

Structural position vectors and symmetries in complex networks

F SCI

Cite as: Chaos 32, 093132 (2022); <https://doi.org/10.1063/5.0107583>

Submitted: 06 July 2022 • Accepted: 11 August 2022 • Published Online: 22 September 2022

 Yong-Shang Long,  Zheng-Meng Zhai,  Ming Tang, et al.

COLLECTIONS

 This paper was selected as Featured

 This paper was selected as Scilight



View Online



Export Citation



CrossMark

APL Machine Learning

Open, quality research for the networking communities

Now Open for Submissions

LEARN MORE

AIP
Publishing

Structural position vectors and symmetries in complex networks



Cite as: Chaos 32, 093132 (2022); doi: 10.1063/5.0107583

Submitted: 6 July 2022 · Accepted: 11 August 2022 ·

Published Online: 22 September 2022



View Online



Export Citation



CrossMark

Yong-Shang Long,¹ Zheng-Meng Zhai,² Ming Tang,^{1,3,a)} Ying Liu,⁴ and Ying-Cheng Lai^{2,b)}

AFFILIATIONS

¹School of Physics and Electronic Science, East China Normal University, Shanghai 200241, China

²School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, Arizona 85287, USA

³Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University, Shanghai 200241, China

⁴School of Computer Science, Southwest Petroleum University, Chengdu 610500, China

^{a)} Author to whom correspondence should be addressed: tangminghan007@gmail.com

^{b)} Electronic mail: Ying-Cheng.Lai@asu.edu

ABSTRACT

Symmetries, due to their fundamental importance to dynamical processes on networks, have attracted a great deal of current research. Finding all symmetric nodes in large complex networks typically relies on automorphism groups from algebraic-group theory, which are solvable in quasipolynomial time. We articulate a conceptually appealing and computationally extremely efficient approach to finding and characterizing all symmetric nodes by introducing a structural position vector (SPV) for each node in networks. We establish the mathematical result that symmetric nodes must have the same SPV value and demonstrate, using six representative complex networks from the real world, that all symmetric nodes in these networks can be found in linear time. Furthermore, the SPVs not only characterize the similarity of nodes but also quantify the nodal influences in propagation dynamics. A caveat is that the proved mathematical result relating the SPV values to nodal symmetries is not sufficient; i.e., nodes having the same SPV values may not be symmetric, which arises in regular networks or networks with a dominant regular component. We point out with an analysis that this caveat is, in fact, shared by the known existing approaches to finding symmetric nodes in the literature. We further argue, with the aid of a mathematical analysis, that our SPV method is generally effective for finding the symmetric nodes in real-world networks that typically do not have a dominant regular component. Our SPV-based framework, therefore, provides a physically intuitive and computationally efficient way to uncover, understand, and exploit symmetric structures in complex networks arising from real-world applications.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0107583>

Symmetric nodes play an important role not only in the structural characterization of complex networks but also in the dynamical processes on the network. For example, the extensively studied phenomenon of cluster synchronization relies on network symmetries. From a computational point of view, the traditional methods based on the algebraic-group theory are NP-hard. To find all the symmetric nodes in large, real-world complex networks in a computationally efficient manner and to characterize the structural similarity of nodes constitute an active area of current research. This article introduces a concept, the nodal structural position vector (SPV), to address the two problems. The intuitive meaning of SPV is that, for nodes that are completely symmetric to each other in the network, their structural “positions” should be identical. This result is established rigorously. In particular, based on the algebraic-group theory,

it can be mathematically proved that a set of symmetric nodes must have the same SPV values—a necessary condition. Extensive simulations of six large complex networks from the real world demonstrate that their known symmetric structures can be accurately found based on the nodal SPV values in linear time. An SPV-based index is introduced to describe the structural similarity among the nodes, leading to an effective algorithm for coarse-graining the network. An SPV-based centrality measure is further defined to quantify the role played by nodes in epidemic spreading dynamics. A caveat of the SPV method is that it will give the wrong result for asymmetric regular networks or networks containing a dominant regular component but without any symmetry, as the SPV values are necessarily equal among all the regular nodes. It is argued that this caveat is, in fact, commonly shared by the existing methods to find the symmetric nodes.

However, it is argued with the aid of a mathematical analysis that the SPV method is generally effective for finding the symmetric nodes in real-world networks that typically do not have a dominant regular component. Taken together, not only does the SPV framework provide a computationally extremely efficient way to find all symmetric nodes in typical complex networks, the SPVs can also be exploited to characterize the structural similarity of nodes and as an effective and physically meaningful measure to quantify the role played by nodes in dynamical processes on the network.

I. INTRODUCTION

Symmetries are ubiquitous in natural systems. In physics, the existence of a continuous symmetry implies the conservation of a physical quantity and, as such, a great deal can be learned about the system without the need for analyzing the system details. For example, if a physical system is invariant with respect to translation in time, the energy of the system is conserved. The same principle has been applied to complex networks for identifying and understanding intricate dynamical phenomena, such as cluster synchronization,¹⁻⁷ which otherwise would be difficult to analyze. Given a complex network, a symmetry implies the existence of a group or a cluster of structurally completely equivalent nodes. For example, in a star network, all leaf nodes are symmetric to each other. Mathematically, that two nodes u and v are symmetric with respect to each other requires the existence of a permutation π such that $\{u, v\}$ is an edge in the network if and only if $\{\pi(u), \pi(v)\}$ is also an edge. When a dynamical process occurs on the network, the nodes in a symmetric cluster, due to their complete equivalence, tend to be more readily synchronized among themselves than with nodes outside the cluster. A purely random network, such as the Erdős-Rényi network,⁸ typically has zero symmetry in the sense that the probability is zero for any two nodes to be equivalent in terms of their connection structure in the network. However, networks in the real world are not purely random, but in fact, they often can possess a large number of symmetric nodes^{2,9-11} that affect not only properties of the network, such as spectrum, redundancy, and robustness, but also the various dynamical processes on the network. For example, in the brain network, symmetries play an important role in the network dynamics, such as remote synchronization¹² and cluster synchronization.³ For network computations, symmetries can be exploited to coarse-grain and reduce the dimensionality of the network, merge symmetric nodes, and generate the so-called entropy graph of the original network¹³ with reduced computational complexity.^{14,15} In complex networks, the mechanisms by which symmetries arise include replicative growth, such as duplication,¹⁶ evolution from basic principles,¹¹ and functional optimization.¹⁷

To take advantage of the network symmetries, an essential task is to find all the symmetric nodes. Finding the symmetric structures of networks is a classical mathematical problem that is NP hard.¹² Previously, a number of methods were proposed for finding the symmetric nodes in networks. For example, the properties of the eigenvectors associated with the degenerate eigenvalues of the network adjacency matrix were exploited to pinpoint the structural

symmetries in a network.^{18,19} However, it is not straightforward to determine the symmetric nodes according to the degenerate eigenvectors, especially those associated with the complex symmetric motifs. Another difficulty is the high computational cost for large networks because of the need to calculate all the eigenvalues and eigenvectors of the adjacency matrix. A method based on cluster synchronization was articulated,^{3,12} but the ability to yield the symmetric nodes depends sensitively on the coupling strength among the nodes. A combinatorial algorithm incorporating the concept of minimally balanced coloring was proposed to locate the symmetric nodes.²⁰ Due to the iterative nature of the algorithm, not only is it necessary to find the classes to which all nodes belong, but the amount of the information provided to the nodes during the iteration process would increase continuously, rendering high the associated computational complexity. In fact, there were traditional methods in this area based on the algebraic-group theory, such as dividing the equivalence class by backtracking search guided by the coloring theory.^{21,22} For those methods, NAUTY stood out as the most efficient algorithm—it can identify all the symmetric structures of a network in polynomial time.²³ In another line of work, symmetries were exploited to eliminate the redundancy of the network through a compression algorithm,¹⁵ where the software *saucy3* based on the NAUTY algorithm was used to find the symmetric structure of the network.^{21,23}

In this paper, we introduce the concept of a nodal structural position vector (SPV) and establish a mathematical theory to demonstrate: (1) SPVs provide a computationally extremely efficient way to find all symmetric nodes in typical complex networks and (2) SPVs can characterize the structural similarity of nodes and serve as an effective and physically meaningful measure to quantify the effects of the nodes in dynamical processes on complex networks. Utilizing the algebraic-group theory, we prove that having the same SPV values for a set of nodes is the necessary condition that they are symmetric with respect to each other. That is, if a set of nodes are symmetric, then their SPV values must be equal. We demonstrate, using six large complex networks from the real world, that the known symmetric structures can be accurately found based on the nodal SPV values. In fact, since the computations required involve only a small number of multiplications between the adjacency matrix (sparse matrix) and a vector, all the symmetric nodes can be found in linear time. We then articulate an SPV-based index to describe the structural similarity among the nodes, which provides an effective algorithm for coarse-graining the network. We further introduce a centrality measure to quantify the nodal spreading influence—the role played by nodes in epidemic spreading dynamics, which is validated using real-world networks. Because our mathematical result relating nodal symmetries to SPV values is not sufficient, there are situations where nodes having the same SPV values are not symmetric, which arises in regular networks or in networks with a dominant regular component. We argue that this caveat is, in fact, shared by the known existing approaches to finding symmetric nodes. With the aid of a mathematical result that we prove as a prerequisite (i.e., if there is a permutation containing two nodes such that all the eigenvectors before and after the permutation are equal or opposite, then the two nodes are symmetric), we argue that our SPV method is generally effective for finding the symmetric nodes in real-world

networks that typically do not have a dominant regular component.

II. THEORY OF STRUCTURAL POSITION VECTORS AND THEIR RELATIONS TO SYMMETRIES

A. Definition of SPVs

Mathematically, the existence of a symmetry or symmetric nodes in the network requires a permutation π such that $\{u, v\}$ is an edge in the network if and only if $\{\pi(u), \pi(v)\}$ is also an edge, where u and v denote a pair of nodes. Given a network, finding the identical nodes is the goal of our SPV algorithm because, for nodes that are completely symmetric to each other in the network, their structural positions are identical. However, it is not known *a priori* which nodes are identical. The steps for finding the SPVs for each and every node in the network are as follows. First, we set an initial or the zeroth-order structural position $l_i^0 = 1$ for each node in the network, where i is the nodal index. For all nodes in the network, their zeroth-order structural positions can be represented by a vector: $\mathbf{L}^0 = [1, 1, \dots, 1]^T$. That is, initially, we disregard the edges and assume that all nodes have the same structural positions. To take into account the edges, we note that the structural position of a node is related to the structural positions of its n th-order neighbors, which is determined by the n th power of the network adjacency matrix \mathcal{A}^n . Multiplying \mathbf{L}^0 from the left by \mathcal{A}^n , we get

$$\mathbf{L}^n = \mathcal{A}^n \cdot \mathbf{L}^0,$$

where \mathbf{L}^n ($n = 1, \dots, \infty$) is a vector containing information about the n th-order structural positions of all nodes in the network, whose i th component is given by

$$L_i^n = \sum_{j=1}^N [\mathcal{A}^n]_{ij} L_j^0,$$

which is the sum of the number of paths of length n to node i from all nodes in the network. Specifically, the first-order structural position of node i is its degree and the ∞ th order is nothing but its eigenvector centrality, the i th component of the eigenvector associated with the largest eigenvalue of the network adjacency matrix.

It is worth emphasizing that the vectors \mathbf{L}^n ($n = 1, \dots, \infty$) contain information about the structural positions of all nodes, but it is not the SPV that is node-specific. To define the SPV for node i , we use the i th component in \mathbf{L}^n (for $n = 1, \dots, \infty$) to obtain $\mathbf{S}_i \equiv (L_i^1, L_i^2, \dots, L_i^\infty)$. For a finite network of size N , the longest path from any node to node i through non-repeated nodes must not be greater than N , and the dimensionality of the adjacency relationship between a node and the other nodes in the adjacency matrix fully characterizing the network structure is also N . These considerations lead us to set the dimensionality of vector \mathbf{S}_i to be N (see below for a proof). We, therefore, have $\mathbf{S}_i \equiv (L_i^1, L_i^2, \dots, L_i^N)$.

Making use of the algebraic-group theory, we can prove that equal SPV values of nodes i and j ,

$$(L_i^1, L_i^2, \dots, L_i^N) = (L_j^1, L_j^2, \dots, L_j^N),$$

is a necessary condition for nodes i and j to be symmetric. That is, if nodes i and j are symmetric, then their SPV values must be

equal. Before we present a sketch of the proof in Sec. II C, we present a physical and intuitive explanation of this mathematical result through toy network models, which suggests that it offers the possibility of finding all the symmetric nodes in a network in a computationally efficient way.

B. Physical understanding of SPVs and its potential use in finding symmetric nodes

Figure 1 presents a simple example to illustrate how the symmetric nodes can be found using the SPVs. The toy network has eight symmetric motifs M_i ($i = 1, 2, \dots, 8$), as shown in Fig. 1(a). In motif M_1 , there are two sets of nodes that are symmetric to each other: $\{28, 29, 30, 31\}$ and $\{26, 27\}$. A set of symmetric nodes form an orbit, e.g., nodes 26 and 27. Motif M_1 , therefore, consists of two orbits with two and four symmetric nodes, respectively, motif M_3 has one orbit of three symmetric nodes, and so on. We say that M_1 and M_3 are represented by the orbits $O_2 \xrightarrow{2} O_4$ and O_3^* , respectively (Appendix A). The orbital representations of the eight symmetric motifs in Fig. 1(a) and their corresponding geometric factors are listed in Table II in Appendix A. The leftmost column in Fig. 1(b) is a diagram of five blocks determined according to the vector \mathbf{L}^1 , where all the nodes in a block have identical component values in \mathbf{L}^1 . Similarly, the columns under \mathbf{C}_2 and \mathbf{C}_3 have 9 and 11 blocks, respectively. The middle column represents the set of intersection between the sets under \mathbf{C}_1 and \mathbf{C}_2 , which is done according to the criterion that all nodes in a block must have identical component values in both \mathbf{L}^1 and \mathbf{L}^2 , although the common value in \mathbf{L}^1 may not equal that in \mathbf{L}^2 . The rightmost column in Fig. 1(b) consists of distinct blocks obtained through the intersection among the sets corresponding to \mathbf{C}_1 , \mathbf{C}_2 , and \mathbf{C}_3 , which already contain all the symmetric nodes of the network as specified in Fig. 1(a). This means that, for this toy network of 36 nodes, three iterations of the initial vector \mathbf{L}^0 already suffice to yield all the symmetric nodes. Note that, while some symmetric nodes have identical row vectors in the adjacency matrix, such as $\{6, 7, 8\}$, $\{18, 19\}$, there are also symmetric nodes with non-identical row vectors, such as $\{26, 27\}$, $\{33, 34\}$. Our SPV algorithm can find both types of symmetric nodes accurately and efficiently.

The workings of our SPV algorithm can be explicitly illustrated using an even simpler toy network of ten nodes, as shown in Fig. 2. Let

$$\mathbf{L}^0 = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1].$$

We have

$$\mathbf{L}^1 = \mathcal{A} \cdot \mathbf{L}^0 = [4, 3, 3, 1, 1, 1, 1, 2, 2, 2],$$

based on which we divide the nodes into four categories,

$$\{1\}, \{2, 3\}, \{4, 5, 6, 7\}, \{8, 9, 10\}.$$

We next obtain

$$\mathbf{L}^2 = \mathcal{A} \cdot \mathbf{L}^1 = [10, 6, 6, 3, 3, 3, 3, 6, 6, 4],$$

based on which the nodes in the network can be divided into four different categories,

$$\{1\}, \{2, 3, 8, 9\}, \{4, 5, 6, 7\}, \{10\}.$$

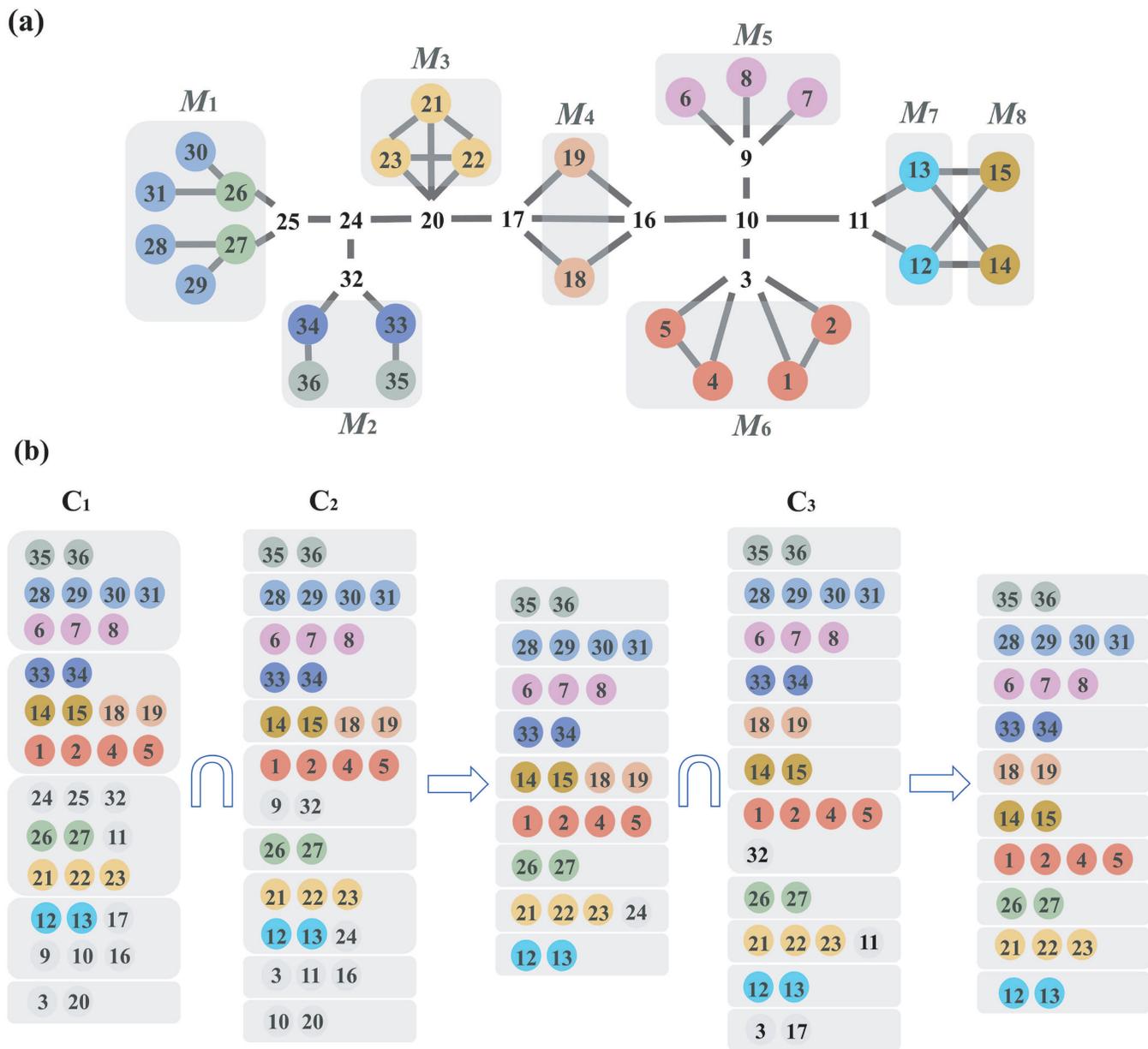


FIG. 1. Schematic representation of finding the symmetric nodes using the SPVs. (a) A network containing eight symmetric motifs. (b) C_n represents the sets of nodal classification by the single n th-order vector L^n , for $n = 1, 2, 3$, for each node in (a). The intersections of the nodal classification yield the symmetric nodes.

