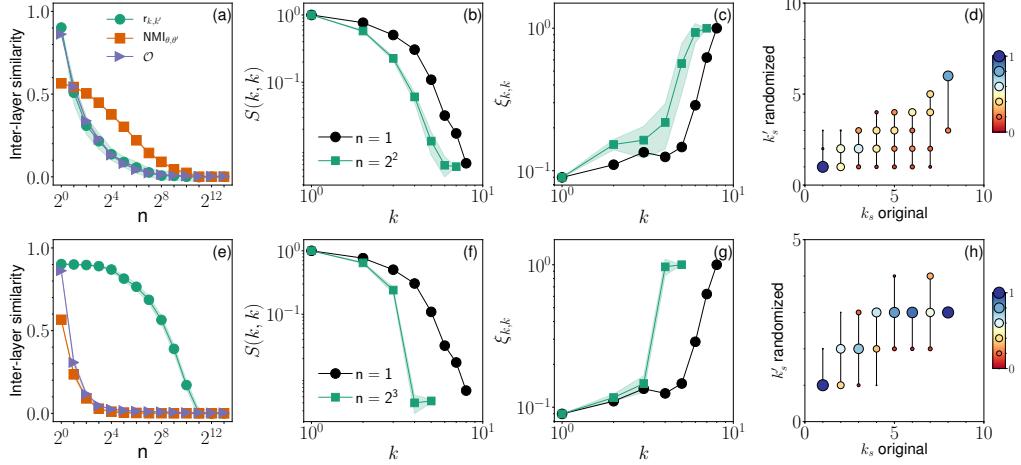


FIG. 16. Inter-layer correlations and the k -core structure of the arXiv1-arXiv4 multiplex network. We analyze the multiplex network consisting of arXiv1 and arXiv4. (a) Different metrics of inter-layer similarity as a function of the group size n used to randomize node labels, thus breaking inter-layer degree correlations. For $n = 1$, node labels of the network are not randomized; full shuffle of node labels is obtained for large n values. We focus here on the case where degree inter-layer correlation is broken, but we preserve inter-layer correlation among nodes' angular coordinates. Metrics of similarities considered here are the Pearson correlation coefficient $r_{k,k'}$ among the degrees of nodes in the two layers; normalized mutual information $NMI_{\theta,\theta'}$ of the angular coordinates of the nodes in the two layers; and edge overlap \mathcal{O} among the two layers. (b) Relative size $S(k,k)$ of the (k,k) -core. The results of the original multiplex network ($n = 1$) are compared with those valid for $n = 4$. The results for $n = 4$ are average values obtained on 100 independent randomizations. Shaded areas identify the region corresponding to one standard deviation away from the average. (c) Same as in panel b, but for the angular coherence $\xi_{k,k}$. (d) Scatter plot of the (k,k) -shell index of nodes in the original vs. the randomized multiplex network. The size of the symbols is proportional to the percentage of points in the scatter plot. (e, f, g and h) Same as in panel a, b, c and d, respectively. We consider here the case where inter-layer correlation among nodes' angular coordinates is destroyed, but inter-layer correlation among nodes' degrees is preserved. The results of the original network are compared with those obtained for $n = 8$.

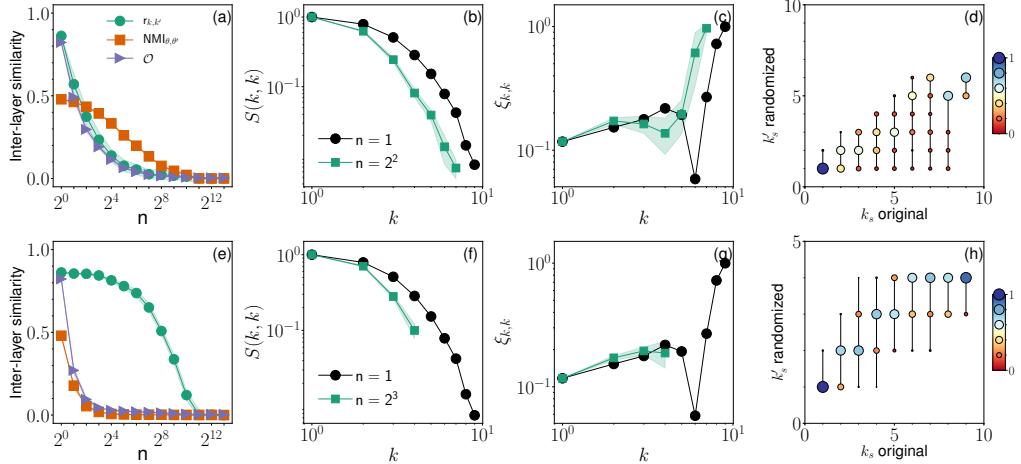


FIG. 17. Inter-layer correlations and the k -core structure of the arXiv2-arXiv4 multiplex network. Same as in Figure 16, but for the multiplex consisting of arXiv2 and arXiv4.

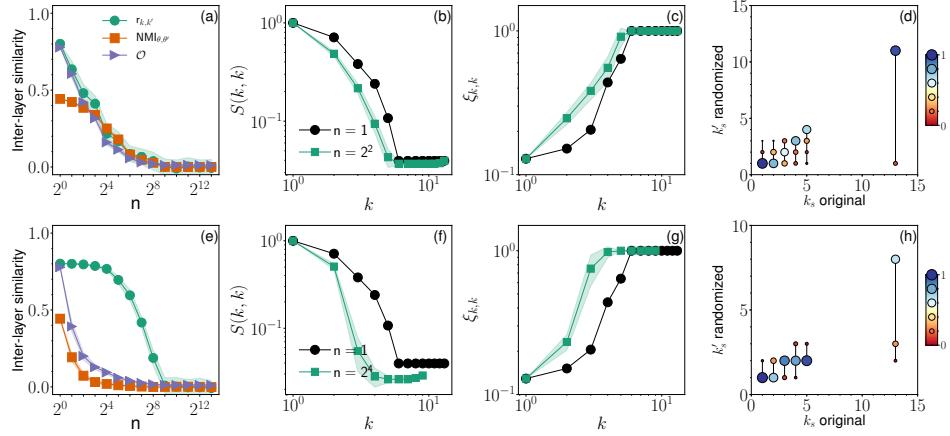


FIG. 18. **Inter-layer correlations and the k-core structure of the arXiv1-arXiv5 multiplex network.** Same as in Figure 16, but for the multiplex consisting of arXiv1 and arXiv5.

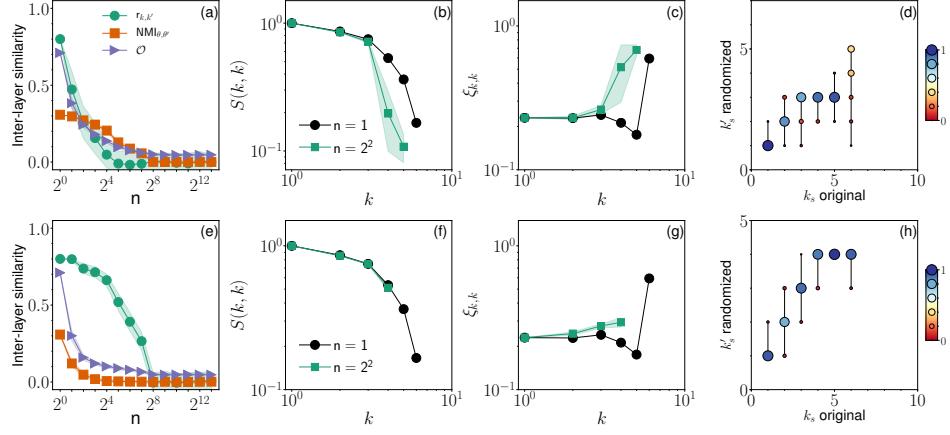


FIG. 19. **Inter-layer correlations and the k-core structure of the C.Elegans2-C.Elegans3 multiplex network.** Same as in Figure 16, but for the multiplex consisting of C.Elegans2 and C.Elegans3.

