

Mathematical Formulation of Multilayer Networks

Manlio De Domenico,¹ Albert Solé-Ribalta,¹ Emanuele Cozzo,² Mikko Kivelä,³ Yamir Moreno,^{2,4,5}
Mason A. Porter,⁶ Sergio Gómez,¹ and Alex Arenas¹

¹*Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, 43007 Tarragona, Spain*

²*Institute for Biocomputation and Physics of Complex Systems (BIFI), University of Zaragoza, Zaragoza 50018, Spain*

³*Oxford Centre for Industrial and Applied Mathematics, Mathematical Institute,
University of Oxford, Oxford OX1 3LB, United Kingdom*

⁴*Department of Theoretical Physics, University of Zaragoza, Zaragoza 50009, Spain*

⁵*Complex Networks and Systems Lagrange Lab, Institute for Scientific Interchange, Turin 10126, Italy*

⁶*Oxford Centre for Industrial and Applied Mathematics, Mathematical Institute and CABDyN Complexity Centre,
University of Oxford, Oxford OX1 3LB, United Kingdom*

(Received 23 July 2013; published 4 December 2013)

A network representation is useful for describing the structure of a large variety of complex systems. However, most real and engineered systems have multiple subsystems and layers of connectivity, and the data produced by such systems are very rich. Achieving a deep understanding of such systems necessitates generalizing “traditional” network theory, and the newfound deluge of data now makes it possible to test increasingly general frameworks for the study of networks. In particular, although adjacency matrices are useful to describe traditional single-layer networks, such a representation is insufficient for the analysis and description of multiplex and time-dependent networks. One must therefore develop a more general mathematical framework to cope with the challenges posed by multilayer complex systems. In this paper, we introduce a tensorial framework to study multilayer networks, and we discuss the generalization of several important network descriptors and dynamical processes—including degree centrality, clustering coefficients, eigenvector centrality, modularity, von Neumann entropy, and diffusion—for this framework. We examine the impact of different choices in constructing these generalizations, and we illustrate how to obtain known results for the special cases of single-layer and multiplex networks. Our tensorial approach will be helpful for tackling pressing problems in multilayer complex systems, such as inferring who is influencing whom (and by which media) in multichannel social networks and developing routing techniques for multimodal transportation systems.

DOI: [10.1103/PhysRevX.3.041022](https://doi.org/10.1103/PhysRevX.3.041022)

Subject Areas: Interdisciplinary Physics

I. INTRODUCTION

The quantitative study of networks is fundamental for the study of complex systems throughout the biological, social, information, engineering, and physical sciences [1–3]. The broad applicability of networks and their success in providing insights into the structure and function of both natural and designed systems have thus generated considerable excitement across myriad scientific disciplines. For example, networks have been used to represent interactions between proteins, friendships between people, hyperlinks between Web pages, and much more. Importantly, several features arise in a diverse variety of networks. For example, many networks constructed from empirical data exhibit heavy-tailed degree distributions, the small-world property, and/or modular structures; such structural features can have important implications for information diffusion, robustness against component failure, and many other considerations [1–3].

Traditional studies of networks generally assume that nodes are connected to each other by a single type of static edge that encapsulates all connections between them. This assumption is almost always a gross oversimplification, and it can lead to misleading results and even the sheer inability to address certain problems. For example, ignoring time dependence throws away the ordering of pairwise human contacts in transmission of diseases [4], and ignoring the presence of multiple types of edges (which is known as “multiplexity” [5]) makes it hard to take into account the simultaneous presence and relevance of multiple modes of transportation or communication [6].

Multiplex networks explicitly incorporate multiple channels of connectivity in a system, and they provide a natural description for systems in which entities have a different set of neighbors in each layer (which can represent, e.g., a task, an activity, or a category). A fundamental aspect of describing multiplex networks is defining and quantifying the interconnectivity between different categories of connections. Examining such interconnectivity is necessary for examining switching between layers in a multilayer system, and the associated interlayer connections in a network are responsible for the emergence of new

Published by the American Physical Society under the terms of the [Creative Commons Attribution 3.0 License](https://creativecommons.org/licenses/by/3.0/). Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

phenomena in multiplex networks. Interlayer connections can generate new structural and dynamical correlations between components of a system, so it is important to develop a framework that takes them into account. Note that multiplex networks are not simply a special case of interdependent networks [7]: In multiplex systems, many or even all of the nodes have a counterpart in each layer, so one can associate a vector of states with each node. For example, a person might currently be logged into Facebook (and hence able to receive information there) but not logged into some other social-networking site. The presence of nodes in multiple layers of a system also entails the possibility of self-interactions. This feature has no counterpart in interdependent networks, which were conceived as interconnected communities within a single, larger network [8,9].

The scientific community has been developing tools for temporal networks for several years [4,10], although much more work remains to be done, and now an increasingly large number of scholars with diverse expertise have turned their attention to studying multiplex networks (and related constructs, such as the aforementioned interdependent networks and so-called “networks of networks”) [7,8,11–33]. Moreover, despite this wealth of recent attention, we note that multiplexity was already highlighted decades ago in fields such as engineering [34,35] and sociology [5,36,37].

To study multiplex and/or temporal networks systematically, it is necessary to develop a precise mathematical representation for them as well as appropriate tools to go with such a representation. In this paper, we develop a mathematical framework for multilayer networks using tensor algebra. Our framework can be used to study all types of multilayer networks (including multiplex networks, temporal networks, cognitive social structures [38], multivariate networks [39], interdependent networks, etc.). To simplify exposition, we will predominantly use the language of multiplex networks in this paper, and we will thus pay particular attention to this case.

There are numerous network diagnostics for which it is desirable to develop multilayer generalizations. In particular, we consider degree centrality, eigenvector centrality, clustering coefficients, modularity, von Neumann entropy, and random walks. Some of these notions have been discussed previously in the context of multiplex networks [13,21,22,24,27,28]. In this paper, we define these notions for adjacency-tensor representations of multilayer networks. Our generalizations of these quantities are natural, and they make it possible to compare these multilayer quantities with their single-layer counterparts in a systematic manner. Such generalizations are particularly important for the examination of new phenomena, such as multiplexity-induced correlations [19] and new dynamical feedbacks [26], which arise when generalizing the usual single-layer networks. In Fig. 1, we show a schematic of multilayer networks. In the left panel, we highlight the

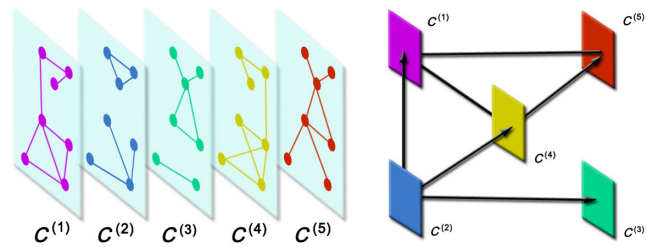


FIG. 1. Schematic of a multilayer network. When studying ordinary networks, one represents the interaction of entities (i.e., nodes) using an adjacency matrix, which encodes the intensity of each pairwise interrelationship. However, as shown in the left panel, entities can interact in different ways (e.g., depending on the environment in which they are embedded). In this panel, each layer of a multilayer network corresponds to a different type of interaction (e.g., social relationships, business collaborations, etc.) and is represented by a different adjacency matrix. As shown in the right panel, the layers can also be interconnected. One thereby obtains a new type of relationship, which complicates the description of networked systems.

different interactions (edges) between entities (nodes) in different layers; in the right panel, we highlight the connections between different layers.

The remainder of this paper is organized as follows. In Sec. II, we represent single-layer (i.e., “monoplex”) networks using a tensorial framework. We extend this framework to multilayer networks in Sec. III, and we discuss several descriptors and diagnostics for both single-layer and multilayer networks in Sec. IV. We conclude in Sec. V.

II. SINGLE-LAYER (MONOPLEX) NETWORKS

Given a set of N objects n_i (where $i = 1, 2, \dots, N$ and $N \in \mathbb{N}$), we associate with each object a state that is represented by a canonical vector in the vector space \mathbb{R}^N . More specifically, let $\mathbf{e}_i \equiv (0, \dots, 0, 1, 0, \dots, 0)^\dagger$, where \dagger is the transposition operator, be the column vector that corresponds to the object n_i (which we call a node). The i th component of \mathbf{e}_i is 1, and all of its other components are 0.

One can relate the objects n_i with each other, and our goal is to find a simple way to indicate the presence and the intensity of such relationships. The most natural choice of vector space for describing the relationship is created using the tensor product (i.e., the Kronecker product) $\mathbb{R}^N \otimes \mathbb{R}^N = \mathbb{R}^{N \times N}$ [40]. Thus, second-order (i.e., rank-2) canonical tensors are defined by $\mathbf{E}_{ij} = \mathbf{e}_i \otimes \mathbf{e}_j^\dagger$ (where $i, j = 1, 2, \dots, N$). Consequently, if w_{ij} indicates the intensity of the relationship from object n_i to object n_j , we can write the relationship tensor as

$$\mathbf{W} = \sum_{i,j=1}^N w_{ij} \mathbf{E}_{ij} = \sum_{i,j=1}^N w_{ij} \mathbf{e}_i \otimes \mathbf{e}_j^\dagger, \quad \mathbf{W} \in \mathbb{R}^N \otimes \mathbb{R}^N. \quad (1)$$

Importantly, the relationships that we have just described can be directed. That is, the intensity of the relationship

from object n_i to object n_j need not be the same as the intensity of the relationship from object n_j to object n_i .

In the context of networks, \mathbf{W} corresponds to an $N \times N$ weight matrix that represents the standard graph of a system that consists of N nodes n_i . This matrix is thus an example of an adjacency tensor, which is the language that we will use in the rest of this paper. To distinguish such simple networks from the more complicated situations (e.g., multiplex networks) that we discuss in this paper, we will use the term monoplex networks to describe such standard networks, which are time independent and possess only a single type of edge that connects their nodes.

Tensors provide a convenient mathematical representation for generalizing ordinary static networks, as they provide a natural way to encapsulate complicated sets of relationships that can also change in time [13,41]. Matrices are rank-2 tensors, so they are inherently limited in the complexity of the relationships that they can capture. One can represent increasingly complicated types of relationships between nodes by considering tensors of higher order. An adjacency tensor can be written using a more compact notation that will be useful for the generalization to multilayer networks that we will discuss later. We will use the covariant notation introduced by Ricci and Levi-Civita in Ref. [42]. In this notation, a row vector $\mathbf{a} \in \mathbb{R}^N$ is given by a covariant vector a_α (where $\alpha = 1, 2, \dots, N$), and the corresponding contravariant vector a^α (i.e., its dual vector) is a column vector in Euclidean space.

To avoid confusion, we will use latin letters i, j, \dots to indicate, for example, the i th vector, the (ij) th tensor, etc., and we will use greek letters α, β, \dots to indicate the components of a vector or a tensor. With this terminology, $e^\alpha(i)$ is the α th component of the i th contravariant canonical vector \mathbf{e}_i in \mathbb{R}^N , and $e_\alpha(j)$ is the α th component of the j th covariant canonical vector in \mathbb{R}^N .

With these conventions, the adjacency tensor \mathbf{W} can be represented as a linear combination of tensors in the canonical basis:

$$W_\beta^\alpha = \sum_{i,j=1}^N w_{ij} e^\alpha(i) e_\beta(j) = \sum_{i,j=1}^N w_{ij} E_\beta^\alpha(ij), \quad (2)$$

where $E_\beta^\alpha(ij) \in \mathbb{R}^{N \times N}$ indicates the tensor in the canonical basis that corresponds to the tensor product of the canonical vectors assigned to nodes n_i and n_j (i.e., it is \mathbf{E}_{ij}).

