

Research

Article submitted to journal

Subject Areas:

mathematical physics

Keywords:

complex networks, network convexity,
network backbones, convex skeletons

Author for correspondence:

Lovro Šubelj

e-mail: lovro.subelj@fri.uni-lj.si

Convex skeletons of complex networks

Lovro Šubelj

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

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 generalisation of a network 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, distances, node position and also community structure, 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.

1. 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 analysing 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] define network convexity through the expansion of randomly grown subsets of nodes to convex subgraphs. In contrast to the hull number, which is an NP-hard problem [9], network convexity has a polynomial computational complexity and has already been successfully used in characterising 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 backboning 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 2, we provide a formal definition of network convexity and review the main results in reference [8] relevant for this work. Next, in section 3, we study the robustness of convexity under random perturbations of network structure. Section 4 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 4, while in section 5 we more thoroughly analyse node position and community structure of convex skeletons, and compare different backbones extracted from the Slovenian computer scientists coauthorship network. Section 6 concludes the paper and gives some final remarks on convex skeletons.

2. 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 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 figure 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 figure 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 characterise networks beyond connected and induced subgraphs [13], also known as motifs [14] and graphlets [15] in networks jargon.

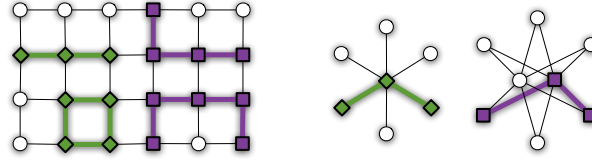


Figure 1. (left) Rectangular lattice with highlighted convex (◆) and non-convex (■) subgraphs. (right) Two star graphs in which every connected triplet of nodes induces either a convex or non-convex subgraph, respectively.

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 figure 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]. 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 initialising 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 Γ_i is the set of its neighbours. 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. We highlight that neither the selection of the initial seed node nor the subsequent choices of including new nodes affect the evolution of the procedure [8]. The algorithmic complexity of a single run in a network with n nodes and m edges is $\mathcal{O}(nm)$ [16].

Figure 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 initialisation step. Initially, the subset S contains a single node i , $S = \{i\}$, which is a convex subset. On the first step, a random neighbour $j \in \Gamma_i$ is added, $S = \{i, j\}$, which is still convex. On the next step, another neighbour $k \in \Gamma_i \cup \Gamma_j \setminus \{i, j\}$ 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)

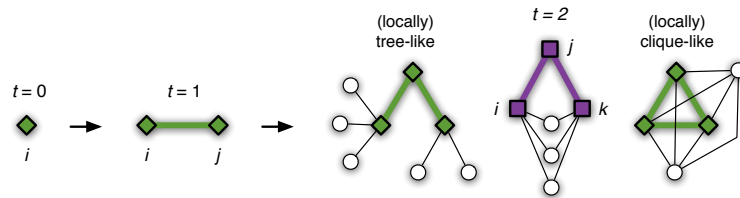


Figure 2. Growth of convex subgraphs (◆) in the first two steps $t \leq 2$ of the convex expansion procedure. Notice that non-convex subgraphs (■) indicate the absence of a (locally) tree-like or clique-like structure.

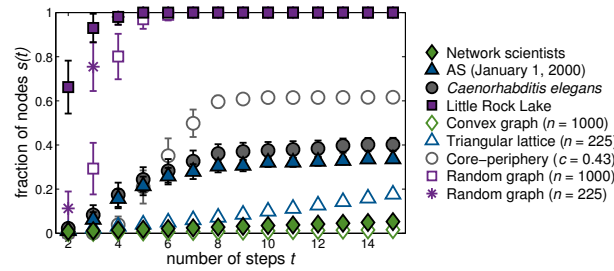


Figure 3. Expansion of convex subgraphs in empirical networks and synthetic graphs with n nodes of which c are core nodes in the core-periphery graphs. The plot shows the fraction of nodes $s(t)$ in the growing convex subgraphs at different steps $t \geq 2$, where $s(t) \approx (t+1)/n$ quantifies the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

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 figure 2 (■) 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 .

