

# Convex skeletons of complex networks

Lovro Šubelj\*

*University of Ljubljana, Faculty of Computer and Information Science, Ljubljana, Slovenia*

(Dated: Monday 18<sup>th</sup> September, 2017)

A convex network can be defined as a network such that every connected induced subgraph includes all the shortest paths between its nodes. Fully convex network would therefore be a collection of cliques stitched together in a tree. In this paper, we study the largest high-convexity part of empirical networks obtained by removing the least number of edges, which we call a convex skeleton. A convex skeleton is a generalization of a spanning tree in which each edge can be replaced by a clique of arbitrary size. We present different approaches for extracting convex skeletons and apply them to social collaboration and protein interactions networks, autonomous systems graphs and food webs. We show that the extracted convex skeletons retain the degree distribution, clustering, connectivity and also distances between the nodes, while making the shortest paths between the nodes largely unique. Moreover, in the Slovenian computer scientists coauthorship network, a convex skeleton retains the strongest ties between the authors, differently from a spanning tree or high-betweenness backbone and high-salience skeleton. A convex skeleton thus represents a simple definition of a network backbone with applications in coauthorship and other social collaboration networks.

## I. INTRODUCTION

An integral part of modern network analysis is understanding the structural properties of real empirical networks through different statistical models and techniques. One such example is the study of shortest paths between the nodes in a network, which has proven very useful in the past. Milgram's small-world experiment is a well-known empirical example [1]. As another example, the frequently used measure of betweenness centrality is also based on the number of shortest paths [2].

A related, but different concept is that of convexity studied in metric graph theory [3–6]. Convexity is a property of a part of a mathematical object that includes all the shortest paths between its units. In the case of graphs or networks, a connected induced subgraph is said to be convex if every shortest path between the nodes of the subgraph lies entirely within the subgraph. Note that while convexity demands the inclusion of all shortest paths between the nodes of the subgraph it makes no implications regarding their length nor number.

Two methodologies for analyzing convexity in networks have been proposed in the literature. Everett and Seidman [7] define the hull number of a network as the size of the smallest subset of nodes whose convex hull is the entire network. Here, convex hull of a subset of nodes is the smallest convex subgraph including those nodes. Furthermore, Marc and Šubelj [8, 9] define network convexity through the expansion of randomly grown subsets of nodes to convex subgraphs. In contrast to the hull number, network convexity has a polynomial computational complexity and has already been successfully used in characterizing the structure of empirical networks. In particular, it measures whether a network is either tree-like or clique-like on all different scales.

In this paper, we study the largest high-convexity part of empirical networks obtained by removing the least number of edges, which we call a convex skeleton. As demonstrated below, a convex skeleton is a collection of cliques or nearly cliques that are stitched together in a tree-like arrangement. We present different approaches for extracting convex skeletons and apply them to social collaboration and protein interactions networks, autonomous systems graphs and food webs. We show that the extracted convex skeletons retain most important structural properties of networks, differently from some popular network backbone techniques [10, 11] such as a spanning tree or high-betweenness backbone [2] and high-salience skeleton [12]. At the same time, a convex skeleton implies a very simple structure consisting only of cliques and a tree. It seems to provide a particularly reasonable abstraction of social collaboration networks, with applications in coauthorship networks and also elsewhere.

The rest of the paper is structured as follows. In Section II, we provide a formal definition of network convexity and review the main results in Ref. [8] relevant for this work. Next, in Section III, we study the robustness of convexity under random perturbations of network structure. Section IV introduces the concept of a convex skeleton and presents different approaches for extracting convex skeletons from empirical networks. These are applied to networks of various type and origin in Section IV, while in Section V we more thoroughly analyse the convex skeletons extracted from the Slovenian computer scientists coauthorship network. Section VI concludes the paper and gives some final remarks on convex skeletons.

## II. CONVEXITY IN NETWORKS

The definition of convexity in networks adopted here is based on the concept of a convex subgraph [3–6]. Let  $S$  be a connected subset of nodes in a network. The subgraph induced by the nodes in  $S$  is convex if all the

---

\* lovro.subelj@fri.uni-lj.si

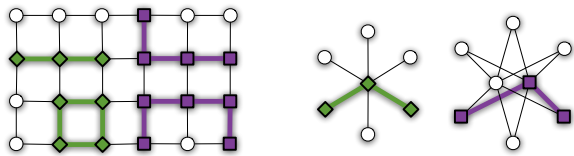


FIG. 1. (*left*) Rectangular lattice with highlighted convex ( $\blacklozenge$ ) and non-convex ( $\blacksquare$ ) subgraphs. (*right*) Two star graphs in which every connected triplet of nodes induces either a convex or non-convex subgraph, respectively.

shortest paths or geodesics between any two nodes in  $S$  are entirely included within  $S$ . For instance, in a tree or a complete graph, every connected subset of nodes  $S$  induces either a convex subtree or a clique, which is always convex. Note that convex subgraphs must include all the shortest paths between its nodes, whereas a subgraph including at least one shortest path between each pair of nodes is called an isometric subgraph.

The left side of Fig. 1 shows different examples of convex and non-convex subgraphs of a rectangular lattice, where any convex subgraph is necessarily a rectangular sublattice. Furthermore, the right side of Fig. 1 shows two seemingly similar, but different graphs consisting of a star graph and a two-star arrangement. While any connected induced subgraph is convex in the star graph, no non-trivial subgraph is convex in the second graph. Convex subgraphs thus characterize networks beyond connected and induced subgraphs [13], also known as motifs [14] and graphlets [15] in networks jargon.

To ease the terminology, we say that a connected subset of nodes  $S$  is convex if it induces a convex subgraph in a given network. Now, assume the subset  $S$  is not convex. The smallest convex subgraph including all the nodes in  $S$  is called the convex hull  $\mathcal{H}(S)$  [3, 4], which is uniquely defined. Obviously,  $\mathcal{H}(S) = S$  only when  $S$  is a convex subset. For example, the convex hull  $\mathcal{H}(S)$  of any connected triplet of nodes  $S$  in the two-star graph in Fig. 1 spans the whole graph, whereas  $\mathcal{H}(S) = S$  for any connected subset  $S$  in the star graph.

The above suggests a possible definition of convexity in networks [8, 9]. One randomly grows connected induced subgraphs or subsets of nodes  $S$  one node at a time and expands them to their convex hulls  $\mathcal{H}(S)$  if needed. Any subset  $S$  generated by this expansion procedure therefore induces a convex subgraph. By observing the growth of subsets  $S$ , one can measure convexity in a network. In networks in which any connected induced subgraph is likely to be convex, the subsets  $S$  would grow slowly, a single node at a time. Such networks are termed convex networks [8]. On the other hand, in non-convex networks such as food webs and random graphs as we show below, the subsets  $S$  expand rapidly and include all the nodes in a network after only a few steps of the procedure.

The details of the convex expansion procedure are as follows. One starts by initializing the subset  $S$  with a randomly selected seed node. Then, on each step of the

procedure, a new node is chosen by following a random edge leading outside of  $S$ . Hence, a node  $i \notin S$  is selected with the probability  $\propto |\Gamma_i \cap S|$ , where  $\Gamma_i$  is the set of its neighbors. Last, the newly selected node  $i$  is added to the subset  $S$ , which is finally expanded to a convex subset as  $\mathcal{H}(S \cup \{i\})$ . The entire process is repeated until the subset  $S$  includes all the nodes in a network. The algorithmic complexity of a single run of the expansion procedure in a network with  $n$  nodes and  $m$  edges is  $\mathcal{O}(nm)$  [16].