The adjacency tensor W_β^α is of mixed type: it is 1-covariant and 1-contravariant. This choice provides an elegant formulation for the subsequent definitions.

III. MULTILAYER NETWORKS

In the previous section, we described a procedure to build an adjacency tensor for a monoplex (i.e., single-layer) network. In general, however, there might be several types of relationships between pairs of nodes n_1, n_2, \dots, n_N , and an adjacency tensor can be used to

represent this situation. In other words, one can think of a more general system represented as a multilayer object in which each type of relationship is encompassed in a single layer \tilde{k} (where $\tilde{k} = 1, 2, \dots, L$) of a system.

We use the term intralayer adjacency tensor for the second-order tensor $W_\beta^\alpha(\tilde{k})$ that indicates the relationships between nodes within the same layer \tilde{k} . The tilde symbol allows us to distinguish indices that correspond to nodes from those that correspond to layers.

We take into account the possibility that a node n_i from layer \tilde{h} can be connected to any other node n_j in any other layer \tilde{k} . To encode information about relationships that incorporate multiple layers, we introduce the second-order interlayer adjacency tensor $C_\beta^\alpha(\tilde{h} \tilde{k})$. Note that $C_\beta^\alpha(\tilde{k} \tilde{k}) = W_\beta^\alpha(\tilde{k})$, so the interlayer adjacency tensor that corresponds to the case in which a pair of layers represents the same layer \tilde{k} is equivalent to the intralayer adjacency tensor of such a layer.

Following an approach similar to that in Sec. II, we introduce the vectors $e^{\tilde{\gamma}}(\tilde{k})$ (where $\tilde{\gamma} = 1, 2, \dots, L$ and $\tilde{k} = 1, 2, \dots, L$) of the canonical basis in the space \mathbb{R}^L , where the greek index indicates the components of the vector and the latin index indicates the k th canonical vector. The tilde symbol on the greek indices allows us to distinguish these indices from the greek indices that correspond to nodes. It is straightforward to construct the second-order tensors $E_\delta^{\tilde{\gamma}}(\tilde{h} \tilde{k}) = e^{\tilde{\gamma}}(\tilde{h}) e_\delta(\tilde{k})$ that represent the canonical basis of the space $\mathbb{R}^{L \times L}$. (We use analogous notation for canonical bases throughout this paper.)

We can write the multilayer adjacency tensor discussed early in this section using a tensor product between the adjacency tensors $C_\beta^\alpha(\tilde{h} \tilde{k})$ and the canonical tensors $E_\delta^{\tilde{\gamma}}(\tilde{h} \tilde{k})$. We obtain

$$\begin{aligned} M_{\beta\delta}^{\alpha\tilde{\gamma}} &= \sum_{\tilde{h}, \tilde{k}=1}^L C_\beta^\alpha(\tilde{h} \tilde{k}) E_\delta^{\tilde{\gamma}}(\tilde{h} \tilde{k}) \\ &= \sum_{\tilde{h}, \tilde{k}=1}^L \left[\sum_{i,j=1}^N w_{ij}(\tilde{h} \tilde{k}) E_\beta^\alpha(ij) \right] E_\delta^{\tilde{\gamma}}(\tilde{h} \tilde{k}) \\ &= \sum_{\tilde{h}, \tilde{k}=1}^L \sum_{i,j=1}^N w_{ij}(\tilde{h} \tilde{k}) \mathcal{E}_{\beta\delta}^{\alpha\tilde{\gamma}}(ij \tilde{h} \tilde{k}), \end{aligned} \quad (3)$$

where $w_{ij}(\tilde{h} \tilde{k})$ are real numbers that indicate the intensity of the relationship (which may not be symmetric) between nodes n_i in layer \tilde{h} and node n_j in layer \tilde{k} , and $\mathcal{E}_{\beta\delta}^{\alpha\tilde{\gamma}}(ij \tilde{h} \tilde{k}) \equiv e^\alpha(i) e_\beta(j) e^{\tilde{\gamma}}(\tilde{h}) e_\delta(\tilde{k})$ indicates the fourth-order (i.e., rank-4) tensors of the canonical basis in the space $\mathbb{R}^{N \times N \times L \times L}$.

The multilayer adjacency tensor $M_{\beta\delta}^{\alpha\tilde{\gamma}}$ is a very general object that can be used to represent a wealth of complicated relationships among nodes. In this paper, we focus on multiplex networks. A multiplex network is a special

type of multilayer network in which the only possible types of interlayer connections are ones in which a given node is connected to its counterpart nodes in the other layers. In many studies of multiplex networks, it is assumed (at least implicitly) that interlayer connections exist between counterpart nodes in all pairs of layers. However, (i) this assumption need not hold, and (ii) it departs from traditional notions of multiplexity [5], which focus on the existence of multiple types of connections and do not preclude entities from possessing only a subset of the available categories of connections. We thus advocate a somewhat more general (and more traditional) definition of multiplex networks.

When describing a multiplex network, the associated interlayer adjacency tensor is diagonal. Importantly, connections between a node and its counterparts can have different weights for different pairs of layers, and interlayer connections can also be different for different entities in a network [43]. For instance, such considerations are important for transportation networks, where one can relate the weight of interlayer connections to the cost of switching between a pair of transportation modes (i.e., layers). For example, at a given station (i.e., node) in a transportation network, it takes time to walk from a train platform to a bus, and it is crucial for transportation companies to measure how long it takes to change transportation modes [44]. See Fig. 2 for schematics of multiplex networks.

As we discussed above, entities in many systems have connections in some layers but not in others. For example, a user of online social networks might have a Facebook account but not use Twitter. Such an individual can thus broadcast and receive information only on a subset of the layers in the multiplex-network representation of the system. In a transportation network, if a station does not exist in a given layer of a multilayer network, then its associated edges also do not exist. The algebra in this paper holds for these situations without any formal modification (one simply assigns the value 0 to associated edges), but one must think carefully about the interpretation of calculations of network diagnostics.

If there is only a single layer, there is no distinction between a monoplex network and a single-layer network, so we can use these terms interchangeably. However, the difference is crucial when studying multilayer networks. Importantly—because it is convenient, for instance, for the implementation of computational algorithms—one can represent the multilayer adjacency tensor $M_{\beta\delta}^{\alpha\gamma}$ as a special rank-2 tensor that one obtains by a process called matricization (which is also known as unfolding and flattening) [45]. The elements of $M_{\beta\delta}^{\alpha\gamma}$, which is defined in the space $\mathbb{R}^{N \times N \times L \times L}$, can be represented as an $N^2 \times L^2$ or an $NL \times NL$ matrix. Flattening a multilayer adjacency tensor can be very helpful. Recent studies on community detection [13,46], diffusion [21], random walks [22], social contagions [27,28], and clustering coefficients [47] on multilayer networks have all used matrix representations of multilayer networks for computational (and occasionally analytical) purposes. For many years, the computational-science community has stressed the importance of developing tools for both tensors and their associated unfoldings (see the review article [45] and numerous references therein) and of examining problems from these complementary perspectives. We hope that this paper will help foster similar achievements in network science.

Another important special case of multilayer adjacency tensors is time-dependent (i.e., “temporal”) networks. Multilayer representations of temporal networks have thus far tended to include only connections between a given node in one layer and its corresponding nodes in its one or two neighboring layers. For example, the numerical calculations in Refs. [13,41] only use these “ordinal” interlayer couplings, which causes the off-diagonal blocks of a flattened adjacency tensor to have nonzero entries only along their diagonals, even though the theoretical formulation in those papers allows more general couplings between layers. Indeed, this restriction does not apply, in general, to temporal networks, as it is important for some applications to consider more general types of interlayer couplings (e.g., if one is considering causal

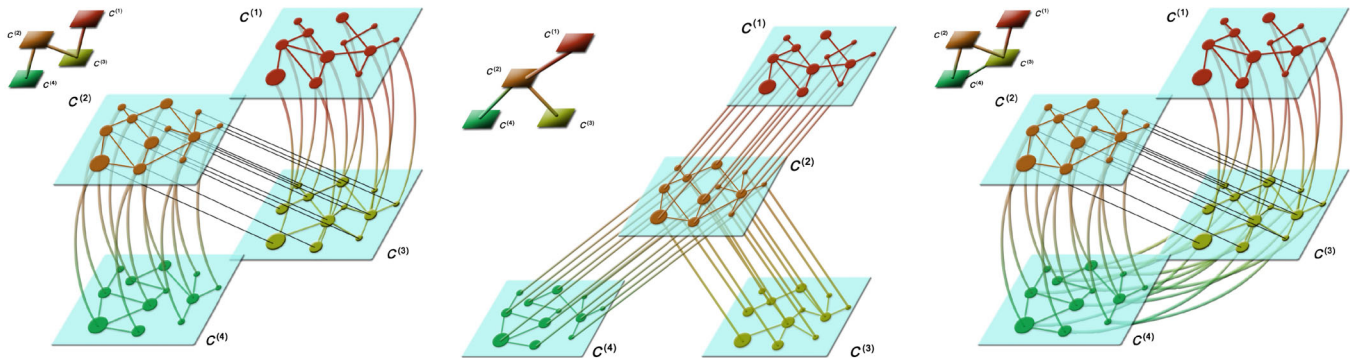


FIG. 2. Schematic of multilayer networks for three different topologies. We show three four-layer multiplex networks (and the corresponding network of layers as an inset in the top-left corners) and recall that each interlayer edge connects a node with one of its counterparts in another layer.

relationships between different nodes or if one wants to consider interlayer coupling over a longer time horizon). In temporal networks that are constructed from coupled time series, the individual layers in a multilayer adjacency tensor tend to be almost completely connected (although the intralayer edges, of course, have different weights). In other cases, such as most of the temporal networks discussed in Ref. [4], there might only be a small number of nonzero connections (which represent, e.g., a small number of phone calls in a particular time window) within a single layer.

IV. NETWORK DESCRIPTORS AND DYNAMICAL PROCESSES

In this section, we examine how to generalize some of the most common network descriptors and dynamical processes for multilayer networks. First, we use our tensorial construction to show that the properties of a multilayer network when it is made up of only a single layer reduce to the corresponding ones for a monoplex network. We obtain these properties using algebraic operations involving the adjacency tensor, canonical vectors, and canonical tensors. We then generalize these results for more general multilayer adjacency tensors.

A. Monoplex networks

Degree centrality.—Consider an undirected and unweighted network, which can be represented using the (symmetric) adjacency tensor W_β^α . Define the 1-vector $u^\alpha = (1, \dots, 1)^\dagger \in \mathbb{R}^N$ whose components are all equal to 1, and let $U_\alpha^\beta = u_\alpha u^\beta$ be the second-order tensor whose elements are all equal to 1. (Throughout this paper, we use analogous notation for all so-called “1-tensors,” whose entries are all equal to 1.) We adopt Einstein summation notation (see Appendix A) and interpret the adjacency tensor as an operator to be applied to the 1-vector.

We thereby calculate the degree centrality vector (or degree vector) $k_\beta = W_\beta^\alpha u_\alpha$ in the space \mathbb{R}^N . It is then straightforward to calculate the degree centrality of node n_i by projecting the degree vector onto the i th canonical vector: $k(i) = k_\beta e^\beta(i)$. Analogously, for an undirected and weighted network, we use the corresponding weighted adjacency tensor W_β^α to define the strength centrality vector (or strength vector) s_β , which can be used to calculate the strength (i.e., weighted degree) [48,49] of each node. With our notation, the mean degree is $\langle k \rangle = (U_\rho^\rho)^{-1} k_\beta u^\beta$, the second moment of the degree is $\langle k^2 \rangle = (U_\rho^\rho)^{-1} k_\beta k^\beta$, and the variance of the degree is $\text{var}(k) = (U_\rho^\rho)^{-1} k_\beta k^\beta - (U_\rho^\rho)^{-2} k_\alpha k_\beta U^{\alpha\beta}$.