The first few steps of the convex expansion procedure thus characterise 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, figure 3 shows the evolution of $s(t)$ in different empirical networks and synthetic graphs. The former include social collaboration networks, protein interactions networks, autonomous systems graphs and food webs. The collaboration networks represent jazz musicians [17], coauthorships between network scientists [18] and Slovenian computer scientists [19]. The protein interactions networks represent interactions of the *Plasmodium falciparum*, *Saccharomyces cerevisiae* and *Caenorhabditis elegans* [20]. The autonomous systems graphs are the Internet maps from 1998, 1999 and 2000 [21]. The food webs represent the species of Little Rock Lake [22] and Florida Bay [23]. A thorough 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 (◇ in figure 3), convex subgraphs grow one node at a time, $s(t) = (t+1)/n$. Similar behaviour is observed for the network scientists coauthorship network. 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 [24] having no characteristic structure (□ in figure 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 figure 3 with a less convex structure than random graphs is the Little Rock Lake food web (■), 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,

$$X = 1 - \sum_{t=1}^{n-1} \max\{\Delta s(t) - 1/n, 0\}. \quad (2.1)$$

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 $X = 1$. In more simple terms, convexity X measures the number of steps of the procedure needed to cover the whole network t' relative to its size n , $X = (t' + 1)/n$. 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 4 in appendix A.

The growth $\Delta s(t)$ shows a notable transition in the autonomous systems graph from 2000 and the *Caenorhabditis elegans* protein interactions network (▲ and ● in figure 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 [25,26]. 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 [8].

3. Convexity under randomisation

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 randomisation by edge rewiring [27].

All empirical networks analysed 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 equation (2.1) is only sensible for connected networks [8]. In disconnected networks, there are no (shortest) paths between 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,

$$\begin{aligned} Xs &= s - s \sum_{t=1}^{sn-1} \max\{\Delta s(t) - 1/sn, 0\} \\ &= s - \sum_{t=1}^{sn-1} \max\{s\Delta s(t) - 1/n, 0\}, \end{aligned} \quad (3.1)$$

while all other details are the same as in equation (2.1). 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 $s = 1$. Note, however, that the measure is not sensible for disconnected networks with $s \ll 1$.

Figure 4 demonstrates the robustness of convexity in different empirical networks and synthetic graphs presented in appendix A. The plot shows the evolution of corrected convexity

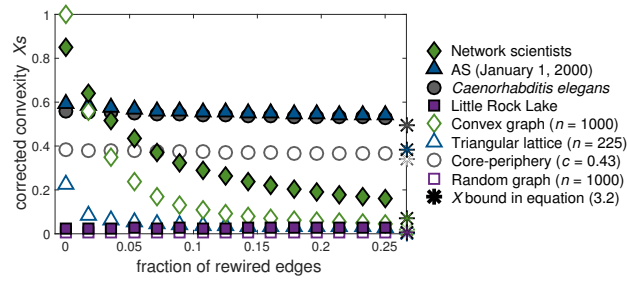


Figure 4. Evolution of convexity in empirical networks and synthetic graphs under degree-preserving randomisation by edge rewiring. The plot shows corrected convexity X_s for different fractions of rewired edges and the asterisks mark the lower bound for X . The symbols are averages over 25 independent randomisations, whereas the errors bars are smaller than the symbol sizes.

X_s under degree-preserving randomisation by edge rewiring [27], 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 [28].

In the case of the Little Rock Lake food web and the Erdős-Rényi random graphs that lack convexity $X_s \approx 0$ (■ and □ in figure 4), 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 figure 4). In particular, after rewiring 5% of the edges, corrected convexity X_s drops to less than half, although this percentage is about a hundred times smaller than what is normally needed to fully randomise a network or graph [29]. Thus, as anticipated above, convexity is a very sensitive property that is not resilient even to small perturbations of network structure.

Different behaviour is observed for the autonomous systems graph from 2000 and the *Caenorhabditis elegans* protein interactions network (▲ and ● in figure 4), where corrected convexity X_s 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 figure 7). Since pendant nodes retain convexity during the convex expansion procedure in section 2, and the rewiring process retains the degrees of the nodes, convexity X is bound by

$$X \geq n_1/n, \quad (3.2)$$

where n_1 is the number of pendant nodes and n the number of all nodes. The bound in equation (3.2) is marked by the asterisks in figure 4 and is non-trivial for scale-free networks [30] 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 analyse the robustness of convexity also under full randomisation, where the bound in equation (3.2) no longer holds.

4. Convex skeletons of networks

Section 3 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 2 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 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.