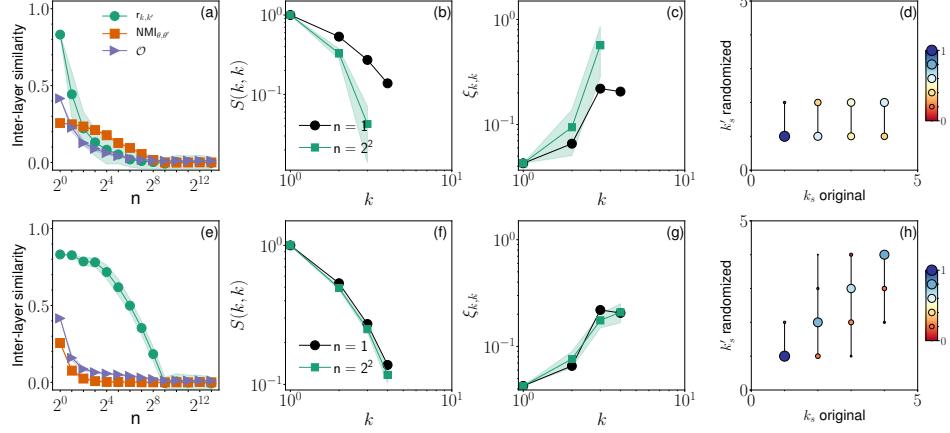


FIG. 20. **Inter-layer correlations and the k-core structure of the Drosophila1-Drosophila2 multiplex network.** Same as in Figure 16, but for the multiplex consisting of Drosophila1 and Drosophila2.

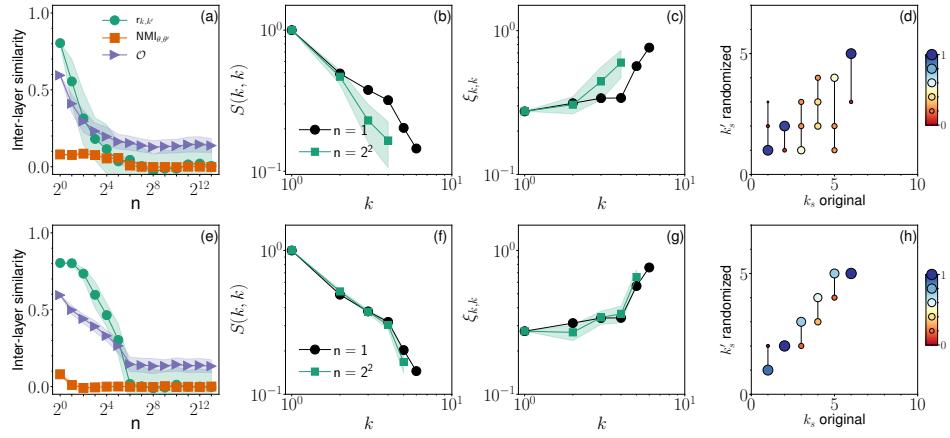


FIG. 21. **Inter-layer correlations and the k -core structure of the Air-Train multiplex network.** Same as in Figure 16, but for the multiplex consisting of Air and Train.

V. k-CORE STRUCTURE OF SYNTHETIC MULTIPLEX NETWORKS

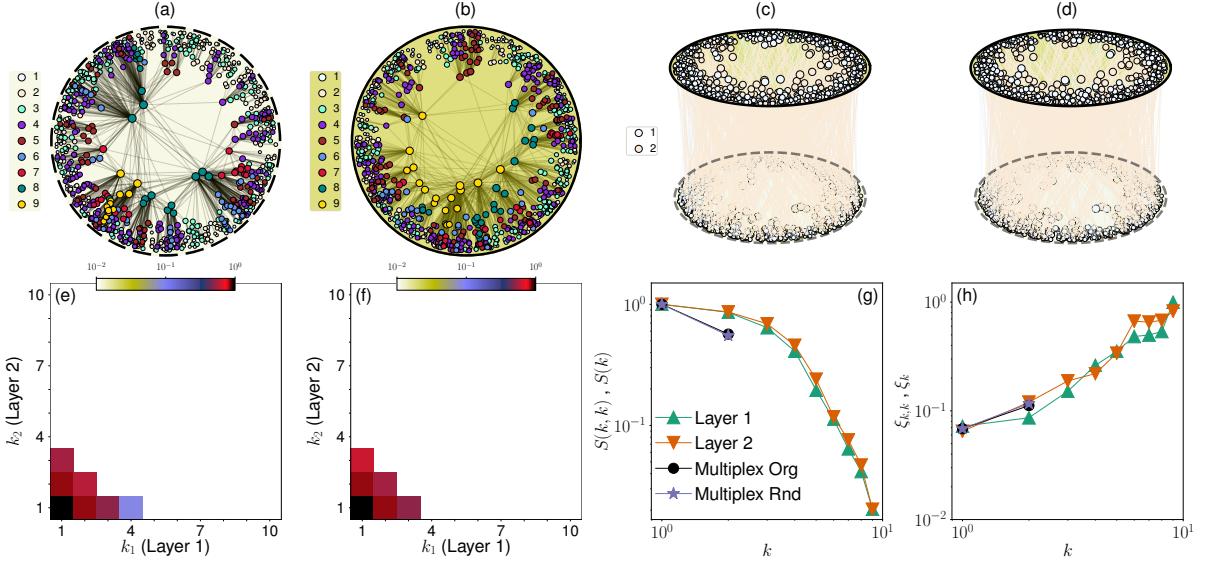


FIG. 22. **k-core structure of synthetic multiplex networks.** Same as in Figure 9, but for a two-layer synthetic multiplex network constructed according to the Geometric Multiplex Model (GMM) with no inter-layer degree and angular correlations ($\nu = g = 0$). Each layer of the multiplex has $N = 1000$ nodes, power-law degree distribution with exponent $\gamma = 2.6$, average degree $\bar{k} \approx 6$, and temperature $T = 0.1$.

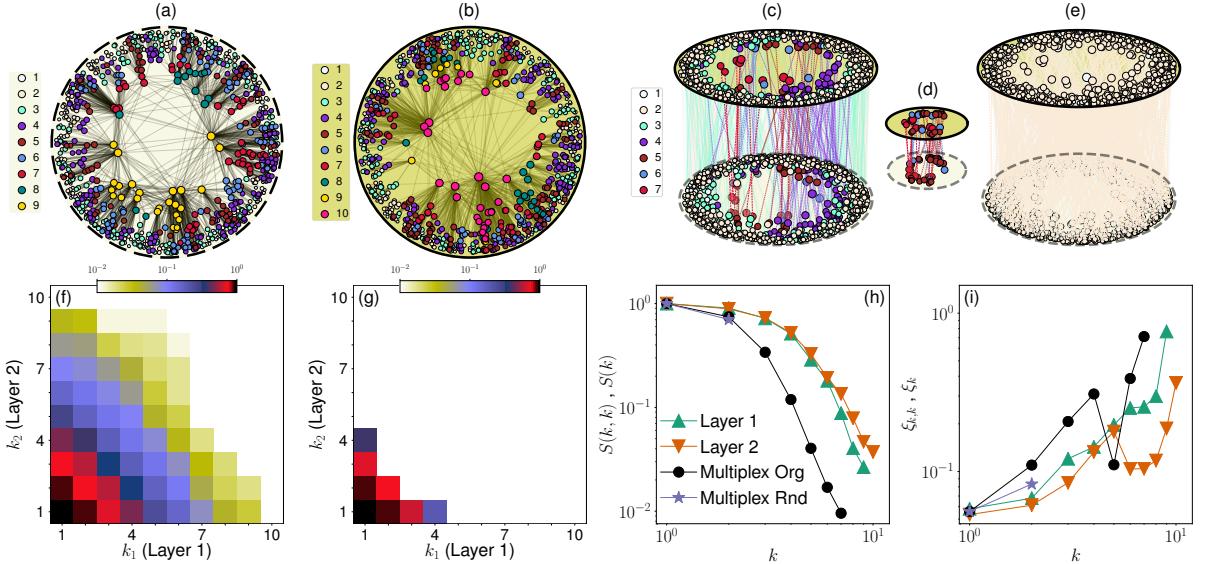


FIG. 23. **k-core structure of synthetic multiplex networks.** Same as in Figure 9, but for a two-layer synthetic multiplex network constructed according to the GMM with intermediate inter-layer degree and angular correlations ($\nu = g = 0.5$). Each layer of the multiplex has the same parameters as in Figure 22. Panel (d) shows the correspondence among nodes belonging to the (k, k) -shells with $k \geq 5$.