Directed networks are also very important, and they illustrate why it is advantageous to introduce contravariant notation. Importantly, in-degree centrality and out-degree centrality are represented using different tensor products.

The in-degree centrality vector is $k_\beta = W_\beta^\alpha u_\alpha$, whereas the out-degree centrality vector is $k^\alpha = W_\beta^\alpha u^\beta$. We then recover the usual definitions for directed networks. For example, the in-degree centrality of node n_i is $k_{\text{in}}(i) = W_\beta^\alpha u_\alpha e^\beta(i)$. In directed and weighted networks, the analogous definition yields the in-strength centrality vector and the out-strength centrality vector. These calculations with directed networks are simple, but they illustrate that the proposed tensor algebra makes it possible to develop a deeper understanding of networks, as the tensor indices are related directly to the directionality of relationships between nodes in a network.

Clustering coefficients.—Clustering coefficients are useful measures of transitivity in a network [50]. For unweighted and undirected networks, the local clustering coefficient of a node n_i is defined as the number of existing edges among the set of its neighboring nodes divided by the total number of possible connections between them [51]. Several different definitions for local clustering coefficients have been developed for weighted and undirected networks [49,52–54] and for directed networks [55]. Given a local clustering coefficient, one can calculate a different global clustering coefficient by averaging over all nodes. Alternatively, one can calculate a global clustering coefficient as the total number of closed triples of nodes (where all three edges are present) divided by the number of connected triples [2].

One can obtain equivalent definitions of clustering coefficients in terms of walks on a network. In standard network theory, suppose that one has an adjacency matrix \mathbf{A} and a positive integer m . Then, each matrix element $(\mathbf{A}^m)_{ij}$ gives the number of distinct walks of length m that start from node n_i and end at node n_j . Therefore, taking $j = i$ and $m = 3$ yields the number of walks of length three that start and end at node n_i . In an unweighted network without self-loops, we thereby obtain the number of distinct three-cycles $t(i)$ that start from node n_i . One then calculates the local clustering coefficient of node n_i by dividing $t(i)$ by the number of three-cycles that would exist if the neighborhood of n_i were completely connected. For example, in an undirected network, one divides $t(i)$ by $k(i)[k(i) - 1]$, which is the number of ways to select two of the neighbors of n_i . In our notation, the value of $(\mathbf{A}^m)_{ij}$ is

$$t(i, j) = W_{\xi_1}^\alpha W_{\xi_2}^{\xi_1} W_{\xi_3}^{\xi_2} \dots W_{\xi_{m-1}}^{\xi_{m-2}} W_{\xi_m}^{\xi_{m-1}} e_\alpha(i) e^\beta(j), \quad (4)$$

which reduces to $t(i) = W_\rho^\alpha W_\sigma^\rho W_\beta^\sigma e_\alpha(i) e^\beta(i)$ for $j = i$ and $m = 3$. One can then define the local clustering coefficient by dividing the number of three-cycles by the number of three-cycles in a network for which the neighborhood of the node n_i is completely connected. We thereby obtain the formula

$$c(W_\beta^\alpha, i) = \frac{W_\rho^\alpha W_\sigma^\rho W_\beta^\sigma e_\alpha(i) e^\beta(i)}{W_\rho^\alpha F_\sigma^\rho W_\beta^\sigma e_\alpha(i) e^\beta(i)}, \quad (5)$$

where

$$F_\sigma^\rho = U_\sigma^\rho - \delta_\sigma^\rho$$

is the adjacency tensor corresponding to a network that includes all edges except for self-loops.

To use the above formulation to calculate a global clustering coefficient of a network, we need to calculate both the total number of three-cycles and the total number of three-cycles that one obtains when the second step of the walk occurs in a complete network. A compact way to express this global clustering coefficient is

$$c(W_\beta^\alpha) = \frac{W_\rho^\alpha W_\sigma^\rho W_\alpha^\sigma}{W_\rho^\alpha F_\sigma^\rho W_\alpha^\sigma}. \quad (6)$$

One can define a clustering coefficient in a weighted network without any changes to Eqs. (5) and (6) by assuming that W_β^α corresponds to the weighted adjacency tensor normalized such that each element of the tensor lies in the interval $[0, 1]$. If weights are not defined within this range, then Eqs. (5) and (6) do need to be modified. One might also wish to modify Eq. (5) to explore generalizations of the several existing weighted clustering coefficients for ordinary networks [52].

We now modify Eq. (6) to consider weighted clustering coefficients more generally. Let \mathcal{N} be a real number that can be used to rescale the elements of the tensor. Define $\tilde{W}_\beta^\alpha = W_\beta^\alpha / \mathcal{N}$, where one can define the normalization \mathcal{N} in various ways. For example, it can come from the maximum (so that $\mathcal{N} = \max_{\alpha,\beta} \{W_\beta^\alpha\}$). It is straightforward to show that $c(\tilde{W}_\beta^\alpha) = c(W_\beta^\alpha) / \mathcal{N}$. Therefore, we redefine the global clustering coefficient $c(W_\beta^\alpha)$ from Eq. (5) using this normalization:

$$c(W_\beta^\alpha) = \mathcal{N}^{-1} \frac{W_\rho^\alpha W_\sigma^\rho W_\alpha^\sigma}{W_\rho^\alpha F_\sigma^\rho W_\alpha^\sigma}. \quad (7)$$

The same argument applies in the case of the local clustering coefficient for weighted networks. The choice of the norm in the normalization factor \mathcal{N} is an important consideration. For example, the choice $\mathcal{N} = \max_{\alpha,\beta} \{W_\beta^\alpha\}$ ensures that Eq. (7) reduces to Eq. (6) for unweighted networks.

Eigenvector and Katz centralities.—Numerous notions of centrality exist to attempt to quantify the importance of nodes (and other components) in a network [5]. For example, a node n_i has a high eigenvector centrality if its neighbors also have high eigenvector centrality, and the recursive nature of this notion yields a vector of centralities that satisfies an eigenvalue problem.

Let \mathbf{A} be the adjacency matrix for an undirected network, \mathbf{v} be a solution of the equation $\mathbf{A}\mathbf{v} = \lambda_1 \mathbf{v}$, and λ_1 be the largest (“leading”) eigenvalue of \mathbf{A} . Thus, \mathbf{v} is the leading eigenvector of \mathbf{A} , and the components of \mathbf{v} give the eigenvector centralities of the nodes. That is, the eigenvector centrality of node n_i is given by v_i [56,57].

In our tensorial formulation, the eigenvector centrality vector is a solution of the tensorial equation

$$W_\beta^\alpha v_\alpha = \lambda_1 v_\beta, \quad (8)$$

and $v_\alpha e^\alpha(i)$ gives the eigenvector centrality of node n_i .

For directed networks, there are two leading eigenvectors, and one needs to take into account the difference between Eq. (8) and its contravariant counterpart. Moreover, nodes with only outgoing edges have an eigenvector centrality of 0 if the above definition is adopted. One way to address this situation is to assign a small amount b of centrality to each node before calculating centrality. One incorporates this modification of eigenvector centrality by finding the leading-eigenvector solution of the eigenvalue problem $\mathbf{v} = a\mathbf{A}\mathbf{v} + b\mathbf{1}$, where $\mathbf{1}$ is a vector in which each entry is a 1. This type of centrality is known as Katz centrality [58]. One often chooses $b = 1$, and we note that Katz centrality is well defined if $\lambda_1^{-1} > a$. Using tensorial notation, we obtain

$$v_\beta = (\delta_\beta^\alpha - aW_\beta^\alpha)^{-1} u_\alpha. \quad (9)$$

To calculate Katz centrality from Eq. (9), we need to calculate the tensor inverse T_β^α , which satisfies the equation $T_\beta^\alpha (\delta_\sigma^\beta - aW_\sigma^\beta) = \delta_\sigma^\alpha$.

Modularity.—It is often useful to decompose networks into disjoint sets (“communities”) of nodes such that (relative to some null model) nodes within each community are densely connected to each other but connections between communities are sparse. Modularity is a network descriptor that can be calculated for any partition of a network into disjoint sets of nodes. Additionally, one can attempt to algorithmically determine a partition that maximizes modularity to identify dense communities in a monoplex network. There are many ways to maximize modularity as well as many other ways to algorithmically detect communities (see the review articles [59,60]). We will consider Newman-Girvan modularity [61], which is the most popular version of modularity and can be written conveniently in matrix notation [62,63]. Let S_a^α be a tensor in $\mathbb{R}^{N \times M}$, where α indexes nodes and a indexes the communities¹ in an undirected network, which can be either weighted or unweighted. The value of a component of S_a^α is defined to be 1 when a node belongs to a particular community and 0 when it does not. We introduce the tensor $B_\beta^\alpha = W_\beta^\alpha - k^\alpha k_\beta / \mathcal{K}$, where $\mathcal{K} = W_\beta^\alpha U_\alpha^\beta$. It follows that the modularity of a network partition is given by the scalar²

$$Q = \frac{1}{\mathcal{K}} S_a^\alpha B_\beta^\alpha S_a^\beta. \quad (10)$$

¹The reader should be careful to not confuse the latter latin index with the indices that we have used thus far.

²Recall that swapping subscripts and superscripts (and hence covariance and contravariance) in a tensor is an implicit use of a transposition operator [40].

To consider a general null model, we write $B_\beta^\alpha = W_\beta^\alpha - P_\beta^\alpha$, where P_β^α is a tensor that encodes the random connections against which one compares a network's actual connections. With this general null-model tensor, modularity is also appropriate for directed networks (although, of course, it is still necessary to choose an appropriate null model).

von Neumann entropy.—The study of entropy in monoplex networks has been used to help characterize complexity in networked systems [64–68]. As an example, let us consider the von Neumann entropy of a monoplex network [69]. Recall that the von Neumann entropy extends the Shannon (information) entropy to quantum systems. In quantum mechanics, the density matrix ρ is a positive semidefinite operator that describes the mixed state of a quantum system, and the von Neumann entropy of ρ is defined by $\mathcal{H}(\rho) = -\text{tr}(\rho \log_2 \rho)$. The eigenvalues of ρ must sum to 1 to have a well-defined measure of entropy.

We also need to recall the (unnormalized) combinatorial Laplacian tensor, which is a well-known object in graph theory (see, e.g., Refs. [70,71] and references therein) and is defined by $L_\beta^\alpha = \Delta_\beta^\alpha - W_\beta^\alpha$, where $\Delta_\beta^\alpha = W_\gamma^\eta u_\eta e^\gamma(\beta) \delta_\beta^\alpha$ is the strength tensor (i.e., a diagonal tensor whose elements represent the strength of the nodes). The combinatorial Laplacian is positive semidefinite, and the trace of the strength tensor is $\Delta = \Delta_\alpha^\alpha$. The eigenvalues of the density tensor $\rho_\beta^\alpha = \Delta^{-1} L_\beta^\alpha$ sum to 1, as required, and they can be used to define the von Neumann entropy of a monoplex network using the formula

$$\mathcal{H}(W_\beta^\alpha) = -\rho_\beta^\alpha \log_2[\rho_\beta^\alpha]. \quad (11)$$

Using the eigendecomposition of the density tensor, one can show that the von Neumann entropy reduces to

$$\mathcal{H}(W_\beta^\alpha) = -\Lambda_\beta^\alpha \log_2[\Lambda_\beta^\alpha], \quad (12)$$

where Λ_β^α is the diagonal tensor whose elements are the eigenvalues of ρ_β^α (see Appendix C).