Fig. 2 demonstrates the growth of convex subgraphs in the first two steps  $t \leq 2$  of the procedure,  $t \in \{0, \dots, n-1\}$ . Note that time  $t$  measures the number of expansion steps, not including the initialization step. Initially, the subset  $S$  contains a single node  $i$ ,  $S = \{i\}$ , which is a convex subset. On the first step, a random neighbor  $j \in \Gamma_i$  is added,  $S = \{i, j\}$ , which is still convex. On the next step, another neighbor  $k \in \Gamma_i \cup \Gamma_j \setminus \{i, j\}$  is selected,  $S = \{i, j, k\}$ . In a network that is (locally) tree-like, the subset  $S$  induces a convex subtree. Similarly, in a (locally) clique-like network, the subset  $S$  induces a triangle and is therefore convex. In any other case, meaning that the subgraph is not a triangle and neither the network is tree-like, the expansion procedure would have to include all the nodes  $\Gamma_i \cup \Gamma_k \setminus \{j\}$  in the right side of Fig. 2 ( $\blacksquare$ ) to keep the subset  $S$  convex. This may, however, demand additional nodes and so on, possibly resulting in a sudden expansion of the subset  $S$ .

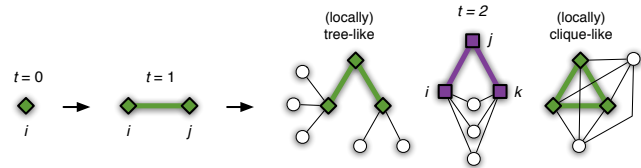


FIG. 2. Growth of convex subgraphs ( $\blacklozenge$ ) in the first two steps  $t \leq 2$  of the convex expansion procedure. Notice that non-convex subgraphs ( $\blacksquare$ ) indicate the absence of a (locally) tree-like or clique-like structure.

The first few steps of the convex expansion procedure thus characterize whether a network is locally either tree-like or clique-like. Furthermore, in the later steps, the absence of any sudden expansion implies that the network is tree-like or clique-like also globally and therefore convex. A general definition of a fully convex network is a collection of cliques of arbitrary size which are stitched together in a tree, while each pair of cliques can overlap in at most one node [8]. This definition includes a complete graph, where the tree consists of a single node, and also a tree, where all the cliques are single edges.

Let  $s(t)$  denote the average fraction of nodes included in the convex subgraphs or subsets  $S$  after  $t$  steps of the procedure. According to the adopted time convention,  $s(t) \geq (t+1)/n$ . In a convex network,  $s(t) \approx (t+1)/n$  for any number of steps  $t$ , whereas  $s(t) \gg (t+1)/n$  already for very small  $t$  in a non-convex network. As an example, Fig. 3 shows the evolution of  $s(t)$  in different empirical networks and synthetic graphs. A thorough

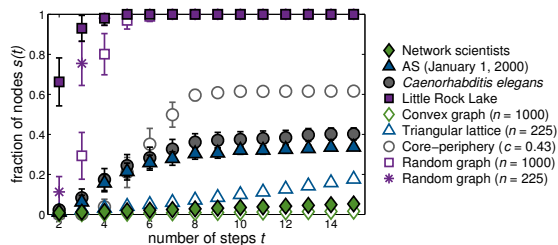


FIG. 3. Expansion of convex subgraphs in empirical networks and synthetic graphs with  $n$  nodes of which  $c$  are core nodes in the core-periphery graphs. The plot shows the fraction of nodes  $s(t)$  in the growing convex subgraphs at different steps  $t \geq 2$ , where  $s(t) \approx (t+1)/n$  quantifies the presence of a convex structure. The symbols are averages over 100 runs of convex expansion, while the errors bars show the 99% confidence intervals.

description of the empirical networks and the details of the construction of synthetic graphs, together with their basic statistics, are given in Appendix A.

In fully convex graphs that are constructed as random trees with every edge expanded to a clique ( $\diamond$  in Fig. 3), convex subgraphs grow one node at a time,  $s(t) = (t+1)/n$ . Similar behavior is observed for the network scientists coauthorship network [17]. Since the network is a projection of a bipartite graph, it is a union of cliques by construction, while its convex structure further suggests that these are connected together in a tree-like arrangement. On the other hand, there is a sudden expansion of convex subgraphs in the Erdős-Rényi random graphs [18] having no characteristic structure ( $\square$  in Fig. 3), where  $s(t) \approx 1$  after less than five steps  $t \leq 5$  in these examples. In fact, the sudden expansion in random graphs occurs at  $t = \ln n / \ln \langle k \rangle$  [8], where  $\langle k \rangle$  is the expected degree of the nodes. The only empirical network in Fig. 3 with a less convex structure than random graphs is the Little Rock Lake food web [19] ( $\blacksquare$ ), due to a division of species into trophic levels.

Network convexity can be assessed quantitatively using a measure of convexity  $X \in [0, 1]$  [8]. Let  $\Delta s(t)$  denote the average increase in the size of convex subgraphs at step  $t$  of the procedure,  $\Delta s(t) = s(t) - s(t-1)$ . Then,

$$X = 1 - \sum_{t=1}^{n-1} \max\{\Delta s(t) - 1/n, 0\}. \quad (1)$$

Convexity  $X$  compares the growth of convex subgraphs  $\Delta s(t)$  in a network with the growth in a fully convex graph, where  $\Delta s(t) = 1/n$  at each step  $t$  and  $X = 1$ . For instance, convexity of the above coauthorship network is  $X = 0.85$ , while  $X \leq 0.03$  for the food web and random graphs. The values of convexity  $X$  for all networks and graphs are reported in Table III in Appendix A.

The growth  $\Delta s(t)$  shows a notable transition in the autonomous systems graph from 2000 [20] and the *Caenorhabditis elegans* protein interactions network [21] ( $\blacktriangle$  and  $\bullet$  in Fig. 3). Convex subgraphs expand relatively

quickly in the first few steps,  $\Delta s(t) \gg 1/n$ , after which the growth settles,  $\Delta s(t) \approx 1/n$ . These are both core-periphery networks with a natural division into a densely connected core surrounded by a sparse disconnected periphery [22, 23]. In such networks, convex subgraphs quickly cover the network core, which is found to be non-convex, while the growth settles when convex subgraphs start to stretch over a convex periphery.

The subset of nodes included in the majority of convex subgraphs after fifteen steps  $t = 15$  as in Fig. 3 is called the network  $c$ -core [8]. For example, the left side of Fig. 4 shows the  $c$ -core of the autonomous systems graph from 1999. Let  $c$  be the fraction of nodes in the network  $c$ -core. Convexity  $X$  can then be approximated as

$$X \approx 1 - c + \frac{2\langle \ell \rangle + 1}{n}, \quad (2)$$

where  $1 - c$  is the fraction of nodes in the convex periphery,  $\langle \ell \rangle$  is the average distance between the nodes measured as the number of edges in the shortest paths and  $t = 2\langle \ell \rangle$  is approximately the number of steps needed for convex subgraphs to cover the network  $c$ -core.

### III. CONVEXITY UNDER RANDOMIZATION

Convexity implies a very specific network structure consisting only of cliques and a tree. Therefore, it is intuitive to expect that convexity is very sensitive to random perturbations of network structure. Indeed, adding a single edge to a tree already creates a cycle, while removing a single edge already destroys a clique. In this section, we study the robustness of convexity in networks and graphs under randomization by edge rewiring [25].