Taking the intersection of the two sets of nodes, we get the symmetric nodes,

$$\{2, 3\}, \{4, 5, 6, 7\}, \{8, 9\}.$$

It can be seen that not only can the symmetric nodes with equal row vectors in the adjacency matrix be identified (i.e., $\{4,5\}$, $\{8,9\}$, $\{6,7\}$), but the symmetric nodes with unequal row vectors can also be found: $\{2,3\}$, $\{4,6\}$, $\{5,7\}$ or $\{4,7\}$, $\{5,6\}$.

C. Sketch of the proof of main mathematical results

The main mathematical results of this paper are two: (1) Nodes x and y are symmetric if there is a permutation containing the two nodes such that all eigenvectors before and after the permutation are equal or opposite. (2) Nodes having equal SPV values constitute stronger necessary conditions for them being symmetric to each other than redundant eigenvectors; i.e., if they are symmetric

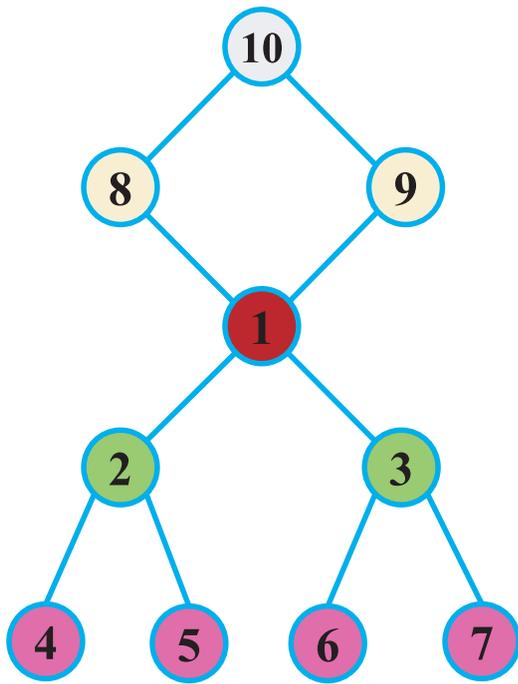


FIG. 2. A toy network of ten nodes, which contains three sets of symmetric nodes.

nodes, then their SPV values must be equal. Also, if their SPV values are equal, there will be redundant eigenvectors about them in the eigenvectors of the adjacency matrix corresponding to the network.

Mathematically, the existence of symmetric nodes in the network stipulates an automorphism π satisfying the property that $\{u, v\}$ is an edge in the network if and only if $\{\pi(u), \pi(v)\}$ is also an edge. All the automorphisms of a network constitute an automorphism group $G(\pi)$.

First, we prove two lemmas.

Lemma 1. Permutation π is an automorphism if and only if $\mathcal{P} \cdot \mathcal{A} = \mathcal{A} \cdot \mathcal{P}$, where \mathcal{A} , π , and $\mathcal{P} = (p_{ij})$ are the adjacency matrix, permutation, and permutation matrix of the network, respectively. When nodes i and j have a permutation relation with each other: $v_i = \pi(v_j)$, the corresponding elements in the permutation matrix are $p_{ij} = p_{ji} = 1$, and all other elements are zero.

Proof. Assume $v_h = \pi(v_i)$, $v_k = \pi(v_j)$, then

$$\begin{cases} (\mathcal{P} \cdot \mathcal{A})_{hj} = \sum p_{hi} a_{ij} = a_{ij}, \\ (\mathcal{A} \cdot \mathcal{P})_{hj} = \sum a_{hi} p_{lj} = a_{hk}; \end{cases} \quad (1)$$

i.e., $\mathcal{P} \cdot \mathcal{A} = \mathcal{A} \cdot \mathcal{P}$ if and only if $a_{ij} = a_{hk}$. That is, permutation π is an automorphism. \square

Lemma 2. (A sufficient condition for nodal symmetry): If there is a permutation matrix \mathcal{P} such that $\mathcal{P} \cdot \mathbf{x} = \pm \mathbf{x}$, where \mathbf{x} is an eigenvector of the adjacency matrix of the network, then the permutation corresponding to \mathcal{P} is an automorphism.

Proof. Let $\mathcal{A} \cdot \mathbf{x} = \lambda \mathbf{x}$, where λ and \mathbf{x} are the eigenvalue and the associated eigenvector of the network adjacency matrix \mathcal{A} ,

respectively. Assume $\mathcal{P} \cdot \mathbf{x} = \pm \mathbf{x}$. Since the permutation matrix \mathcal{P} is derived from the identity matrix through a series of elementary transformations, \mathcal{P} is invertible. Left multiplying both sides of the characteristic equation $\mathcal{A} \cdot \mathbf{x} = \lambda \mathbf{x}$ by \mathcal{P} , we get $\mathcal{P} \cdot \mathcal{A} \cdot \mathbf{x} = \lambda \mathcal{P} \cdot \mathbf{x}$. With $\mathcal{P} \cdot \mathbf{x} = \pm \mathbf{x}$, we get

$$\mathcal{P} \cdot \mathcal{A} \cdot \mathbf{x} = \pm \lambda \mathbf{x} = \pm \mathcal{A} \cdot \mathbf{x} = \mathcal{A} \cdot \mathcal{P} \cdot \mathbf{x}, \quad (2)$$

which gives $(\mathcal{P} \cdot \mathcal{A} - \mathcal{A} \cdot \mathcal{P}) \cdot \mathbf{x} = 0$. Because \mathcal{A} is an $N \times N$ symmetric matrix, there must be N linearly independent eigenvectors, denoted as $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$. The matrix $\mathcal{X} \equiv (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ has a full rank. We, therefore, have $(\mathcal{P} \cdot \mathcal{A} - \mathcal{A} \cdot \mathcal{P}) \cdot \mathcal{X} = 0$. Expressing the matrix $(\mathcal{P} \cdot \mathcal{A} - \mathcal{A} \cdot \mathcal{P})$ in a row form, we get

$$(\mathcal{P} \cdot \mathcal{A} - \mathcal{A} \cdot \mathcal{P}) \cdot \mathcal{X} = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_N \end{bmatrix} \mathcal{X} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (3)$$

Since matrix \mathcal{X} has a full rank, the equations $\phi_i \mathcal{X} = 0$ have trivial solutions only: $\phi_i = 0$. We, therefore, have $\mathcal{P} \cdot \mathcal{A} = \mathcal{A} \cdot \mathcal{P}$. According to Lemma 1, the permutation corresponding to the permutation matrix \mathcal{P} is an automorphism. \square

We now sketch the steps involved in the proof of the stronger necessary condition for nodal symmetry than redundant eigenvectors: if two nodes are symmetric, their SPV values must be equal. Also, if SPVs of two nodes are equal, there will be redundant eigenvectors about them in the eigenvectors of the adjacency matrix corresponding to the network. (A detailed proof can be found in Appendix B.)

We first prove that the SPV equality is a necessary condition for node symmetry. Assuming that nodes x and y are symmetrical, it can be known from Lemma 1 that the number of neighbors of nodes x and y is equal (i.e., the degrees of nodes x and y are equal), and there must be one-to-one neighbors symmetrical to each other. Suppose the neighbors of nodes x and y are x_i, y_i ($i = 1, 2, \dots, r$), respectively. Without loss of generality, let x_i, y_i ($i = 1, 2, \dots, r$) be symmetrical to each other. We use mathematical induction to prove that if nodes x and y are symmetrical, then $L_x^n = L_y^n$, $n = 1, 2, \dots, N$. By definition of SPV, we have

$$L_x^n = \sum_{j=1}^N [\mathcal{A}^n]_{xj} L_j^0 = \sum_{j=1}^N \mathcal{A}_{xj} L_j^{n-1}. \quad (4)$$

Obviously, when $n = 1$, L_x^n and L_y^n represent the degrees of nodes x and y , respectively, then $L_x^n = L_y^n$. Similarly, $L_{x_i}^n = L_{y_i}^n$, $i = 1, 2, \dots, r$.

Assuming that when $n = N - 1$, the $(N - 1)$ th-order structural positions of symmetric nodes are equal; i.e., $L_x^{N-1} = L_y^{N-1}$ and $L_{x_i}^{N-1} = L_{y_i}^{N-1}$, $i = 1, 2, \dots, r$. When $n = N$, by Eq. (4),

$$L_x^N = \sum_{i=1}^r \mathcal{A}_{x,x_i} L_{x_i}^{N-1} \quad (5)$$

and

$$L_y^N = \sum_{i=1}^r \mathcal{A}_{y,y_i} L_{y_i}^{N-1}, \quad (6)$$

where $\mathcal{A}_{x,x_i} = \mathcal{A}_{y,y_i} = 1$ and $L_{x_i}^{N-1} = L_{y_i}^{N-1}$. Then, $L_x^N = L_y^N$. It can be known from mathematical induction that if nodes x and y are symmetrical, then $L_x^n = L_y^n$, $n = 1, 2, \dots, N$. That is, the SPV equality is a necessary condition for node symmetry.

Now, we prove if the SPVs of two nodes are equal, there will be redundant eigenvectors about them in the eigenvectors of the adjacency matrix corresponding to the network.

Assume that the SPVs of the nodes x and y are equal: $L_x^n = L_y^n$ for $n = 1, 2, \dots, N$. Because \mathcal{A} is a symmetric matrix, it must have N linearly independent eigenvectors. The initial vector $\mathbf{L}^0 = (1, 1, \dots, 1)^T$ can then be linearly represented by the eigenvectors $\boldsymbol{\eta}_i$ for $i = 1, 2, \dots, N$: $\mathbf{L}^0 = \sum_{i=1}^N t_i \boldsymbol{\eta}_i$. From $\mathbf{L}^n = \mathcal{A}^n \cdot \mathbf{L}^0$, we get

$$\begin{cases} \mathbf{L}^1 = \mathcal{A} \cdot \mathbf{L}^0 = \mathcal{A}(\sum_{i=1}^N t_i \boldsymbol{\eta}_i) = \sum_{i=1}^N t_i \lambda_i \boldsymbol{\eta}_i, \\ \mathbf{L}^2 = \mathcal{A}^2 \cdot \mathbf{L}^0 = \mathcal{A}^2(\sum_{i=1}^N t_i \boldsymbol{\eta}_i) = \sum_{i=1}^N t_i \lambda_i^2 \boldsymbol{\eta}_i, \\ \vdots \\ \mathbf{L}^N = \mathcal{A}^N \cdot \mathbf{L}^0 = \mathcal{A}^N(\sum_{i=1}^N t_i \boldsymbol{\eta}_i) = \sum_{i=1}^N t_i \lambda_i^N \boldsymbol{\eta}_i. \end{cases} \quad (7)$$

The equality $L_x^n = L_y^n$ for $n = 1, 2, \dots, N$ gives

$$\begin{cases} \sum_{i=1}^N t_i \lambda_i \eta_{i,x} = \sum_{i=1}^N t_i \lambda_i \eta_{i,y}, \\ \sum_{i=1}^N t_i \lambda_i^2 \eta_{i,x} = \sum_{i=1}^N t_i \lambda_i^2 \eta_{i,y}, \\ \vdots \\ \sum_{i=1}^N t_i \lambda_i^N \eta_{i,x} = \sum_{i=1}^N t_i \lambda_i^N \eta_{i,y} \end{cases} \quad (8)$$

or

$$\begin{cases} \sum_{i=1}^N t_i \lambda_i (\eta_{i,x} - \eta_{i,y}) = 0, \\ \sum_{i=1}^N t_i \lambda_i^2 (\eta_{i,x} - \eta_{i,y}) = 0, \\ \vdots \\ \sum_{i=1}^N t_i \lambda_i^N (\eta_{i,x} - \eta_{i,y}) = 0. \end{cases} \quad (9)$$

The determinant of this set of linear equations in $(\eta_{i,x} - \eta_{i,y})$ is

$$C = \begin{vmatrix} t_1 \lambda_1 & t_2 \lambda_2 & \dots & t_N \lambda_N \\ t_1 \lambda_1^2 & t_2 \lambda_2^2 & \dots & t_N \lambda_N^2 \\ t_1 \lambda_1^3 & t_2 \lambda_2^3 & \dots & t_N \lambda_N^3 \\ \vdots & \vdots & \ddots & \vdots \\ t_1 \lambda_1^N & t_2 \lambda_2^N & \dots & t_N \lambda_N^N \end{vmatrix} = t_1 t_2 \dots t_N \lambda_1 \lambda_2 \dots \lambda_N \begin{vmatrix} 1 & 1 & \dots & 1 \\ \lambda_1 & \lambda_2 & \dots & \lambda_N \\ \lambda_1^2 & \lambda_2^2 & \dots & \lambda_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_1^{N-1} & \lambda_2^{N-1} & \dots & \lambda_N^{N-1} \end{vmatrix}, \quad (10)$$

which is the Vandermonde determinant. We have $C = t_1 t_2 \dots t_N \lambda_1 \lambda_2 \dots \lambda_N \prod_{1 \leq j < i \leq N} (\lambda_i - \lambda_j)$. If all eigenvalues of \mathcal{A} are simple, zero is not its eigenvalue and $t_i \neq 0$ for $i = 1, 2, \dots, N$, then we have $C \neq 0$. The only solution of $(\eta_{i,x} - \eta_{i,y})$ is zero; therefore, $\eta_{i,x} = \eta_{i,y}$. The permutation of nodes x and y implies that all eigenvalues of \mathcal{A} must satisfy $\mathcal{P} \cdot \boldsymbol{\eta} = \boldsymbol{\eta}$. By Lemma 2, the permutation corresponding to the permutation matrix \mathcal{P} is an automorphism; therefore, the nodes x and y are symmetric to each other.

However, when the adjacency matrix \mathcal{A} has zero or repeated eigenvalues or there is k ($k \in \{1, 2, \dots, N\}$) such that $t_k = 0$, the nodes x and y cannot be deduced to be symmetric to each other. However, the adjacency matrix \mathcal{A} has redundant eigenvector components about nodes x and y in that the sum of their eigencomponents is zero and the eigencomponents of remaining nodes are all zero (Appendix B). A redundant eigenvector is a necessary condition for node symmetry^{18,19} and is often used to find symmetric nodes in the network.¹⁹ We conclude that SPV equality is a stronger necessary condition for nodal symmetry than redundant eigenvectors. In addition, the computational complexity of an SPV method is lower than that of a redundant eigenvector method.

III. APPLICATIONS OF AN SPV METHOD TO REAL-WORLD NETWORKS

We apply the SPV theory to finding the symmetric motifs in real-world networks. Since the SPV method is only a necessary but not sufficient condition for nodal symmetries, some additional criterion should be used to guarantee that the SPV solutions do indeed give the symmetric nodes in the network. For example, if the underlying network contains a large regular component, then the SPV algorithm would stipulate that the nodes in this component are symmetric nodes, which can be untrue. We use Lemma 2 proven in Sec. II C as the additional criterion to deal with such regular components provided that they exist. However, real-world networks typically do not contain any sizable regular components, rendering unnecessary to involve the use of Lemma 2. That is, for many real-world networks, the computationally efficient SPV alone is sufficient to determine the symmetric nodes. We remark that the need to use an additional criterion to deal with networks containing a large regular component is shared by the existing methods in the literature, such as those based redundant eigenvectors^{18,19} and cluster synchronization^{2,12} as well as the combinatorial algorithm incorporating the concept of minimally balanced coloring.²⁰ (A detailed analysis of the common caveat among three existing approaches to finding network symmetries can be found in Appendix C.)

To demonstrate the general applicability of our SPV method to a variety of real-world networks, we test six such networks: two information networks (Odlis.net²⁴ and p2p-Gnutella²⁵), the social network Facebook,²⁶ a cooperative network CA-GrQc,²⁷ the biological network Yeast,²⁸ and a voting network Wiki-Vote,²⁹ whose structural properties are listed in Table III in Appendix D. By calculating the SPVs of the nodes, we have succeeded in finding all the symmetric motifs in those networks in linear time. The results are presented in Table IV in Appendix D and in Fig. 3.