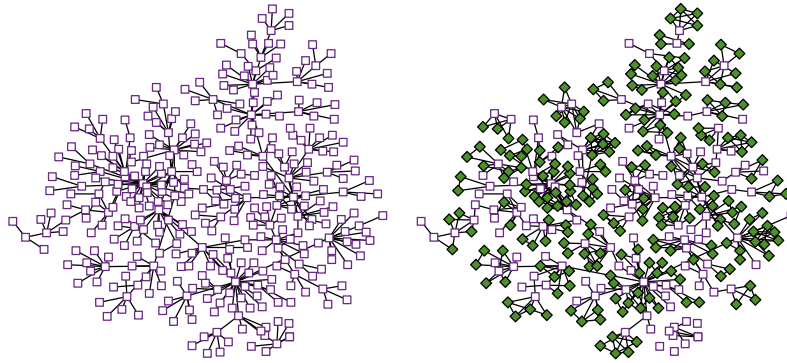


Figure 5. (left) A spanning tree and (right) a convex skeleton of the network scientists coauthorship network extracted as in table 1. The nodes with the clustering coefficient above $C > 0.5$ are represented with \blacklozenge and the remaining nodes with \blacksquare , while the layout was computed with the Large Graph Layout [31].

(a) Definition of convex skeleton

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 2, a fully convex skeleton would be 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 generalisation of a network spanning tree in which each edge can be replaced by a clique of arbitrary size (see the right side of figure 5). 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 equation (3.1).

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, figure 5 shows realisations of a spanning tree and a convex skeleton of the network scientists coauthorship network. Consider the average node clustering coefficient $\langle C \rangle$ [32] 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.

(b) Extraction of convex skeletons

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.

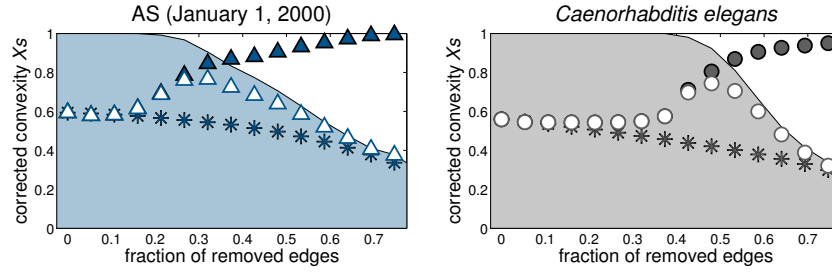


Figure 6. Evolution of convexity in core-periphery networks under targeted removal of edges based on the node c-centrality. The plots show the fraction of nodes in the largest connected component s (filled areas), convexity X (▲ and ●) and corrected convexity X_s (△ and ○), while the asterisks mark the value of X_s under random removal of edges. The symbols are averages over 25 independent removals, whereas the errors bars are not visible.

The asterisks in figure 6 show the evolution of corrected convexity X_s 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 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 figure 6 break apart before convexity X increases, resulting in a monotonically decreasing corrected convexity X_s . The same decreasing trend is observed also for targeted removal of edges based on the standard measures of centrality [2,33,34].

As mentioned in section 2, the main source of non-convexity in these networks is the existence of a non-convex core called the network c-core [8]. The c-core is defined as the subset of nodes included in the majority of convex subgraphs after fifteen steps $t = 15$ of the convex expansion procedure as in figure 3. The particular choice of the threshold $t = 15$ does not impact the results as long as $t \geq 2\langle\ell\rangle$ [8], where $\langle\ell\rangle$ is the average distance between the nodes. For example, the left side of figure 7 shows the c-core of the autonomous systems graph from 1999. 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 centre 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 centre of the network c-core.

Let p_i be the probability that node i is included in the c-core, which is estimated as the fraction of times node i is included in convex subgraphs after fifteen steps $t = 15$ of convex expansion. 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 Γ_i is the neighbourhood 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

$$\begin{aligned} c_i &= \sum_{j \in \Gamma_i} p_j - \sum_{j \in \Gamma_i} 1 - p_j \\ &= -k_i + 2 \sum_{j \in \Gamma_i} p_j, \end{aligned} \quad (4.1)$$

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, equation (4.1) reduces to $c_i = k_i$.