All empirical networks analyzed in the paper are either connected or reduced to the largest connected component, while all synthetic graphs are almost fully connected. Random rewiring of edges, however, can certainly disconnect the networks and graphs. This presents a problem because the measure of convexity  $X$  in Eq. (1) is only sensible for connected networks [8]. In disconnected networks, there are no (shortest) paths between

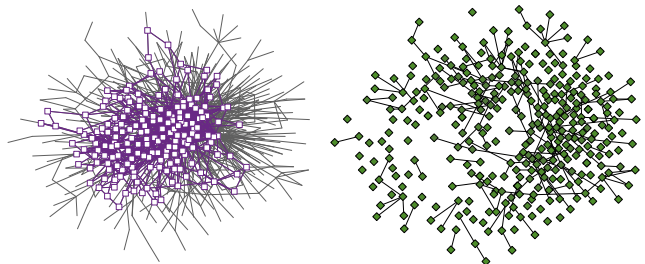


FIG. 4. Division of the autonomous systems graph from 1999 into (left) the  $c$ -core represented with  $\square$  and (right) a convex periphery represented with  $\diamond$ . The layout was computed from the full network with the Large Graph Layout [24] and is consistent between the figures.

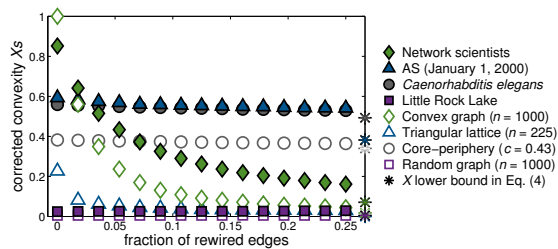


FIG. 5. Evolution of convexity in empirical networks and synthetic graphs under degree-preserving randomization by edge rewiring. The plot shows corrected convexity  $Xs$  for different fractions of rewired edges and the asterisks mark the lower bound for  $X$ . The symbols are averages over 25 independent randomizations, whereas the errors bars are smaller than the symbol sizes.

the nodes of different connected components, thus any argument regarding the inclusion of shortest paths is ill-posed. In fact, the concept of a convex subgraph, which presents the basis of convexity  $X$ , can only be defined for connected subgraphs, whereas any disconnected subgraph is necessarily non-convex [3, 4].

We here propose a corrected measure of convexity suitable also for disconnected networks that contain one large connected component. Corrected convexity written as  $Xs$  is defined as the product  $sX$ , where  $s$  is the fraction of nodes in the largest connected component and convexity  $X$  is estimated only from the largest connected component. Hence,

$$Xs = s - \sum_{t=1}^{sn-1} \max\{s\Delta s(t) - 1/n, 0\}, \quad (3)$$

while all other details are the same as in Eq. (1). In contrast to convexity  $X$ , corrected convexity  $Xs$  explicitly demands connectivity between the nodes. It is consistent with the definition of a convex subgraph, under which any disconnected subgraph is non-convex, and equivalent to convexity  $X$  for connected networks, where  $s = 1$ .

Fig. 5 demonstrates the robustness of convexity in different empirical networks and synthetic graphs presented in Appendix A. The plot shows the evolution of corrected convexity  $Xs$  under degree-preserving randomization by edge rewiring [25], where randomly selected edges are rewired by swapping one of their endpoints. We ensure simple networks and graphs by forbidding any parallel edges or self-edges during the rewiring. At the limit, this process generates a random graph with the same degree sequence as the original network or graph [26].

In the case of the Little Rock Lake food web and the Erdős-Rényi random graphs that lack convexity  $Xs \approx 0$  (■ and □ in Fig. 5), rewiring of edges obviously has no effect. On the other hand, rewiring only a very small fraction of edges already destroys convexity in the network scientists coauthorship network and convex graphs (◆ and ◇ in Fig. 5). In particular, after rewiring 5% of the edges, corrected convexity  $Xs$  drops to less than

half, although this percentage is about a hundred times smaller than what is normally needed to fully randomize a network or graph [27]. Thus, as anticipated above, convexity is a very sensitive property that is not resilient even to small perturbations of network structure.

Different behavior is observed for the autonomous systems graph from 2000 and the *Caenorhabditis elegans* protein interactions network (▲ and ● in Fig. 5), where corrected convexity  $Xs$  decreases only moderately. This can be explained by a large number of pendant nodes in these networks, which are nodes with degree one that are connected to the rest of a network by a single edge (see the right side of Fig. 4). Since pendant nodes retain convexity during the convex expansion procedure in Section II, and the rewiring process retains the degrees of the nodes, convexity  $X$  is bound by

$$X \geq \frac{n_1}{n}, \quad (4)$$

where  $n_1$  is the number of pendant nodes and  $n$  the number of all nodes. The bound in Eq. (4) is marked by the asterisks in Fig. 5 and is non-trivial for scale-free networks [28] due to a large number of nodes with small degree. For example, convexity in the protein interactions network and the corresponding core-periphery graphs (● and ○) can be almost entirely explained by the pendant nodes, while some further dependencies exist in the autonomous systems graph (▲). In Appendix B, we analyze the robustness of convexity also under full randomization, where the bound in Eq. (4) no longer holds.

#### IV. CONVEX SKELETONS OF NETWORKS

Section III reveals the sensitivity of convexity in networks, which is not resilient even to small random perturbations of network structure. Therefore, any convexity observed in empirical networks in Section II might actually be much higher if measured differently, since this fact can be obscured already by a small number of “random” edges that decrease convexity. In this section, we

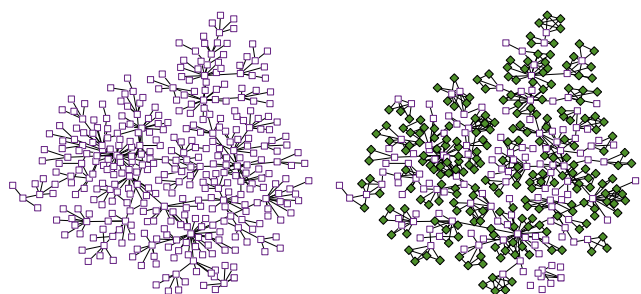


FIG. 6. (left) A spanning tree and (right) a convex skeleton of the network scientists coauthorship network extracted as in Table I. The nodes with the clustering coefficient above  $C > 0.5$  are represented with ◆ and the remaining nodes with □, while other details are the same as in Fig. 4.

investigate this hypothesis by studying the largest high-convexity part of networks obtained by removing the least number of edges, which we call a convex skeleton. Hence, a convex skeleton is defined as the largest part of a network such that every connected subset of nodes induces a convex subgraph.

By definition of a convex network given in Section II, a fully convex skeleton would be as a collection of cliques glued together in a tree. Each two cliques of a convex skeleton can overlap in at most one node and each cycle is retained in some clique. A convex skeleton is thus a generalization of a network spanning tree in which each edge can be replaced by a clique of arbitrary size (see the right side of Fig. 6). This implies that every connected network contains at least one convex skeleton, which is a spanning tree, while there are usually many others. Note that we here allow a convex skeleton to contain also nearly cliques with some of the edges missing that are connected in a tree-like manner, whereas any deviation from the ideal structure is measured by corrected convexity  $Xs$  defined in Eq. (3).

Both a spanning tree and a convex skeleton represent a very simple definition of a network backbone [10, 11] consisting only of cliques and a tree. However, in contrast to a spanning tree, a convex skeleton retains important structural properties of networks by preserving their cliques. For example, Fig. 6 shows realizations of a spanning tree and a convex skeleton of the network scientists coauthorship network presented in Appendix A. Consider the average node clustering coefficient  $\langle C \rangle$  [29] with the node clustering coefficient defined as  $C_i = \frac{2t_i}{k_i(k_i-1)}$ , where  $t_i$  is the number of triangles including node  $i$  and  $k_i$  is its degree. Since every coauthorship between three authors is a triangle in a coauthorship network, and larger coauthorships cause larger cliques with many triangles, the network is locally very dense with  $\langle C \rangle = 0.74$ . This property is retained in a convex skeleton where  $\langle C \rangle = 0.75$  on average, while corrected convexity equals  $Xs = 0.95$ . On the other hand,  $\langle C \rangle = 0$  and  $Xs = 1$  for any spanning tree. Furthermore, a convex skeleton consists of more than 90% of the edges in this network, differently from a spanning tree that includes only 41% of the edges.