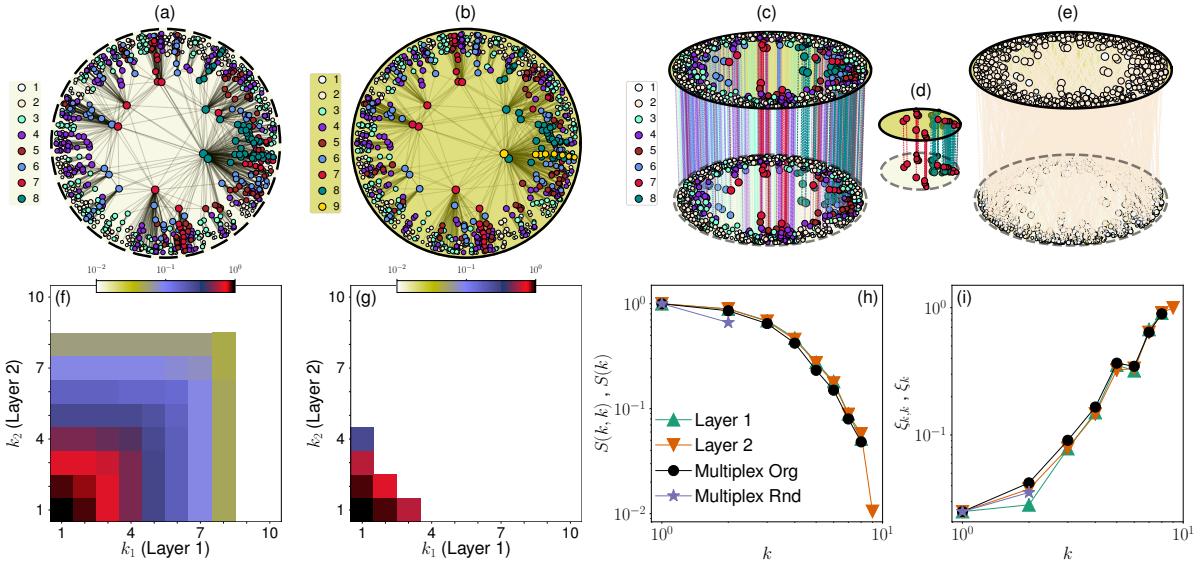


FIG. 24. **k**-core structure of synthetic multiplex networks. Same as in Figure 9, but for a two-layer synthetic multiplex network constructed according to the GMM with maximal inter-layer degree and angular correlations ($\nu = g = 1$). Each layer of the multiplex has the same parameters as in Figure 22. Panel (d) shows the correspondence among nodes belonging to the (k, k) -shells with $k \geq 7$.

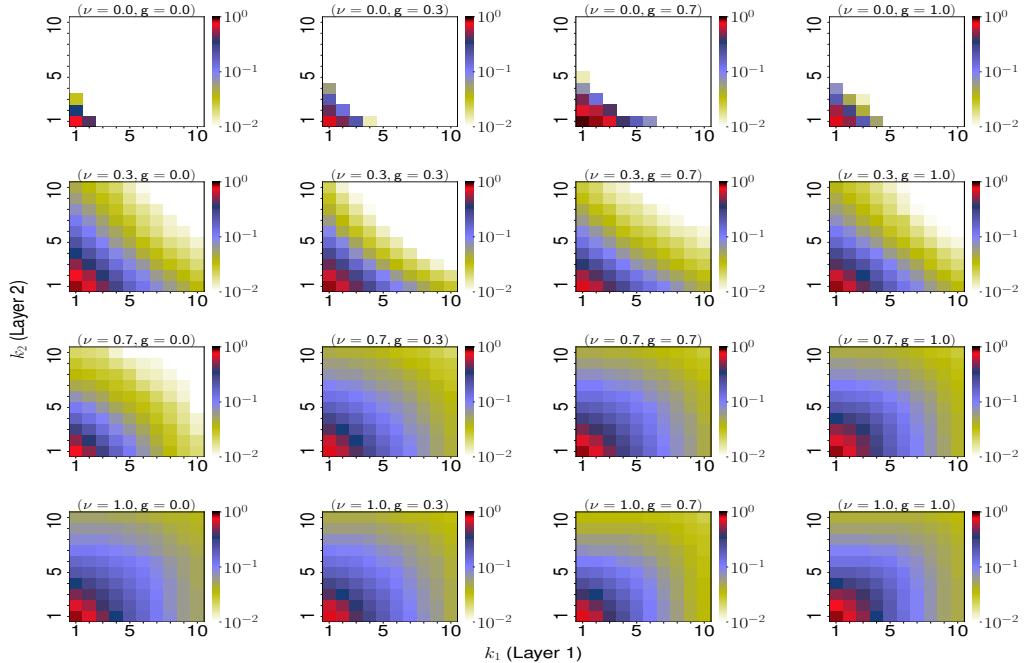


FIG. 25. **k**-core of synthetic multiplex networks. We study here the effect of degree and angular correlations on the relative size $S(k_1, k_2)$ of the (k_1, k_2) -core, in two-layer synthetic multiplexes constructed according to the GMM. Results are shown for different combinations of the inter-layer degree and angular correlation strength parameters $\nu \in [0, 1]$ and $g \in [0, 1]$. Each layer has $N = 1000$ nodes, power-law degree distribution with exponent $\gamma = 2.2$, average degree $\bar{k} \approx 6$, and temperature $T = 0.5$.

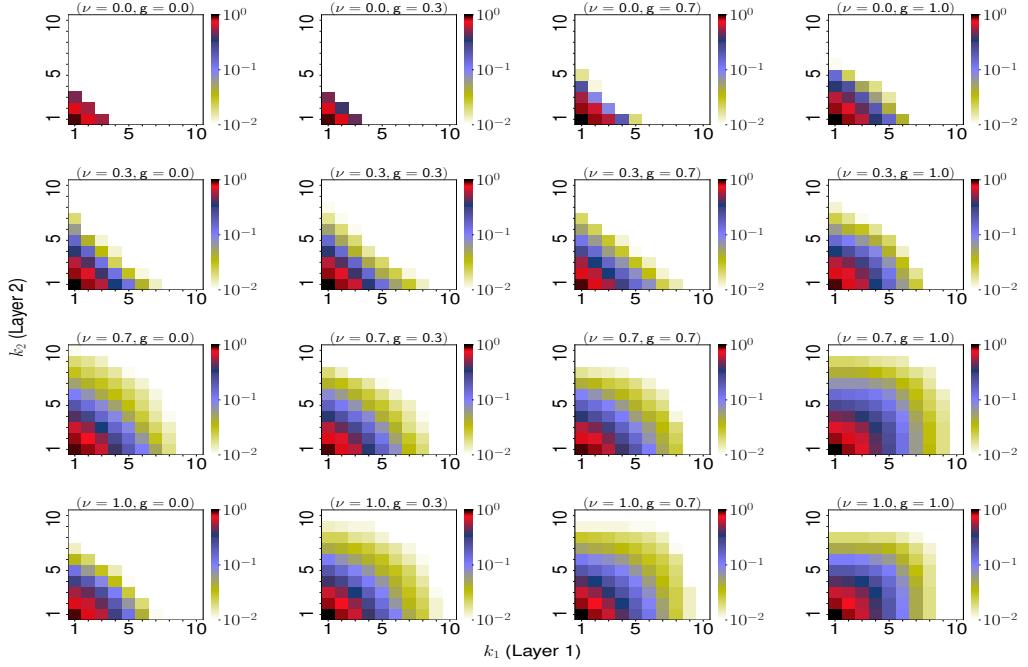


FIG. 26. **k-core of synthetic multiplex networks.** Same as in Figure 25, but for a different value of the degree exponent $\gamma = 2.6$. All other model parameters are identical to those used in Figure 25.