The symbols in figure 6 show the evolution of convexity under targeted removal of edges based on the node c-centrality. The edges are removed in decreasing order of $c_i + c_j$, where i and

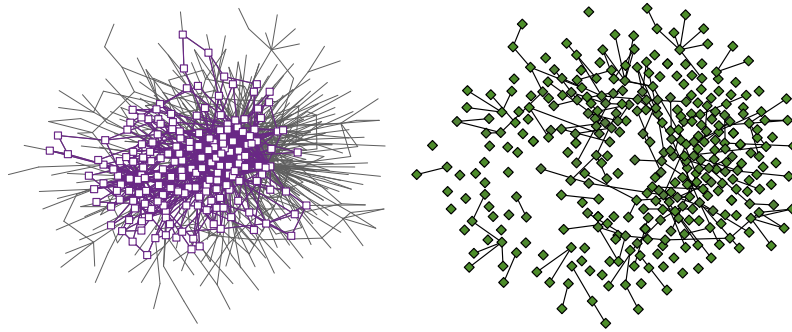


Figure 7. Division of the autonomous systems graph from 1999 into (left) the c-core represented with \square and (right) a convex periphery represented with \diamond . The layout was computed from the full network and is consistent between the figures, while other details are the same as in figure 5.

j are the endpoints of an edge. We remove a single edge at a time and recalculate the c-centrality of nodes according to equation (4.1) 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 figure 6, 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 optimising 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, optimising 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 optimise the properties of a network that should ideally be retained in a convex skeleton such as the average node clustering coefficient $\langle C \rangle$ [35]. Consider an edge with endpoints i and j , and let ΔC_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 become disconnected.

The extraction of convex skeletons is obviously non-deterministic. To see this, consider a ring graph, where a convex skeleton can be obtained by removing an arbitrary edge. Convex skeletons extracted from empirical networks are still remarkably deterministic. Figure 8 shows multidimensional scaling maps of different empirical networks, the corresponding Erdős-Rényi random graphs [24], network spanning trees and convex skeletons. The empirical networks include the network scientists coauthorship network, the autonomous systems graph from 1999 and the *Saccharomyces cerevisiae* protein interactions network. The maps are performed over graph edit distance GED defined as the number of edges that need to be inserted or deleted to translate one graph or network to another.

The maps show that convex skeletons are much closer to empirical networks than spanning trees and random graphs. More importantly, the extracted convex skeletons are very similar to

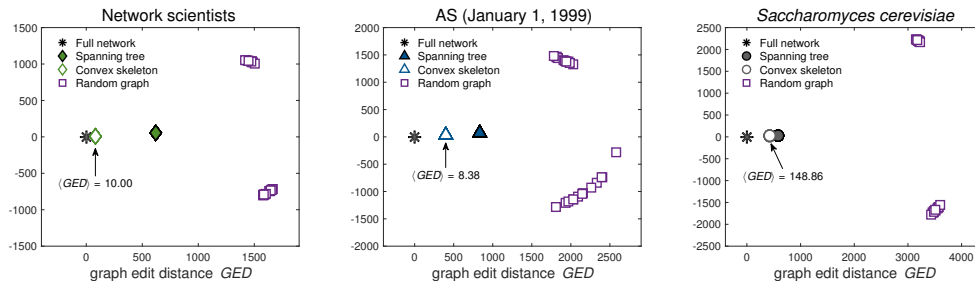


Figure 8. Multidimensional scaling maps of empirical networks, random graphs, spanning trees and convex skeletons extracted as in table 1. The maps are performed over graph edit distance GED between 25 independent realisations and have been translated thus the networks appear in the origin. Note that the symbols for spanning trees and convex skeletons overlap. The annotations show the average distance $\langle GED \rangle$ between the convex skeletons.

one another. The average distance $\langle GED \rangle$ between the convex skeletons is only 1.20% of the edges for the coauthorship network, 0.99% for the autonomous systems graph and 9.45% for the protein interactions network. The particular structure of empirical networks therefore allows for a robust extraction of convex skeletons, in contrast to say a ring graph. As shown in the remaining of the paper, convex skeletons also retain the node degree distribution, network clustering and connectivity, distances between the nodes, node position and network community structure.

In a more broader context, removal of network edges as for extracting convex skeletons is a popular technique for analysing the structure and dynamics of empirical networks. For instance, the emergence of connectivity in a network can be studied by random removal of edges known as inverse bond percolation in physics literature [36]. Edges that destroy the connectivity of a network are called bridges [37]. Moreover, random and targeted removal of edges is also commonly adopted for network sampling [38], sparsification or simplification [11] and for revealing network hierarchy [39]. Yet another example is removal of spurious edges in a network while retaining its basic properties [40] or highlighting specific properties [41].