A. Identification of symmetric nodes using SPVs

Theoretically, when calculating the SPVs of a network of size N , it is necessary to obtain all \mathbf{L}^n vectors for $n = 1, \dots, N$. However, if the network has a small diameter (as in many real-world networks³⁰), \mathbf{L}^n tends to converge fast, typically requiring only a few iterations of $(\mathcal{A}^n \cdot \mathbf{L}^0)$. This means that, computationally, our method for finding the symmetric nodes can be extremely efficient. To quantify the computational efficiency, we define the following accuracy measure ρ to find the smallest n value, denoted as n^* , for

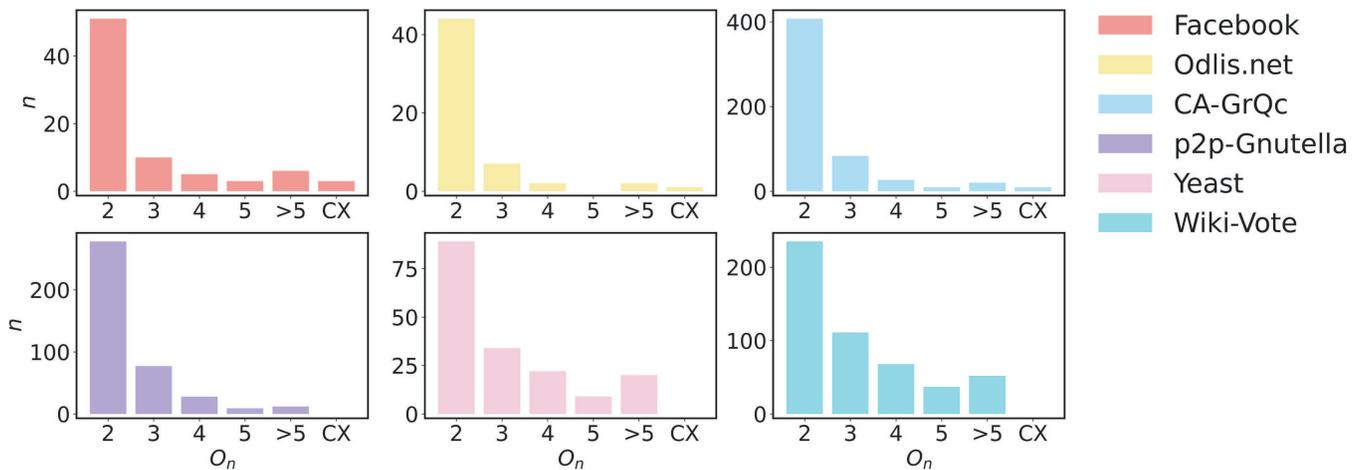


FIG. 3. All symmetric motifs in six real-world networks. In each panel, the abscissa O_n is the number of nodes contained in the symmetric motifs and “CX” denotes the complex symmetric motifs. The ordinate is the number of symmetric motifs.

which all symmetric structures in the network can be completely identified through the first n th-order SPVs,

$$\rho = \frac{\min\{U, Q\}}{\max\{U, Q\}} \frac{1}{U} \sum_{k=1}^U \frac{n_k}{N_k}, \quad (11)$$

where U is the number of non-trivial orbits in the symmetric structure of the network (see Appendix A for a definition of “orbits”) and Q is the number of nontrivial classes obtained through

$$C^1 \cap C^2 \cap \dots \cap C^n,$$

where “nontrivial” means that there are at least two elements in the class [e.g., the distinct blocks in the rightmost column in Fig. 1(b)]. The quantity n_k ($k = 1, \dots, U$) is the number of nodes in the k th orbit of the network, and N_k is the number of nodes in the class (block) containing the k th orbit. By definition, we have $\rho \leq 1$. The larger the value of ρ is, the higher is the accuracy of the identified symmetric nodes. The perfect case $\rho = 1$ means that all symmetric nodes can be identified using the first n orders SPVs.

Figure 4 shows the accuracy measure ρ vs n for the six real-world networks listed in Table III. It can be seen that 100% accuracy as characterized by $\rho = 1$ is achieved for $n \geq 5$ for all six networks. For Facebook and a Wiki-Vote network, this is achieved even for $n \geq 3$. The value of n^* is, therefore, exceedingly small for all cases tested, indicating that in actual calculations, it is only necessary to use the vectors $\mathbf{L}^n = \mathcal{A}^n \cdot \mathbf{L}^0$, including the order- n^* structural positions of all nodes to identify the symmetric nodes. The main computational cost comes from iterative multiplications of the network adjacency matrix $\mathcal{A}^{N \times N}$ with the vector \mathbf{L}^n for $n = (1, \dots, n^*)$. Since the computational complexity of each iteration is $\mathcal{O}(N\langle k \rangle)$, where $\langle k \rangle$ is the average degree of the network, the computational cost of finding all the symmetric structures of the network is $\mathcal{O}(n^*N\langle k \rangle)$. Networks in the real world often have sparse structure properties:

($k \ll N$); therefore, the computational complexity of the algorithm is only proportional to the size of the network (in linear time).

B. Quantifying nodal structure similarity based on SPVs

The similarity measure between two nodes based on their common neighbors has been widely used in problems, such as link prediction and recommendation systems.³¹ The use of a single centrality measure is usually not sufficient to describe the structural similarity between different nodes, as two nodes can have identical values for certain centrality but differ dramatically in other measures. [For example, nodes 10 and 12 in Fig. 1(a) have the same degree but their betweenness centrality values are quite different.] Symmetrical nodes are expected to have exactly the same SPV and centrality measures,¹⁴ such as H-index,³² PageRank,³³ k-core,³⁴ eigenvector centrality,³⁵ and betweenness centrality.³⁶ Intuitively, two nodes with similar (not equal) SPVs are structurally similar; therefore, we define the following similarity index r , the Euclidean distance between the structural position vectors of nodes i and j :

$$r = \sqrt{\sum_{n=1}^N (\tilde{L}_i^n - \tilde{L}_j^n)^2}, \quad (12)$$

where \tilde{L}_i^n ($n = 1, 2, \dots, N$) is the n th component of the SPV of node i , normalized by the largest n th component value among all the nodes: $\tilde{L}_i^n = L_i^n / \max(L_1^n, L_2^n, \dots, L_N^n)$. A small value of r indicates a relatively high degree of similarity between the structures of nodes i and j . The results in Fig. 4 indicate that SPVs of size at most about five are needed to accurately identify all symmetric nodes in the six real-world networks. It, therefore, suffices to use the following truncated SPV with six components, $\tilde{\mathbf{S}}_i^* \equiv (\tilde{L}_i^1, \tilde{L}_i^2, \dots, \tilde{L}_i^6)^T$, to calculate the similarity measure r .

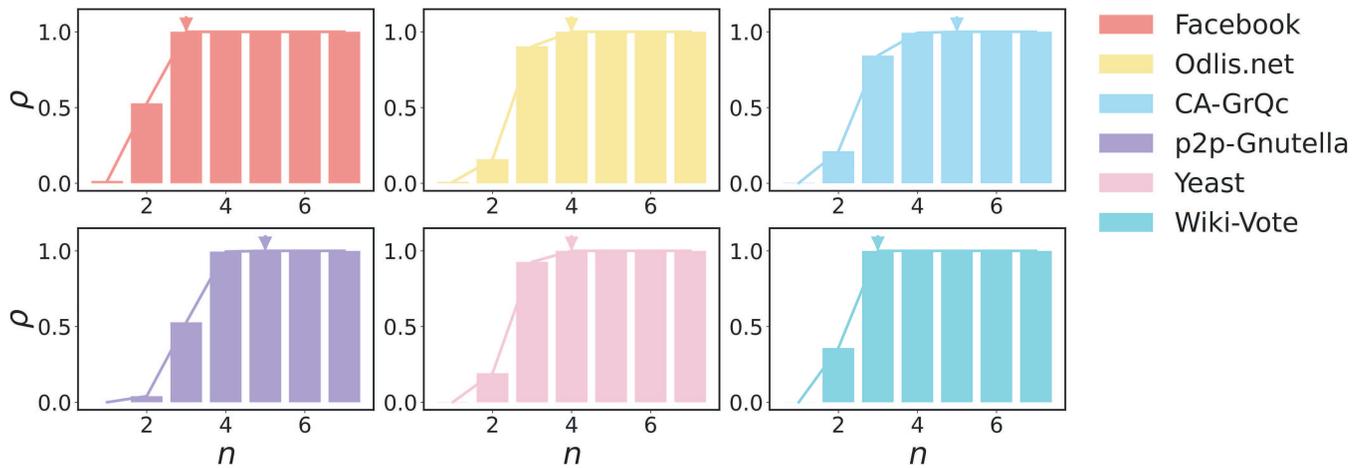


FIG. 4. Accuracy measure ρ of identifying all symmetric nodes vs the order- n SPVs for the six real-world networks in Table III. For Facebook and Wiki-Vote, the values of n^* , the minimum order required to have all symmetric nodes in the network correctly identified, are three. For Odlis.net and Yeast network, $n^* = 4$. For CA-GrQc and p2p-Gnutella networks, $n^* = 5$. The very small values of n^* across all the networks is a strong indication that the SPV-based framework is computationally extremely efficient for finding all the symmetric nodes in large complex networks.

As an application, we consider the problem of clustering nodes with similar structures. Our point is that the method based on the similarity index r defined in Eq. (12) performs better than many previously known methods. To demonstrate this, we use the k -mean algorithm to cluster the nodes in a complex network into \tilde{N} classes. To verify the effect of clustering, we coarse-grain the network and use the properties of the center of each class to approximate the properties of nodes in this class. For comparative analysis, five clustering methods, respectively, based on degree, betweenness, eigenvector centrality, closeness centrality, and clustering coefficient are considered. We define the coarse-grained error value E as

$$E = \sum_{c=1}^{\tilde{N}} \sum_{j=1}^{n_c} \sqrt{(z_{c_j} - Z_c)^2}, \quad (13)$$

where \tilde{N} is the number of clusters, n_c is the number of nodes in the c th cluster, z_{c_j} is a conventional statistical measure (e.g., degree, betweenness, closeness centrality, etc.) of node j in the c th cluster, and Z_c is the statistical measure of the center of the c th cluster. Given a network, we cluster all the nodes using each of the six clustering methods. Each clustering method produces errors for the five conventional statistical measures. Except for the clustering method based on similarity index r , only the errors of the four statistical measures are calculated in the other five clustering methods. For example, if we coarse-grain the network using the nodal degree, we calculate the errors from betweenness, eigenvector centrality, closeness centrality, and clustering coefficient. (In this case, the error from the nodal degree must be minimum by definition; therefore, including this error in the comparison is not meaningful.) We then normalize these coarse-grained error values to obtain the relative errors. Figure 5 shows, for the six real-world networks in Table III, the relative errors for the coarse-grained networks obtained from each of the six clustering methods, where the coarse-grained scale

is $\tilde{N} = N(k)$, with $N(k)$ being the scale of a coarse-grained network based on the nodal degree. It can be seen that the relative node-clustering errors associated with three or four statistical measures are minimized when clustering nodes are based on similarity index r . In particular, the relative errors from degree, eigenvector centrality, and clustering coefficient are the smallest for all networks. In addition, the relative error of closeness centrality is the smallest for Odlis.net, p2p-Gnutella, Yeast, and Wiki-Vote for the similarity index r based coarse-grained networks. For the remaining two networks, the relative error from closeness centrality can reach the second smallest. No conventional clustering method can generate node clustering as accurate as our similarity measure r , indicating that the SPVs can better describe the structural similarity of nodes than many existing conventional methods.

C. Quantifying role of nodes in propagation dynamics based on SPVs

Propagation is a fundamental type of dynamical processes in real-world networks, with examples ranging from epidemic spreading in social networks^{37,38} and diffusion of crisis in financial networks^{39,40} to cascading failures in complex networks⁴¹⁻⁴⁵ and signal transmission in neural networks.^{46,47} Measuring/quantifying the influence of nodes in propagation and identifying the nodes that play a critical role in the processes are issues with significant basic and applied values in network science. Conventionally, nodal centralities, such as the degree, the H-index, and the k -core, are used for these purposes. The principle of coarse-graining stipulates that nodes with the same degree, the H-index, or the k -core have identical influences. Our result is that the SPVs provide an alternative and potentially more powerful way to measure the nodal influences in the propagation dynamics. This is based on the intuition that the more similar the structures of nodes, the closer are their influences.

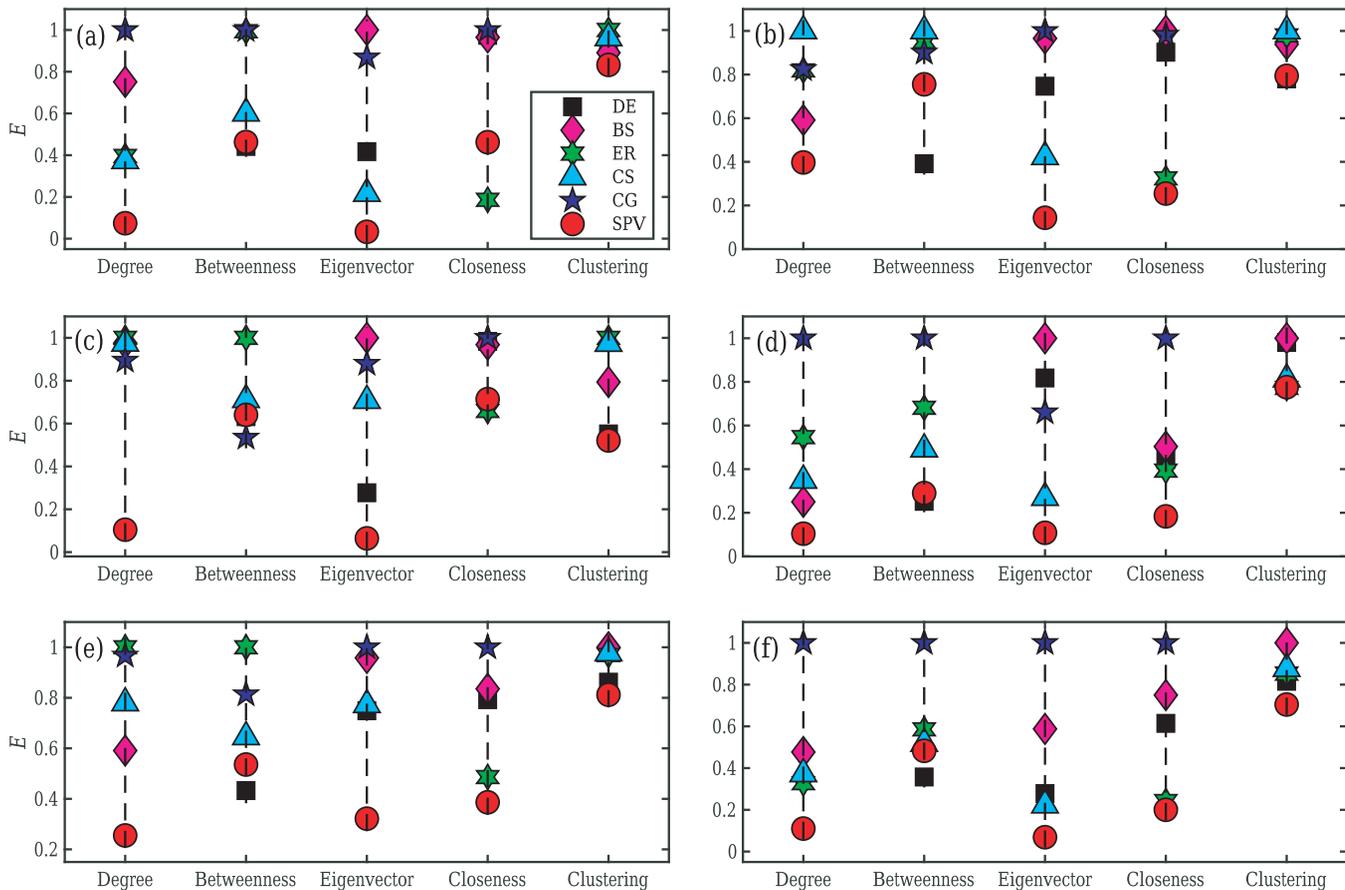


FIG. 5. Relative errors of node clustering by different clustering methods. (a)–(f) Respective results from six real-world networks: Facebook, Odlis.net, CA-GrQc, p2p-Gnutella, Yeast, and Wiki-Vote. Black squares, pink diamonds, light green hexagon stars, cyan triangles, blue pentagrams, and red circles represent the relative errors associated with the abscissa when clustering nodes based on degree (DE), betweenness (BS), eigenvector (ER), closeness centrality (CS), clustering centrality (CG), and structural position vector (SPV), respectively.

Following our approach to quantifying nodal structure similarity, we cluster the network using the similarity index r and calculate the two-norm value b_i of the truncated SPVs \tilde{S}_i^* of the center of each class. This leads to the cluster centrality b_i based on the SPVs, which can be used to quantify the influence of nodes in this class. When the network is not clustered, b_i is the SPV centrality.

To demonstrate the SPV-based quantification of nodal influences in propagation dynamics, we use the classic SIR (susceptible–infected–refractory) model. In the simulations, we set each node i as the origin of the spreading dynamics and calculate the fraction R_i of the recovered nodes. We average over 5000 independent runs to obtain the mean R_i value that characterizes the propagation influence of node i . We, therefore, have two sequences for all nodes in the network: the cluster centrality sequence (b_1, b_2, \dots, b_N) and the propagation influence sequence (R_1, R_2, \dots, R_N) ; therefore, their correlation can be calculated using, e.g., Kendall’s correlation coefficient τ , where $-1 \leq \tau \leq 1$ (see Appendix E for a definition).