Diffusion and random walks.—A random walk is the simplest dynamical process that can occur on a monoplex network, and random walks can be used to approximate other types of diffusion [2,72]. Diffusion is also relevant for many other types of dynamical processes (e.g., for some types of synchronization [73]).

Let $x_\alpha(t)$ denote a state vector of nodes at time t . The diffusion equation is

$$\frac{dx_\beta(t)}{dt} = D[W_\beta^\alpha x_\alpha(t) - W_\gamma^\alpha u_\alpha e^\gamma(\beta) x_\beta(t)], \quad (13)$$

where D is a diffusion constant. Recall that $s_\gamma = W_\gamma^\alpha u_\alpha$ is the strength vector and that $s_\gamma e^\gamma(\beta) x_\beta(t) = s_\gamma e^\gamma(\beta) \times \delta_\beta^\alpha x_\alpha(t)$. We obtain the following covariant diffusion law on monoplex networks:

$$\frac{dx_\beta(t)}{dt} = -DL_\beta^\alpha x_\alpha(t), \quad (14)$$

where $L_\beta^\alpha = W_\gamma^\eta u_\eta e^\gamma(\beta) \delta_\beta^\alpha - W_\beta^\alpha$ is the combinatorial Laplacian tensor. The solution of Eq. (14) is $x_\beta(t) = x_\alpha(0) e^{-DL_\beta^\alpha t}$.

Random walks on monoplex networks [2,72,74] have attracted considerable interest because they are both important and easy to interpret. They have yielded important insights on a huge variety of applications and can be studied analytically. For example, random walks have been used to rank Web pages [75] and sports teams [76], optimize searches [77], investigate the efficiency of network navigation [78,79], characterize cyclic structures in networks [80], and coarse grain networks to illuminate mesoscale features such as community structure [81–83].

In this paper, we consider a discrete-time random walk. Let T_β^α denote the tensor of transition probabilities between pairs of nodes, and let $p_\alpha(t)$ denote the vector of probabilities to find a walker at each node. Hence, the covariant master equation that governs the discrete-time evolution of the probability from time t to time $t+1$ is $p_\beta(t+1) = T_\beta^\alpha p_\alpha(t)$. One can rewrite this master equation in terms of evolving probability rates as $\dot{p}_\beta(t) = -\bar{L}_\beta^\alpha p_\alpha(t)$, where $\bar{L}_\beta^\alpha = \delta_\beta^\alpha - T_\beta^\alpha$ is the normalized Laplacian tensor. The normalized Laplacian governs the evolution of the probability-rate vector for random walks.

B. Multilayer networks

Because of its structure, a multilayer network can incorporate a lot of information. Before generalizing the descriptors that we discussed for monoplex networks, we discuss some algebraic operations that can be employed to extract useful information from an adjacency tensor.

Contraction.—Tensor contractions yield interesting quantities that are invariants when the indices are repeated (see Appendix A). For instance, one can obtain the number of nodes in a network (which is an invariant quantity) by contracting the Kronecker tensor $N = \delta_\alpha^\alpha$, where we have again used Einstein summation convention. For unweighted networks, one obtains the number of edges (which is another invariant) by calculating the scalar product between the adjacency tensor W_β^α with the dual 1-tensor U_α^β (whose components are all equal to 1).

Single-layer extraction.—In some applications, it can be useful to extract a specific layer (e.g., the \tilde{r} th one) from a multilayer network. Using tensorial algebra, this operation is equivalent to projecting the tensor $M_{\beta\delta}^{\alpha\tilde{\gamma}}$ to the canonical tensor $E_{\tilde{\gamma}}^{\delta\tilde{\gamma}}(\tilde{r}\tilde{r})$ that corresponds to this particular layer. The second-order canonical tensors in $\mathbb{R}^{L \times L}$ form an orthonormal basis, so the product $E_{\tilde{\delta}}^{\tilde{\gamma}}(\tilde{h}\tilde{k}) E_{\tilde{\gamma}}^{\delta\tilde{\gamma}}(\tilde{r}\tilde{r})$ equals 1 for $\tilde{h} = \tilde{k} = \tilde{r}$ and it equals 0 otherwise. Therefore, we use Eq. (3) to write

$$M_{\beta\delta}^{\alpha\tilde{\gamma}} E_{\tilde{\gamma}}^{\delta}(\tilde{r}\tilde{r}) = C_{\beta}^{\alpha}(\tilde{r}\tilde{r}) = W_{\beta}^{\alpha}(\tilde{r}), \quad (15)$$

which is the desired adjacency tensor that corresponds to layer \tilde{r} . Clearly, it is possible to use an analogous procedure to extract any other tensor (e.g., ones that give interlayer relationships). In practical applications, for example, it might be useful to extract the tensors that describe interlayer connections between pairs of layers in the multilayer network to compare the strengths of the couplings between them. Another important application, which we discuss later, is the calculation of multilayer clustering coefficients.

Projected and overlay monoplex networks.—In some cases, one constructs a monoplex network by aggregating multiple networks. Such aggregation is useful in many situations. For example, the first step in studying a temporal network is often to aggregate over time. When studying social networks, one often aggregates over different types of relationships, different communication platforms, and/or different social circles. To project a multilayer network onto a weighted single-layer network, we multiply the corresponding tensor by the 1-tensor $U_{\tilde{\gamma}}^{\delta}$. The projected monoplex network $P_{\beta}^{\alpha} = M_{\beta\delta}^{\alpha\tilde{\gamma}} U_{\tilde{\gamma}}^{\delta}$ that we obtain is

$$P_{\beta}^{\alpha} = \sum_{\tilde{h}, \tilde{k}=1}^L C_{\beta}^{\alpha}(\tilde{h}\tilde{k}). \quad (16)$$

Importantly, a projected monoplex network is different from the weighted monoplex network (which one might call an overlay monoplex network) that one obtains from a multilayer network by summing the edges over all layers for each node. In particular, the overlay network ignores the non-negligible contribution of interlayer connections, which are also important for quantifying the properties of a multilayer network. One obtains the overlay network from a multilayer adjacency tensor by contracting the indices corresponding to the layer components:

$$O_{\beta}^{\alpha} = M_{\beta\tilde{\gamma}}^{\alpha\tilde{\gamma}}. \quad (17)$$

In Fig. 3, we show schematics to illustrate the difference between projected and overlay monoplex networks.

Network of layers.—It can be useful to construct a global observable to help understand relations between layers at a macroscopic level. For instance, suppose that two layers have more interconnected nodes than other pairs of layers. Perhaps there are no connections across a given pair of layers. One can build a network of layers to help understand the structure of such interconnections. Such a network is weighted, although the weighting procedure is application dependent. As an example, let us consider the most intuitive weighting procedure: For each pair of layers $(\tilde{h}\tilde{k})$, we sum all of the weights in the connections between their nodes to obtain edge weights of $q_{\tilde{h}\tilde{k}} = C_{\beta}^{\alpha}(\tilde{h}\tilde{k}) U_{\alpha}^{\beta}$. For the special case of multiplex networks with unit weights between pairs of nodes in different layers, we obtain $q_{\tilde{h}\tilde{k}} = N$ if layers \tilde{h}

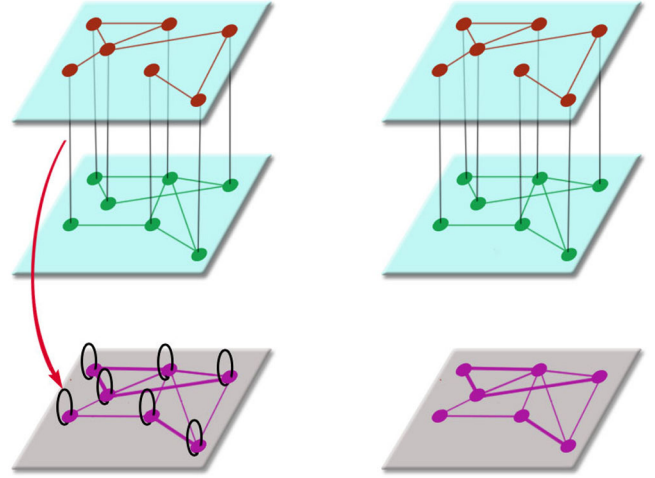


FIG. 3. Schematics of (left) projected and (right) overlay monoplex networks obtained from a multilayer network. Both types of single-layer networks are weighted, but their edges have different weights. One obtains an overlay network using the contraction operator and thus neglects the influence of interlayer connections.

and \tilde{k} are connected. The resulting weighted adjacency tensor of layers in the space $\mathbb{R}^{L \times L}$ is

$$\Psi_{\delta}^{\tilde{\gamma}} = \sum_{\tilde{h}, \tilde{k}=1}^L q_{\tilde{h}\tilde{k}} E_{\delta}^{\tilde{\gamma}}(\tilde{h}\tilde{k}). \quad (18)$$

Hence, one can calculate $\Psi_{\delta}^{\tilde{\gamma}}$ from the multilayer adjacency tensor with the formula

$$\Psi_{\delta}^{\tilde{\gamma}} = M_{\beta\delta}^{\alpha\tilde{\gamma}} U_{\alpha}^{\beta}. \quad (19)$$

One can then normalize the resulting tensor in a way that is appropriate for the application of interest. For multiplex networks, for example, the most sensible normalization constant is typically the number of layers L . In the insets (in the top-left corners) of the panels of Fig. 2, we show representations for networks of layers that correspond to three different topologies of connections between layers.

Degree centrality.—We now show how to compute degree centrality for multilayer networks by performing the same projections from the case of monoplex networks using 1-tensors of an appropriate order. (Recall that a 1-tensor is a tensor that contains a 1 in every component.) We thereby obtain a multidegree centrality vector $K^{\alpha} = M_{\beta\delta}^{\alpha\tilde{\gamma}} U_{\tilde{\gamma}}^{\delta} u^{\beta}$. After some algebra, we obtain

$$K^{\alpha} = \sum_{\tilde{h}, \tilde{k}=1}^L k^{\alpha}(\tilde{h}\tilde{k}), \quad (20)$$

where $k^{\alpha}(\tilde{h}\tilde{k})$ is the degree centrality vector that corresponds to connections between layers \tilde{h} and \tilde{k} . Even in the special case of multiplex networks, it is already evident that K^{α} differs from the degree centrality vector that one

would obtain by simply projecting all layers of a multilayer network onto a single weighted network.

The definitions of mean degree, second moment, and variance are analogous to the corresponding monoplex-network counterparts, except that one uses K^α instead of k^α .

Clustering coefficients.—For multilayer networks, it is nontrivial to define a clustering coefficient using triangles as a measure of transitivity. As shown in Fig. 4, a closed set of three nodes might not exist on any single layer, but transitivity can still arise as a consequence of multiplexity. In the left panel, for example, suppose that nodes A and B are friends with node C but not with each other, but that nodes A and B still have a social tie because they work at the same company (but node C does not). In this situation, it is necessary for connections to exist on multiple layers for it to be possible to transfer information from any one node to any other node.