In what follows, we first present different informed and direct heuristic approaches for extracting convex skeletons from empirical networks by targeted removal of edges. Next, we study the properties of the extracted convex skeletons and show that these retain the degree distribution, clustering, connectivity and also distances between the nodes, while at the same time make the shortest paths between the nodes largely unique. This is because any two nodes in a convex skeleton are either part of the same clique and thus connected by an edge or there exists a unique shortest path traversing the tree.

The asterisks in Fig. 7 show the evolution of corrected convexity  $Xs$  in two core-periphery networks under random removal of edges. These are the autonomous systems graph from 2000 and the *Caenorhabditis elegans* protein interactions network. Note that any strategy for

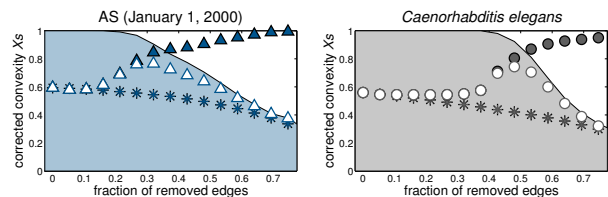


FIG. 7. Evolution of convexity in core-periphery networks under targeted removal of edges based on the node c-centrality. The plots show the fraction of nodes in the largest connected component  $s$  (filled areas), convexity  $X$  (▲ and ●) and corrected convexity  $Xs$  (△ and ○), while the asterisks mark the value of  $Xs$  under random removal of edges. The symbols are averages over 25 independent removals, whereas the errors bars are not visible.

removing the edges of a network increases non-corrected convexity  $X$  as the network becomes increasingly sparse and thus start to resemble a tree. Yet, the network also starts falling apart decreasing the fraction of nodes in the largest connected component  $s$ . Under random removal of edges, the networks in Fig. 7 break apart before convexity  $X$  increases, resulting in a monotonically decreasing corrected convexity  $Xs$ . The same decreasing trend is observed also for targeted removal of edges based on the standard measures of centrality [2, 30].

As discussed in Section II, the main source of non-convexity in these networks is the existence of a non-convex c-core [8]. The removal of edges should therefore target only the edges between the nodes in the c-core, since the remaining periphery is already convex. However, removing the edges between the nodes in the border of the c-core would redirect the shortest paths between the peripheral nodes through the center of the non-convex c-core, decreasing convexity also in the periphery. Thus, in order to extract a convex skeleton from core-periphery networks, one should target only the edges between the nodes in the center of the network c-core.

Let  $p_i$  be the probability that node  $i$  is included in the c-core as defined alongside Eq. (2). The distribution of  $p$  is extremely bimodal with peaks at  $p \approx 0$  and 1 [8]. We define the central nodes of the c-core as those connecting many other nodes in the c-core estimated by  $\sum_{j \in \Gamma_i} p_j$ , where  $\Gamma_i$  is the neighborhood of node  $i$ , but not the nodes in the periphery estimated by  $\sum_{j \in \Gamma_i} 1 - p_j$ . Hence, the c-core centrality or c-centrality of node  $i$  is defined as

$$\begin{aligned} c_i &= \sum_{j \in \Gamma_i} p_j - \sum_{j \in \Gamma_i} 1 - p_j \\ &= -k_i + 2 \sum_{j \in \Gamma_i} p_j, \end{aligned} \quad (5)$$

where  $k_i$  is the degree of node  $i$ . Notice that nodes with high c-centrality are usually not high-degree nodes. Still, in a non-convex network or a random graph where  $p_i = 1$  for all the nodes, Eq. (5) reduces to  $c_i = k_i$ .

The symbols in Fig. 7 show the evolution of convexity under targeted removal of edges based on the node

TABLE I. Statistics of empirical networks (N), spanning trees (ST) and convex skeletons (CS) extracted by targeted removal of edges based on the node clustering coefficient (we remove 1% of edges at a time or 0.1% for protein interactions networks). The statistics show the average node degree  $\langle k \rangle$  and clustering coefficient  $\langle C \rangle$ , the average number of shortest paths or geodesics  $\langle \sigma \rangle$  and corrected convexity  $Xs$ . The values are averages over 25 independent realizations and 100 runs of convex expansion.

Class	Network	degree $\langle k \rangle$			clustering $\langle C \rangle$			geodesics $\langle \sigma \rangle$			convexity $Xs$		
		N	CS	ST	N	CS	ST	N	CS	ST	N	CS	ST
Collaboration	Jazz musicians	27.70	11.06	1.99	0.62	0.81	0.00	9.71	1.97	1.00	0.12	0.84	1.00
	Network scientists	4.82	4.40	1.99	0.74	0.75	0.00	2.66	1.47	1.00	0.85	0.95	1.00
	Computer scientists	4.75	3.25	1.99	0.48	0.54	0.00	4.08	1.42	1.00	0.64	0.95	1.00
Protein interactions	<i>Plasmodium falciparum</i>	4.15	2.26	2.00	0.02	0.07	0.00	3.71	1.77	1.00	0.43	0.95	1.00
	<i>Saccharomyces cerevisiae</i>	2.67	2.16	2.00	0.07	0.10	0.00	2.58	1.19	1.00	0.68	0.88	1.00
	<i>Caenorhabditis elegans</i>	4.14	2.43	2.00	0.06	0.12	0.00	6.79	3.03	1.00	0.56	0.85	1.00
Autonomous systems	AS (January 1, 1998)	3.50	2.70	2.00	0.18	0.21	0.00	3.87	2.32	1.00	0.66	0.91	1.00
	AS (January 1, 1999)	4.58	3.18	2.00	0.18	0.27	0.00	3.54	2.05	1.00	0.49	0.95	1.00
	AS (January 1, 2000)	3.94	2.96	2.00	0.20	0.25	0.00	4.81	3.07	1.00	0.59	0.90	1.00
Food webs	Little Rock Lake	26.60	4.31	1.99	0.32	0.69	0.00	22.13	4.32	1.00	0.02	0.82	1.00
	Florida Bay (wet)	32.42	3.98	1.98	0.33	0.79	0.00	9.17	1.37	1.00	0.03	0.92	1.00
	Florida Bay (dry)	32.91	5.02	1.98	0.33	0.82	0.00	9.37	1.65	1.00	0.03	0.93	1.00

c-centrality. The edges are removed in decreasing order of  $c_i + c_j$ , where  $i$  and  $j$  are the endpoints of an edge. We remove a single edge at a time and recalculate the c-centrality of nodes according to Eq. (5) after each step. Observe that convexity  $X$  ( $\blacktriangle$  and  $\bullet$ ) increases before the networks start falling apart, resulting in a peak in corrected convexity  $Xs$  ( $\triangle$  and  $\circ$ ). This peak provides a natural way to extract a convex skeleton. For instance, after removing about a third of the edges in the autonomous system graph in the left side of Fig. 7, corrected convexity increases from  $Xs = 0.59$  to  $0.77$ , when  $X = 0.85$  and  $s = 0.91$ . Moreover, as we show below, the network actually contains a convex skeleton with  $Xs = 0.90$  that retains more than 75% of its edges.