A large value of τ indicates a higher accuracy of the centrality in ranking the nodal influences. Figure 6 shows, for the six real-world networks, Kendall’s correlation coefficient between different cluster centralities and the propagation influence as \tilde{N} , the number of clusters increases. It can be seen that the cluster centralities are generally suitable for ranking the nodal influences. For example, for the network Odlis.net, Fig. 6(b) gives $\tau \geq 0.85$, and, for the Yeast network in Fig. 6(e), we have $\tau \geq 0.8$. Within a certain range, as the number of clusters increases, the correlation increases correspondingly. When the number of clusters is sufficiently large (e.g., $\tilde{N} \geq 50$), the correlation value plateaus. For comparison, we also calculate the correlation coefficient between the propagation influence sequence and each of the three conventional coarse-graining methods: degree, H-index, and k-core, where each method gives only a single value for each network. As shown in Fig. 6, for all six real-world networks, the ranking performance of the SPV-based cluster centrality is consistently and significantly better than those of the

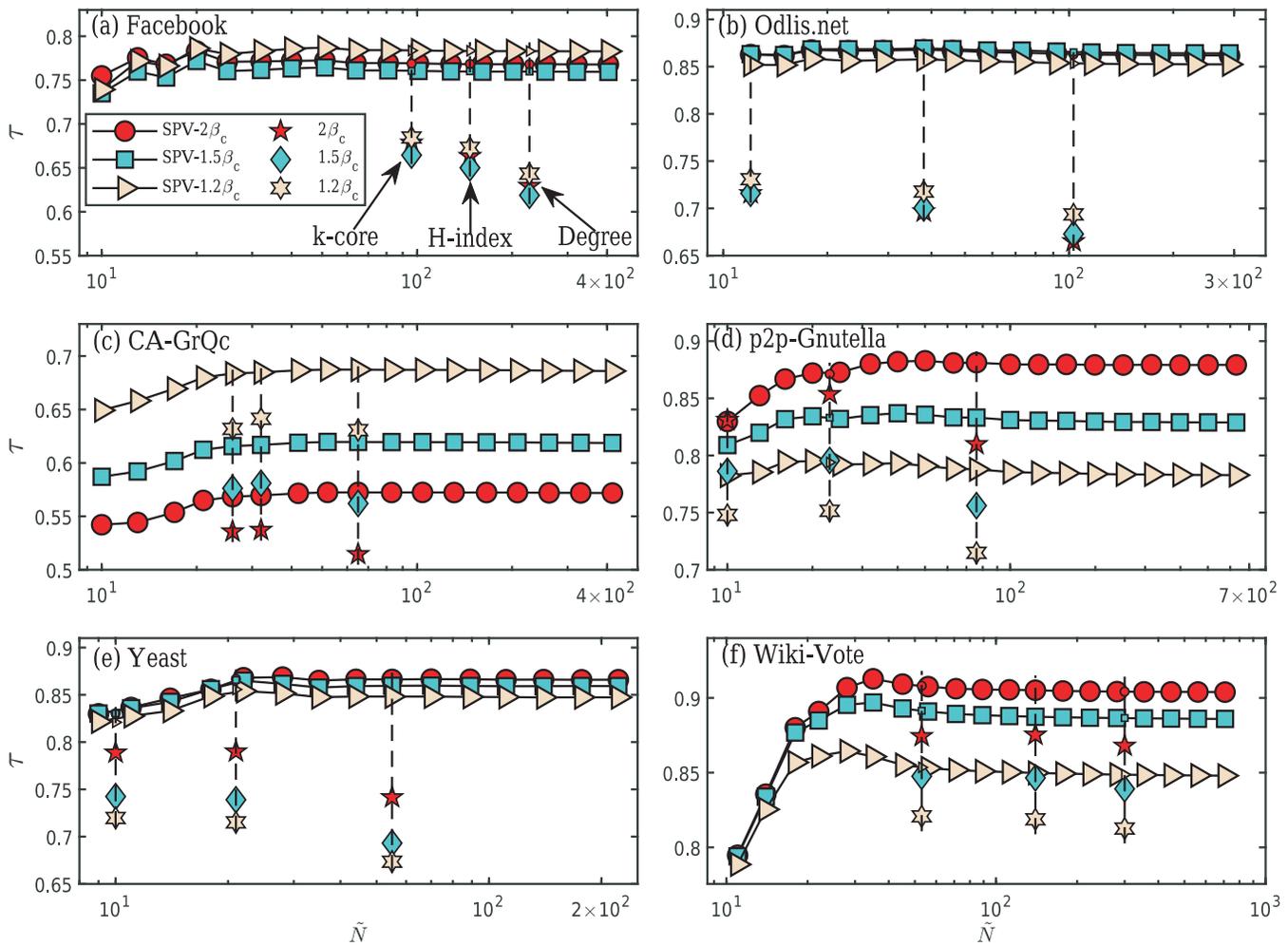


FIG. 6. Kendall's τ correlation coefficient for the six real-world networks. Shown is the τ value between the SPV-based cluster centrality and the nodal influence R vs \tilde{N} , the number of clusters in the SPV-based coarse-grained network for $0.001N \leq \tilde{N} \leq 0.1N$, with N being the size of the whole network. The light yellow triangles, light green squares, and red circles, respectively, represent the correlation coefficient τ between the SPV-based cluster centrality and the node influence R for infection rate $\beta = 1.2\beta_c$, $1.5\beta_c$, and $2\beta_c$ in the SIR dynamics. For comparison, τ values between three conventional centralities (degree, H-index, and k-core) and R are shown. (For each conventional centrality, the corresponding coarse-grained network has a single value of \tilde{N} .) The light yellow six-pointed star, the light green diamond, and the red five-pointed star, respectively, represent the τ value between the three conventional centralities and R for $\beta = 1.2\beta_c$, $1.5\beta_c$, and $2\beta_c$.

three conventional methods in characterizing the nodal influences in epidemic spreading.

As the number of clusters exceeds about 50 ($\tilde{N} \geq 50$), the cluster centrality achieves a high accuracy in ranking the nodal influence. Note that, for $\tilde{N} = N$, the cluster centrality becomes the actual SPV centrality. Let $SPV(\tilde{N})$ denote the cluster centrality for the coarse-grained network of \tilde{N} clusters, where $SPV \equiv SPV(\tilde{N} = N)$. To systematically compare the nodal ranking performances of SPV and $SPV(\tilde{N})$ with those of the six conventional nodal ranking methods (i.e., those based on the degree, H-index, k-core, closeness, betweenness, and eigenvector centrality), we choose \tilde{N} to

be the number of clusters obtained from the degree. The results for the six real-world networks are listed in Table I, where the basic parameter β (the infection rate) associated with SIR dynamics is set to be $\beta = 1.2\beta_c$ with β_c being the propagation threshold value (Appendix E). We see that SPV and $SPV(\tilde{N})$ have approximately the same accuracy that is generally higher than the accuracies of the conventional nodal ranking methods. There are a small number of exceptions. For example, in the p2p-Gnutella network, the eigenvector and closeness centralities have a higher correlation with the propagation influence than that with the SPV-based methods. This is due to the fact that the network has no apparent local

TABLE I. Kendall's τ correlation coefficient between node influence R and eight indices for $\beta = 1.2\beta_c$. Boldface denotes the largest T value in each row.

Networks	Degree	H-index	k-core	Closeness	Betweenness	Eigenvector	SPV	SPV(\tilde{N})
Facebook	0.6434	0.6729	0.6846	0.4286	0.4331	0.6511	0.7869	0.7834
Odlis.net	0.6939	0.7178	0.7309	0.7801	0.5406	0.8307	0.8514	0.8577
CA-GrQc	0.6307	0.6413	0.6318	0.5806	0.3587	0.5965	0.6856	0.6847
p2p-Gnutella	0.7148	0.7516	0.7480	0.8309	0.6533	0.8452	0.7825	0.7939
Yeast	0.6734	0.7151	0.7197	0.8216	0.5646	0.8272	0.8476	0.8556
Wiki-Vote	0.8126	0.8188	0.8207	0.8257	0.7431	0.8508	0.8509	0.8498

structures. Additional results for $\beta = 1.5\beta_c$ and $\beta = 2\beta_c$ can be found in Tables V and VI in Appendix E, respectively. We find that, except for the two networks p2p-Gnutella and CA-GrQc, the SPV-based cluster centralities have the best ranking performance. [In the CA-GrQc network, there is a high clustering coefficient c and assortativity r , indicating a core-periphery structure.⁴⁸ As a result, indices that reflect the importance of the core nodes (such as k-core, closeness centrality, eigenvector centrality) can better predict the spreading influence of nodes.]

In general, the influence of a node depends on the adjacency of different orders, corresponding to the components of each order in the SPV. The contribution of each order of adjacency is different.⁴⁹ Motivated by this, we further study the relative importance of each order component in the SPV to the influence of the node. Specifically, when clustering the nodes, we adjust the dimension of the truncated SPV and the weight of each order component. As shown in Figs. 10 and 11 in Appendix F, within a certain range, the more dimensions that the truncated SPV possesses, the better is the achieved ranking performance. Figures 12 and 13 in Appendix F show that, in a network with a small average path length, the importance of L^2 is larger than that of L^3 . In networks with a larger average path length, the conclusion is the opposite (Appendix F).

IV. GENERAL APPLICABILITY OF AN SPV METHOD TO FIND SYMMETRIC NODES IN REAL-WORLD NETWORKS

We have proved that the SPVs provide a necessary but not sufficient criterion for finding the symmetric nodes. This limitation is particularly pronounced for networks having a large regular component. One example is the classic Frucht network⁵⁰ as shown in Fig. 7(a). It is a regular network of 12 nodes and 18 edges, where each node has the same degree three, but the network has no symmetry. A slight modification can be considered, where two nodes are added with one having a link to each and every node in the Frucht network, leading to an irregular network but with a large regular component, which still has no symmetry, as shown in Fig. 7(b). For these two networks, all nodes in the original Frucht network will have the same SPV values, but this does not mean that the nodes are symmetric with respect to each other, as the SPV criterion is not sufficient. In spite of this deficiency, a key point worth emphasizing is that our SPV method is generally applicable for finding the symmetric nodes in real-world networks, as they typically do not contain large regular components.

To gain insights into developing a mathematical argument for this point, we first consider the modified Frucht network shown in Fig. 7(b) and prove that all eigenvalues except the largest one are the same as those from the original Frucht network, and the associated eigenvectors are, therefore, redundant eigenvectors (see Appendix C for a discussion of the concept of redundant eigenvectors).

Proof. Let λ_i ($i = 1, 2, \dots, 11$) be the eigenvalues (excluding the largest one) in the Frucht network and $\eta_i = [x_{i,1}, x_{i,2}, \dots, x_{i,12}]$ be the corresponding eigenvectors. Let

$$\mathcal{A}^{12 \times 12} \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \end{bmatrix} = \lambda_i \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \end{bmatrix}. \quad (14)$$

We can argue that the modified Frucht network in Fig. 7(b) has eigenvalues λ_i with the corresponding eigenvectors $\eta_i = [x_{i,1}, x_{i,2}, \dots, x_{i,12}, 0, 0]$. The starting point is the following equation:

$$\begin{bmatrix} \mathcal{A}^{12 \times 14} \\ \mathcal{A}^{2 \times 14} \end{bmatrix} \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \\ 0 \\ 0 \end{bmatrix} = \lambda_i \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \\ 0 \\ 0 \end{bmatrix}. \quad (15)$$

Separating Eq. (15) into two equations as

$$\mathcal{A}^{12 \times 14} \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \\ 0 \\ 0 \end{bmatrix} = \lambda_i \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \\ 0 \\ 0 \end{bmatrix} \quad (16)$$

and

$$\mathcal{A}^{2 \times 14} \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \\ 0 \\ 0 \end{bmatrix} = \lambda_i \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,12} \\ 0 \\ 0 \end{bmatrix}, \quad (17)$$

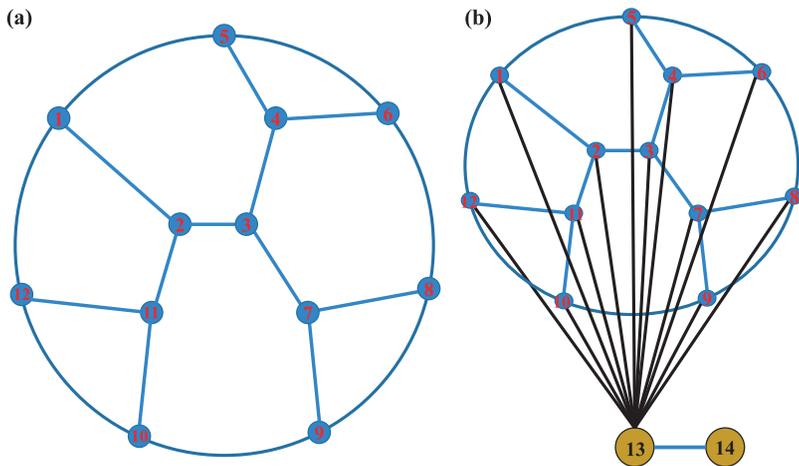


FIG. 7. Examples of regular networks to which our SPV method and previous some methods for finding symmetric nodes are not applicable. (a) The classic Frucht network—a regular network but with no symmetry. (b) A modified network that contains the Frucht network as a large regular component.

we see that Eq. (16) is equivalent to Eq. (14), indicating that Eq. (16) holds. Equation (17) is equivalent to

$$\begin{cases} \lambda_i \cdot 0 = \sum_{j=1}^{12} x_{ij} + 0, \\ \lambda_i \cdot 0 = 0. \end{cases} \quad (18)$$

In the Frucht network, because of the identity $\sum_{j=1}^{12} x_{ij} = 0$, Eqs. (15), (17), and (18) successively hold, indicating that λ_i and $[x_{i1}, x_{i2}, \dots, x_{i12}, 0, 0]$ are indeed the eigenvalues and eigenvectors of the network in Fig. 7(b), respectively. This means that the network containing a large regular component inherits all the eigenvalues and eigenvectors of the regular subnetwork except the largest eigenvalue and its corresponding eigenvector. All the “inherited” eigenvectors satisfy the same property $\sum_{j=1}^{14} x_{ij} = 0$. Following the argument in Appendix C, we have that using the redundant eigenvector to find the symmetric nodes fails for the modified Frucht network. □

In fact, Lemma 2 in the main text stipulates that the nodes in the modified Frucht network are asymmetric. In general, for a network that contains a relatively large regular component, application of our SPV method would give the wrong result that the nodes in the regular component are symmetric. The reason is similar to that of the redundant eigenvector method. Specifically, the SPV method can ensure that the sum of the eigencomponents corresponding to nodes with an equal SPV value is zero and the rest of the eigencomponents are all zero. While symmetric nodes have this property, the nodes of a regular subnetwork possess this property, too, in spite of lack of any symmetry. Our point is that large regular components, such as the one in the modified Frucht network, are rare in real-world networks.

To ensure that the candidate symmetric nodes identified by the SPV method do not include the asymmetric nodes in some special structures, such as a regular subnetwork, we exclude two classes of candidate symmetric structures found by the SPV method that contain only one orbit: all nodes that are fully connected or not connected in the candidate symmetric structure because the nodes are naturally symmetric in these two cases. These two structures are

called basic symmetric motifs. If the remaining candidate symmetric nodes found are symmetric nodes, the structure formed by them belongs to some complex symmetric motifs.

In real networks, often, there are only a few complex symmetric motifs.¹¹ Such complex symmetric motifs are composed of a small number of nodes, and nodes in the small regular subnetwork are all symmetric with respect to each other. Thus, when the SPV method is used to find the candidate symmetric nodes, only an exceedingly small number of special subnetwork structures need to be retested using Lemma 2, which hardly affects the computational complexity of the SPV method when being applied to real-world networks. While nodes with equal SPVs may not be symmetric, often, they have quite similar local topologies; therefore, the use of SPVs for coarse-grained networks is not affected.

We now prove that the nodes in the two basic symmetric motifs found by the SPV method must be symmetric.

Proof. By the SPV method, we can obtain two types of eigenvectors. In the first type, nodes with equal SPVs have equal eigencomponents. Applying an arbitrary permutation between the nodes, we have that the eigenvectors satisfy $P\eta = \eta$. The second type of eigenvectors is, in fact, redundant eigenvectors, for which the eigencomponents corresponding to the nodes in the two basic symmetric motifs have no restrictions except the constraint $\sum_{i=1}^K x_i = 0$, where K is the number of nodes in the motif. □

For $K = 2n$, where n is a positive integer greater than one, this gives $2n - 1$ linearly independent eigenvectors. From the nodal set $\{q_1, q_2, \dots, q_K\}$, we pair the nodes arbitrarily. Without loss of generality, we write the pairing sequence as

$$(q_1, q_2), (q_3, q_4), \dots, (q_{2j-1}, q_{2j}), \dots, (q_{2n-1}, q_{2n}).$$

We then permute these paired nodes. We can prove that, with permutation under the above rule, all the eigenvectors of matrix \mathcal{A} satisfy $P\eta = \pm\eta$. It is, therefore, necessary to construct $2n - 1$ eigenvectors that satisfy $P\eta = \pm\eta$.

For the j th eigenvector ($j = 1, 2, \dots, n$), let $(\eta_{j,q_{2j-1}}, \eta_{j,q_{2j}}) = (1, -1)$ and the eigencomponents of the remaining $K - 2$ nodes in nodal set V_1 be zero. We obtain n K -dimensional vectors. Since, for the second type of eigenvectors, except for nodal set V_1 , the

eigencomponents of the rest of the nodes are zero, we can simply expand them into $|V|$ -dimensional eigenvectors as

$$\begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & -1 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix}. \tag{19}$$

The n eigenvectors obtained by this method are linearly independent, which can be proved, as follows.

Proof. Denote each row of the matrix on the right side of Eq. (19), from top to bottom, as $\eta_1, \eta_2, \dots, \eta_n$, respectively. There exist k_1, k_2, \dots, k_n such that $k_1\eta_1 + k_2\eta_2 + \dots + k_n\eta_n = 0$. We then have

$$\begin{aligned} &(k_1, -k_1, k_2, -k_2, \dots, k_n, -k_n, 0, 0, \dots, 0, 0) \\ &= (0, 0, 0, 0, \dots, 0, 0, 0, 0, \dots, 0, 0). \end{aligned} \tag{20}$$

We have $k_1 = k_2 = \dots = k_n = 0$, indicating that the group of vectors $\eta_1, \eta_2, \dots, \eta_n$ are linearly independent, which are n linearly independent eigenvectors that satisfy $P\eta = -\eta$. \square

We search for $n - 1$ linearly independent eigenvectors that satisfy $P\eta = \eta$, which are also linearly independent of the n eigenvectors found above. For the j th eigenvector ($j = 1, 2, \dots, n$), let $(\eta_{j,q1}, \eta_{j,q2}) = (1, 1), (\eta_{j,q2j-1}, \eta_{j,q2j}) = (-1, -1)$ and the eigencomponents of the remaining $K - 4$ nodes in nodal set V_1 be zero. This leads to $n - 1$ K -dimensional vectors. Similarly, we expand them into $|V|$ -dimensional eigenvectors, as in formula (19). Apparently, the eigenvectors so constructed satisfy $P\eta = \eta$. Here, we prove that the newly constructed $n - 1$ eigenvectors are linearly independent of each other and are also linearly independent of the n eigenvectors found above.