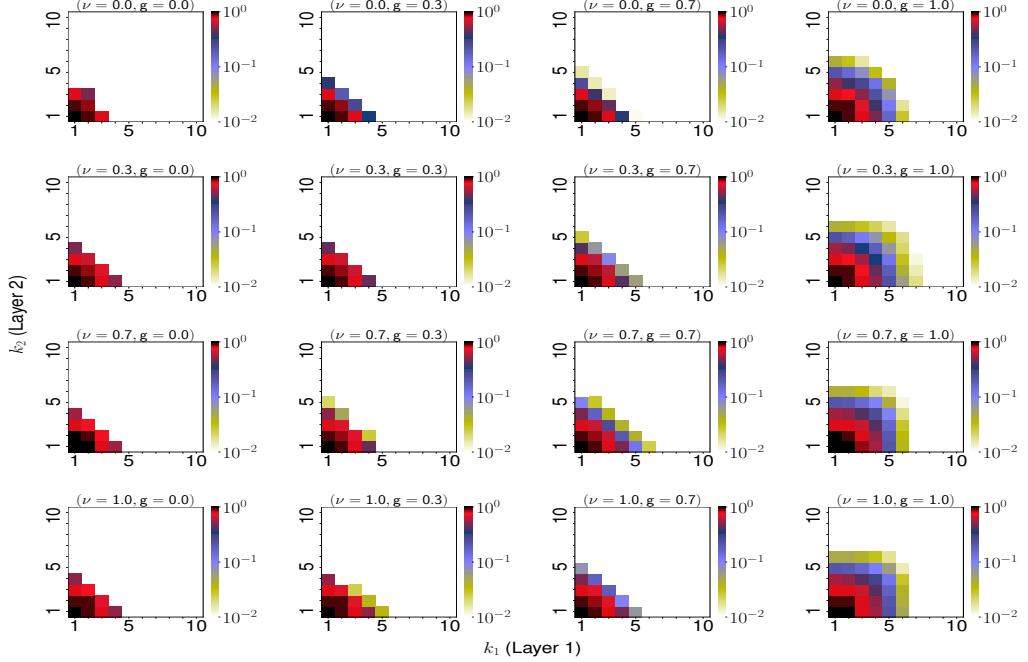


FIG. 27. **k-core of synthetic multiplex networks.** Same as in Figure 25, but for a different value of the degree exponent $\gamma = 3.5$. All other model parameters are identical to those used in Figure 25.

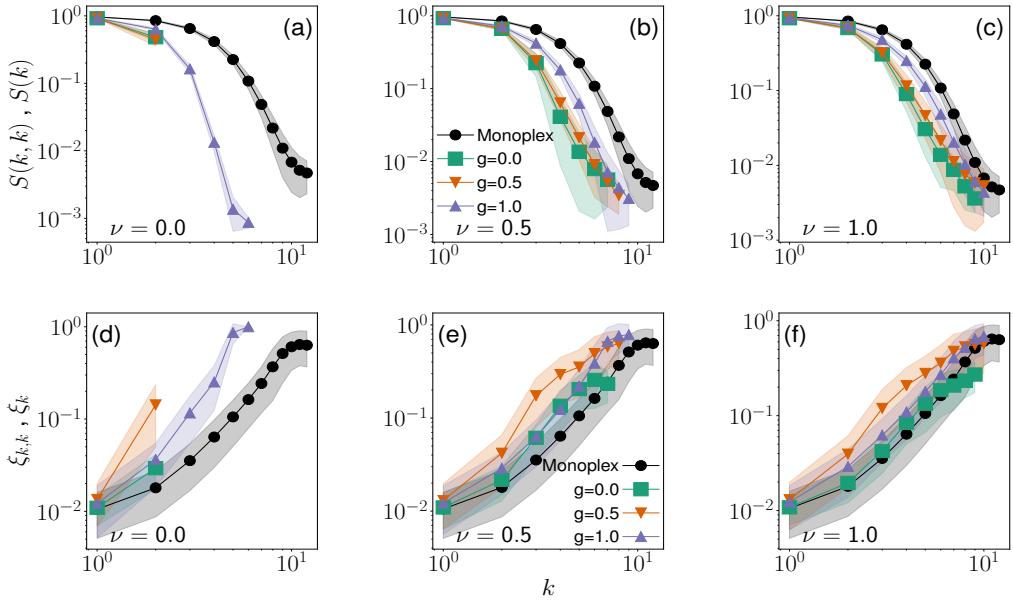


FIG. 28. k -core structure of synthetic multiplex networks. We study here the effect of degree and angular correlations on the size of the (k, k) -core $S(k, k)$ and its coherence $\xi_{k,k}$, in two-layer synthetic multiplex networks constructed according to the GMM. Each layer of the multiplex has $N = 10000$ nodes, power-law degree distribution with exponent $\gamma = 2.6$, average degree $\bar{k} \approx 6$, and temperature $T = 0.5$ (i.e., average clustering coefficient $\bar{c} = 0.4$). We consider various combinations of the degree and angular correlation strength parameters ν and g . Results in each case are obtained by taking the average value over 100 realizations. Shaded areas denote regions corresponding to one standard deviation away from the average. (a) Relative size $S(k, k)$ of the (k, k) -core as a function of the threshold k . The curve corresponding to the monoplex is obtained by measuring $S(k)$ for the k -core of the individual layers, and then taking the average value. (b and c) Same as in panel a, but for different choices of the model parameters. (d, e and f) We consider the same data as in panels a, b, and c, respectively, but we monitor the metrics of angular coherence $\xi_{k,k}$ and ξ_k as functions of the threshold value k .

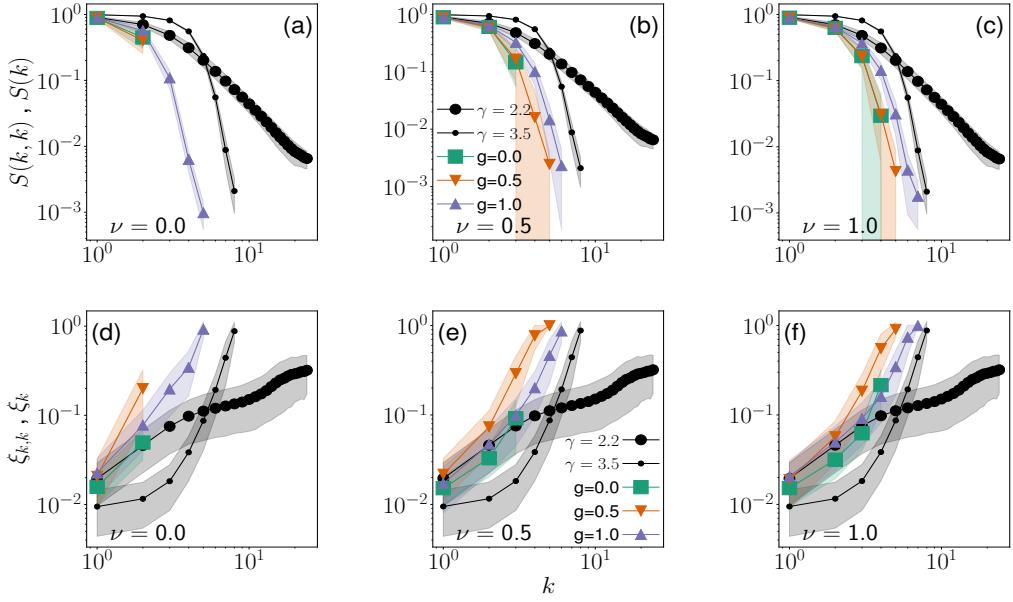


FIG. 29. k -core structure of synthetic multiplex networks. Same as in Figure 28, but for power-law degree distribution with exponent $\gamma = 2.2$ in one layer and $\gamma = 3.5$ in the other.