As with monoplex networks, we start by defining the integer power of an adjacency tensor and use a contraction operation. For instance, one can calculate the square of $M_{\beta\delta}^{\alpha\gamma}$ by constructing the eighth-order (i.e., rank-8) tensor

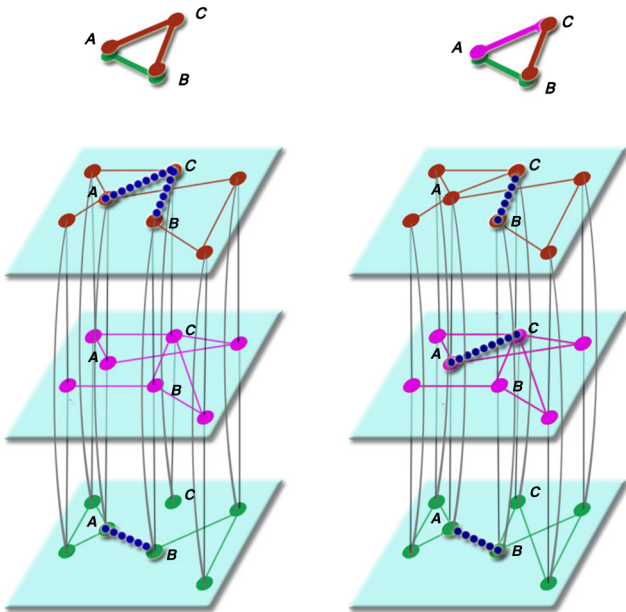


FIG. 4. Schematic of closing triangles in multiplex networks. Triangles can be closed using intralayer connections from different layers. In the figure, we show two different situations that can arise. For example, the left panel might represent a multilayer social network in which nodes A and B are friends of node C but are not friends with each other (first layer), but nodes A and B still have a social tie because they work in the same company even though node C does not work there (third layer). In the right panel, one might imagine that each layer corresponds to a different online social network: Perhaps node B tweets about an item; node C sees node B 's post on Twitter (first layer) and then posts the same item on Facebook (second layer); node A then sees this post and blogs about it (third layer), and node B reads this blog entry.

$M_{\beta\delta}^{\alpha\gamma} M_{\sigma\eta}^{\rho\epsilon}$ and then contracting β with ρ and δ with ϵ . One computes higher powers analogously. We define a global clustering coefficient on a multilayer adjacency tensor by generalizing Eq. (6) for fourth-order tensors:

$$c(M_{\beta\delta}^{\alpha\gamma}) = \mathcal{N}^{-1} \frac{M_{\beta\delta}^{\alpha\gamma} M_{\epsilon\eta}^{\beta\delta} M_{\alpha\gamma}^{\epsilon\eta}}{M_{\beta\delta}^{\alpha\gamma} F_{\epsilon\eta}^{\beta\delta} M_{\alpha\gamma}^{\epsilon\eta}}, \quad (21)$$

where we again define $F_{\epsilon\eta}^{\beta\delta} = U_{\epsilon\eta}^{\beta\delta} - \delta_{\epsilon\eta}^{\beta\delta}$ as the adjacency tensor of a complete multilayer network (without self-edges). The choice of the normalization factor \mathcal{N} is again arbitrary, but the choice $\mathcal{N} = \max_{\alpha,\beta,\gamma,\delta} \{M_{\beta\delta}^{\alpha\gamma}\}$ ensures that Eq. (21) is well defined for both weighted and unweighted multilayer networks.

The tensor contractions in Eq. (21) count all of the three-cycles, including ones in which a walk goes through any combination of interlayer and intralayer connections. Thus, for multiplex networks with categorical layers, Eq. (21) counts not only fully intralayer three-cycles but also the interlayer three-cycles that are induced by the connection of nodes to their counterparts in all of the other layers.

A more traditional, and also simpler, approach to calculating a global clustering coefficient of a multilayer network is to project it onto a single weighted network [i.e., the overlay network O_β^α defined by Eq. (17)] and then calculate a clustering coefficient for the resulting network. In this case, we obtain

$$c(O_\beta^\alpha) = \mathcal{M}^{-1} \frac{M_{\beta\gamma}^{\alpha\gamma} M_{\epsilon\delta}^{\beta\delta} M_{\alpha\eta}^{\epsilon\eta}}{M_{\beta\gamma}^{\alpha\gamma} F_{\epsilon\delta}^{\beta\delta} M_{\alpha\eta}^{\epsilon\eta}}, \quad (22)$$

where $\mathcal{M} = \max_{\alpha,\beta} \{M_{\beta\gamma}^{\alpha\gamma}\} / L$. In the chosen normalization, note that we need to include a factor for the number of layers L because the construction of the overlay network discards the information about the number of layers. For example, adding an empty layer to a multilayer network does not affect the resulting overlay network, but it increases the number of possible walks that one must consider in Eq. (22). In general, the clustering coefficient in Eq. (22) is different from the one in Eq. (21) because Eq. (22) discards all interlayer connections. However, there are some special cases when Eq. (22) and Eq. (21) take the same value. In particular, these equations yield the same value when there is only a single layer or, more generally, when there are no interlayer edges and all of the intralayer networks are exactly the same.

The global clustering coefficient defined in Eq. (22) sums the contributions of three-cycles for which each step of a walk is on the same layer and those for which a walk traverses two or three layers. We decompose this clustering coefficient to separately count the contributions of three-cycles that take place on one, two, and three layers [47]. Using our tensorial framework, we modify Eq. (22) as follows:

$$c(M_{\beta\delta}^{\alpha\gamma}, w^\Omega) = \frac{M_{\beta\delta}^{\alpha\gamma} M_{\gamma\sigma}^{\beta\bar{\rho}} M_{\alpha\bar{\epsilon}}^{\gamma\bar{\eta}} \sum_{\tilde{h}, \tilde{k}, \tilde{l}=1}^L E_{\tilde{\gamma}}^{\delta}(\tilde{h}, \tilde{h}) E_{\tilde{\rho}}^{\bar{\sigma}}(\tilde{k}, \tilde{k}) E_{\tilde{\eta}}^{\bar{\epsilon}}(\tilde{l}, \tilde{l}) \delta_{\Omega}(\tilde{h}, \tilde{k}, \tilde{l}) w^{\Omega}}{M_{\beta\delta}^{\alpha\gamma} F_{\gamma\sigma}^{\beta\bar{\rho}} M_{\alpha\bar{\epsilon}}^{\gamma\bar{\eta}} \sum_{\tilde{h}, \tilde{k}, \tilde{l}=1}^L E_{\tilde{\gamma}}^{\delta}(\tilde{h}, \tilde{h}) E_{\tilde{\rho}}^{\bar{\sigma}}(\tilde{k}, \tilde{k}) E_{\tilde{\eta}}^{\bar{\epsilon}}(\tilde{l}, \tilde{l}) \delta_{\Omega}(\tilde{h}, \tilde{k}, \tilde{l}) w^{\Omega}}, \quad (23)$$

where we have employed layer extraction operations (see our earlier discussion), w^Ω is a vector that weights the contribution of three-cycles that span Ω layers, and δ_Ω (where $\Omega = 1, 2, 3$) is a function that selects the cycles with Ω layers:

$$\begin{aligned} \delta_1(\tilde{h}, \tilde{k}, \tilde{l}) &= \delta_{\tilde{h}\tilde{l}} \delta_{\tilde{h}\tilde{k}} \delta_{\tilde{k}\tilde{l}}, \\ \delta_2(\tilde{h}, \tilde{k}, \tilde{l}) &= (1 - \delta_{\tilde{h}\tilde{l}}) \delta_{\tilde{h}\tilde{k}} + (1 - \delta_{\tilde{k}\tilde{l}}) \delta_{\tilde{h}\tilde{k}} + (1 - \delta_{\tilde{h}\tilde{k}}) \delta_{\tilde{k}\tilde{l}}, \\ \delta_3(\tilde{h}, \tilde{k}, \tilde{l}) &= (1 - \delta_{\tilde{h}\tilde{l}})(1 - \delta_{\tilde{k}\tilde{l}})(1 - \delta_{\tilde{h}\tilde{k}}). \end{aligned}$$

We recover Eq. (22) with the choice $w^\Omega = (1/3, 1/3, 1/3)$. [We note that $c(M_{\beta\delta}^{\alpha\gamma}, w^\Omega) = c(M_{\beta\delta}^{\alpha\gamma}, c_0 w^\Omega)$ for any nonzero constant c_0 , although we still normalize w^Ω by convention.] By contrast, with the choice of $w^\Omega = (1, 0, 0)$, we only consider three-cycles in which every step of a walk is on the same layer.

Importantly, we can define the weight vector w^Ω in the clustering coefficient in Eq. (23) so that it takes into account that there might be some cost for crossing different layers. As discussed in Ref. [47], one determines this ‘‘cost’’ based on the dynamics and the application of interest. For example, if one is studying a dynamical process whose time scale is very fast compared to the time scale (i.e., the cost) associated with changing layers, then it is desirable to consider the contribution from only one layer (i.e., the one in which the dynamical process occurs). For other dynamical processes, it is compulsory to also include contributions from two or three layers. To give a brief example, consider transportation at the King’s Cross/St. Pancras station in London. This station includes a node in a layer that describes travel via London’s metropolitan transportation system, a node in a layer for train travel within England, and a node for international train travel. A relevant cost is then related to how long it takes to travel between different parts of the station [44]. One needs to consider such intrastation travel time in comparison to the schedule times of several transportation mechanisms. By contrast, there is typically very little cost associated with a person seeing information on Facebook and then posting it on Twitter.

In the above definition, the entries of $F_{\epsilon\bar{\eta}}^{\beta\bar{\delta}}$ are all equal to 1 except for self-edges. Sometimes, we need to instead use a tensor $\check{F}_{\epsilon\bar{\eta}}^{\beta\bar{\delta}}$ that we construct by setting some of the off-diagonal entries of $F_{\epsilon\bar{\eta}}^{\beta\bar{\delta}}$ to 0. If the original multilayer network cannot have a particular edge, then the tensor $\check{F}_{\epsilon\bar{\eta}}^{\beta\bar{\delta}}$ needs to have a 0 in its corresponding entry. For example, we need to use such 0 entries for multilayer networks whose structural constraints forbid the existence of some nodes in certain layers or forbid certain inter- and intralayer edges. It is also necessary for multiplex networks, for which

interlayer edges can only exist between nodes and their counterparts in other layers. Note, however, that using the tensor $\check{F}_{\epsilon\bar{\eta}}^{\beta\bar{\delta}}$ instead of $F_{\epsilon\bar{\eta}}^{\beta\bar{\delta}}$ influences the normalization of clustering coefficients, as it affects the set of potential three-cycles that can exist in a multilayer network.

Eigenvector centrality.—Generalizing eigenvector centrality for a multilayer network is not trivial, and there are several possible ways to do it [24].

References [21,22,27,28] recently introduced the concepts of supra-adjacency (i.e., ‘‘superadjacency’’) and supra-Laplacian (i.e., super-Laplacian) matrices to formulate and solve eigenvalue problems in multiplex networks. Such supramatrices correspond to unique unfoldings of corresponding fourth-order tensors to obtain square matrices. It is worth noting that the tensorial space in which the multilayer adjacency tensor exists is $\mathbb{R}^{N \times N \times L \times L}$, and there exists a unique unfolding—up to the $L!$ permutations of diagonal blocks of size $N \times N$ in the resulting space—that provides a square supra-adjacency tensor defined in $\mathbb{R}^{NL \times NL}$. We now exploit the same idea by arguing that a supraeigenvector corresponds to a rank-1 unfolding of a second-order ‘‘eigensensor’’ $V_{\alpha\bar{\gamma}}$. According to this unique mapping, if λ_1 is the largest eigenvalue and $V_{\alpha\bar{\gamma}}$ is the corresponding eigensensor, then it follows that

$$M_{\beta\delta}^{\alpha\gamma} V_{\alpha\bar{\gamma}} = \lambda_1 V_{\beta\bar{\delta}}. \quad (24)$$

Therefore, similarly to monoplex networks, one can calculate the leading eigensensor $V_{\alpha\bar{\gamma}}$ iteratively. Start with a tensor $X_{\alpha\bar{\gamma}}(t=0)$, which we can take to be $X_{\alpha\bar{\gamma}}(t=0) = U_{\alpha\bar{\gamma}}$. By writing $X_{\alpha\bar{\gamma}}(0)$ as a linear combination of the second-order eigensensors and by observing that $X_{\alpha\bar{\gamma}}(t) = (M_{\beta\delta}^{\alpha\gamma})^t X_{\alpha\bar{\gamma}}(0)$, one can show that $X_{\alpha\bar{\gamma}}(t)$ is proportional to $V_{\alpha\bar{\gamma}}$ in the $t \rightarrow \infty$ limit. The convergence of this approach is ensured by the existence of the unfolding of M , since the iterative procedure is equivalent to the one applied to the corresponding supramatrices.