Proof. The matrix composed of the newly constructed $n - 1$ vectors is

$$\begin{pmatrix} 1 & 1 & -1 & -1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 0 & 0 & -1 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & 0 & 0 & 0 & 0 & \cdots & -0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & \cdots & -1 & -1 \end{pmatrix} \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix}. \tag{21}$$

Denote each row of the matrix as $\eta_{n+1}, \eta_{n+2}, \dots, \eta_{2n-1}$, respectively, from top to bottom. There exist $k_1, k_2, \dots, k_{2n-1}$ such that $k_1\eta_1 + k_2\eta_2 + \dots + k_{2n-1}\eta_{2n-1} = 0$. We have

$$\begin{cases} k_1 + \sum_{l=n+1}^{2n-1} k_l = 0, \\ -k_1 + \sum_{l=n+1}^{2n-1} k_l = 0 \end{cases} \tag{22}$$

and

$$\begin{cases} k_{j+1} - k_{n+j} = 0, \\ -k_{j+1} - k_{n+j} = 0, \end{cases} \quad j = 1, 2, \dots, n - 1. \tag{23}$$

From Eqs. (22) and (23), we have $k_1 = k_2 = \dots = k_{2n-1} = 0$, which means that the group of vectors $\eta_1, \eta_2, \dots, \eta_{2n-1}$ are linearly independent of each other, leading to $2n - 1$ linearly independent eigenvectors. Among them, n eigenvectors satisfy $P\eta = -\eta$ and $n - 1$ eigenvectors satisfy $P\eta = \eta$.

For $K = 2n + 1$, where n is a positive integer, there are $2n$ linearly independent eigenvectors. Similarly, we perform node pairing and permutation to construct the $2n$ eigenvectors. The first $2n - 1$ eigenvectors are constructed in the same way as for $K = 2n$, and the last eigenvector is $\eta_{2n} = (1, 1, \dots, 1, 1, -2n, 0, 0, \dots, 0, 0)$. The eigenvector η_{2n} is then linearly independent of the above $2n - 1$ eigenvectors, and we have $P\eta = \eta$.

According to the permutation method explained, all eigenvectors of the network satisfy $P\eta = \pm\eta$. Since we pair nodes arbitrarily, permuting any two nodes can make all the eigenvalues satisfy $P\eta = \pm\eta$. According to Lemma 2 in the main text, the permutation corresponding to the permutation matrix P is an automorphism; that is, nodes in the two basic symmetric motifs found by the SPV method must be symmetric. \square

V. DISCUSSION

Symmetric structures are fundamental to dynamical processes on complex networks, rendering important accurately identifying these structures. While the algebraic-group theory based methods can find all the symmetric nodes in the network in quasipolynomial time, these methods do not provide a statistical measure to describe whether the two nodes are symmetrical. To articulate a statistical measure that can lead to an efficient method to find all the symmetric nodes in linear time and can characterize the structural similarity of nodes and serve to quantify the roles of nodes is the goal of the present work. In particular, we have introduced the nodal SPVs, which are defined through the adjacency relationships in the network. Mathematically, by employing the algebraic-group theory, we transform the interplay between SPVs and network symmetries into an eigenvector problem and prove that having equal SPV values is a necessary condition for the underlying nodes to be symmetric: if the nodes are symmetric, then their SPV values must be equal. This is a stronger necessary condition for nodal symmetry than that based on, e.g., the redundant eigenvectors as reported in the literature. Utilizing real-world networks, we have demonstrated that our SPV-based method allows all symmetric structures to be found with a small number of iterations of the SPV in computation time that is proportional to the network size. More importantly, our SPV-based framework enables the role of the nodal influence in dynamical processes to be quantified and differentiated.

A caveat of our SPV method is that it provides a necessary but not a sufficient condition for the symmetric nodes. There are regular networks, such as the Frucht network, or networks that contain a dominant regular component, which have no symmetries, but the SPV values are equal among the nodes. The SPV method is, therefore, not applicable to such networks. We argue that this deficiency is, in fact, shared by some existing methods for finding the network symmetries. In the real world, networks that contain a dominant component are rare. We present strong numerical evidence and a mathematical argument that our SPV method is generally applicable to finding symmetric nodes in real-world networks.

The similarity between two nodes based on common neighbors has been widely used in link prediction and recommendation systems. Since the symmetric nodes have exactly the same structural characteristics, our SPV framework provides a natural way to define the structural similarity. In particular, defining a structural similarity index or centrality, we cluster the nodes in the network and find that the SPV-based clustering method is remarkably effective in coarse-graining the network in that it outperforms the previous clustering methods based on the traditional centralities, such as the degree, eigenvector centrality, H-index, and closeness and betweenness centralities. More importantly, from the standpoint of dynamical processes, our SPV-based clustering centrality can be used to measure and quantify the roles of nodes in propagation dynamics. A detailed calculation using six real-world networks indicates that, in most cases, the SPV-based centrality outperforms the conventional centralities by a large margin in predicting the nodal influences on propagation dynamics.

Taken together, our work provides a conceptually appealing and computationally extremely efficient framework to find symmetric nodes in large complex networks, totally bypassing the sophisticated steps in the conventional methods based on automorphism groups. The method also leads to statistical characterization of nodal symmetries, with direct applications to coarse-graining of complex networks and cluster synchronization that may occur among remote nodes. While large regular components are rare in real networks, the isomorphism and automorphism of regular networks have always been the focus and a difficult aspect of graph theory.^{51,52} From a theoretical point of view, to characterize the structural position vectors in networks that contain a dominant regular component to distinguish the asymmetric nodes is worth studying.

ACKNOWLEDGMENTS

This work was supported by the National Ten Thousand Talents Plan Youth Top Talent project, the National Natural Science Foundation of China (Grant Nos. 82161148012, 11975099, and 61802321), and the Science and Technology Commission of Shanghai Municipality (Grant No. 14DZ2260800). The work at Arizona State University was supported by the Office of Naval Research through Grant No. N00014-21-1-2323.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Yong-Shang Long: Conceptualization (equal); Investigation (equal); Writing – original draft (equal); Writing – review & editing (equal). **Zheng-Meng Zhai:** Conceptualization (equal); Software (equal). **Ming Tang:** Conceptualization (equal); Funding acquisition (equal); Investigation (equal). **Ying Liu:** Writing – review & editing (equal). **Ying-Cheng Lai:** Conceptualization (equal); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

All relevant computer codes and data are available from the corresponding author upon reasonable request.

APPENDIX A: SYMMETRY AND AUTOMORPHISM GROUPS

A network is a graph $G(V, E)$ with vertex set V and edge set E , where a pair of nodes are adjacent if there is an edge between them. An automorphism is a permutation of the vertices of the network, which preserves the adjacency. The combinations of a set of automorphisms form a group $\mathcal{G} = \text{Aut}(G)$ that describes the symmetries of the network.⁵³ A network with a nontrivial automorphism group is symmetric.

Consider the permutations of a set of N nodes $Y = v_1, v_2, \dots, v_N$. The support of a permutation p is the set of points that p moves: $\text{supp}(p) = \{v_i \in V(G) | p(v_i) \neq v_i\}$. Two sets of automorphisms, P and Q , are support-disjoint if every pair of automorphisms $p \in P$ and $q \in Q$ has disjoint support. Similarly, the automorphism subgroups G_P and G_Q generated by P and Q are support-disjoint if P and Q are support-disjoint. The independent action of automorphism subgroups provides a way to factorize the automorphism groups of complex networks into “irreducible building blocks.”¹⁰ In particular, let S be a set of generators of the

TABLE II. Orbital representation for the symmetric motifs and the corresponding geometric factors. The meanings of the symbols are as follows: O_α^* specifies that the orbit contains α nodes that are connected with each other; O_α means that the orbit contains α nodes, and there is no connection between them; O_α^μ signifies that the orbit contains α nodes with μ edges between them; and $O_\alpha \xrightarrow{\gamma} O_{\alpha'}$ means that the symmetric motifs contain two orbits, and each node in the first orbit is connected to γ nodes in the second orbit. If $\gamma = \alpha'$, the symmetric motif is decomposed into two independent symmetric motifs, e.g., M_7 and M_8 in Fig. 3(a). The group of all permutations of n objects is denoted as S_n , and \wr is the wreath product.⁵⁴ Table IV lists that there are mainly two symmetric motifs in the six real-world networks: O_α^* and O_α , which are the basic symmetric motifs. There are a few symmetric motifs, such as O_α^μ and $O_\alpha \xrightarrow{\gamma} O_{\alpha'}$, which are the complex symmetric motifs.

Symmetric motif	Orbital representation	Geometric factor
M_1	$O_2 \xrightarrow{2} O_4$	$S_2 \wr S_2$
M_2	$O_2 \xrightarrow{1} O_2$	$S_2 \wr S_1$ or S_2
M_3	O_3^*	S_3
M_4, M_7, M_8	O_2	S_2
M_5	O_3	S_3
M_6	O_4^2	$S_2 \wr S_2$

automorphism groups $\text{Aut}(G)$. Suppose that we partition S into n support-disjoint subsets: $S = S_1 \cup S_2 \cdots S_n$ such that each S_i cannot itself be decomposed into smaller support-disjoint subsets. H_i is called the subgroup generated by S_i , and every H_i commutes with all others. A direct product decomposition of the automorphism groups $\text{Aut}(G)$ is

$$\text{Aut}(G) = H_1 \times H_2 \times \cdots \times H_n. \tag{A1}$$

The decomposition of an automorphism group in Eq. (A1) is unique and irreducible.¹⁰ The direct product factorization given in Eq. (A1) is the geometric decomposition of $\text{Aut}(G)$ and each factor H_i a geometric component. The induced subgraph on a set of vertices $X \subset V$ is the graph obtained by taking X and any edges whose end points are in X . The induced subgraph on the support of a geometric factor H_i is called the symmetric motif, denoted as M_{H_i} . A number of common symmetric motifs in the real-world networks are shown in Fig. 3. The orbital representation of the symmetric motifs therein and how the geometric factors of the automorphism group are related to different symmetric motifs are described in Table II. For every vertex $v \in V$, the set of vertices to which v maps under the action of the automorphism group $\mathcal{G} = \text{Aut}(G)$ is called a G -orbit of v , denoted as $\Delta(v)$. More formally, we have $\Delta(v) = \{g \cdot v \in V : g \in \text{Aut}(G)\}$. A symmetric motif generally consists of single or multiple orbits.

APPENDIX B: THE SPV EQUALITY IS A SUFFICIENT CONDITION FOR THE ADJACENCY MATRIX CORRESPONDING TO A NETWORK TO HAVE REDUNDANT EIGENVECTORS

We proceed to prove the theorem in Sec. II C: if the SPV of two nodes is equal, there will be redundant eigenvectors about them in the eigenvectors of the adjacency matrix corresponding to the network. If the SPVs of nodes x and y are equal, their corresponding

eigencomponents satisfy the following equations:

$$\begin{cases} \sum_{i=1}^N c_i \lambda_i (\eta_{i,x} - \eta_{i,y}) = 0, \\ \sum_{i=1}^N c_i \lambda_i^2 (\eta_{i,x} - \eta_{i,y}) = 0, \\ \vdots \\ \sum_{i=1}^N c_i \lambda_i^N (\eta_{i,x} - \eta_{i,y}) = 0, \end{cases} \tag{B1}$$

where some of the eigenvalues of the network adjacency matrix can be zero. Without loss of generality, we arbitrarily assume $\lambda_1 = 0, c_2 = 0$. In general, a large network has repeated eigenvalues. If the k th and the $(k + 1)$ th eigenvalues are equal: $\lambda_k = \lambda_{k+1}$, Eq. (B1) becomes

$$\begin{cases} \sum_{i=3}^{k-1} c_i \lambda_i (\eta_{i,x} - \eta_{i,y}) + \lambda_k (c_k (\eta_{k,x} - \eta_{k,y}) + c_{k+1} (\eta_{k+1,x} - \eta_{k+1,y})) \\ \quad + \sum_{i=k+2}^N c_i \lambda_i (\eta_{i,x} - \eta_{i,y}) = 0, \\ \sum_{i=3}^{k-1} c_i \lambda_i^2 (\eta_{i,x} - \eta_{i,y}) + \lambda_k^2 (c_k (\eta_{k,x} - \eta_{k,y}) + c_{k+1} (\eta_{k+1,x} - \eta_{k+1,y})) \\ \quad + \sum_{i=k+2}^N c_i \lambda_i^2 (\eta_{i,x} - \eta_{i,y}) = 0, \\ \vdots \\ \sum_{i=3}^{k-1} c_i \lambda_i^N (\eta_{i,x} - \eta_{i,y}) + \lambda_k^N (c_k (\eta_{k,x} - \eta_{k,y}) + c_{k+1} (\eta_{k+1,x} - \eta_{k+1,y})) \\ \quad + \sum_{i=k+2}^N c_i \lambda_i^N (\eta_{i,x} - \eta_{i,y}) = 0. \end{cases} \tag{B2}$$

We take the first $(N - 3)$ equations from Eq. (B2) to form a new set of equations. Regarding $\eta_{i,x} - \eta_{i,y}$ for $i = 3, 4, \dots, k - 1, k + 2, k + 3, \dots, N$ and $c_k (\eta_{k,x} - \eta_{k,y}) + c_{k+1} (\eta_{k+1,x} - \eta_{k+1,y})$ for $i = k, k + 1$ as unknown quantities, we have that the determinant of the coefficient of the new set of equations is

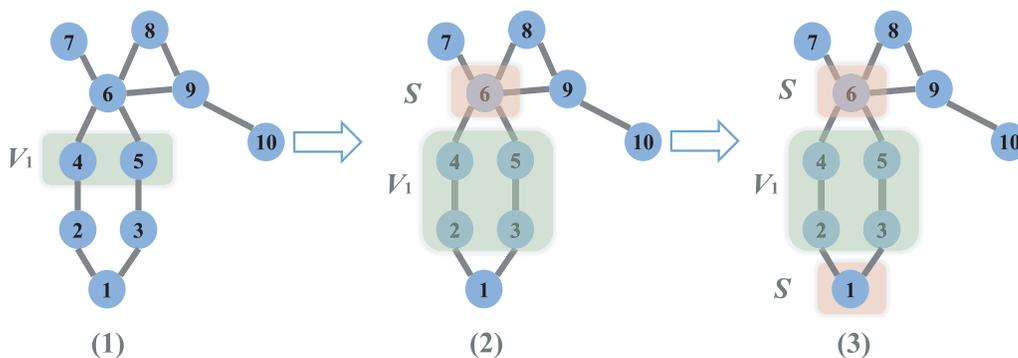


FIG. 8. Schematic illustration of determining nodal sets V_1 and S . (1) Take nodes 4 and 5 as an example. Let $V_1 = \{4, 5\}$. The characteristic equations of the two nodes are $\lambda_i \eta_{i,4} = \eta_{i,2} + \eta_{i,6}$ (e1) and $\lambda_i \eta_{i,5} = \eta_{i,3} + \eta_{i,6}$ (e2), respectively. (2) $S = \{6\}$. Substitute the characteristic equations of nodes 2 and 3 into (e1) and (e2), respectively, to get $(\lambda_i^2 - 1)\eta_{i,4} = \eta_{i,1} + \lambda_i \eta_{i,6}$, $(\lambda_i^2 - 1)\eta_{i,5} = \eta_{i,1} + \lambda_i \eta_{i,6}$, and $V_1 = \{2, 3, 4, 5\}$. (3) $S = \{1, 6\}$. The characteristic components of nodes 4 and 5 can be represented by the characteristic components of node set S , where $V_1 = \{2, 3, 4, 5\}$ and $S = \{1, 6\}$.

$$C' = \begin{pmatrix} c_3\lambda_3 & c_4\lambda_4 & \cdots & c_{k-1}\lambda_{k-1} & \lambda_k & c_{k+2}\lambda_{k+2} & \cdots & c_N\lambda_N \\ c_3\lambda_3^2 & c_4\lambda_4^2 & \cdots & c_{k-1}\lambda_{k-1}^2 & \lambda_k^2 & c_{k+2}\lambda_{k+2}^2 & \cdots & c_N\lambda_N^2 \\ c_3\lambda_3^3 & c_4\lambda_4^3 & \cdots & c_{k-1}\lambda_{k-1}^3 & \lambda_k^3 & c_{k+2}\lambda_{k+2}^3 & \cdots & c_N\lambda_N^3 \\ \vdots & \vdots \\ c_3\lambda_3^{N-3} & c_4\lambda_4^{N-3} & \cdots & c_{k-1}\lambda_{k-1}^{N-3} & \lambda_k^{N-3} & c_{k+2}\lambda_{k+2}^{N-3} & \cdots & c_N\lambda_N^{N-3} \end{pmatrix} = c_3c_4 \cdots c_{k-1}c_{k+2} \cdots c_N\lambda_3\lambda_4 \cdots \lambda_k\lambda_{k+2} \cdots \lambda_N \cdot \begin{pmatrix} 1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \\ \lambda_3 & \lambda_4 & \cdots & \lambda_{k-1} & \lambda_k & \lambda_{k+2} & \cdots & \lambda_N \\ \lambda_3^2 & \lambda_4^2 & \cdots & \lambda_{k-1}^2 & \lambda_k^2 & \lambda_{k+2}^2 & \cdots & \lambda_N^2 \\ \vdots & \vdots \\ \lambda_3^{N-4} & \lambda_4^{N-4} & \cdots & \lambda_{k-1}^{N-4} & \lambda_k^{N-4} & \lambda_{k+2}^{N-4} & \cdots & \lambda_N^{N-4} \end{pmatrix}.$$

Because the multiplicity of eigenvalues has been eliminated, we have $\lambda_3 \neq \lambda_4 \neq \cdots \neq \lambda_k \neq \lambda_{k+2} \neq \cdots \neq \lambda_N$ and $c_i \neq 0$ for $i = 3, 4, \dots, k - 1, k + 2, k + 3, \dots, N$; therefore, $C' \neq 0$. The only solution of Eq. (B2) is trivial. In the eigenvectors representing L^0 , insofar as the corresponding eigenvalues are not zero, the eigenvectors corresponding to nodes x and y are equal.