In the remaining, we consider also approaches for targeted removal of edges that try to extract a convex skeleton more directly. However, directly optimizing corrected convexity  $Xs$  in a network comes with a number of problems. First, the computation of convexity  $X$  is inevitably quadratic operation with  $\mathcal{O}(nm)$  [16], where  $n$  is the number of nodes and  $m$  the number of edges. The complexity of removal of edges would therefore be at least cubic  $\mathcal{O}(nm^2)$  and applicable only to very small networks. Second, random variations in the estimation of convexity  $X$  are often larger than the changes resulting from the removal of a single edge. Third, optimizing corrected convexity  $Xs$  does not guarantee a convex skeleton that retains important properties of a network, since any network spanning tree is also convex with  $Xs = 1$ .

A better approach is to directly optimize the properties of a network that should ideally be retained in a convex skeleton such as the average node clustering coefficient  $\langle C \rangle$ . Consider an edge with endpoints  $i$  and  $j$ , and let  $\Delta C_i$  denote the change in the clustering coefficient of node  $i$  after the removal of the mentioned edge. In order to extract a convex skeleton, we remove the edges from a network in a decreasing order of  $\Delta C_i + \Delta C_j$ , while ensuring that the network stays connected. Nevertheless,

since we remove between 0.1% and 1% of edges at a time from the networks below, while we ensure the connectivity only for the removal of individual edges, these can still become disconnected.

Table I shows selected statistics of convex skeletons extracted from different social collaboration and protein interactions networks, autonomous systems graphs and food webs. The details of the networks are given in Appendix A. For an easier comparison, Table I also shows the statistics of network spanning trees [31], although all these directly follow from the definition of a tree.

The average corrected convexity of the extracted convex skeletons over all networks is  $\langle Xs \rangle = 0.90$ , which can be considered as high-convexity parts of these networks. Besides, this implies that the skeletons are almost fully connected as the fraction of nodes in the largest connected component is  $s \geq Xs$ . We stress that the adopted process of edge removal does not optimize convexity directly, thus a convex skeleton is an emerging property of these networks. The fact might be particularly interesting as the removal process is almost the exact opposite of small-world network models [29, 32], where the edges of a ring lattice are randomly rewired or new random edges are added. In contrast to a ring lattice, a convex skeleton suggest a tree-like structure. This has important implications for the distances between the nodes. While the average distance  $\langle \ell \rangle$  in a ring on  $n$  nodes is in  $\mathcal{O}(n)$  [33],  $\langle \ell \rangle = \mathcal{O}(\sqrt{n})$  in a random tree [34] and  $\langle \ell \rangle = \mathcal{O}(\ln n)$  in a balanced tree, consistently with small-world networks.

The average node clustering coefficient  $\langle C \rangle$  in the extracted convex skeletons is expectedly higher than in the full networks, although obviously no new triangles are created. Furthermore, the shortest paths or geodesics between the nodes become largely unique, with the average number of geodesics  $\langle \sigma \rangle < 2$  in most cases. At the same time, the average distance between the nodes  $\langle \ell \rangle$  remains comparable, due to an underlying tree-like structure in a convex skeleton. A convex skeleton might thus represent

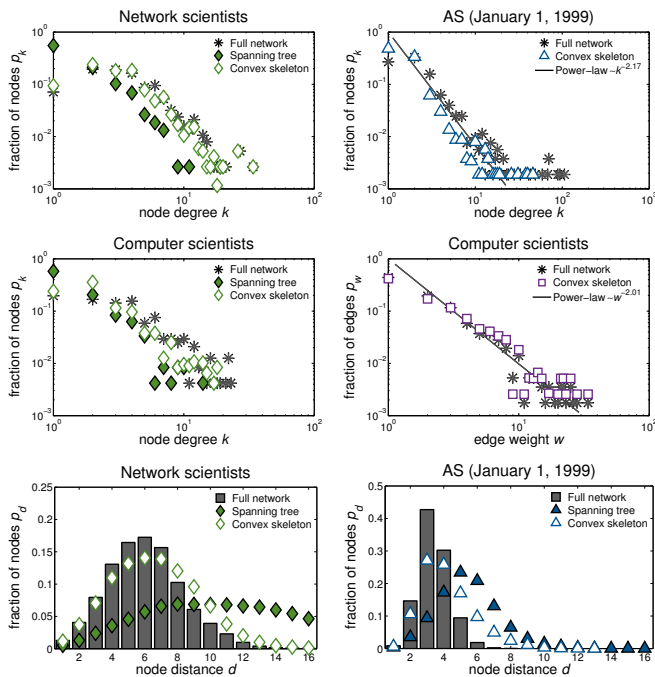


FIG. 8. Node and edge distributions of empirical networks, spanning trees and convex skeletons. The plots show the fractions of (top) nodes  $p_k$  with degree  $k$ , (middle right) edges  $p_w$  with weight  $w$  and (bottom) pairs of nodes  $p_d$  at distance  $d$  over 25 independent realizations, whereas the errors bars are not shown. The power-law distributions are maximum likelihood estimations [36] for the full networks.

a suitable model for studying different dynamic processes on networks such as navigation and synchronization [35].

Table I also shows the average node degree  $\langle k \rangle$ , which allows to estimate the fraction of edges retained in the convex skeletons. On the one hand, this fraction is larger than 90% in the network scientists coauthorship network, while 70-75% in the computer scientists coauthorship network and the autonomous systems graphs. On the other hand, one has to remove 85-90% of the edges in the food webs to reveal a convex skeleton. This can be anticipated from a large redundancy of geodesics in food webs needed for an ecosystem to survive. For instance, the average number of geodesics in the Little Rock Lake food web is  $\langle \sigma \rangle > 20$ . A convex skeleton therefore does not represent a very sensible abstraction of food webs.

Finally, Fig. 8 shows different node distributions of the extracted convex skeletons and spanning trees. Notice that the convex skeletons preserve the distributions of node degrees  $p_k$  in the full networks (top row), regardless of whether these are heavy-tailed or contain a clear peak. Moreover, the convex skeletons largely retain the distributions of distances between the nodes  $p_d$  (bottom row). In contrast to the convex skeletons, the lack of cliques in the spanning trees noticeably increases the distances, while low-degree nodes become overrepresented and high-degree nodes become underrepresented. This is best observed in the case of the network scientists coau-

thorship network ( $\blacklozenge$  and  $\blacktriangle$ ). In Appendix C, we compare the node distributions of convex skeletons and spanning trees also in larger networks, where the conclusions are exactly the same.

## V. COAUTHORSHIP NETWORKS

A convex skeleton represents a simple definition of a network backbone [10, 11] with a very plain structure consisting only of cliques and a tree. Nevertheless, as shown in Section IV, convex skeletons retain most important structural properties of empirical networks. They seem to provide a particularly reasonable abstraction of different social collaboration networks such as coauthorship networks [37], since these are unions of cliques by construction. In this section, we more thoroughly analyse convex skeletons extracted from the Slovenian computer scientists coauthorship network [38] and compare them with other network backbone techniques.

The coauthorship network was collected from the Slovenian national research database SICRIS [39] in 2010. We here analyse only the largest connected component of the network representing coauthorships between 239 computer scientists, which is about a third of all registered computer scientists at that time. Since each scientist in the SICRIS database is represented by a unique identifier, no name disambiguation was needed. Note, however, that the main purpose of the SICRIS database is not the archiving of scientific contributions, but the evaluation of the performance of scientists, which may reflect certain policy needs [40]. Further details about the network are given in Appendix A.

Table II shows the standard statistics of the network, and the extracted spanning trees and convex skeletons. The results are consistent with Section IV. While both techniques produce a high-convexity backbone with  $X_s \geq 0.95$  and retain the connectivity of the network  $s \approx 1$ , only the convex skeletons preserve the average node clustering coefficient  $\langle C \rangle = 0.54$  and to a large extent also the distances between the nodes  $\langle \ell \rangle = 7.06$ . Moreover,

TABLE II. Summary statistics of backbones of the computer scientists coauthorship network. These are the average node degree  $\langle k \rangle$  and clustering coefficient  $\langle C \rangle$ , the average distance between the nodes  $\langle \ell \rangle$ , modularity  $Q$  of the classification of authors into fields, corrected convexity  $X_s$  and the fraction of nodes in the largest connected component  $s$ . The values are averages over 25 independent realizations.