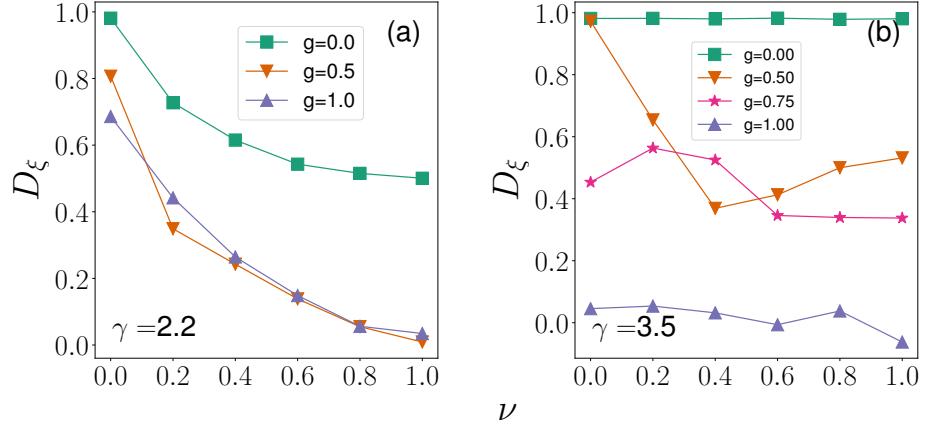


FIG. 30. **Quantifying the effect of inter-layer degree and similarity correlations in the k-core structure of synthetic multiplex networks.** (a and b) Relative difference $D_\xi = [\sum_k \xi_k - \sum_k \xi_{k,k}] / \sum_k \xi_k$ between the angular coherences ξ_k and $\xi_{k,k}$ for the networks constructed in Figures 9a and 9b in the main text. Error bars are not shown for clarity.

-
- [1] “KONECT – openflights network dataset,” <http://konect.uni-koblenz.de/networks/openflights> (2016).
- [2] J. Kunegis, in Proc. Int. Conf. on World Wide Web Companion, pp. 1343–135 (2013).
- [3] Guillermo García-Pérez, Marián Boguñá, and M. Ángeles Serrano, “Multiscale unfolding of real networks by geometric renormalization,” *Nature Physics* **14**, 583–589 (2018).
- [4] Shin-ya Takemura, Arjun Bharioke, Zhiyuan Lu, Aljoscha Nern, Shiv Vitaladevuni, Patricia K. Rivlin, William T. Katz, Donald J. Olbris, Stephen M. Plaza, Philip Winston, Ting Zhao, Jane Anne Horne, Richard D. Fetter, Satoko Takemura, Katerina Blazek, Lei-Ann Chang, Omotara Ogundeyi, Mathew A. Saunders, Victor Shapiro, Christopher Sigmund, Gerald M. Rubin, Louis K. Scheffer, Ian A. Meinertzhangen, and Dmitri B. Chklovskii, “A visual motion detection circuit suggested by drosophila connectomics,” *Nature* **500**, 175 EP – (2013).
- [5] M. Ángeles Serrano, Marián Boguñá, and Francesc Sagués, “Uncovering the hidden geometry behind metabolic networks,” *Mol. BioSyst.* **8**, 843–850 (2012).
- [6] Thomas Rolland, Murat Taşan, Benoit Charlotteaux, Samuel J. Pevzner, Quan Zhong, Nidhi Sahni, Song Yi, Irma Lemmens, Celia Fontanillo, Roberto Mosca, Atanas Kamburov, Susan D. Ghiassian, Xinpeng Yang, Lila Ghamsari, Dawit Balcha, Bridget E. Begg, Pascal Braun, Marc Brehme, Martin P. Broly, Anne-Ruxandra Carvunis, Dan Convery-Zupan, Roser Corominas, Jasmin Coulombe-Huntington, Elizabeth Dann, Matija Dreze, Amélie Dricot, Changyu Fan, Eric Franzosa, Fana Gebreab, Bryan J. Gutierrez, Madeleine F. Hardy, Mike Jin, Shuli Kang, Ruth Kiros, Guan Ning Lin, Katja Luck, Andrew MacWilliams, Jörg Menche, Ryan R. Murray, Alexandre Palagi, Matthew M. Poulin, Xavier Rambout, John Rasla, Patrick Reichert, Viviana Romero, Elien Ruysinck, Julie M. Sahalie, Annemarie Scholz, Akash A. Shah, Amitabh Sharma, Yun Shen, Kerstin Spirohn, Stanley Tam, Alexander O. Tejeda, Shelly A. Trigg, Jean-Claude Twizere, Kerwin Vega, Jennifer Walsh, Michael E. Cusick, Yu Xia, Albert-László Barabási, Lilia M. Iakoucheva, Patrick Aloy, Javier De Las Rivas, Jan Tavernier, Michael A. Calderwood, David E. Hill, Tong Hao, Frederick P. Roth, and Marc Vidal, “A proteome-scale map of the human interactome network,” *Cell* **159**, 1212 – 1226 (2014).
- [7] Bryan Klimt and Yiming Yang, “Introducing the enron corpus,” in *CEAS 2004 - First Conference on Email and Anti-Spam, July 30-31, 2004, Mountain View, California, USA* (2004).
- [8] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney, “Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters,” *Internet Mathematics* **6**, 29–123 (2009), <https://doi.org/10.1080/15427951.2009.10129177>.
- [9] k. claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, *IEEE DHS Cybersecurity Applications and Technologies Conference for Homeland Security (CATCH)* (2009) pp. 205–211.
- [10] Joan Serrà, Álvaro Corral, Marián Boguñá, Martí Haro, and Josep Ll Arcos, “Measuring the evolution of contemporary western popular music,” *Scientific Reports* **2**, 521 EP – (2012), article.
- [11] M. Ángeles Serrano, Marián Boguñá, and Alessandro Vespignani, “Extracting the multiscale backbone of complex weighted networks,” *Proceedings of the National Academy of Sciences* **106**, 6483–6488 (2009), <https://www.pnas.org/content/106/16/6483.full.pdf>.
- [12] Ron Milo, Shalev Itzkovitz, Nadav Kashtan, Reuven Levitt, Shai Shen-Orr, Inbal Ayzenststat, Michal Sheffer, and Uri Alon, “Superfamilies of evolved and designed networks,” *Science* **303**, 1538–1542 (2004), <https://science.sciencemag.org/content/303/5663/1538.full.pdf>.
- [13] Manlio De Domenico, Andrea Lancichinetti, Alex Arenas, and Martin Rosvall, “Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems,” *Phys. Rev. X* **5**, 011027 (2015).
- [14] Kaj-Kolja Kleineberg, Marián Boguñá, M. Ángeles Serrano, and Fragkiskos Papadopoulos, “Hidden geometric correlations in real multiplex?networks,” *Nature Physics* **12**, 1076 EP – (2016).
- [15] Beth L. Chen, David H. Hall, and Dmitri B. Chklovskii, “Wiring optimization can relate neuronal structure and function,” *Proceedings of the National Academy of Sciences* **103**, 4723–4728 (2006), <https://www.pnas.org/content/103/12/4723.full.pdf>.
- [16] Manlio De Domenico, Mason A. Porter, and Alex Arenas, “MuxViz: a tool for multilayer analysis and visualization of networks,” *Journal of Complex Networks* **3**, 159–176 (2014), <http://oup.prod.sis.lan/comnet/article-pdf/3/2/159/1070864/cnu038.pdf>.
- [17] Chris Stark, Bobby-Joe Breitkreutz, Teresa Reguly, Lorrie Boucher, Ashton Breitkreutz, and Mike Tyers, “BioGRID: a general repository for interaction datasets,” *Nucleic Acids Research* **34**, D535–D539 (2006), http://oup.prod.sis.lan/nar/article-pdf/34/suppl_1/D535/3925435/gkj109.pdf.
- [18] Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora, “Structural reducibility of multilayer networks,” *Nature Communications* **6**, 6864 (2015).
- [19] “The IPv4 and IPv6 Topology Datasets,” http://www.caida.org/data/active/ipv4_routed_topology_aslinks_dataset.xml and https://www.caida.org/data/active/ipv6_allpref_topology_dataset.xml. (2015).
- [20] Arda Halu, Satyam Mukherjee, and Ginestra Bianconi, “Emergence of overlap in ensembles of spatial multiplexes and statistical mechanics of spatial interacting network ensembles,” *Phys. Rev. E* **89**, 012806 (2014).