For nodes x and y with equal SPVs, we further analyze the relationship between their eigenvectors in the remaining eigenvectors. In particular, by transforming and simplifying the characteristic equations $\mathcal{A} \cdot \eta = \lambda\eta$, the eigenvectors corresponding to nodes x and y can be represented by the eigenvectors of any identical nodal set S . We assume that the eigenvectors corresponding to nodes x and y are represented by the eigenvectors of node s_c and write

$$\varphi_0(\lambda_i)\eta_{i,x} = \varphi_1(\lambda_i)\eta_{i,s_c} \tag{B3}$$

$$\psi_0(\lambda_i)\eta_{i,y} = \psi_1(\lambda_i)\eta_{i,s_c} \tag{B4}$$

or

$$\frac{\varphi_0(\lambda_i)}{\psi_0(\lambda_i)}\eta_{i,x} = \frac{\varphi_1(\lambda_i)}{\psi_1(\lambda_i)}\eta_{i,y}. \tag{B5}$$

Since $\varphi_0(\lambda_i)/\psi_0(\lambda_i)$ and $\varphi_1(\lambda_i)/\psi_1(\lambda_i)$ are polynomials in λ_i , we let $T_0(\lambda_i) \equiv \varphi_0(\lambda_i)/\psi_0(\lambda_i)$ and $T_1(\lambda_i) \equiv \varphi_1(\lambda_i)/\psi_1(\lambda_i)$. Equation (B5) becomes

$$T_0(\lambda_i)\eta_{i,x} = T_1(\lambda_i)\eta_{i,y}. \tag{B6}$$

If the eigenvalue λ_i associated with the eigenvector used to represent L^0 is non-zero, we have $\eta_{i,x} = \eta_{i,y}$ and

$$T_0(\lambda_i) = T_1(\lambda_i). \tag{B7}$$

It is worth noting that the condition for Eq. (B7) to hold does not limit the value of λ_i (except zero), indicating that the equality $T_0(\lambda_i) = T_1(\lambda_i)$ does not depend on the value of λ_i . That is, the polynomials $T_0(\lambda_i)$ and $T_1(\lambda_i)$ are exactly the same. Substituting $\varphi_0(\lambda_i) = T_0(\lambda_i)\psi_0(\lambda_i)$ and $\varphi_1(\lambda_i) = T_1(\lambda_i)\psi_1(\lambda_i)$ into Eq. (B3) gives

$$T_0(\lambda_i)\psi_0(\lambda_i)\eta_{i,x} = T_1(\lambda_i)\psi_1(\lambda_i)\eta_{i,s_c} \tag{B8}$$

or

$$\psi_0(\lambda_i)\eta_{i,x} = \psi_1(\lambda_i)\eta_{i,s_c}. \tag{B9}$$

From Eqs. (B3) and (B9), we see that the eigenvectors corresponding to nodes x and y can be represented by the eigenvectors of any identical nodal set S , and the form of representation is exactly the same.

To find the identical node set S , we define the characteristic equation of node j as $\lambda_i\eta_{i,j} = \sum_{i \in \Lambda_j} \eta_{i,i}$, where $i = 1, 2, \dots, N$ and Λ_j is the set of neighboring nodes of j . That is, the eigenvector of node j can be expressed by the eigenvectors of all its neighboring nodes. The process, as illustrated in Fig. 8, consists of the following four steps.

1. Analyze the characteristic equation $\lambda_i\eta_{i,x} = \sum_{j=1}^{k_x} \eta_{i,x_j}$ of node x with degree k_x , where x_j is a neighboring node of x . If all neighbors of x are removed, node x will be separated from the network. Denote $V_1 = \{x\}$.
2. Substitute the characteristic equation of a neighboring node x_1 of x into the characteristic equation of x so that the eigenvectors of node x are represented by the eigenvectors of its neighbors (excluding x_1) and the neighbors of x_1 . If the nodes used to represent the eigenvectors of x are removed, the local structure formed by nodes x and x_1 must be separated from the network. Denote $V_1 = \{x, x_1\}$.
3. Continue the iterative process in step 2 until the eigenvectors of node x are represented by the eigenvectors of nodal set S . If the characteristic equation of node m is used for iteration, then m is added to V_1 and $V_1 = \{x, x_1, \dots\}$. Remove nodal set S . The local structure $local_G(V_1, E_1)$ formed by V_1 is separated from the network, where E_1 is the set of internal edges of the node set V_1 .
4. Two possible situations can arise in determining nodal set S : two nodes may or may not have common neighboring nodes of different orders. The first case is where nodes x and y have common neighboring nodes. In this case, simplify the characteristic equations of nodes x and y according to step 2. If x and y have a common neighboring set $s_{1,xy}$, then $S = \{s_{1,xy}\}$. In this case,

simplify the characteristic equations of x and y with the characteristic equations of non-common neighboring nodes and add these nodes to the set V_1 . If there is a second-order common neighboring set $s_{2,xy}$ between nodes x and y , add them to the set S to get $S = \{s_{1,xy}, s_{2,xy}\}$. Continue the process until the eigencomponents of x and y can be represented by the eigencomponents of S . The second case is where nodes x and y have no n th-order common neighbors. In this case, all neighbors of x constitute nodal set S .

Let the identical nodal set be $S = \{s_1, s_2, \dots, s_k\}$. Nodes of the network can then be divided into three sets: V_1, S , and the remaining nodal set $V/\{V_1, S\}$. We perform elementary row transformation on matrix \mathcal{A} and divide it into blocks by row to get

$$\mathcal{A} \cdot \boldsymbol{\eta} = \begin{bmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \mathcal{A}_3 \end{bmatrix} \cdot \boldsymbol{\eta} = \lambda \boldsymbol{\eta}, \tag{B10}$$

where $\mathcal{A}_1, \mathcal{A}_2$, and \mathcal{A}_3 are, respectively, $|V_1| \times N, |S| \times N$, and $|V/\{V_1, S\}| \times N$ adjacency matrices of the local structures formed by nodal sets $V_1, S, V/\{V_1, S\}$ together with their neighbors. This analysis indicates that the eigencomponents of nodes x and y can be represented by the eigencomponents of nodal set $S = \{s_1, s_2, \dots, s_k\}$ through the characteristic equations $\mathcal{A}_1 \boldsymbol{\eta} = \lambda \boldsymbol{\eta}$. Since we have proved that the eigencomponents corresponding to x and y can be represented by the eigencomponents of any identical nodal set S , the representation is exactly the same. We, therefore, have

$$f_0(\lambda_i) \eta_{i,x} = f_1(\lambda_i) \eta_{i,s_1} + f_2(\lambda_i) \eta_{i,s_2} + \dots + f_k(\lambda_i) \eta_{i,s_k}, \tag{B11}$$

$$f_0(\lambda_i) \eta_{i,y} = f_1(\lambda_i) \eta_{i,s_1} + f_2(\lambda_i) \eta_{i,s_2} + \dots + f_k(\lambda_i) \eta_{i,s_k}. \tag{B12}$$

If $f_0(\lambda_i) \neq 0$, then $\eta_{i,x} = \eta_{i,y}$. Permuting nodes x and y , we have that the eigenvectors associated with such eigenvalues satisfy $P\boldsymbol{\eta} = \boldsymbol{\eta}$.

We now discuss the case $f_0(\lambda_i) = 0$. It can be seen from Eqs. (B11) and (B12) that the eigencomponents of nodes x and y are free variables. In order to find the relationship between the eigencomponents of x and y , we further analyze the characteristic equation $\mathcal{A} \cdot \boldsymbol{\eta} = \lambda \boldsymbol{\eta}$. To do this, we move x and y out of nodal set V_1 and divide all nodes in the network into two sets: $V_1/\{x, y\}$ and $\{V/V_1, x, y\}$. The characteristic equation can be written as

$$\mathcal{A} \cdot \boldsymbol{\eta} = \begin{bmatrix} \mathcal{A}'_1 \\ \mathcal{A}'_2 \end{bmatrix} \cdot \boldsymbol{\eta} = \lambda \boldsymbol{\eta}, \tag{B13}$$

where \mathcal{A}'_1 and \mathcal{A}'_2 are, respectively, $|V_1/\{x, y\}| \times N$ - and $|\{V/V_1, x, y\}| \times N$ -dimensional adjacency matrices of the local structures formed by nodal sets $V_1/\{x, y\}$ and $\{V/V_1, x, y\}$ as well as their neighbors. The eigencomponents of $V_1/\{x, y\}$ can be represented by the eigencomponents of the set $\{x, y, S\}$ through the characteristic equation $\mathcal{A}'_1 \boldsymbol{\eta} = \lambda \boldsymbol{\eta}$. From Eqs. (B11) and (B12), the eigencomponents of x and y are related to the eigencomponents of nodal set S .

We are, therefore, led to analyze the characteristic equation of any node $s_i \in S: \lambda_i \eta_{i,s_i} = \sum_{j \in \Lambda_{s_i}} \eta_{i,j}$, where $i = 1, 2, \dots, N$ and Λ_{s_i} is the set of neighbors of node s_i . We divide the neighbors of s_i into three parts: the first part is a subset of nodal set V_1 , the second part belongs to the set S , and the third part is contained in the set $V/\{V_1, S\}$. Since the eigencomponents of the nodes belonging to the

set V_1 can be represented by the eigencomponents of the set $\{x, y, S\}$, the characteristic equation of node s_i can be written as

$$F_0(\lambda_i) \eta_{i,s_i} = G_0(\lambda_i) \eta_{i,x} + G_1(\lambda_i) \eta_{i,y} + \sum_{s_j \in S/s_i} F_j(\lambda_i) \eta_{i,s_j} + \sum_{j=1}^m H_j(\lambda_i) \eta_{i,r_j}, \tag{B14}$$

where nodal set $R(r_1, r_2, \dots, r_m)$ are the neighbors of node s_i in the set $V/\{V_1, S\}$. Rearranging the terms, we get

$$G_0(\lambda_i) \eta_{i,x} + G_1(\lambda_i) \eta_{i,y} = F_0(\lambda_i) \eta_{i,s_i} - \sum_{s_j \in S/s_i} F_j(\lambda_i) \eta_{i,s_j} - \sum_{j=1}^m H_j(\lambda_i) \eta_{i,r_j}. \tag{B15}$$

Let $E \equiv F_0(\lambda_i) \eta_{i,s_i} - \sum_{s_j \in S/s_i} F_j(\lambda_i) \eta_{i,s_j} - \sum_{j=1}^m H_j(\lambda_i) \eta_{i,r_j}$. Equation (B15) leads to

$$G_0(\lambda_i) \eta_{i,x} = E - G_1(\lambda_i) \eta_{i,y}, \tag{B16}$$

$$G_1(\lambda_i) \eta_{i,y} = E - G_0(\lambda_i) \eta_{i,x}. \tag{B17}$$

Multiplying both sides of Eqs. (B11) and (B12) by $G_0(\lambda_i)$ and $G_1(\lambda_i)$, respectively, and substituting Eqs. (B16) and (B17) into the resulting equations, we get

$$f_0(\lambda_i) G_1(\lambda_i) \eta_{i,y} = f_0(\lambda_i) E - (f_1(\lambda_i) \eta_{i,s_1} + f_2(\lambda_i) \eta_{i,s_2} + \dots + f_k(\lambda_i) \eta_{i,s_k}) G_0(\lambda_i), \tag{B18}$$

$$f_0(\lambda_i) G_0(\lambda_i) \eta_{i,x} = f_0(\lambda_i) E - (f_1(\lambda_i) \eta_{i,s_1} + f_2(\lambda_i) \eta_{i,s_2} + \dots + f_k(\lambda_i) \eta_{i,s_k}) G_1(\lambda_i). \tag{B19}$$

Since we have proved that the eigencomponents corresponding to nodes x and y can be represented by the eigencomponents of any identical nodal set and the way of representation is exactly the same, we have $G_0(\lambda_i) = G_1(\lambda_i)$. Equation (B14) can be further simplified to

$$F_0(\lambda_i) \eta_{i,s_i} = G_0(\lambda_i) (\eta_{i,x} + \eta_{i,y}) + \sum_{s_j \in S/s_i} F_j(\lambda_i) \eta_{i,s_j} + \sum_{j=1}^m H_j(\lambda_i) \eta_{i,r_j}. \tag{B20}$$

The above analysis indicates that Eqs. (B11) and (B12) are actually the consequence of the characteristic equations $\mathcal{A}_1 \boldsymbol{\eta} = \lambda \boldsymbol{\eta}$. As a result, the eigenvalue satisfying $f_0(\lambda_i) = 0$ is determined by the local structure $local_G(V_1, E_1)$ of nodal set V_1 only. Likewise, the eigenvector associated with this eigenvalue is related to the local structure $local_G(V_1, E_1)$ formed by V_1 only; therefore, the eigencomponents of nodal set V/V_1 can be directly set to zero. Substituting the values of these components into Eq. (B20), we have that the left side of this equation is zero, so are the second and third terms on the right side. We, therefore, have

$$G_0(\lambda_i) (\eta_{i,x} + \eta_{i,y}) = 0 \text{ or } \eta_{i,x} + \eta_{i,y} = 0.$$

Apparently, for $f_0(\lambda_i) \neq 0$, the local structure $local_G(V_1, E_1)$ is not sufficient to determine this eigenvalue; therefore, it must be related to the structure outside nodal set V_1 . The eigenvectors of matrix \mathcal{A} can be classified into two categories. The first category is $f_0(\lambda_i) \neq 0$,

and the eigenvectors are determined by the eigencomponents not only of nodal set V_1 but also of the set V/V_1 . The eigencomponents of nodes x and y satisfy $\eta_{ix} = \eta_{iy}$. In the second category defined by $f_0(\lambda_i) = 0$, the eigenvectors are related to the eigencomponents of nodal set V_1 only, and the eigencomponents of nodal set V/V_1 are all zero. In this case, the eigencomponents of x and y satisfy $\eta_{ix} + \eta_{iy} = 0$.

If there are nodes x' and y' with the same SPV in nodal set $V_1/\{x, y\}$, their characteristic components have similar properties to those of nodes x and y . In particular, for the first type of eigenvectors, the eigencomponents of x' and y' satisfy $\eta_{ix'} = \eta_{iy'}$. For the second type, we have $\eta_{ix'} + \eta_{iy'} = 0$. If there is a node x'' in nodal set $V_1/\{x, y\}$, whose SPV is not equal to that of any other node in the set, then using the characteristic equations $\mathcal{A}_1 \eta = \lambda \eta$, the eigencomponents of x'' are represented by those of nodal set S . Similarly, consider the eigencomponents of node x'' for the two types of eigenvectors. In the first type, the value of the eigencomponent of x'' can be arbitrary. In the second type, according to Eq. (B14) or (B20), we have $\eta_{ix''} = 0$.

Since the SPVs of nodes x and y are equal, we can know that the eigencomponents of nodes x and y satisfy two types of relationships: when the eigencomponents are not determined by the local structure, the components of node x and y are equal; when the eigenvectors are determined by the local structure where nodes x and y are located, the sum of eigencomponents of nodes x and y is 0 and the eigencomponents of nodes outside the local structure are all 0. The second type of eigenvectors is called redundant eigenvectors, which is a necessary condition for node symmetry.¹⁸ Also, some scholars used the redundant eigenvectors to find symmetric structures in the network.^{18,19} SPV equality is a stronger necessary condition for node symmetry than redundant eigenvectors.

However, there may be some special structures in networks whose eigenvectors also satisfy the special property of redundant eigenvectors, but nodes in the structures are asymmetric, such as regular graphs (see Appendix C). Therefore, when we use the SPV method to find the symmetric structure, we need to discriminate the special network structures. Lemma 2 in the main text provides a rigorous way to identify symmetric nodes, and we can use it to re-identify the special structures. Almost all algorithms for finding symmetric structures in networks lack rigorous theoretical proofs.^{2,12,18,20,23} However, it does not affect the practicality of the algorithm.

APPENDIX C: A COMMON CAVEAT AMONG SOME EXISTING APPROACHES TO FINDING NETWORK SYMMETRIES

We have emphasized in the main text that our SPV method represents a necessary but not sufficient criterion for finding the symmetric nodes. Here, using the Frucht network⁵⁰ in Fig. 7(a), we argue that the same deficiency arises in three previous methods on finding symmetric nodes, namely, the redundant eigenvector method,^{18,19} the combinatorial algorithm incorporating the concept of minimally balanced coloring,²⁰ and the cluster synchronization method.^{2,12}

We consider the classic Frucht graph in Fig. 7(a) in the main text. To see if the redundant eigenvector method^{18,19} can find the symmetric nodes, we note that the degrees of all nodes in a regular

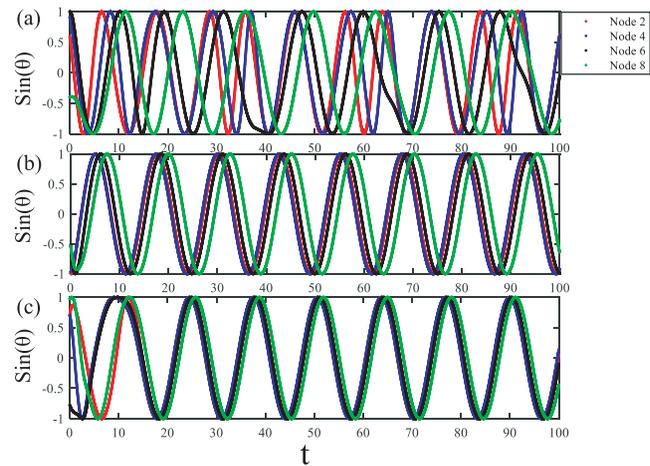


FIG. 9. Emergence of cluster and global synchronization in the Kuramoto model with synchronous frustration implemented on the Frucht network. Shown are the phase evolution of nodes 2, 4, 6, and 8 for (a) $\lambda = 0.2$, (b) $\lambda = 0.3$, and (c) $\lambda = 0.5$.

graph of size N are equal and the eigenvector corresponding to the largest eigenvalue is $\eta_N = [1, 1, \dots, 1]_N$. The rest of the eigenvectors satisfy the equality $\sum_{j=1}^N \eta_{ij} = 0$ ($i = 1, 2, \dots, N - 1$), where η_{ij} is the j th component of the i th eigenvector. For a small regular graph, such as a 2-regular graph with three nodes (a triangle), a 2-regular graph with 4 nodes (a quadrilateral), etc., each node is a symmetric node. However, for a larger regular graph, nodes are often asymmetric, as exemplified by the Frucht network in Fig. 7(a) where no nodes are symmetric. By the redundant eigenvector method, the eigencomponents of the redundant eigenvectors corresponding to the symmetric nodes are those whose sum is zero, while the remaining eigencomponents are all zero. Except for the eigenvector corresponding to the largest eigenvalue, all eigenvectors of a regular graph satisfy this property. The method would then give the wrong result that all the nodes in the Frucht network are symmetric nodes.