Backbone	$\langle k \rangle$	$\langle \ell \rangle$	$\langle C \rangle$	$Q$	$X_s$	$s$
Full network	4.75	4.58	0.48	0.33	0.64	1.00
Spanning tree	1.99	10.11	0.00	0.32	1.00	1.00
Edge betweenness	3.25	4.63	0.18	0.29	0.69	1.00
	3.25	2.44	0.40	0.36	0.35	0.64
Saliency skeleton	3.10	4.78	0.05	0.30	0.57	1.00
Convex skeleton	3.25	7.06	0.54	0.38	0.95	0.99

the convex skeletons retain the node degree distribution  $p_k$  of the network ( $\diamond$  in the middle row of Fig. 8).

Table II further shows the statistics of two alternative network backbone techniques based on edge betweenness [2] and salience [12]. The betweenness of an edge is defined as the fraction of shortest paths between all nodes that traverse the edge, while the salience of an edge is defined as the fraction of shortest path trees originating from each node that include the edge. Consistently with a convex skeleton, both techniques define a network backbone based on the shortest paths between the nodes. In contrast, the techniques are based on the number of shortest paths and not their inclusion as in a convex skeleton. The salience of the edges was computed using the implementation provided in Refs. [11, 31], while we use our own implementation for edge betweenness [41].

The distribution of edge salience is bimodal [12], which gives a natural definition of a high-salience skeleton. On the other hand, the distribution of edge betweenness is heavy-tailed in the concerned coauthorship network. In Table II, we therefore report the statistics of a high-betweenness backbone with the same number of edges as in the convex skeletons (first row) and, for comparison, also a low-betweenness backbone (second row).

Consider first the high-salience skeleton and the high-betweenness backbone. Both techniques retain the connectivity of the network  $s = 1$  and short distances between the nodes  $\langle \ell \rangle \approx 4.71$ , while corrected convexity remains comparable with the full network  $Xs \approx 0.63$ . Yet, the average node clustering coefficient drops to  $\langle C \rangle \approx 0.12$ , since the backbones do not preserve the cliques in the network. This is because the edges embedded in the cliques are not bridges between different parts of the network that would support many shortest paths. These edges are in fact included in the low-betweenness backbone with  $\langle C \rangle = 0.40$ , but which disconnects the network as  $s = 0.64$ . In summary, out of all the network backbones considered, only the convex skeletons retain the important structural properties of this coauthorship network. Besides, the convex skeletons preserve also the power-law distribution of the weights of the coauthorship ties  $p_w$ , where the weight  $w$  of a tie is defined as the number of coauthored papers ( $\square$  in the middle row of Fig. 8).

A realization of a convex skeleton is shown in the top of Fig. 9, where node symbols represent the primary research fields of the authors. One can visually observe the underlying tree structure of the convex skeleton and also the cliques formed by the authors from the same research field. Furthermore, the bottom of Fig. 9 compares the properties of different backbones that one might find desirable for an abstraction of a coauthorship network. These are the average weight of the coauthorship ties  $\langle w \rangle$  in the backbones, modularity  $Q$  [42] of the field classification that compares the number of ties between the authors from the same field with the expected number of such ties and the average fraction of inter-field ties in the remaining network after removal of the backbones. In all cases, these properties are enhanced in a convex skeleton

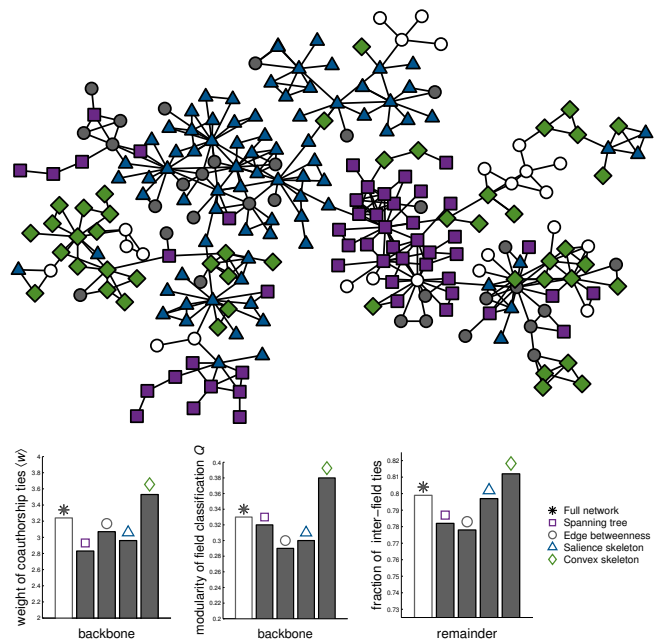


FIG. 9. (top) A convex skeleton of the computer scientists coauthorship network. The node symbols represent primary research fields of the authors: computer theory ( $\diamond$ ), information systems ( $\blacksquare$ ), intelligent systems ( $\blacktriangle$ ), programming technologies ( $\circ$ ) and other ( $\bullet$ ). (bottom) Comparison of network backbones showing the average weight  $\langle w \rangle$  of the coauthorship ties in a backbone and modularity  $Q$  of the field classification, and the fraction of inter-field ties in the remaining network. The histograms are averages over 25 independent realizations.

( $\diamond$ ) compared to the full network, differently from a spanning tree ( $\square$ ), high-betweenness backbone ( $\circ$ ) and also high-salience skeleton ( $\triangle$ ). Preliminary results on coauthorship networks from the SICRIS database for other scientific disciplines are consistent with above, while detailed results will be reported elsewhere.

## VI. CONCLUSIONS

In this paper, we have introduced a network backbone technique called a convex skeleton. A convex skeleton is a collection of cliques glued together in a tree and as such represents a generalization of a network spanning tree. We have extracted convex skeletons from empirical networks of various type and origin, and shown that these retain the degree distribution, clustering, connectivity and distances between the nodes, while at the same time make the shortest paths in a network largely unique. This is in contrast with a spanning tree or high-betweenness backbone [2] and high-salience skeleton [12]. A convex skeleton seems to provide a particularly reasonable abstraction of social collaboration networks such as coauthorship networks, with applications in network modeling, visualization, navigation and possibly also elsewhere.

A convex skeleton can be seen as a particular case of a



network sampling or sparsification technique [10, 43, 44], but with diverging goals. At the one hand, sparsification techniques try to simplify the network by removing redundant or spurious edges [11, 45], while at the same time keep its structural properties intact. Therefore, one would ideally like to remove as many edges as possible. On the other hand, a convex skeleton tries to retain as many edges as possible, since the simplicity is implied by its plain structure. It should, however, be mentioned that the approach for extracting convex skeletons adopted here is just one example and likely much more effective techniques exist. Besides, we were interested only in revealing a convex skeleton by removal of edges, while adding edges in order to obtain a fully convex network might be more suitable for some applications.

One probably noticed that the concept of convexity is not actually needed in order to extract a convex skeleton. Then, why bother the reader with the definition of convexity in the first place? The reason is that convexity allows one to measure how far the resulting structure is from a tree of cliques, which would be hard to estimate otherwise. Finally, it is reasonable to expect that trees of cliques have been studied in networks context before, either in graph theory or network science literature. Nevertheless, we are convinced that our definition based on convexity is novel and unique. We are eager to see whether the ideas developed in this paper will prove useful for better understanding of network structure and stimulate new applications of network theory.