Chapter Eight

Conclusions

In my Ph.D. thesis I considered percolation and some of its variants (bond percolation, site percolation, core percolation, k -core percolation, targeted attack and optimal percolation) as processes taking place on both monoplex and multiplex networks. The focus in all the cases was the interplay between the structure of a network and the dynamics which is running on top of it. The main results of the thesis can be summarized as follows:

- **Power laws in noncritical neuronal systems.** In this study bond percolation is used in order to model neuronal avalanches. We studies analytically and numerically distribution of avalanches in different network topologies. The main finding is that some specific network topologies can induce power law distributions of avalanches without the system being at critical point.
- **Generalization of core percolation.** We defined a new k -core decomposition for monoplex networks. We studied its phase transition for random synthetic networks with Poisson and scale free degree distributions using theoretical and numerical techniques. Finally, we presented G k -core decomposition of the networks as a new network layout that can have potential implications on singling out influential nodes.
- **Observability transition on multiplex networks.** We generalized observability phase transition of monoplex networks to multiplex networks. We developed the exact

solution of this model for synthetic locally tree-like multiplex networks. We also applied the model to real multiplex networks and observed that inter-layer correlations as major players in multiplex networks, make them observable.

- **Hyperbolic embedding and community structure.** We unveiled the analogy between hyperbolic embedding and community structure of complex networks. This analogy helps us to understand robustness of real multiplex networks in the language of community correlated structures.
- **Optimal percolation on multiplex networks.** We generalized optimal percolation (graph dismantling) to multiplex networks. We designed several heuristic algorithms to solve this problem approximately on both real and synthetic multiplex networks. Finally, we highlighted that inter-layer correlations such as edge overlap plays a big role in the robustness of multiplex networks.
- **k -core of real multiplex networks.** We observed that real multiplex networks non-trivially have a rich k -core structure. Using randomization techniques to decrease inter-layer correlations, we observed that the rich k -core of real multiplex networks originates due to inter-layer correlations. To quantify these correlations we used hyperbolic network framework. We concluded that in the case of networks with homogeneous degree distributions, angular correlations matter more. However, in the case of networks with broad degree distributions radial (degree) correlations play an important role.

Finally, I conclude this chapter by following remark. When considering a dynamic taking place on top of a monoplex or multiplex networks, either real or random, there are many factors affecting the outcome. Topology of the network, inter-layer correlations, parameters of the model dynamic which is running on top of either monoplex or multiplex networks, etc. are examples of these factors.

REFERENCES

- [1] Albert-László Barabási et al. *Network science*. Cambridge university press, 2016.
- [2] Mark Newman. *Networks*. Oxford university press, 2018.
- [3] S. Boccaletti et al. “Complex networks: Structure and dynamics”. In: *Physics Reports* 424.4 (2006), pp. 175–308. ISSN: 0370-1573.
- [4] Mark EJ Newman. “The structure and function of complex networks”. In: *SIAM review* 45.2 (2003), pp. 167–256.
- [5] Sergey N Dorogovtsev, Alexander V Goltsev, and José FF Mendes. “Critical phenomena in complex networks”. In: *Reviews of Modern Physics* 80.4 (2008), p. 1275.
- [6] Alain Barrat, Marc Barthelemy, and Alessandro Vespignani. *Dynamical processes on complex networks*. Cambridge university press, 2008.
- [7] Romualdo Pastor-Satorras and Alessandro Vespignani. “Epidemic spreading in scale-free networks”. In: *Physical review letters* 86.14 (2001), p. 3200.
- [8] Romualdo Pastor-Satorras et al. “Epidemic processes in complex networks”. In: *Reviews of modern physics* 87.3 (2015), p. 925.
- [9] Sergey V Buldyrev et al. “Catastrophic cascade of failures in interdependent networks”. In: *Nature* 464.7291 (2010), pp. 1025–1028.
- [10] Duncan S Callaway et al. “Network robustness and fragility: Percolation on random graphs”. In: *Physical review letters* 85.25 (2000), p. 5468.
- [11] Sergey N Dorogovtsev, Alexander V Goltsev, and Jose Ferreira F Mendes. “K-core organization of complex networks”. In: *Physical review letters* 96.4 (2006), p. 040601.
- [12] Dimitris Achlioptas, Raissa M D’Souza, and Joel Spencer. “Explosive percolation in random networks”. In: *Science* 323.5920 (2009), pp. 1453–1455.
- [13] GJ Baxter et al. “Avalanche collapse of interdependent networks”. In: *Physical review letters* 109.24 (2012), p. 248701.
- [14] Saeed Osat, Ali Faqeeh, and Filippo Radicchi. “Optimal percolation on multiplex networks”. In: *Nature communications* 8.1 (2017), pp. 1–7.

- [15] Sergei N Dorogovtsev and José FF Mendes. *Evolution of networks: From biological nets to the Internet and WWW*. OUP Oxford, 2013.
- [16] Santo Fortunato. “Community detection in graphs”. In: *Physics reports* 486.3-5 (2010), pp. 75–174.
- [17] Mason A Porter and James P Gleeson. “Dynamical systems on networks”. In: *Frontiers in Applied Dynamical Systems: Reviews and Tutorials* 4 (2016).
- [18] Jean-Jacques E Slotine, Weiping Li, et al. *Applied nonlinear control*. Vol. 199. 1. Prentice hall Englewood Cliffs, NJ, 1991.
- [19] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. “Controllability of complex networks”. In: *nature* 473.7346 (2011), pp. 167–173.
- [20] Yang-Yu Liu and Albert-László Barabási. “Control principles of complex systems”. In: *Reviews of Modern Physics* 88.3 (2016), p. 035006.
- [21] Ali Faqeeh et al. “Emergence of power laws in noncritical neuronal systems”. In: *Physical Review E* 100.1 (2019), p. 010401.
- [22] N. Azimi-Tafreshi, S. Osat, and S. N. Dorogovtsev. “Generalization of core percolation on complex networks”. In: *Phys. Rev. E* 99 (2 Feb. 2019), p. 022312.
- [23] Saeed Osat and Filippo Radicchi. “Observability transition in multiplex networks”. In: *Physica A: Statistical Mechanics and its Applications* 503 (2018), pp. 745–761. ISSN: 0378-4371.
- [24] Ali Faqeeh, Saeed Osat, and Filippo Radicchi. “Characterizing the analogy between hyperbolic embedding and community structure of complex networks”. In: *Physical review letters* 121.9 (2018), p. 098301.
- [25] Saeed Osat, Filippo Radicchi, and Fragkiskos Papadopoulos. “ k -core structure of real multiplex networks”. In: *Phys. Rev. Research* 2 (2 May 2020), p. 023176.
- [26] Stephen P Borgatti, Martin G Everett, and Jeffrey C Johnson. *Analyzing social networks*. Sage, 2018.
- [27] Ed Bullmore and Olaf Sporns. “Complex brain networks: graph theoretical analysis of structural and functional systems”. In: *Nature reviews neuroscience* 10.3 (2009), pp. 186–198.
- [28] Ginestra Bianconi. *Multilayer networks: structure and function*. Oxford university press, 2018.
- [29] Stefano Boccaletti et al. “The structure and dynamics of multilayer networks”. In: *Physics Reports* 544.1 (2014), pp. 1–122.