(c) Properties of convex skeletons

Table 1 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 1 also shows the statistics of network spanning trees [42], 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 optimise 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 [32,43], 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 $\mathcal{O}(n)$ [44], $\langle \ell \rangle = \mathcal{O}(\sqrt{n})$ in a random tree [45] 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

Table 1. 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 X_s . The values are averages over 25 independent realisations and 100 runs of convex expansion.

Class	Network	degree $\langle k \rangle$			clustering $\langle C \rangle$			geodesics $\langle \sigma \rangle$			convexity X_s		
		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

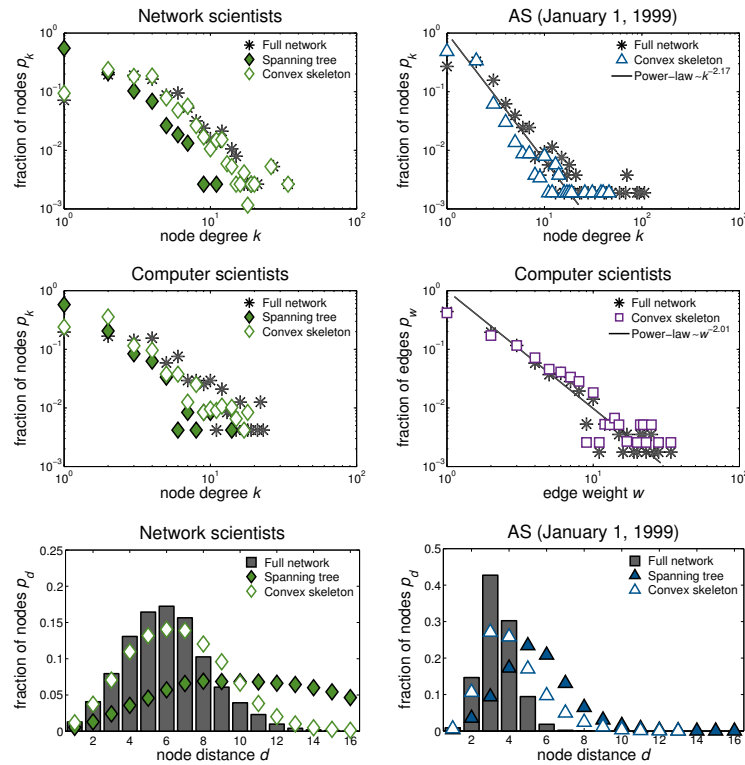


Figure 9. Node and edge distributions of empirical networks, spanning trees and convex skeletons. The plots show the fractions of (top) nodes p_k with degree k , (middle right) edges p_w with weight w and (bottom) pairs of nodes p_d at distance d over 25 independent realisations, whereas the error bars are not shown. The power-law distributions are maximum likelihood estimations [47] for the full networks.

convex skeleton might thus represent a suitable model for studying different dynamic processes on networks such as navigation and synchronisation [46].

Table 1 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, figure 9 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 coauthorship network (◆ and ◇). 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.