## ACKNOWLEDGMENTS

The authors thank Tilen Marc, Vincent Traag, Ludo Waltman, Zoran Levanić and Dalibor Fiala for helpful comments and discussions, Tadej Ciglarič for implementation of network convexity and Michele Coscia for implementations of network backbones. The authors also acknowledge a stimulating environment at the Centre for Science and Technology Studies of Leiden University where part of this paper was written. This work has been supported in part by the Slovenian Research Agency under the program P2-0359 and by the European Union COST Action number CA15109.

## Appendix A: Networks and graphs

In this appendix, we provide the details of the empirical networks and synthetic graphs analyzed in the paper. The former include three examples of social collaboration networks, protein interactions networks, autonomous systems graphs and food webs. In all cases, the networks have been represented with simple undirected graphs and reduced to their largest connected components.

The collaboration networks represent jazz musicians that played in a band between 1912 and 1940 obtained from the Red Hot Jazz archive [46], coauthorships be-

TABLE III. Statistics of empirical networks and synthetic graphs analyzed in the paper. These show the number of nodes  $n$ , the average node degree  $\langle k \rangle$  and clustering coefficient  $\langle C \rangle$ , the average distance between the nodes  $\langle \ell \rangle$  and convexity  $X$ . The values are averages over 25 independent realizations.

Network	$n$	$\langle k \rangle$	$\langle \ell \rangle$	$\langle C \rangle$	$X$
Jazz musicians	198	27.70	2.24	0.62	0.12
Network scientists	379	4.82	6.04	0.74	0.85
Computer scientists	239	4.75	4.58	0.48	0.64
<i>Plasmodium falciparum</i>	1158	4.15	4.24	0.02	0.43
<i>Saccharomyces cerevisiae</i>	1458	2.67	6.81	0.07	0.68
<i>Caenorhabditis elegans</i>	3747	4.14	4.32	0.06	0.56
AS (January 1, 1998)	3213	3.50	3.77	0.18	0.66
AS (January 1, 1999)	531	4.58	3.39	0.18	0.49
AS (January 1, 2000)	3570	3.94	3.80	0.20	0.59
Little Rock Lake	183	26.60	2.15	0.32	0.02
Florida Bay (wet)	128	32.42	1.78	0.33	0.03
Florida Bay (dry)	128	32.91	1.77	0.33	0.03
Random graph	2500	10.00	3.65	0.00	0.00
	1000	10.00	3.26	0.01	0.01
	225	10.00	2.60	0.05	0.03
Triangular lattice	225	5.48	8.07	0.43	0.23
Rectangular lattice	225	3.73	10.00	0.00	0.13
Core-periphery graph	3747	4.48	5.22	0.00	0.39
	2500	5.97	10.23	0.90	1.00
Convex graph	1000	5.97	8.40	0.90	1.00
	225	6.01	5.44	0.90	1.00

tween network scientists parsed from the bibliographies of two review papers in 2006 [17] and coauthorships between Slovenian computer scientists up to 2010 extracted from the SICRIS database [38]. The protein interactions networks represent recorded protein-protein interactions of the parasite *Plasmodium falciparum*, yeast *Saccharomyces cerevisiae* and nematode *Caenorhabditis elegans*. The networks were collected from the BioGRID repository in 2016 [21]. The autonomous systems graphs are the Internet maps at the level of autonomous systems [20] on the first day of 1998, 1999 and 2000 reconstructed from the University of Oregon Route Views project [47]. The food webs represent observed predator-prey relationships between the species of Little Rock Lake [19] and Florida Bay in wet and dry season [48].

Table III shows the standard statistics of the networks including the number of nodes  $n$ , the average node degree  $\langle k \rangle = 2m/n$ , where  $m$  is the number of edges, the average node clustering coefficient  $\langle C \rangle$  [29] with the clustering coefficient of node  $i$  defined as  $C_i = \frac{2t_i}{k_i(k_i-1)}$ , where  $t_i$  is the number of triangles including node  $i$  and  $k_i$  is its degree, the average distance between the nodes  $\langle \ell \rangle = \frac{2}{n(n-1)} \sum_{i < j} d_{ij}$ , where  $d_{ij}$  is the number of edges in the shortest paths between nodes  $i$  and  $j$ , and convexity  $X$  defined in Eq. (1) in Section II.

Table III also shows the statistics of the analyzed synthetic graphs. First, these include triangular and rectangular lattices with the side of 15 nodes and the Erdős-Rényi random graphs [18] with  $n = 225, 1000$  or 2500

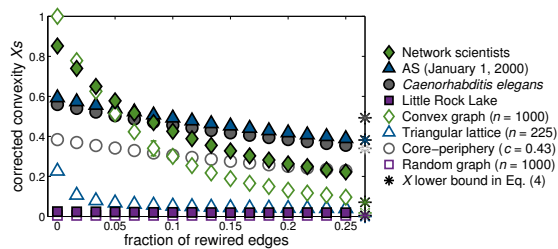


FIG. 10. Evolution of convexity in empirical networks and synthetic graphs under full randomization by edge rewiring, while other details are the same as in Fig. 5.

nodes and the average degree  $\langle k \rangle = 10$  to ensure the graphs are almost fully connected. We generate simple graphs by forbidding any parallel edges or self-edges.

Next, core-periphery graphs are based on the core-periphery structure of the *Caenorhabditis elegans* protein interactions network. Let  $n = 3747$  be the number of nodes in the network and  $cn$  the number of nodes in its  $c$ -core as defined alongside Eq. (2) with  $c = 0.43$ . We generate random graphs with two sets of  $cn$  and  $(1 - c)n$  nodes, and the probability of an edge between the nodes equal to the density within or between the network  $c$ -core and periphery. Because this leaves a non-trivial fraction of nodes isolated, any isolated node is reattached to the remaining nodes by a random edge. The average degree in the graphs is thus larger than in the corresponding network,  $\langle k \rangle = 4.48 > 4.14$ . Nevertheless, the reattachment process actually creates pendant nodes that can only increase convexity in the generated graphs as in Eq. (4), which is still lower than in the network,  $X = 0.39 < 0.56$ .

Finally, convex graphs with  $n \approx 225, 1000$  or  $2500$  nodes are constructed as follows. Since these are collections of cliques connected in a tree, we first generate a random tree on  $tn$  nodes, where  $t \in [2/n, 1]$  is a parameter different than in the main text. Starting with a single node, we add  $tn - 1$  nodes one at a time, while each node is connected to a tree by a random edge. Then, every edge of the constructed tree is expanded to a clique on  $k$  nodes by adding  $k - 2$  new nodes, where  $k$  is sampled independently for each edge from the interval  $[2, \frac{2(n-1)}{tn-1}]$ . We set  $t = 0.25$  to match the characteristic size of cliques in empirical networks [49]. Note, however, that this construction process cannot generate all possible convex graphs. The graphs also lack scale-free degree distribution [28], although this could be imposed by an appropriate preferential attachment mechanism [50].

## Appendix B: Convexity under full randomization

In Section III, we study the robustness of convexity in different empirical networks and synthetic graphs under degree-preserving randomization by edge rewiring [25]. In this appendix, we consider also full randomization, where endpoints of randomly selected edges are rewired to random other nodes. We ensure simple networks and graphs by forbidding any parallel edges or self-edges during the rewiring. At the limit, this process generates the Erdős-Rényi random graphs [18] that are non-convex.

Fig. 10 shows the evolution of corrected convexity  $X_s$  in Eq. (3) under full randomization. The results are conceptually the same as for degree-preserving randomization in Fig. 5, with the only difference that convexity  $X$  is now no longer bound by Eq. (4).