For the combinatorial algorithm incorporating the concept of minimally balanced coloring,²⁰ we note that all nodes are first assigned the same color, and the colors are then reassigned according to the adjacency. In particular, if the neighbors of two nodes have the same color and the numbers of their neighbors with the same color are identical, then the two nodes are regarded as belonging to the same class and are assigned the same color. The process continues until color changes no longer occur. Nodes with the same color are regarded as symmetric to each other. However, for the Frucht network, regardless of the number of iterations, all nodes would be deemed to belong to the same class. The method would then lead to the incorrect result that all nodes are symmetric to each other.

For the Frucht network, the method based on cluster synchronization to determine node symmetry fails too. To demonstrate this, we use the Kuramoto model with synchronous frustration,¹²

$$\dot{\theta} = \omega_i + \lambda \sum_{j=1}^N a_{ij} \sin(\theta_j - \theta_i - \alpha), \quad (C1)$$

where (a_{ij}) are the elements of the network adjacency matrix, ω_i and θ_i are the frequency and phase of node i , respectively, λ is the coupling strength, and α is the phase frustration parameter. We set $\alpha = 0.02$, $\omega_i \in (0, 1)$, $i = 1, 2, \dots, 12$, and the initial phases $\theta_i \in (-\pi, \pi)$, $i = 1, 2, \dots, 12$. Figure 9 presents the phase-time plots of randomly selected nodes (2,4,6,8) for different values of the coupling strength. For $\lambda = 0.2$, there is no synchronization among the nodes, as shown in Fig. 9(a). For $\lambda = 0.3$, phase synchronization among the nodes occurs, where the phases between nodes 2 and 4 are closer, as shown in Fig. 9(b). According to this cluster synchronization behavior, nodes 2 and 4 would be deemed symmetric, which is wrong. For $\omega = 0.5$, global synchronization among all nodes is achieved, implying that all nodes in the Frucht network are symmetric, which is wrong too.

APPENDIX D: STRUCTURAL PROPERTIES OF SIX REAL-WORLD NETWORKS AND THEIR MOTIFS

Table III lists the structural properties of the six real-world networks studied. Table IV lists all the symmetric motifs identified by the SPV algorithm.

APPENDIX E: SIR MODEL AND KENDALL'S τ CORRELATION COEFFICIENT

1. SIR model

The standard SIR (susceptible–infected–recovered) model⁵⁵ has been widely used in simulating epidemic spreading and information propagation/diffusion dynamics. At each time step, an infected node makes contact with its neighbors and each susceptible neighbor is infected with the probability β . The infected node enters into the state of recovery or removal with the probability λ . It is convenient to set $\lambda = 1$.

To quantify the spreading influence of node i , we initiate the spreading process with i being the infected seed and all other nodes being susceptible. The spreading process stops when there are no infected nodes. We record the fraction of recovered nodes R_i as the spreading influence of the seed node. According to the heterogeneous mean-field theory,^{56,57} the epidemic threshold in the SIR model is given by $\beta_c = \langle k \rangle / (\langle k^2 \rangle - \langle k \rangle)$, where $\langle k \rangle$ is the average degree and $\langle k^2 \rangle$ is the second moment of the random degree variable. In our simulations, we set $\beta = 1.2\beta_c, 1.5\beta_c, 2\beta_c$. To eliminate

TABLE III. Structural properties of six real-world networks.

Networks	$ V $	$ E $	$\langle k \rangle$	$\langle d \rangle$	c	r
Facebook	4039	88 234	43.69	3.69	0.606	0.064
Odlis.net	2909	16 388	11.27	3.17	0.296	-0.173
CA-GrQc	4158	13 422	6.46	6.05	0.557	0.639
p2p-Gnutella	6301	20 779	6.60	4.64	0.011	0.035
Yeast	2224	6 609	5.94	4.38	0.138	-0.105
Wiki-Vote	7066	100 736	28.51	3.25	0.142	0.083

the fluctuations of R_i , we average the results over 5000 independent runs.

2. Kendall's τ correlation coefficient

The Kendall's τ correlation coefficient is a widely used statistical measure for ranking the correlation. Consider two sequences associated with $|V|$ samples: $x = (x_1, x_2, \dots, x_{|V|})$ and $y = (y_1, y_2, \dots, y_{|V|})$. A pair of orders (x_i, x_j) and (y_i, y_j) are concordant if both $x_i > x_j$ and $y_i > y_j$ or if both $x_i < x_j$ and $y_i < y_j$ are discordant if $x_i > x_j$ and $y_i < y_j$ or if both $x_i < x_j$ and $y_i > y_j$. If $x_i = x_j$ or $y_i = y_j$, the pair is neither concordant nor discordant. As there are $|V|(|V| - 1)/2$ pairs of samples in total, the Kendall's τ correlation coefficient is defined as

$$\tau = 2(\delta^+ - \delta^-) / [|V|(|V| - 1)], \tag{E1}$$

where δ^+ and δ^- are the numbers of concordant and discordant pairs, respectively.

Tables V and VI present the Kendall's τ correlation coefficient between node influence R and eight indices for $\beta = 1.5\beta_c$ and $\beta = 2\beta_c$, respectively.

APPENDIX F: IMPACT OF DIMENSION OF TRUNCATED SPV AND ITS COMPONENT WEIGHTS ON NODAL PROPAGATION INFLUENCE

In general, the influence of a node depends on the adjacency of different orders, corresponding to the components of each order in the SPV. The contribution of each order of adjacency is different.⁴⁹ Motivated by this, we further study the relative importance of each order component in the SPV to the nodal propagation influence. We first set the dimension of \tilde{S}_i^* to be n : $\tilde{S}_i^* = (L_i^1, L_i^2, \dots, L_i^n)$ for

TABLE IV. Symmetric motifs of the six real networks found through SPVs.

Network	Symmetric motif
Facebook	$(O_2^*)^{33} + (O_2)^{18} + (O_3^*)^9 + O_3 + (O_4^*)^4 + O_4 + (O_5^*)^2 + O_5 + (O_7)^2 + O_9 + O_{11} + O_{13} + O_{14} + (O_4^*)^2 + O_6^3$
Odlis.net	$(O_2^*)^{14} + (O_2)^{30} + (O_3)^7 + (O_4)^2 + O_8 + O_{15} + (O_2 \xrightarrow{1} O_2^*)$
CA-GrQc	$(O_2^*)^{279} + (O_2)^{128} + (O_3^*)^{57} + (O_3)^{26} + (O_4^*)^{17} + (O_4)^{10} + (O_5^*)^7 + (O_5)^2 + (O_6^*)^5 + (O_6)^3 + O_7^* + O_8^* + O_9^* + O_{10}^*$
p2p-Gnutella	$O_2^* + (O_2)^{277} + (O_3)^{77} + (O_4)^{28} + (O_5)^9 + O_6 + (O_7)^3 + O_8 + O_9 + (O_{10})^4 + O_{26} + O_{32}$
Yeast	$(O_2^*)^2 + (O_2)^{87} + (O_3)^{34} + (O_4)^{22} + (O_5)^9 + (O_6)^9 + (O_7)^5 + (O_8)^3 + O_9 + (O_{10})^2$
Wiki-Vote	$(O_2)^{235} + (O_3)^{111} + (O_4)^{68} + (O_5)^{37} + (O_6)^{15} + (O_7)^{13} + (O_8)^5 + (O_9)^4 + (O_{10})^2 + (O_{12})^2 + (O_{13})^2 + O_{14} + O_{15} + O_{17} + O_{19} + O_{20} + O_{24} + O_{26} + O_{29} + O_{32}$

TABLE V. Kendall's τ correlation coefficient between node influence R and eight indices for $\beta = 1.5\beta_c$. Boldface denotes the largest T value in each row.

Networks	Degree	H-index	k-core	Closeness	Betweenness	Eigenvector	SPV	SPV($\tilde{N} = N(k)$)
Facebook	0.6187	0.6501	0.6645	0.4633	0.4219	0.6715	0.7633	0.7602
Odlis.net	0.6729	0.7004	0.7160	0.7915	0.5203	0.8505	0.8640	0.8693
CA-GrQc	0.5620	0.5810	0.5762	0.6284	0.3273	0.6477	0.6184	0.6168
p2p-Gnutella	0.7561	0.7957	0.7862	0.8813	0.6910	0.8228	0.8286	0.8330
Yeast	0.6932	0.7390	0.7424	0.8345	0.5762	0.8121	0.8599	0.8657
Wiki-Vote	0.8392	0.8465	0.8476	0.8543	0.7624	0.8853	0.8888	0.8874

TABLE VI. Kendall's τ correlation coefficient between the nodal influence R and eight indices for $\beta = 2\beta_c$. Boldface denotes the largest T value in each row.

Networks	Degree	H-index	k-core	Closeness	Betweenness	Eigenvector	SPV	SPV($\tilde{N} = N(k)$)
Facebook	0.6293	0.6635	0.6786	0.4955	0.4262	0.6579	0.7712	0.7685
Odlis.net	0.6651	0.6966	0.7160	0.7864	0.5101	0.8626	0.8632	0.8678
CA-GrQc	0.5146	0.5372	0.5358	0.6919	0.3172	0.6624	0.5718	0.5695
p2p-Gnutella	0.8099	0.8537	0.8309	0.9183	0.7385	0.7608	0.8794	0.8712
Yeast	0.7416	0.7900	0.7888	0.8152	0.6022	0.7689	0.8666	0.8669
Wiki-Vote	0.8680	0.8753	0.8744	0.8568	0.7789	0.8900	0.9038	0.9051

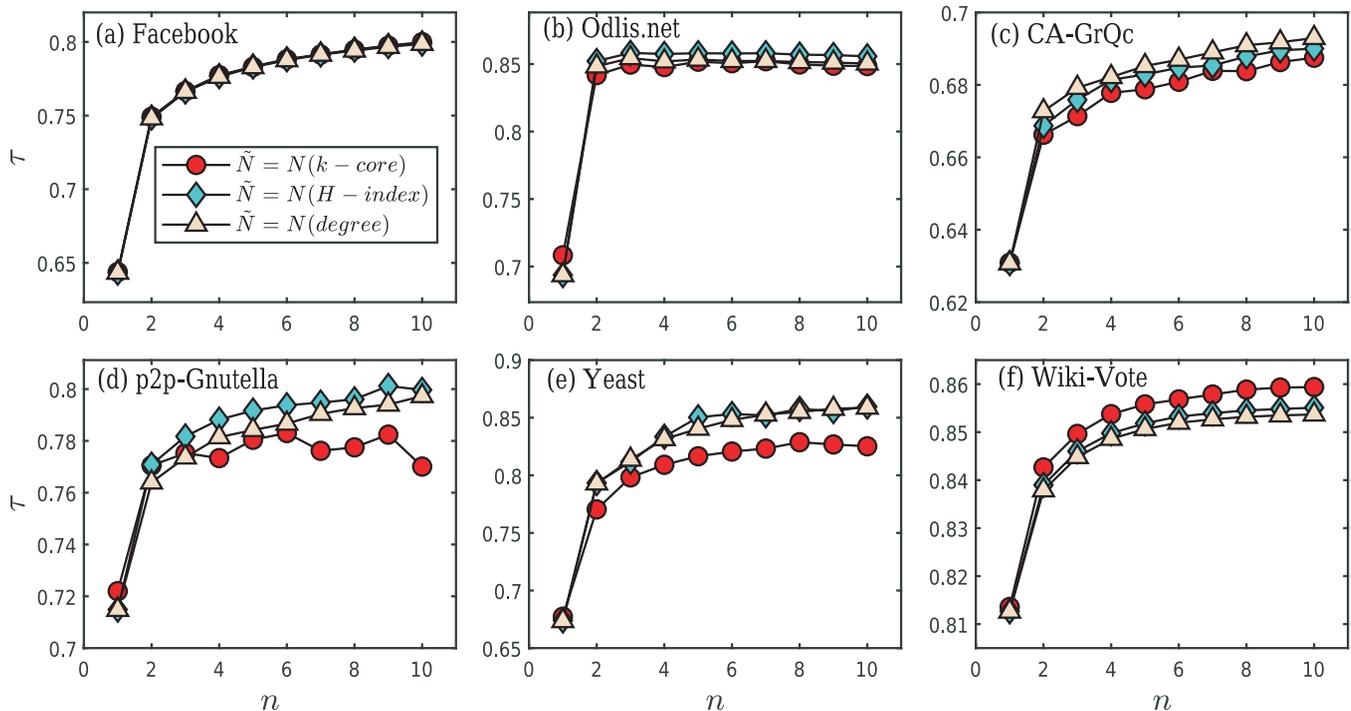


FIG. 10. Kendall's τ correlation coefficient between the SPV-based cluster centrality and nodal influence R as a function of the dimension n of the SPV for $\beta = 1.2\beta_c$.

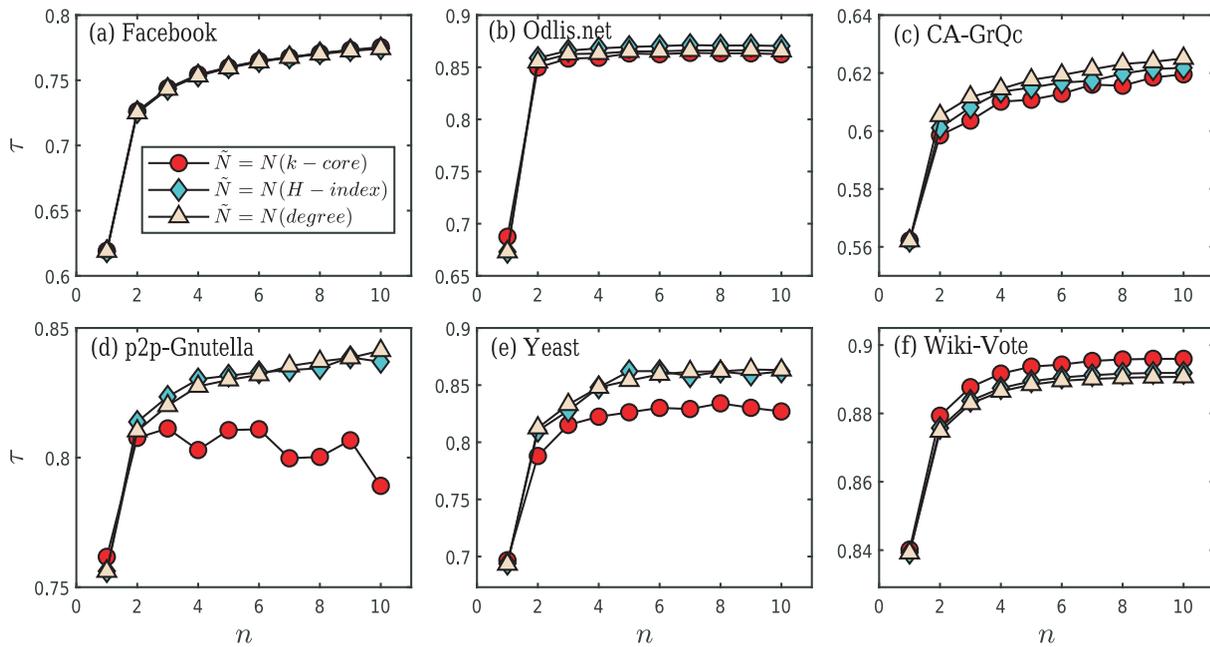


FIG. 11. Kendall's τ correlation coefficient between the SPV-based cluster centrality and nodal influence R as a function of the dimension n of the SPV for $\beta = 1.5\beta_c$.