- [30] Béla Bollobás and Bollobás Béla. *Random graphs*. 73. Cambridge university press, 2001.
- [31] Paul Erdős and Alfréd Rényi. “On the evolution of random graphs”. In: *Publ. Math. Inst. Hung. Acad. Sci* 5.1 (1960), pp. 17–60.
- [32] Luis A Nunes Amaral et al. “Classes of small-world networks”. In: *Proceedings of the national academy of sciences* 97.21 (2000), pp. 11149–11152.
- [33] Albert-László Barabási and Réka Albert. “Emergence of scaling in random networks”. In: *science* 286.5439 (1999), pp. 509–512.
- [34] Mark EJ Newman. “Random graphs with clustering”. In: *Physical review letters* 103.5 (2009), p. 058701.
- [35] Michelle Girvan and Mark EJ Newman. “Community structure in social and biological networks”. In: *Proceedings of the national academy of sciences* 99.12 (2002), pp. 7821–7826.
- [36] Guido Caldarelli et al. “Scale-free networks from varying vertex intrinsic fitness”. In: *Physical review letters* 89.25 (2002), p. 258702.
- [37] Michael Molloy and Bruce Reed. “A critical point for random graphs with a given degree sequence”. In: *Random structures & algorithms* 6.2-3 (1995), pp. 161–180.
- [38] Dmitri Krioukov et al. “Hyperbolic geometry of complex networks”. In: *Physical Review E* 82.3 (2010), p. 036106.
- [39] Anna D Broido and Aaron Clauset. “Scale-free networks are rare”. In: *Nature communications* 10.1 (2019), pp. 1–10.
- [40] M Angeles Serrano, Dmitri Krioukov, and Marián Boguná. “Self-similarity of complex networks and hidden metric spaces”. In: *Physical review letters* 100.7 (2008), p. 078701.
- [41] Michael Szell, Renaud Lambiotte, and Stefan Thurner. “Multirelational organization of large-scale social networks in an online world”. In: *Proceedings of the National Academy of Sciences* 107.31 (2010), pp. 13636–13641.
- [42] Peter J Mucha et al. “Community structure in time-dependent, multiscale, and multiplex networks”. In: *science* 328.5980 (2010), pp. 876–878.
- [43] Alessio Cardillo et al. “Emergence of network features from multiplexity”. In: *Scientific reports* 3 (2013), p. 1344.
- [44] Manlio De Domenico et al. “Mathematical formulation of multilayer networks”. In: *Physical Review X* 3.4 (2013), p. 041022.

- [45] Renaud Lambiotte, Martin Rosvall, and Ingo Scholtes. “From networks to optimal higher-order models of complex systems”. In: *Nature physics* 15.4 (2019), pp. 313–320.
- [46] Kyu-Min Lee, Byungjoon Min, and Kwang-Il Goh. “Towards real-world complexity: an introduction to multiplex networks”. In: *The European Physical Journal B* 88.2 (2015), p. 48.
- [47] Federico Battiston, Vincenzo Nicosia, and Vito Latora. “Structural measures for multiplex networks”. In: *Physical Review E* 89.3 (2014), p. 032804.
- [48] Saulo DS Reis et al. “Avoiding catastrophic failure in correlated networks of networks”. In: *Nature Physics* 10.10 (2014), pp. 762–767.
- [49] Filippo Radicchi. “Percolation in real interdependent networks”. In: *Nature Physics* 11.7 (2015), pp. 597–602.
- [50] Kaj-Kolja Kleineberg et al. “Hidden geometric correlations in real multiplex networks”. In: *Nature Physics* 12.11 (2016), pp. 1076–1081.
- [51] Kaj-Kolja Kleineberg et al. “Geometric correlations mitigate the extreme vulnerability of multiplex networks against targeted attacks”. In: *Physical review letters* 118.21 (2017), p. 218301.
- [52] Mikko Kivelä et al. “Multilayer networks”. In: *Journal of complex networks* 2.3 (2014), pp. 203–271.
- [53] Manlio De Domenico et al. “Structural reducibility of multilayer networks”. In: *Nature communications* 6.1 (2015), pp. 1–9.
- [54] Manlio De Domenico et al. “The physics of spreading processes in multilayer networks”. In: *Nature Physics* 12.10 (2016), pp. 901–906.
- [55] Sergio Gomez et al. “Diffusion dynamics on multiplex networks”. In: *Physical review letters* 110.2 (2013), p. 028701.
- [56] Seung-Woo Son et al. “Percolation theory on interdependent networks based on epidemic spreading”. In: *EPL (Europhysics Letters)* 97.1 (Jan. 2012), p. 16006.
- [57] Manlio De Domenico et al. “Navigability of interconnected networks under random failures”. In: *Proceedings of the National Academy of Sciences* 111.23 (2014), pp. 8351–8356.
- [58] Clara Granell, Sergio Gómez, and Alex Arenas. “Dynamical interplay between awareness and epidemic spreading in multiplex networks”. In: *Physical review letters* 111.12 (2013), p. 128701.

- [59] Charo I del Genio et al. “Synchronization in networks with multiple interaction layers”. In: *Science advances* 2.11 (2016), e1601679.
- [60] Roni Parshani, Sergey V Buldyrev, and Shlomo Havlin. “Interdependent networks: Reducing the coupling strength leads to a change from a first to second order percolation transition”. In: *Physical review letters* 105.4 (2010), p. 048701.
- [61] Filippo Radicchi and Alex Arenas. “Abrupt transition in the structural formation of interconnected networks”. In: *Nature Physics* 9.11 (2013), pp. 717–720.
- [62] Filippo Radicchi and Ginestra Bianconi. “Redundant interdependencies boost the robustness of multiplex networks”. In: *Physical Review X* 7.1 (2017), p. 011013.
- [63] Byungjoon Min et al. “Network robustness of multiplex networks with interlayer degree correlations”. In: *Physical Review E* 89.4 (2014), p. 042811.
- [64] N Azimi-Tafreshi, Sergey N Dorogovtsev, and José FF Mendes. “Core organization of directed complex networks”. In: *Physical Review E* 87.3 (2013), p. 032815.
- [65] Manlio De Domenico et al. “Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems”. In: *Physical Review X* 5.1 (2015), p. 011027.
- [66] Pan Zhang and Christopher Moore. “Scalable detection of statistically significant communities and hierarchies, using message passing for modularity”. In: *Proceedings of the National Academy of Sciences* 111.51 (2014), pp. 18144–18149.
- [67] Fragkiskos Papadopoulos et al. “Popularity versus similarity in growing networks”. In: *Nature* 489.7417 (2012), pp. 537–540.
- [68] Marian Boguna, Dmitri Krioukov, and Kimberly C Claffy. “Navigability of complex networks”. In: *Nature Physics* 5.1 (2009), pp. 74–80.
- [69] Guillermo García-Pérez, M Angeles Serrano, and Marián Boguñá. “Soft communities in similarity space”. In: *Journal of Statistical Physics* 173.3-4 (2018), pp. 775–782.
- [70] Guillermo García-Pérez, Marián Boguñá, and M Ángeles Serrano. “Multiscale unfolding of real networks by geometric renormalization”. In: *Nature Physics* 14.6 (2018), pp. 583–589.
- [71] Lev Davidovich Landau and Evgenii Mikhailovich Lifshitz. *Course of theoretical physics*. Elsevier, 2013.
- [72] Julia M Yeomans. *Statistical mechanics of phase transitions*. Clarendon Press, 1992.
- [73] J M Kosterlitz and D J Thouless. “Ordering, metastability and phase transitions in two-dimensional systems”. In: *Journal of Physics C: Solid State Physics* 6.7 (Apr. 1973), pp. 1181–1203.