We thereby obtain a multilayer generalization of Bonacich’s eigenvector centrality [56,57]:

$$V_{\beta\bar{\delta}} = \lambda_1^{-1} M_{\beta\delta}^{\alpha\gamma} V_{\alpha\bar{\gamma}}. \quad (25)$$

The monoplex notion of eigencentality grants importance to a node based on its connection to other nodes. One needs to be careful about both intralayer and interlayer connections when interpreting the results of calculating a multilayer generalization of it. For example, the intralayer connections in one layer might be more important than those in others. For interlayer connections, one might ponder how much of a ‘‘bonus’’ an entity earns based on its

presence in multiple layers. (Contrast this situation with the cost that we discussed previously in the context of transportation networks.) For instance, many Web services attempt to measure the influence of people on social media by combining information from multiple online social networks, and one can choose which communication modes (i.e., layers) to include. Moreover, by considering an overlay monoplex network or a projection monoplex network, it is possible to derive separate centrality scores for different layers. In the above example, an individual who possesses different centrality scores in different network layers would reflect the different levels of importance that that person has on different social media.

Modularity.—A multilayer generalization of modularity was derived in Ref. [13] by considering random walks on networks. Let $S_a^{\alpha\tilde{\rho}}$ be a tensor in $\mathbb{R}^{N \times L \times M}$, where $(\alpha, \tilde{\rho})$ indexes nodes and a indexes the communities in an undirected multilayer network, which can be either weighted or unweighted. The value of a component of $S_a^{\alpha\tilde{\rho}}$ is defined to be 1 when a node belongs to a particular community and 0 when it does not. We introduce the tensor $B_{\beta\tilde{\sigma}}^{\alpha\tilde{\rho}} = W_{\beta\tilde{\sigma}}^{\alpha\tilde{\rho}} - P_{\beta\tilde{\sigma}}^{\alpha\tilde{\rho}}$, where $\mathcal{K} = W_{\beta\tilde{\sigma}}^{\alpha\tilde{\rho}} U_{\alpha\tilde{\rho}}^{\beta\tilde{\sigma}}$, and $P_{\beta\tilde{\sigma}}^{\alpha\tilde{\rho}}$ is a null-model tensor that encodes the random connections against which one compares a multilayer network's actual connections. It follows that the modularity of a partition of a multilayer network is given by the scalar

$$Q = \frac{1}{\mathcal{K}} S_{\alpha\tilde{\rho}}^a B_{\beta\tilde{\sigma}}^{\alpha\tilde{\rho}} S_a^{\beta\tilde{\sigma}}. \quad (26)$$

There are numerous choices for the null-model tensor $P_{\beta\tilde{\sigma}}^{\alpha\tilde{\rho}}$. The null models discussed in Refs. [13,41,84] give special cases of the multilayer modularity in Eq. (26).

von Neumann entropy.—To generalize the definition of von Neumann entropy to multilayer networks, we need to generalize the definition of the Laplacian tensor. Such an extension is not trivial because one needs to consider eigenvalues of a fourth-order tensor.

As we showed previously when generalizing eigenvector centrality, the existence of a unique unfolding into supramatrices allows one to define and solve the eigenvalue problem

$$L_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} V_{\alpha\tilde{\gamma}} = \lambda V_{\beta\tilde{\delta}}, \quad (27)$$

where $L_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} = \Delta_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} - M_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}$ is the multilayer Laplacian tensor, $\Delta_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} = M_{\rho\tilde{\sigma}}^{\eta\tilde{\epsilon}} U_{\eta\tilde{\epsilon}}^{\rho\tilde{\sigma}} (\beta\tilde{\delta}) \delta_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}$ is the multistrength tensor (i.e., the rank-4 counterpart of the strength tensor Δ_{β}^{α} that we defined previously for monoplex networks), λ is an eigenvalue, and $V_{\alpha\tilde{\gamma}}$ is its corresponding eigentensor (i.e., the unfolded rank-1 supraeigenvector). We note that there are at most NL different eigenvalues and corresponding eigentensors. (Additionally, note that the 1-tensor that appears in the definition of the multistrength tensor has two subscripts and that the canonical-basis tensor has two superscripts.)

Let $\Delta = \Delta_{\alpha\tilde{\gamma}}^{\alpha\tilde{\gamma}}$ be the trace of the multistrength tensor. The eigenvalues of the multilayer density tensor $\rho_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} = \Delta^{-1} L_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}$ sum to 1, so we can use them to define the von Neumann entropy of a multilayer network as

$$\mathcal{H}(M) = -\Lambda_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} \log_2[\Lambda_{\alpha\tilde{\gamma}}^{\beta\tilde{\delta}}], \quad (28)$$

where $\Lambda_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}$ is the diagonal tensor whose elements are the eigenvalues of $\rho_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}}$.

Diffusion and random walks.—Diffusion in multiplex networks was investigated recently in Ref. [21]. A diffusion equation for multilayer networks needs to include terms that account for interlayer diffusion. Let $X_{\alpha\tilde{\gamma}}(t)$ denote the state tensor of nodes in each layer at time t . The simplest diffusion equation for a multilayer network is then

$$\frac{dX_{\beta\tilde{\delta}}(t)}{dt} = M_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} X_{\alpha\tilde{\gamma}}(t) - M_{\rho\tilde{\sigma}}^{\alpha\tilde{\gamma}} U_{\alpha\tilde{\rho}}^{\rho\tilde{\sigma}} E^{\rho\tilde{\sigma}} (\beta\tilde{\delta}) X_{\beta\tilde{\delta}}(t). \quad (29)$$

As in the case of monoplex networks, we introduce the multilayer combinatorial Laplacian

$$L_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} = M_{\rho\tilde{\sigma}}^{\eta\tilde{\epsilon}} U_{\eta\tilde{\epsilon}}^{\rho\tilde{\sigma}} E^{\rho\tilde{\sigma}} (\beta\tilde{\delta}) \delta_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} - M_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} \quad (30)$$

to obtain the following covariant diffusion equation for multilayer networks:

$$\frac{dX_{\beta\tilde{\delta}}(t)}{dt} = -L_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} X_{\alpha\tilde{\gamma}}(t). \quad (31)$$

The solution of Eq. (31) is $X_{\beta\tilde{\delta}}(t) = X_{\alpha\tilde{\gamma}}(0) e^{-L_{\beta\tilde{\delta}}^{\alpha\tilde{\gamma}} t}$, which is a natural generalization of the result for monoplex networks.

The study of random walks is important for many applications in multilayer networks. For instance, they were used to derive multilayer modularity [13] and to develop optimized exploration strategies [22]. As we illustrate in Fig. 5, a random walk on a multilayer network induces nontrivial effects because the presence of interlayer connections affects its navigation of a networked system. As with monoplex networks, we consider discrete-time

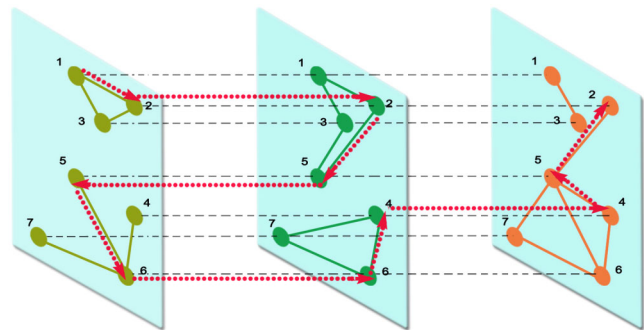


FIG. 5. Schematic of a random walk (dotted trajectories) in a multiplex network. A walker can jump between nodes within the same layer, or it might switch to another layer. This illustration evinces how multiplexity allows a random walker to move between nodes that belong to different (disconnected) components on a given layer.

random walks. Let $T_{\beta\delta}^{\alpha\gamma}$ denote the tensor of transition probabilities for jumping between pairs of nodes and switching between pairs of layers, and let $p_{\alpha\gamma}(t)$ be the time-dependent tensor that gives the probability to find a walker at a particular node in a particular layer. Hence, the covariant master equation that governs the discrete-time evolution of the probability from time t to time $t + 1$ is $p_{\beta\delta}(t + 1) = T_{\beta\delta}^{\alpha\gamma} p_{\alpha\gamma}(t)$. We rewrite this master equation in terms of evolving probability rates to obtain $\dot{p}_{\beta\delta}(t) = -\bar{L}_{\beta\delta}^{\alpha\gamma} p_{\alpha\gamma}(t)$, where $\bar{L}_{\beta\delta}^{\alpha\gamma} = \delta_{\beta\delta}^{\alpha\gamma} - T_{\beta\delta}^{\alpha\gamma}$ is the normalized Laplacian tensor.

V. CONCLUSIONS AND DISCUSSION

In this paper, we developed a tensorial framework to study general multilayer networks. We discussed the generalization of several important network descriptors—including degree centrality, clustering coefficients, eigenvector centrality, and modularity—for our multilayer framework. We examined different choices that one can make in developing such generalizations, and we also demonstrated how our formalism yields results for monoplex and multiplex networks as special cases.

As we have discussed in detail, our multilayer formalism provides natural generalizations of network descriptors. Consequently, it allows systematic comparisons of multilayer diagnostics with their single-layer counterparts. As we have also illustrated (e.g., for global clustering coefficients), our formalism also allows systematic comparisons between different ways of generalizing familiar network concepts. Such comparisons are particularly important for the examination of new phenomena, such as multiplexity-induced correlations [19], that arise when generalizing beyond the usual single-layer networks. One can obtain new insights even for simple descriptors like (directed) degree centrality, for which the tensor indices in our formulation are related directly to the directionality of relationships between nodes in a multilayer network.

The mathematical formalism that we have introduced can be generalized further by considering higher-order (i.e., higher-rank) tensors. This generalization will provide a systematic means to investigate networks that are, for example, both time dependent and multiplex.

Our tensorial framework is an important step toward the development of a unified theoretical framework for studying networks with arbitrary complexities (including multiplexity, time dependence, and more). When faced with generalizing the usual adjacency matrices to incorporate a feature such as multiplexity, different scholars have employed different notation and terminology, and it is thus desirable to construct a unified framework to unify the language for studying networks. Moreover, in addition to defining mathematical notation that simplifies the handling and generalization of previously known diagnostics

on networks, a tensorial framework also offers the opportunity to unravel new properties that remain hidden when using the classical approach of adjacency matrices. We hope to construct a proper geometrical interpretation for tensorial representations of networks and to ultimately obtain an operational theory of dynamics both on and of networks. This perspective has led to significant advances in other areas of physics, and we believe that it will also be important for the study of networks.