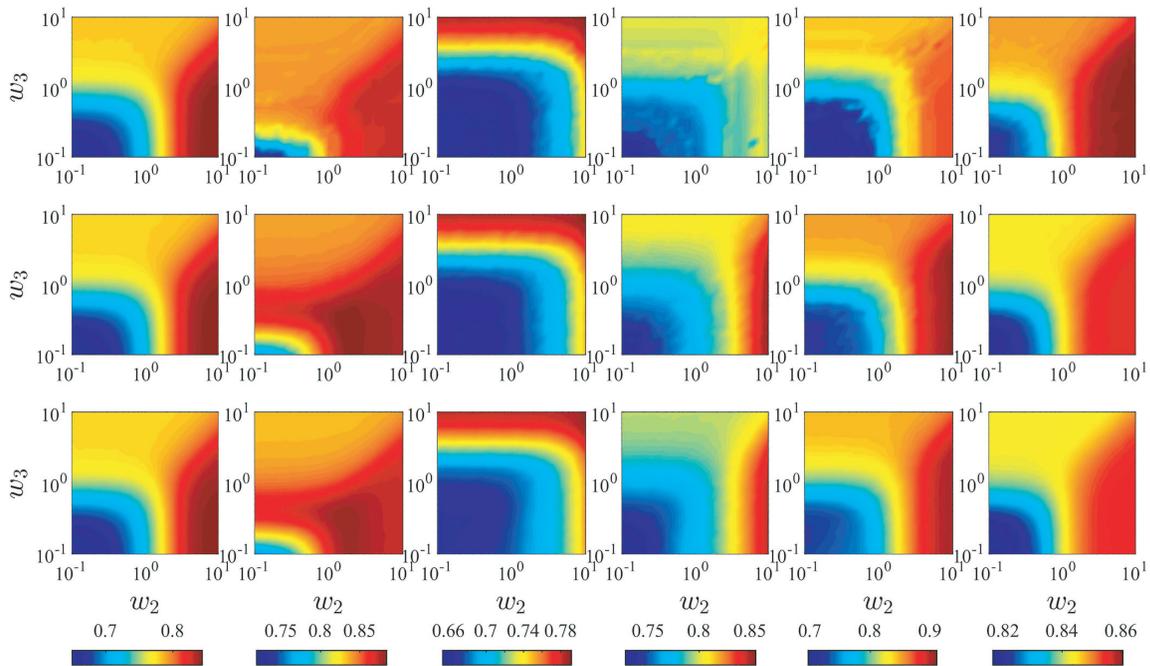


FIG. 12. Effect of SPV component weights on nodal propagation influence. Shown is Kendall's τ correlation coefficient between the SPV-based cluster centrality and nodal influence R vs the weights of the second and third components of the truncated SPV (i.e., the second- and third-order structural positions), denoted as w_2 and w_3 , respectively, for $\beta = 1.2\beta_c$. From left to right, the networks in each column are Facebook, Odlis.net, CA-GrQc, p2p-Gnutella, Yeast, and Wiki-Vote. The scale of the coarse-grained network in each row from top to bottom is $\tilde{N} = N(k - core)$, $\tilde{N} = N(H - index)$, and $\tilde{N} = N(k)$.

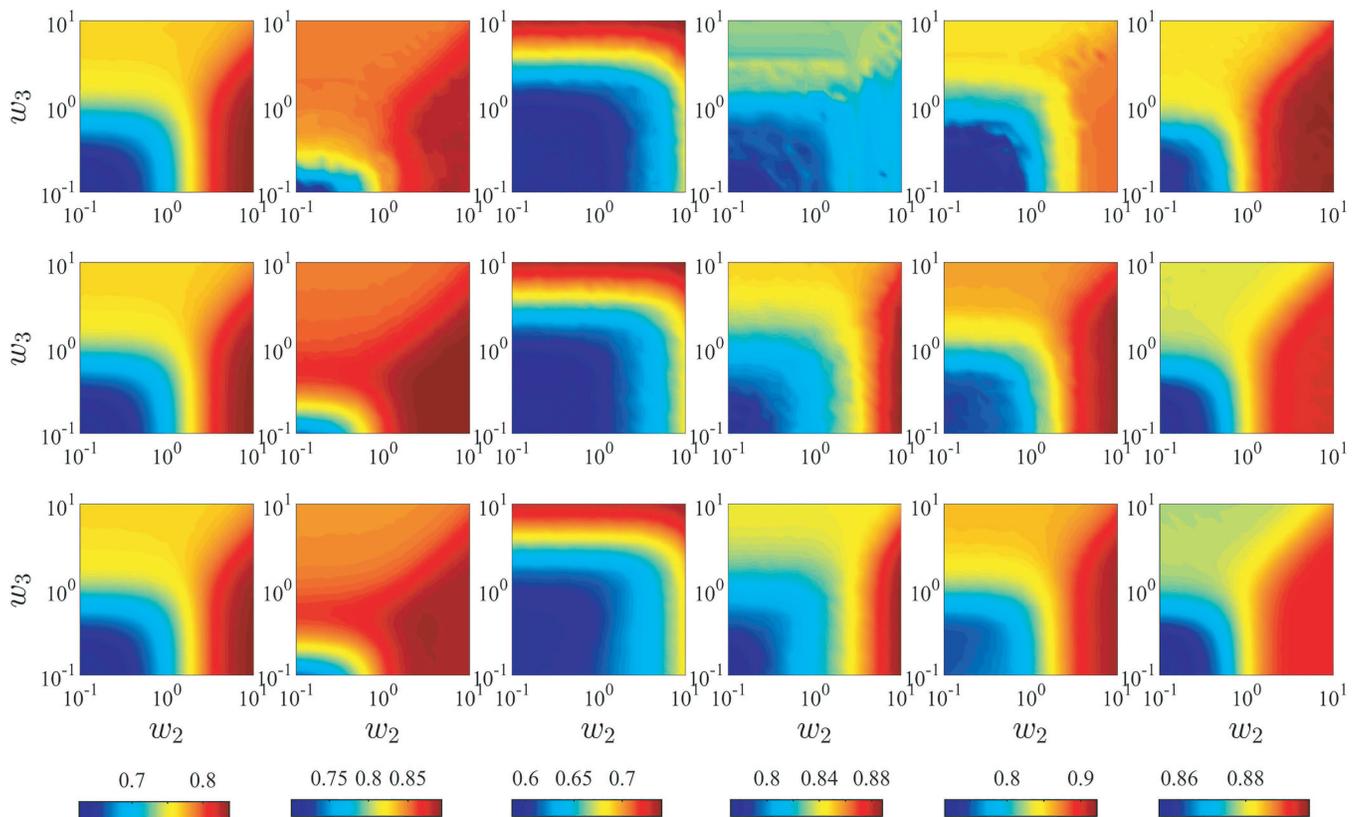


FIG. 13. Effect of SPV component weights on nodal propagation influence. The legends are the same as those in supplementary Fig. 12 except that the value of parameter β is different: $\beta = 1.5\beta_c$.

$n = 1, 2, \dots, 10$. The scale of the coarse-grained network is set to be $\tilde{N} = N(k), N(H\text{-index}),$ and $N(k\text{-core}),$ respectively. To be representative, we consider two cases where the propagation rate is above the threshold: $\beta = 1.2\beta_c$ and $1.5\beta_c$. We then assign weights to different order components of $\tilde{\mathbf{S}}_i^*, \tilde{\mathbf{S}}_i^* = (w_1 L_i^1, w_2 L_i^2, \dots, w_n L_i^n)$.

Figures 10 and 11 show that, when using the SPV-based cluster centrality index to measure the influence of a node, the performance improves with the dimension of the SPV and then saturates. For example, if the SPV is one-dimensional, the only information it contains is the nodal degree; therefore, the performance is relatively poor. However, a three-dimensional SPV can already predict the nodal influence well, but a higher-dimensional SPV only leads to incremental improvement in the performance. In Figs. 10(d) and 11(d), for $\tilde{N} = N(k - \text{core}),$ the performance curve is irregular due mainly to the reason that the scale of the coarse-grained network is too small; therefore, the performance of the SPV-based cluster centrality is not stable. In this case, increasing the dimension of the SPV does not improve the performance significantly. When the scale of the coarse-grained network is large, as the dimensionality increases, the level of classification becomes more detailed. In this case, the performance of cluster centrality tends to improve with the dimension before reaching a plateau.

We also study the relative importance of each component of the SPV to nodal propagation influence. Typically, the first three components of the SPV suffice to quantify the nodal influence. Generally, the nodal degree (the first-order component of SPV) to some extent reflects the propagation influence of nodes. However, the extent to which the two-, three-, and a higher-dimensional structure of the SPV reflect the influence of nodes is worth elucidating. For this purpose, we study the relative importance of the two- and three-dimensional SPVs to nodal influence. In particular, we fix $w_1 = 1$ and adjust (w_2, w_3) in the range $0.1 \leq (w_2, w_3) \leq 10$. Supplementary Figs. 12 and 13 show that, except for the CA-GrQc network, when the SPV-based cluster centrality is used to quantify the nodal influence, the optimal region is $w_2 > 1, w_3 < 1$. We also find that increasing the weight of the second-order structural position has a larger impact on the nodal influence than increasing the weight of the third-order structural position. In the CA-GrQc network, the optimal area is in $w_3 > 1,$ and increasing the weight of the third-order structural position has a larger impact on the ranking performance of cluster centrality. These results indicate that, when the average path length of the network is small, the contribution of the second-order adjacency relationship to the nodal influence is greater than that of the third-order relationship. When the

average path length of the network is large (e.g., the CA-GrQc network whose average path length is $\langle d \rangle = 6.05$), the contribution of the node's third-order adjacency relationship to the nodal influence is greater than that of the second-order relationship. (The average path lengths of the six real-world networks are listed in Table III.)

REFERENCES

- ¹I. Stewart, M. Golubitsky, and M. Pivato, "Symmetry groupoids and patterns of synchrony in coupled cell networks," *SIAM J. Appl. Dyn. Syst.* **2**, 609–646 (2003).
- ²L. M. Pecora, F. Sorrentino, A. M. Hagerstrom, T. E. Murphy, and R. Roy, "Cluster synchronization and isolated desynchronization in complex networks with symmetries," *Nat. Commun.* **5**, 4079 (2014).
- ³F. Sorrentino, L. M. Pecora, A. M. Hagerstrom, T. E. Murphy, and R. Roy, "Complete characterization of the stability of cluster synchronization in complex dynamical networks," *Sci. Adv.* **2**, e1501737 (2016).
- ⁴F. Sorrentino and L. Pecora, "Approximate cluster synchronization in networks with symmetries and parameter mismatches," *Chaos* **26**, 094823 (2016).
- ⁵Y. Wang, L. Wang, H. Fan, and X. Wang, "Cluster synchronization in networked nonidentical chaotic oscillators," *Chaos* **29**, 093118 (2019).
- ⁶F. Della Rossa, L. Pecora, K. Blaha, A. Shirin, I. Klickstein, and F. Sorrentino, "Symmetries and cluster synchronization in multilayer networks," *Nat. Commun.* **11**, 3179 (2020).
- ⁷L. Wang, Y. Guo, Y. Wang, H. Fan, and X. Wang, "Pinning control of cluster synchronization in regular networks," *Phys. Rev. Res.* **2**, 023084 (2020).
- ⁸P. Erdős and A. Rényi, "On random graphs I," *Publ. Math. Debrecen* **6**, 290–291 (1959).
- ⁹B. D. MacArthur and R. J. Sánchez-García, "Spectral characteristics of network redundancy," *Phys. Rev. E* **80**, 026117 (2009).
- ¹⁰B. D. MacArthur, R. J. Sánchez-García, and J. W. Anderson, "Symmetry in complex networks," *Discrete Appl. Math.* **156**, 3525–3531 (2008).
- ¹¹Y. Xiao, M. Xiong, W. Wang, and H. Wang, "Emergence of symmetry in complex networks," *Phys. Rev. E* **77**, 066108 (2008).
- ¹²V. Nicosia, M. Valencia, M. Chavez, A. Diaz-Guilera, and V. Latora, "Remote synchronization reveals network symmetries and functional modules," *Phys. Rev. Lett.* **110**, 174102 (2013).
- ¹³F. Sorrentino, A. B. Siddique, and L. M. Pecora, "Symmetries in the time-averaged dynamics of networks: Reducing unnecessary complexity through minimal network models," *Chaos* **29**, 011101 (2019).
- ¹⁴Y. Xiao, B. D. MacArthur, H. Wang, M. Xiong, and W. Wang, "Network quotients: Structural skeletons of complex systems," *Phys. Rev. E* **78**, 046102 (2008).
- ¹⁵R. J. Sánchez-García, "Exploiting symmetry in network analysis," *Commun. Phys.* **3**, 87 (2020).
- ¹⁶F. Chung, L. Lu, T. G. Dewey, and D. J. Galas, "Duplication models for biological networks," *J. Comp. Biol.* **10**, 677–687 (2003).
- ¹⁷T. Nishikawa and A. E. Motter, "Network-complement transitions, symmetries, and cluster synchronization," *Chaos* **26**, 094818 (2016).
- ¹⁸L. Marrec and S. Jalan, "Analysing degeneracies in networks spectra," *Europhys. Lett.* **117**, 48001 (2017).
- ¹⁹P. Shinde, L. Marrec, A. Rai, A. Yadav, R. Kumar, M. Ivanchenko, A. Zaikin, and S. Jalan, "Symmetry in cancer networks identified: Proposal for multicancer biomarkers," *Netw. Sci.* **7**, 541–555 (2019).
- ²⁰I. Belykh and M. Hasler, "Mesoscale and clusters of synchrony in networks of bursting neurons," *Chaos* **21**, 016106 (2011).
- ²¹B. D. McKay, "Practical graph isomorphism," *Congr. Numer.* **30**, 45–87 (1981).
- ²²B. D. McKay, "Backtrack programming and the graph isomorphism problem," Ph.D. thesis (University of Melbourne, 1976).
- ²³B. D. McKay, "Computing automorphisms and canonical labellings of graphs," in *Combinatorial Mathematics* (Springer, 1978), pp. 223–232.
- ²⁴J. M. Reitz, *Online Dictionary for Library and Information Science* (Western Connecticut State University, Danbury, CT, 2014).
- ²⁵M. Ripeanu and I. Foster, "Mapping the Gnutella network: Macroscopic properties of large-scale peer-to-peer systems," in *International Workshop on Peer-to-Peer Systems* (Springer, 2002), pp. 85–93.
- ²⁶J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *NIPS* (Curran Associates Inc., 2012), Vol. 2012, pp. 548–56.
- ²⁷J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Trans. Knowl. Discov. Data* **1**, 2 (2007).
- ²⁸D. 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**, 2443–2450 (2003).
- ²⁹J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proceedings of the 19th International Conference on World Wide Web* (ACM, 2010), pp. 641–650.
- ³⁰S. Xu, P. Wang, and J. Lü, "Iterative neighbour-information gathering for ranking nodes in complex networks," *Sci. Rep.* **7**, 1–13 (2017).
- ³¹L. Lü, C.-H. Jin, and T. Zhou, "Similarity index based on local paths for link prediction of complex networks," *Phys. Rev. E* **80**, 046122 (2009).
- ³²J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proc. Natl. Acad. Sci. U.S.A.* **102**, 16569–16572 (2005).
- ³³M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Oxford, UK, 2010).
- ³⁴S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, "K-core organization of complex networks," *Phys. Rev. Lett.* **96**, 040601 (2006).
- ³⁵P. Bonacich, "Power and centrality: A family of measures," *Am. J. Sociol.* **92**, 1170–1182 (1987).
- ³⁶L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry* **40**, 35–41 (1977).
- ³⁷P. S. Arunachalam, A. C. Walls, N. Golden, C. Atyeo, S. Fischinger, C. Li, P. Aye, M. J. Navarro, L. Lai, V. V. Edara *et al.*, "Adjuvanting a subunit COVID-19 vaccine to induce protective immunity," *Nature* **594**, 253–258 (2021).
- ³⁸M. Feng, S.-M. Cai, M. Tang, and Y.-C. Lai, "Equivalence and its invalidation between non-Markovian and Markovian spreading dynamics on complex networks," *Nat. Commun.* **10**, 3748 (2019).
- ³⁹T. C. Silva, S. R. S. Souza, and B. M. Tabak, "Monitoring vulnerability and impact diffusion in financial networks," *J. Econ. Dyn. Cont.* **76**, 109–135 (2017).
- ⁴⁰Z.-H. Lin, M. Feng, M. Tang, Z. Liu, C. Xu, P. M. Hui, and Y.-C. Lai, "Non-Markovian recovery makes complex networks more resilient against large-scale failures," *Nat. Commun.* **11**, 2490 (2020).
- ⁴¹A. E. Motter and Y.-C. Lai, "Cascade-based attacks on complex networks," *Phys. Rev. E* **66**, 065102 (2002).
- ⁴²L. Zhao, K. Park, and Y.-C. Lai, "Attack vulnerability of scale-free networks due to cascading breakdown," *Phys. Rev. E* **70**, 035101 (2004).
- ⁴³L. Zhao, K. Park, Y.-C. Lai, and N. Ye, "Tolerance of scale-free networks against attack-induced cascades," *Phys. Rev. E* **72**, 025104 (2005).
- ⁴⁴B. Schäfer, D. Witthaut, M. Timme, and V. Latora, "Dynamically induced cascading failures in power grids," *Nat. Commun.* **9**, 1975 (2018).
- ⁴⁵R.-R. Liu, C.-X. Jia, and Y.-C. Lai, "Asymmetry in interdependence makes a multilayer system more robust against cascading failures," *Phys. Rev. E* **100**, 052306 (2019).
- ⁴⁶A. Avena-Koenigsberger, B. Misić, and O. Sporns, "Communication dynamics in complex brain networks," *Nat. Rev. Neurosci.* **19**, 17 (2018).
- ⁴⁷A. G. Reza and J.-K. K. Rhee, "Nonlinear equalizer based on neural networks for PAM-4 signal transmission using DML," *IEEE Photon. Technol. Lett.* **30**, 1416–1419 (2018).
- ⁴⁸S. Kojaku and N. Masuda, "Core-periphery structure requires something else in the network," *New J. Phys.* **20**, 043012 (2018).
- ⁴⁹Y. Liu, M. Tang, T. Zhou, and Y. Do, "Identify influential spreaders in complex networks, the role of neighborhood," *Physica A* **452**, 289–298 (2016).
- ⁵⁰R. Frucht, "Graphs of degree three with a given abstract group," *Can. J. Math.* **1**, 365–378 (1949).
- ⁵¹B. Kivva, "On the spectral gap and the automorphism group of distance-regular graphs," *J. Comb. Theory Ser. B* **149**, 161–197 (2021).
- ⁵²L. Babai, X. Chen, X. Sun, S.-H. Teng, and J. Wilmes, "Faster canonical forms for strongly regular graphs," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science* (IEEE, 2013), pp. 157–166.
- ⁵³B. Bollobás, *Modern Graph Theory* (Springer Science & Business Media, 2013), Vol. 184.

⁵⁴J. J. Rotman, *An Introduction to the Theory of Groups* (Springer Science & Business Media, 2012), Vol. 148.

⁵⁵R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, "Epidemic processes in complex networks," *Rev. Mod. Phys.* **87**, 925 (2015).

⁵⁶M. E. Newman, "Spread of epidemic disease on networks," *Phys. Rev. E* **66**, 016128 (2002).

⁵⁷C. Castellano and R. Pastor-Satorras, "Thresholds for epidemic spreading in networks," *Phys. Rev. Lett.* **105**, 218701 (2010).