- [74] Herbert S Wilf. *generatingfunctionology*. CRC press, 2005.
- [75] Dietrich Stauffer and Amnon Aharony. *Introduction to percolation theory*. CRC press, 2018.
- [76] Abbas Ali Saberi. “Recent advances in percolation theory and its applications”. In: *Physics Reports* 578 (2015). Recent advances in percolation theory and its applications, pp. 1–32. ISSN: 0370-1573.
- [77] Muhammad Sahimi. *Applications of percolation theory*. CRC Press, 1994.
- [78] Scott Kirkpatrick. “Percolation and Conduction”. In: *Rev. Mod. Phys.* 45 (4 Oct. 1973), pp. 574–588.
- [79] M. B. Isichenko. “Percolation, statistical topography, and transport in random media”. In: *Rev. Mod. Phys.* 64 (4 Oct. 1992), pp. 961–1043.
- [80] J. Machta. “Phase transitions in fractal porous media”. In: *Phys. Rev. Lett.* 66 (2 Jan. 1991), pp. 169–172.
- [81] Paul J. Flory. “Molecular Size Distribution in Three Dimensional Polymers. I. Gelation1”. In: *Journal of the American Chemical Society* 63.11 (Nov. 1941), pp. 3083–3090. ISSN: 0002-7863.
- [82] Walter H. Stockmayer. “Theory of Molecular Size Distribution and Gel Formation in Branched-Chain Polymers”. In: *The Journal of Chemical Physics* 11.2 (1943), pp. 45–55.
- [83] M. E. J. Newman. “Spread of epidemic disease on networks”. In: *Phys. Rev. E* 66 (1 July 2002), p. 016128.
- [84] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. “Random graphs with arbitrary degree distributions and their applications”. In: *Phys. Rev. E* 64 (2 July 2001), p. 026118.
- [85] Filippo Radicchi. “Predicting percolation thresholds in networks”. In: *Physical Review E* 91.1 (2015), p. 010801.
- [86] Dante R Chialvo. “Emergent complex neural dynamics”. In: *Nature physics* 6.10 (2010), pp. 744–750.
- [87] Miguel A Munoz. “Colloquium: Criticality and dynamical scaling in living systems”. In: *Reviews of Modern Physics* 90.3 (2018), p. 031001.
- [88] John M Beggs and Nicholas Timme. “Being critical of criticality in the brain”. In: *Frontiers in physiology* 3 (2012), p. 163.
- [89] John M Beggs and Dietmar Plenz. “Neuronal avalanches in neocortical circuits”. In: *Journal of neuroscience* 23.35 (2003), pp. 11167–11177.

- [90] Nir Friedman et al. “Universal critical dynamics in high resolution neuronal avalanche data”. In: *Physical review letters* 108.20 (2012), p. 208102.
- [91] Dante R Chialvo. “Critical brain networks”. In: *Physica A: Statistical Mechanics and its Applications* 340.4 (2004), pp. 756–765.
- [92] John M Beggs. “The criticality hypothesis: how local cortical networks might optimize information processing”. In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 366.1864 (2008), pp. 329–343.
- [93] Osame Kinouchi and Mauro Copelli. “Optimal dynamical range of excitable networks at criticality”. In: *Nature physics* 2.5 (2006), pp. 348–351.
- [94] Clayton Haldeman and John M Beggs. “Critical branching captures activity in living neural networks and maximizes the number of metastable states”. In: *Physical review letters* 94.5 (2005), p. 058101.
- [95] James P Sethna, Karin A Dahmen, and Christopher R Myers. “Crackling noise”. In: *Nature* 410.6825 (2001), pp. 242–250.
- [96] James P Gleeson and Rick Durrett. “Temporal profiles of avalanches on networks”. In: *Nature communications* 8.1 (2017), pp. 1–13.
- [97] David Kempe, Jon Kleinberg, and Éva Tardos. “Maximizing the spread of influence through a social network”. In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2003, pp. 137–146.
- [98] Jacob Goldenberg, Barak Libai, and Eitan Muller. “Talk of the network: A complex systems look at the underlying process of word-of-mouth”. In: *Marketing letters* 12.3 (2001), pp. 211–223.
- [99] Stephen B Seidman. “Network structure and minimum degree”. In: *Social networks* 5.3 (1983), pp. 269–287.
- [100] J Ignacio Alvarez-Hamelin et al. “Large scale networks fingerprinting and visualization using the k-core decomposition”. In: *Advances in neural information processing systems*. 2006, pp. 41–50.
- [101] Maksim Kitsak et al. “Identification of influential spreaders in complex networks”. In: *Nature physics* 6.11 (2010), pp. 888–893.
- [102] Flaviano Morone, Gino Del Ferraro, and Hernán A Makse. “The k-core as a predictor of structural collapse in mutualistic ecosystems”. In: *Nature physics* 15.1 (2019), pp. 95–102.
- [103] Claudio Castellano and Romualdo Pastor-Satorras. “Relating topological determinants of complex networks to their spectral properties: Structural and dynamical effects”. In: *Physical Review X* 7.4 (2017), p. 041024.

- [104] Richard M Karp and Michael Sipser. “Maximum matching in sparse random graphs”. In: *22nd Annual Symposium on Foundations of Computer Science (sfcs 1981)*. IEEE. 1981, pp. 364–375.
- [105] Yang-Yu Liu et al. “Core percolation on complex networks”. In: *Physical review letters* 109.20 (2012), p. 205703.
- [106] Lenka Zdeborová and Marc Mézard. “The number of matchings in random graphs”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2006.05 (2006), P05003.
- [107] Martin Weigt and Alexander K Hartmann. “Number of guards needed by a museum: A phase transition in vertex covering of random graphs”. In: *Physical review letters* 84.26 (2000), p. 6118.
- [108] Tao Jia and Márton Pósfai. “Connecting core percolation and controllability of complex networks”. In: *Scientific reports* 4 (2014), p. 5379.
- [109] Yang Yang, Jianhui Wang, and Adilson E. Motter. “Network Observability Transitions”. In: *Phys. Rev. Lett.* 109 (25 Dec. 2012), p. 258701.
- [110] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. “Observability of complex systems”. In: *Proceedings of the National Academy of Sciences* 110.7 (2013), pp. 2460–2465.
- [111] Yang Yang and Filippo Radicchi. “Observability transition in real networks”. In: *Physical Review E* 94.3 (2016), p. 030301.
- [112] Ginestra Bianconi and Filippo Radicchi. “Percolation in real multiplex networks”. In: *Physical Review E* 94.6 (2016), p. 060301.
- [113] Reuven Cohen et al. “Breakdown of the Internet under Intentional Attack”. In: *Phys. Rev. Lett.* 86 (16 Apr. 2001), pp. 3682–3685.
- [114] Flaviano Morone and Hernán A Makse. “Influence maximization in complex networks through optimal percolation”. In: *Nature* 524.7563 (2015), pp. 65–68.
- [115] Reuven Cohen et al. “Resilience of the Internet to Random Breakdowns”. In: *Phys. Rev. Lett.* 85 (21 Nov. 2000), pp. 4626–4628.
- [116] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. “Benchmark graphs for testing community detection algorithms”. In: *Phys. Rev. E* 78 (4 Oct. 2008), p. 046110.
- [117] Alfredo Braunstein et al. “Network dismantling”. In: *Proceedings of the National Academy of Sciences* 113.44 (2016), pp. 12368–12373. ISSN: 0027-8424.
- [118] Xiao-Long Ren et al. “Generalized network dismantling”. In: *Proceedings of the National Academy of Sciences* 116.14 (2019), pp. 6554–6559. ISSN: 0027-8424.

- [119] Reuven Cohen, Shlomo Havlin, and Daniel ben-Avraham. “Efficient Immunization Strategies for Computer Networks and Populations”. In: *Phys. Rev. Lett.* 91 (24 Dec. 2003), p. 247901.
- [120] Pau Clusella et al. “Immunization and Targeted Destruction of Networks using Explosive Percolation”. In: *Phys. Rev. Lett.* 117 (20 Nov. 2016), p. 208301.
- [121] Lenka Zdeborová, Pan Zhang, and Hai-Jun Zhou. “Fast and simple decycling and dismantling of networks”. In: *Scientific reports* 6 (2016), p. 37954.
- [122] Salomon Mugisha and Hai-Jun Zhou. “Identifying optimal targets of network attack by belief propagation”. In: *Phys. Rev. E* 94 (1 July 2016), p. 012305.
- [123] G. J. Baxter, G. Timár, and J. F. F. Mendes. “Targeted damage to interdependent networks”. In: *Phys. Rev. E* 98 (3 Sept. 2018), p. 032307.
- [124] N. Azimi-Tafreshi, J. Gómez-Gardeñes, and S. N. Dorogovtsev. “ k –core percolation on multiplex networks”. In: *Phys. Rev. E* 90 (3 Sept. 2014), p. 032816.