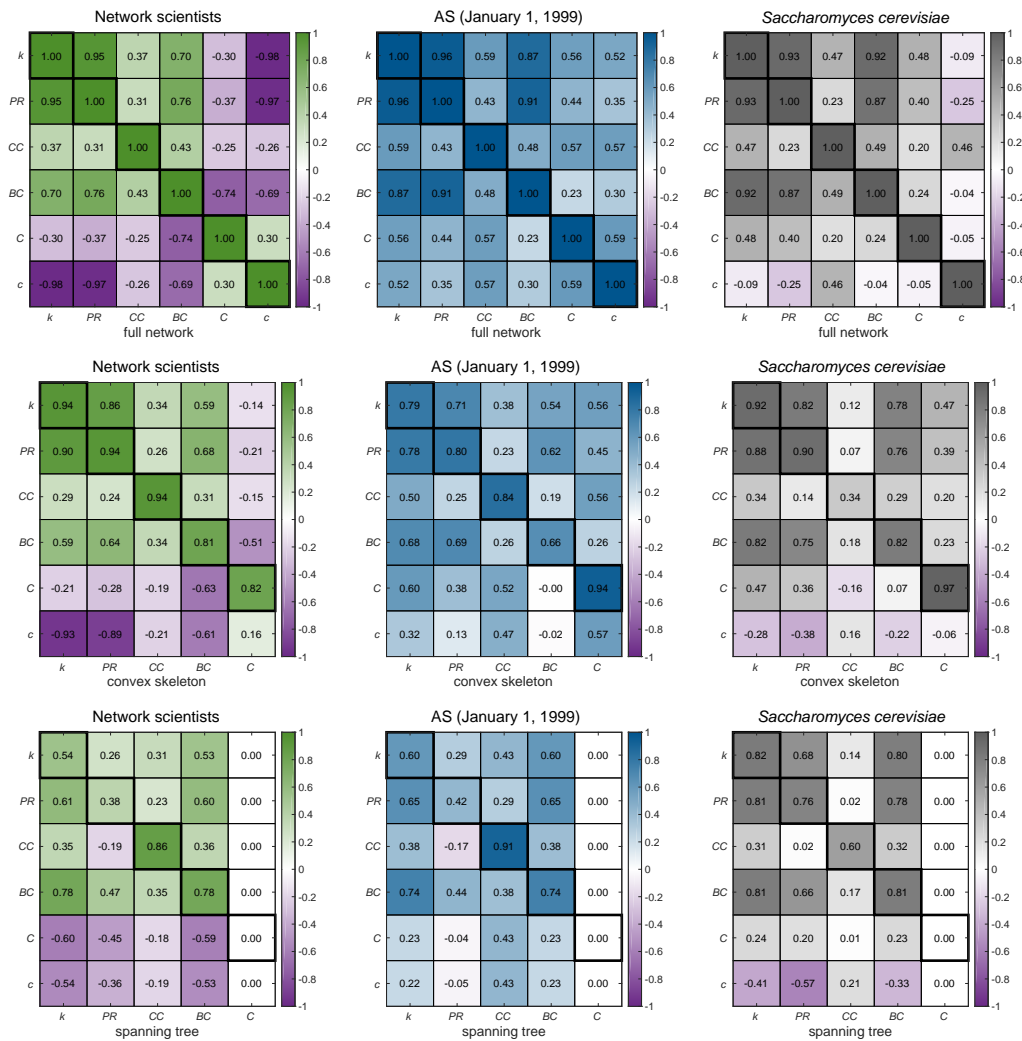


Figure 10. Heat maps showing Pearson correlation coefficient between the measures of node position (*top*) in empirical networks, and between networks and (*middle*) convex skeletons or (*bottom*) spanning trees in rows and columns, respectively. The measures include the node degree k and PageRank score PR , the closeness and betweenness centralities CC and BC , and the clustering coefficient C and c -centrality c . Note that c -centrality c is only reasonable for full networks and is therefore omitted for convex skeletons and spanning trees. The values are averages over 25 independent realisations.

5. Applications of convex skeletons

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 4, convex skeletons retain important structural properties of empirical networks. In this section, we more thoroughly analyse also node position and community structure of convex skeletons, and compare different backbones extracted from the Slovenian computer scientists coauthorship network.

Table 2. Comparison between community structure of empirical networks (N), and spanning trees (ST) and convex skeletons (CS). The structure is revealed with the map equation algorithm, and compared through the number of communities, normalised mutual information *NMI* (higher is better) and normalised variation of information *NVI* (lower is better). The values are averages over 25 independent realisations.

Class	Network	# communities			<i>NMI</i>		<i>NVI</i>	
		N	CS	ST	CS	ST	CS	ST
Collaboration	Jazz musicians	6.0	26.6	25.0	0.49	0.36	0.57	0.75
	Network scientists	37.3	40.5	50.0	0.96	0.87	0.06	0.22
	Computer scientists	24.0	28.0	31.3	0.88	0.81	0.19	0.33
Protein interactions	<i>Plasmodium falciparum</i>	138.0	132.8	140.2	0.83	0.77	0.33	0.45
	<i>Saccharomyces cerevisiae</i>	162.7	164.1	175.2	0.93	0.91	0.13	0.18
	<i>Caenorhabditis elegans</i>	286.7	321.3	348.0	0.86	0.82	0.25	0.32
Autonomous systems	AS (January 1, 1998)	230.4	285.7	285.3	0.86	0.75	0.25	0.41
	AS (January 1, 1999)	70.2	74.9	80.0	0.76	0.64	0.40	0.61
	AS (January 1, 2000)	271.3	358.8	338.9	0.84	0.74	0.28	0.46