ACKNOWLEDGMENTS

All authors were supported by the European Commission FET-Proactive Project PLEXMATH (Grant No. 317614). A. A., M. D. D., S. G., and A. S. were also supported by the Generalitat de Catalunya 2009-SGR-838. A. A. also acknowledges financial support from the ICREA Academia and the James S. McDonnell Foundation, and S. G. and A. A. were supported by FIS2012-38266. Y. M. was also supported by MINECO through Grant No. FIS2011-25167 and by Comunidad de Aragón (Spain) through a grant to the group FENOL. M. A. P. acknowledges a grant (No. EP/J001759/1) from the EPSRC. We thank Peter Mucha and two anonymous referees for useful comments.

APPENDIX A: EINSTEIN SUMMATION CONVENTION

Einstein notation is a summation convention, which we adopt to reduce the notational complexity in our tensorial equations, that is applied to repeated indices in operations that involve tensors. For example, we use this convention in the left-hand sides of the following equations:

$$\begin{aligned} A_\alpha^\alpha &= \sum_{\alpha=1}^N A_{\alpha\alpha}^\alpha, & A^\alpha B_\alpha &= \sum_{\alpha=1}^N A^\alpha B_\alpha, \\ A_\beta^\alpha B_\gamma^\beta &= \sum_{\beta=1}^N A_\beta^\alpha B_\gamma^\beta, & A_\beta^\alpha B_\alpha^\beta &= \sum_{\alpha=1}^N \sum_{\beta=1}^N A_\beta^\alpha B_\alpha^\beta, \end{aligned}$$

whose right-hand sides include the summation signs explicitly. It is straightforward to use this convention for the product of any number of tensors of any order. Repeated indices, such that one index is a subscript and the other is a superscript, are equivalent to perform a tensorial operation known as a contraction. Contracting indices reduces the order of a tensor by 2. For instance, the contraction of the second-order tensor A_β^α is the scalar A_α^α , and the second-order tensors $A_\beta^\alpha B_\delta^\beta$ and $A_\beta^\delta B_\delta^\beta$ are obtained by contracting the fourth-order tensor $A_\beta^\alpha B_\delta^\beta$.

It is important to adopt Einstein summation on repeated indices in a way that is unambiguous. For example, $A_\alpha^\alpha = A_\beta^\beta$, but $A_\beta^\alpha B_\gamma^\beta C_\alpha^\gamma$ is not equivalent to $A_\beta^\alpha B_\beta^\beta C_\alpha^\beta$ because of the ambiguity about the index β in the second term of $A_\beta^\alpha B_\beta^\beta C_\alpha^\beta$. Specifically, it is not clear if the contraction B_β^β

should be calculated before the product with the other tensors, or vice versa. Another situation that deserves particular attention is equations that involve a ratio between tensorial products, where one should separately apply the Einstein convention to the numerator and the denominator. Thus, one should not perform products between tensors B_β^α and C_β^α with repeated indices α and β in cases like

$$\frac{A_\beta^\alpha B_\alpha^\beta}{C_\beta^\alpha A_\alpha^\beta}. \quad (\text{A1})$$

For example, see Eq. (5) in the main text.

APPENDIX B: DEFINITION OF THE TENSOR EXPONENTIAL AND LOGARITHM

The exponential of a tensor B_β^α is a tensor A_β^α such that $e^{B_\beta^\alpha} = A_\beta^\alpha$. The tensor exponential is defined by the power series [85]

$$e^{B_\beta^\alpha} = \sum_{m=0}^{\infty} \frac{1}{m!} (B_\beta^\alpha)^m, \quad (\text{B1})$$

where

$$(B_\beta^\alpha)^m = B_{\gamma_1}^\alpha B_{\gamma_2}^{\gamma_1} B_{\gamma_3}^{\gamma_2} \cdots B_{\beta}^{\gamma_{m-1}}. \quad (\text{B2})$$

A complete discussion of the properties of the tensor exponential is beyond the scope of the present paper. However, we show an example of how to calculate it for diagonalizable tensors.

Let B_β^α be a diagonalizable tensor. In other words, there exists a diagonal tensor D_β^α , whose elements are the eigenvalues of B_β^α , and a tensor J_β^α , whose columns are the eigenvectors of B_β^α , such that $B_\beta^\alpha = J_\sigma^\alpha D_\tau^\sigma (J_\beta^\tau)^{-1}$. It follows that

$$\begin{aligned} (B_\beta^\alpha)^m &= J_\sigma^\alpha D_\tau^\sigma (J_{\gamma_1}^\tau)^{-1} J_{\gamma_1}^{\gamma_2} D_{\tau_1}^{\sigma_1} (J_{\gamma_2}^{\tau_1})^{-1} \cdots J_{\sigma_{m-1}}^{\gamma_{m-1}} D_{\tau_{m-1}}^{\sigma_{m-1}} (J_{\beta}^{\tau_{m-1}})^{-1} \\ &= J_\sigma^\alpha (D_\tau^\sigma)^m (J_\beta^\tau)^{-1} \end{aligned} \quad (\text{B3})$$

and

$$e^{B_\beta^\alpha} = J_\sigma^\alpha \left[\sum_{m=0}^{\infty} \frac{1}{m!} (D_\tau^\sigma)^m \right] (J_\beta^\tau)^{-1} = J_\sigma^\alpha e^{D_\tau^\sigma} (J_\beta^\tau)^{-1}. \quad (\text{B4})$$

The exponential of a diagonal tensor is the tensor obtained by exponentiating each of the diagonal elements D_β^α , and it is straightforward to calculate $e^{D_\beta^\alpha}$.

The logarithm of a tensor A_β^α is defined as the tensor B_β^α that satisfies the relation $e^{B_\beta^\alpha} = A_\beta^\alpha$. It is straightforward to show for a diagonal tensor A_β^α that

$$\log[A_\beta^\alpha] = J_\sigma^\alpha [\log D_\tau^\sigma] (J_\beta^\tau)^{-1}, \quad (\text{B5})$$

where D_β^α is the diagonal tensor whose elements are the eigenvalues of A_β^α , and J_β^α is a tensor whose columns are the eigenvectors of A_β^α .

APPENDIX C: DERIVATION OF VON NEUMANN ENTROPY

The von Neumann entropy of a monoplex network is defined by Eq. (11). Let ξ_α be the i th eigenvector ($i = 1, 2, \dots, N$) of the density tensor ρ_β^α , and let Ξ_β^α be the tensor of eigenvectors. The density tensor is defined by rescaling the combinatorial Laplacian, so it has positive diagonal entries and nonpositive off-diagonal entries. It is positive semidefinite and has non-negative eigenvalues. We diagonalize the density tensor to obtain $\rho_\beta^\alpha = \Xi_\sigma^\alpha \Lambda_\tau^\sigma (\Xi_\beta^\tau)^{-1}$, where Λ_β^α is a diagonal tensor whose elements are the eigenvalues of ρ_β^α . These eigenvalues are equal to the eigenvalues of the combinatorial Laplacian tensor rescaled by the scalar Δ^{-1} , where $\Delta = \Delta_\alpha^\alpha$ and Δ_β^α is the strength tensor. It follows that

$$\begin{aligned} \rho_\beta^\alpha \log_2[\rho_\beta^\alpha] &= (\Xi_\sigma^\alpha \Lambda_\tau^\sigma (\Xi_\beta^\tau)^{-1}) (\Xi_\gamma^\sigma [\log_2 \Lambda_\tau^\sigma] (\Xi_\alpha^\gamma)^{-1}) \\ &= \Xi_\sigma^\alpha \Lambda_\tau^\sigma [\log_2 \Lambda_\tau^\sigma] (\Xi_\alpha^\sigma)^{-1} = \Lambda_\epsilon^\sigma [\log_2 \Lambda_\epsilon^\sigma], \end{aligned} \quad (\text{C1})$$

where we have exploited the relation $\Xi_\sigma^\alpha (\Xi_\alpha^\sigma)^{-1} = \delta_\sigma^\epsilon$. We obtain Eq. (12) by multiplying both sides of Eq. (C1) by -1 .

-
- [1] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, *Complex Networks: Structure and Dynamics*, *Phys. Rep.* **424**, 175 (2006).
 - [2] M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Oxford, England, 2010).
 - [3] E. Estrada, *The Structure of Complex Networks: Theory and Applications* (Oxford University Press, Oxford, England, 2011).
 - [4] P. Holme and J. Saramäki, *Temporal Networks*, *Phys. Rep.* **519**, 97 (2012).
 - [5] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Structural Analysis in the Social Sciences Vol. 8 (Cambridge University Press, Cambridge, England, 1994).
 - [6] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, *Multilayer Networks*, [arXiv:1309.7233](https://arxiv.org/abs/1309.7233).
 - [7] J. Gao, S. V. Buldyrev, H. E. Stanley, and S. Havlin, *Networks Formed from Interdependent Networks*, *Nat. Phys.* **8**, 40 (2012).
 - [8] 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).
 - [9] M. Dickison, S. Havlin, and H. E. Stanley, *Epidemics on Interconnected Networks*, *Phys. Rev. E* **85**, 066109 (2012).
 - [10] *Temporal Networks*, edited by P. Holme and J. Saramäki (Springer, New York, 2013).