## Appendix C: Node distributions of convex skeletons

In Section IV, we analyse different node distributions of convex skeletons and spanning trees extracted from smaller empirical networks. In this appendix, we consider also two larger networks, namely the autonomous systems graph from 2000 and the *Caenorhabditis elegans* protein interactions network. The top row of Fig. 11 shows the node degree distributions  $p_k$  of full networks (asterisks) and the extracted convex skeletons ( $\Delta$  and  $\circ$ ). These are largely consistent and seem to follow a power-law  $k^{-\gamma}$  with  $\gamma \approx 2.3$  [36]. Furthermore, the convex skeletons preserve the distributions of distances between the nodes  $p_d$  shown in the bottom row of Fig. 11. On the other hand, the distances between the nodes increase significantly in the spanning trees ( $\blacktriangle$  and  $\bullet$ ).

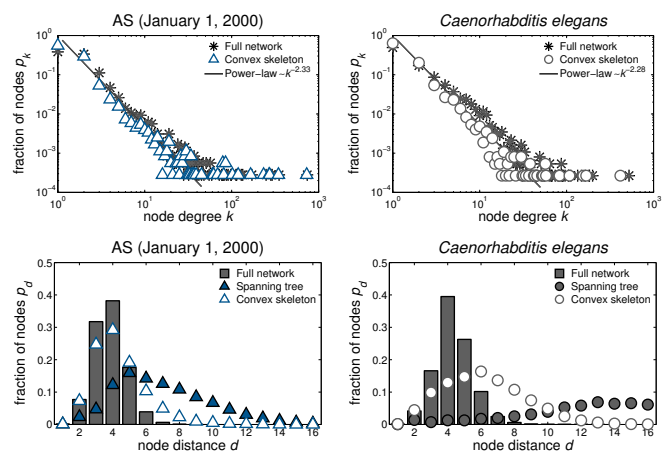


FIG. 11. Node distributions of core-periphery networks and particular realizations of spanning trees and convex skeletons, while other details are the same as in Fig. 8.

- 
- [1] S. Milgram, *Psychol. Today* **1**, 60 (1967).
- [2] L. Freeman, *Sociometry* **40**, 35 (1977).
- [3] F. Harary and J. Nieminen, *J. Differential Geom.* **16**, 185 (1981).
- [4] M. Farber and R. Jamison, *SIAM J. Algebra. Discr.* **7**, 433 (1986).
- [5] M. L. J. Van de Vel, *Theory of Convex Structures* (North-Holland, Amsterdam, 1993).
- [6] I. M. Pelayo, *Geodesic Convexity in Graphs* (Springer, New York, 2013).
- [7] M. G. Everett and S. B. Seidman, *Discrete Math.* **57**, 217 (1985).
- [8] T. Marc and L. Šubelj, *Netw. Sci.* , 1 (2017).
- [9] L. Šubelj and T. Marc, in *Proceedings of the International Conference on Complex Networks* (Dubrovnik, Croatia, 2017) pp. 14–16.
- [10] M. Hamann, G. Lindner, H. Meyerhenke, C. L. Staudt, and D. Wagner, *Soc. Netw. Anal. Min.* **6**, 22 (2016).
- [11] M. Coscia and F. Neffke, in *Proceedings of the IEEE International Conference on Data Engineering* (San Diego, CA, USA, 2017) pp. 425–436.
- [12] D. Grady, C. Thiemann, and D. Brockmann, *Nat. Commun.* **3**, 864 (2012).
- [13] V. Batagelj, in *Proceedings of the International Conference on Interfaces between Mathematics, Chemistry and Computer Science* (Dubrovnik, Yugoslavia, 1988) pp. 25–39.
- [14] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, *Science* **298**, 824 (2002).
- [15] N. Pržulj, D. G. Corneil, and I. Jurisica, *Bioinformatics* **20**, 3508 (2004).
- [16] T. Ciglaric, “Network Convexity,” <https://github.com/t4c1/Graph-Convexity> (2017).
- [17] M. E. J. Newman, *Phys. Rev. E* **74**, 036104 (2006).
- [18] P. Erdős and A. Rényi, *Publ. Math. Debrecen* **6**, 290 (1959).
- [19] N. D. Martinez, *Ecol. Monogr.* **61**, 367 (1991).
- [20] J. Leskovec, J. Kleinberg, and C. Faloutsos, *ACM Trans. Knowl. Discov. Data* **1**, 1 (2007).
- [21] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, *Nucl. Acids Res.* **34**, 535 (2006).
- [22] S. P. Borgatti and M. G. Everett, *Soc. Networks* **21**, 375 (2000).
- [23] P. Holme, *Phys. Rev. E* **72**, 046111 (2005).
- [24] A. T. Adai, S. V. Date, S. Wieland, and E. M. Marcotte, *J. Mol. Biol.* **340**, 179 (2004).
- [25] S. Maslov and K. Sneppen, *Science* **296**, 910 (2002).
- [26] M. E. J. Newman, S. H. Strogatz, and D. J. Watts, *Phys. Rev. E* **64**, 026118 (2001).
- [27] J. Ray, A. Pinar, and C. Seshadhri, *J. Complex Netw.* **3**, 204 (2015).
- [28] A.-L. Barabási and R. Albert, *Science* **286**, 509 (1999).
- [29] D. J. Watts and S. H. Strogatz, *Nature* **393**, 440 (1998).
- [30] L. C. Freeman, *Soc. Networks* **1**, 215 (1979).
- [31] M. Coscia, “Network Backboning,” [http://www.michelecoscia.com/?page\\_id=287](http://www.michelecoscia.com/?page_id=287) (2017).
- [32] M. E. J. Newman and D. J. Watts, *Phys. Rev. E* **60**, 7332 (1999).
- [33] M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Oxford, 2010).
- [34] A. Meir and J. W. Moon, *J. Comb. Theory* **8**, 99 (1970).
- [35] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical Processes on Complex Networks* (Cambridge University Press, Cambridge, 2008).
- [36] A. Clauset, C. R. Shalizi, and M. E. J. Newman, *SIAM Rev.* **51**, 661 (2009).
- [37] M. E. J. Newman, *P. Natl. Acad. Sci. USA* **98**, 404 (2001).
- [38] N. Blagus, L. Šubelj, and M. Bajec, *Physica A* **391**, 2794 (2012).
- [39] “SICRIS Database,” <http://www.sicris.si> (2010).
- [40] R. Rodela, *Scientometrics* **109**, 2151 (2016).
- [41] U. Brandes, *J. Math. Sociol.* **25**, 163 (2001).
- [42] M. Girvan and M. E. J. Newman, *P. Natl. Acad. Sci. USA* **99**, 7821 (2002).
- [43] J. Leskovec and C. Faloutsos, in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Philadelphia, PA, USA, 2006) pp. 631–636.
- [44] N. Blagus, L. Šubelj, and M. Bajec, *Physica A* **477**, 136 (2017).
- [45] R. Guimera and M. Sales-Pardo, *P. Natl. Acad. Sci. USA* **106**, 22073 (2009).
- [46] P. Gleiser and L. Danon, *Advs. Complex Syst.* **6**, 565 (2003).
- [47] J. Leskovec and A. Krevl, “SNAP Datasets,” <http://snap.stanford.edu/data> (2014).
- [48] V. Batagelj and A. Mrvar, “Pajek Datasets,” <http://vlado.fmf.uni-lj.si/pub/networks/data> (2006).
- [49] N. Slater, R. Itzhack, and Y. Louzoun, *Netw. Sci.* **2**, 387 (2014).
- [50] L. Hébert-Dufresne, A. Allard, V. Marceau, P.-A. Noël, and L. J. Dubé, *Phys. Rev. Lett.* **107**, 158702 (2011).