- [11] K. Lewis, J. Kaufman, M. Gonzalez, A. Wimmer, and N. Christakis, *Tastes, Ties, and Time: A New Social Network Dataset Using Facebook.com*, *Soc. Networks* **30**, 330 (2008).
- [12] E. A. Leicht and R. M. D'Souza, *Percolation on Interacting Networks*, [arXiv:0907.0894](https://arxiv.org/abs/0907.0894).
- [13] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, *Community Structure in Time-Dependent, Multiscale, and Multiplex Networks*, *Science* **328**, 876 (2010).
- [14] P. Kaluza, A. Kölzsch, M. T. Gastner, and B. Blasius, *The Complex Network of Global Cargo Ship Movements*, *J. R. Soc. Interface* **7**, 1093 (2010).
- [15] R. Criado, M. Romance, and M. Vela-Pérez, *Hyperstructures: A New Approach to Complex Systems*, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **20**, 877 (2010).
- [16] J. Gao, S. V. Buldyrev, S. Havlin, and H. E. Stanley, *Robustness of a Network of Networks*, *Phys. Rev. Lett.* **107**, 195701 (2011).
- [17] C. D. Brummitt, R. M. D'Souza, and E. A. Leicht, *Suppressing Cascades of Load in Interdependent Networks*, *Proc. Natl. Acad. Sci. U.S.A.* **109**, E680 (2012).
- [18] O. Yağan and V. Gligor, *Analysis of Complex Contagions in Random Multiplex Networks*, *Phys. Rev. E* **86**, 036103 (2012).
- [19] K.-M. Lee, J. Y. Kim, W.-K. Cho, K.-I. Goh, and I.-M. Kim, *Correlated Multiplexity and Connectivity of Multiplex Random Networks*, *New J. Phys.* **14**, 033027 (2012).
- [20] C. D. Brummitt, K.-M. Lee, and K.-I. Goh, *Multiplexity-Facilitated Cascades in Networks*, *Phys. Rev. E* **85**, 045102(R) (2012).
- [21] S. Gómez, A. Díaz-Guilera, J. Gómez-Gardeñes, C. J. Pérez-Vicente, Y. Moreno, and A. Arenas, *Diffusion Dynamics on Multiplex Networks*, *Phys. Rev. Lett.* **110**, 028701 (2013).
- [22] M. De Domenico, A. Sole, S. Gomez, and A. Arenas, *Random Walks on Multiplex Networks*, [arXiv:1306.0519](https://arxiv.org/abs/1306.0519).
- [23] G. Bianconi, *Statistical Mechanics of Multiplex Networks: Entropy and Overlap*, *Phys. Rev. E* **87**, 062806 (2013).
- [24] L. Solá, M. Romance, R. Criado, J. Flores, A. Garcia del Amo, and S. Boccaletti, *Eigenvector Centrality of Nodes in Multiplex Networks*, *Chaos* **23**, 033131 (2013).
- [25] A. Halu, R. J. Mondragon, P. Panzarasa, and G. Bianconi, *Multiplex PageRank*, [arXiv:1306.3576](https://arxiv.org/abs/1306.3576).
- [26] E. Cozzo, A. Arenas, and Y. Moreno, *Stability of Boolean Multilevel Networks*, *Phys. Rev. E* **86**, 036115 (2012).
- [27] C. Granell, S. Gómez, and A. Arenas, *Dynamical Interplay between Awareness and Epidemic Spreading in Multiplex Networks*, *Phys. Rev. Lett.* **111**, 128701 (2013).
- [28] E. Cozzo, R. A. Baños, S. Meloni, and Y. Moreno, *Contact-Based Social Contagion in Multiplex Networks*, [arXiv:1307.1656](https://arxiv.org/abs/1307.1656).
- [29] J. Gómez-Gardeñes, I. Reinares, A. Arenas, and L. M. Floría, *Evolution of Cooperation in Multiplex Networks*, *Sci. Rep.* **2**, 620 (2012).
- [30] A. Cardillo, J. Gómez-Gardeñes, M. Zanin, M. Romance, D. Papo, F. del Pozo, and S. Boccaletti, *Emergence of Network Features from Multiplexity*, *Sci. Rep.* **3**, 1344 (2013).
- [31] R. Criado, J. Flores, A. García del Amo, J. Gómez-Gardeñes, and M. Romance, *A Mathematical Model for Networks with Structures in the Mesoscale*, *Int. J. Comput. Math.* **89**, 291 (2012).
- [32] F. Radicchi and A. Arenas, *Abrupt Transition in the Structural Formation of Interconnected Networks*, *Nat. Phys.* **9**, 717 (2013).
- [33] A. Solé-Ribalta, M. De Domenico, N. E. Kouvaris, A. Díaz-Guilera, S. Gómez, and A. Arenas, *Spectral Properties of the Laplacian of Multiplex Networks*, *Phys. Rev. E* **88**, 032807 (2013).
- [34] S. E. Chang, H. A. Seligson, and R. T. Eguchi, Multidisciplinary Center for Earthquake Engineering Research (MCEER) Technical Report No. NCEER-96-0011, 1996.
- [35] R. G. Little, *Controlling Cascading Failure: Understanding the Vulnerabilities of Interconnected Infrastructures*, *J. Urban Technol.* **9**, 109 (2002).
- [36] L. M. Verbrugge, *Multiplexity in Adult Friendships*, *Social Forces* **57**, 1286 (1979).
- [37] J. S. Coleman, *Social Capital in the Creation of Human Capital*, *Am. J. Sociology* **94**, S95 (1988).
- [38] D. Krackhardt, *Cognitive Social Structures*, *Soc. Networks* **9**, 109 (1987).
- [39] P. Pattison and S. Wasserman, *Logit Models and Logistic Regressions for Social Networks: II. Multivariate Relations*, *Brit. J. Math. Stat. Psychol.* **52**, 169 (1999).
- [40] R. Abraham, J. E. Marsden, and T. Ratiu, *Manifolds, Tensor Analysis, and Applications*, Applied Mathematical Sciences Vol. 75 (Springer-Verlag, New York, 1988), 2nd ed.
- [41] D. S. Bassett, M. A. Porter, N. F. Wymbs, S. T. Grafton, J. M. Carlson, and P. J. Mucha, *Robust Detection of Dynamic Community Structure in Networks*, *Chaos* **23**, 013142 (2013).
- [42] M. M. G. Ricci and T. Levi-Civita, *Méthodes de calcul différentiel absolu et leurs applications*, *Math. Ann.* **54**, 125 (1900).
- [43] B. Min and K.-I. Goh, *Layer-Crossing Overhead and Information Spreading in Multiplex Social Networks*, [arXiv:1307.2967](https://arxiv.org/abs/1307.2967).
- [44] D. Horne (private communication).
- [45] T. G. Kolda and B. W. Bader, *Tensor Decompositions and Applications*, *SIAM Rev.* **51**, 455 (2009).
- [46] P. Ronhovde and Z. Nussinov, *Multiresolution Community Detection for Megascale Networks by Information-Based Replica Correlations*, *Phys. Rev. E* **80**, 016109 (2009).
- [47] E. Cozzo, M. Kivelä, M. De Domenico, A. Solé-Ribalta, A. Arenas, S. Gómez, M. A. Porter, and Y. Moreno, *Clustering Coefficients in Multiplex Networks*, [arXiv:1307.6780](https://arxiv.org/abs/1307.6780).
- [48] S.-H. Yook, H. Jeong, A.-L. Barabási, and Y. Tu, *Weighted Evolving Networks*, *Phys. Rev. Lett.* **86**, 5835 (2001).
- [49] A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani, *The Architecture of Complex Weighted Networks*, *Proc. Natl. Acad. Sci. U.S.A.* **101**, 3747 (2004).
- [50] M. E. J. Newman, *Properties of Highly Clustered Networks*, *Phys. Rev. E* **68**, 026121 (2003).
- [51] D. J. Watts and S. H. Strogatz, *Collective Dynamics of "Small-World" Networks*, *Nature (London)* **393**, 440 (1998).

- [52] J. Saramäki, M. Kivelä, J.-P. Onnela, K. Kaski, and J. Kertész, *Generalizations of the Clustering Coefficient to Weighted Complex Networks*, *Phys. Rev. E* **75**, 027105 (2007).
- [53] S.E. Ahnert, D. Garlaschelli, T.M.A. Fink, and G. Caldarelli, *Ensemble Approach to the Analysis of Weighted Networks*, *Phys. Rev. E* **76**, 016101 (2007).
- [54] T. Opsahl and P. Panzarasa, *Clustering in Weighted Networks*, *Soc. Networks* **31**, 155 (2009).
- [55] G. Fagiolo, *Clustering in Complex Directed Networks*, *Phys. Rev. E* **76**, 026107 (2007).
- [56] P. Bonacich, *Technique for Analyzing Overlapping Memberships*, *Sociol. Method.* **4**, 176 (1972).
- [57] P. Bonacich, *Factoring and Weighing Approaches to Status Scores and Clique Identification*, *J. Math. Sociol.* **2**, 113 (1972).
- [58] L. Katz, *A New Status Index Derived from Sociometric Analysis*, *Psychometrika* **18**, 39 (1953).
- [59] M.A. Porter, J.-P. Onnela, and P.J. Mucha, *Communities in Networks*, *Not. Am. Math. Soc.* **56**, 1082 (2009) [<http://www.ams.org/notices/200909/rtx090901082p.pdf>].
- [60] S. Fortunato, *Community Detection in Graphs*, *Phys. Rep.* **486**, 75 (2010).
- [61] M.E.J. Newman and M. Girvan, *Finding and Evaluating Community Structure in Networks*, *Phys. Rev. E* **69**, 026113 (2004).
- [62] M.E.J. Newman, *Modularity and Community Structure in Networks*, *Proc. Natl. Acad. Sci. U.S.A.* **103**, 8577 (2006).
- [63] M.E.J. Newman, *Finding Community Structure in Networks Using the Eigenvectors of Matrices*, *Phys. Rev. E* **74**, 036104 (2006).
- [64] R.V. Solé and S. Valverde, in *Complex Networks* (Springer, New York, 2004), p. 189–207.
- [65] G. Bianconi, *The Entropy of Randomized Network Ensembles*, *Europhys. Lett.* **81**, 28005 (2008).
- [66] K. Anand and G. Bianconi, *Entropy Measures for Networks: Toward an Information Theory of Complex Topologies*, *Phys. Rev. E* **80**, 045102(R) (2009).
- [67] G. Bianconi, *Entropy of Network Ensembles*, *Phys. Rev. E* **79**, 036114 (2009).
- [68] S. Johnson, J.J. Torres, J. Marro, and M.A. Muñoz, *Entropic Origin of Disassortativity in Complex Networks*, *Phys. Rev. Lett.* **104**, 108702 (2010).
- [69] S.L. Braunstein, S. Ghosh, and S. Severini, *The Laplacian of a Graph as a Density Matrix: A Basic Combinatorial Approach to Separability of Mixed States*, *Ann. Combin.* **10**, 291 (2006).
- [70] B. Mohar, in *Graph Theory, Combinatorics, and Applications* (Wiley, Hoboken, NJ, 1991), Vol. 2, p. 871–898.
- [71] C.D. Godsil and G. Royle, *Algebraic Graph Theory* (Springer-Verlag, New York, 2001), Vol. 207.
- [72] F.R.K. Chung, *Spectral Graph Theory* (American Mathematical Society, Providence, RI, 1997), 2nd ed.
- [73] A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou, *Synchronization in Complex Networks*, *Phys. Rep.* **469**, 93 (2008).
- [74] J.D. Noh and H. Rieger, *Random Walks on Complex Networks*, *Phys. Rev. Lett.* **92**, 118701 (2004).
- [75] S. Brin and L. Page, in *Proceedings of the Seventh International World-Wide Web Conference (WWW 1998)* (1998), <http://ilpubs.stanford.edu:8090/361/>.
- [76] T. Callaghan, P.J. Mucha, and M.A. Porter, *Random Walker Ranking for NCAA Division I-A Football*, *Am. Math. Mon.* **114**, 761 (2007) [<http://www.maa.org/publications/periodicals/american-mathematical-monthly/american-mathematical-monthly-november-2007>].
- [77] G.M. Viswanathan, S.V. Buldyrev, S. Havlin, M.G.E. da Luz, E.P. Raposo, and H.E. Stanley, *Optimizing the Success of Random Searches*, *Nature (London)* **401**, 911 (1999).
- [78] S.-J. Yang, *Exploring Complex Networks by Walking on Them*, *Phys. Rev. E* **71**, 016107 (2005).
- [79] L. da Fontoura Costa and G. Travieso, *Exploring Complex Networks through Random Walks*, *Phys. Rev. E* **75**, 016102 (2007).
- [80] H.D. Rozenfeld, J.E. Kirk, E.M. Bollt, and D. ben-Avraham, *Statistics of Cycles: How Loopy Is Your Network?*, *J. Phys. A* **38**, 4589 (2005).
- [81] D. Gfeller and P. De Los Rios, *Spectral Coarse Graining of Complex Networks*, *Phys. Rev. Lett.* **99**, 038701 (2007).
- [82] M. Rosvall and C. T. Bergstrom, *An Information-Theoretic Framework for Resolving Community Structure in Complex Networks*, *Proc. Natl. Acad. Sci. U.S.A.* **104**, 7327 (2007).
- [83] R. Lambiotte, J.-C. Delvenne, and M. Barahona, *Laplacian Dynamics and Multiscale Modular Structure in Networks*, [arXiv:0812.1770](https://arxiv.org/abs/0812.1770).
- [84] N.F. Wymbs, D.S. Bassett, P.J. Mucha, M.A. Porter, and S.T. Grafton, *Differential Recruitment of the Sensorimotor Putamen and Frontoparietal Cortex During Motor Chunking in Humans*, *Neuron* **74**, 936 (2012).
- [85] M.W. Hirsch and S. Smale, *Differential Equations, Dynamical Systems, and Linear Algebra* (Academic, Waltham, MA, 1974).