(a) Node position in convex skeletons

Figure 10 shows the Pearson correlation coefficient between the measures of node position in the network scientists coauthorship network, the autonomous systems graph from 1999 and the *Saccharomyces cerevisiae* protein interactions network. The networks are more thoroughly described in appendix A. The measures include different node centralities and link analysis algorithms. These are node degree and link analysis algorithm PageRank [48] defined as the probability that a random walker with teleports visits a node. Next, we consider geodesics-based measures called node closeness centrality [33] defined as the average reciprocal distance to other nodes and betweenness centrality [2] defined as the fraction of geodesics between all nodes that pass through a node. Last, we include also node clustering coefficient [32] already introduced in section 4 and c-centrality defined in equation (4.1).

The top row of figure 10 shows that spectral centrality measures node degree and PageRank score are highly correlated, as it is common in undirected networks. Similar holds for betweenness centrality. On the other hand, closeness centrality and clustering coefficient capture different aspects of node position in these networks. Finally, c-centrality is negatively correlated with spectral and betweenness centralities in the coauthorship network, moderately correlated with most measures in the autonomous systems graph, while no strong correlations are observed in the protein interactions network. Hence, c-centrality introduced in this paper is different from standard measures of centrality and link analysis, capturing different aspects of node position.

The middle row of figure 10 further shows correlations between the node position in empirical networks and convex skeletons. Notice that all considered aspects of node position are retained in convex skeletons with values of the correlation coefficient between 0.8 and 0.95 in most cases. In contrast, the center of the protein interactions network shifts in the convex skeletons resulting in weak correlation 0.34 of closeness centrality. In the bottom row of figure 10, we also compare the node position in networks to spanning trees, which do not retain the clustering. Consequently, the correlation of clustering coefficient is undefined shown as empty cells in the heat maps. Similar results are obtained with other network backboning techniques [2,11,12].

(b) Community structure of convex skeletons

We continue by comparing the community structure of empirical networks, and spanning trees and convex skeletons. The networks include different social collaboration and protein interactions networks, and autonomous systems graphs described in appendix A. The community structure is revealed with the map equation algorithm [49] that identifies communities by compressing the

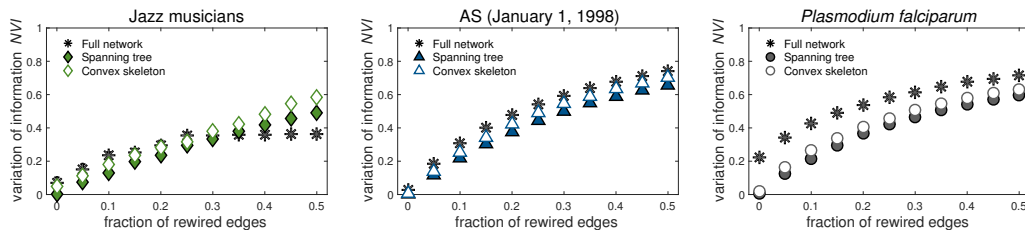


Figure 11. Robustness of community structure of empirical networks, spanning trees and convex skeletons revealed with the map equation algorithm. The plots show the distances between the community structures measured by normalised variation of information NVI for different fractions of rewired edges. The symbols are averages over 25 independent realisations, whereas the errors bars are not visible.

trajectory of a random walker on a network. Note that the algorithm highlights just one specific facet of community structure [50], which has however proved useful in the past [51–53].

Table 2 first shows the number of communities revealed in networks and backbones. These are comparable in all cases but one. In the network of collaborations between jazz musicians, the number of communities in the backbones is considerably higher than in the network, which is due to a high density of the network and much lower density of the backbones. Next, table 2 shows normalised mutual information [54] between the community structures of networks and backbones. In the case of independent structures, the measure equals zero, and is equal to one, when the structures are identical. The results show that the communities in networks are largely retained in the convex skeletons with values between 0.85 and 0.95 in most cases. Differently from convex skeletons, the values are consistently lower in the spanning trees. Table 2 also shows the values of normalised variation of information [55], which is a measure of distance in the space of partitions, thus lower is better. The values further confirm the observations from above.

Figure 11 demonstrates also the robustness of community structure in selected networks and backbones. The plots show the distance between community structures under degree-preserving randomisation [27] measured by normalised variation of information. Notice that both backbones strengthen the community structure, making it more robust to random perturbations of the edges. This is most notably observed in the protein interactions network. In summary, convex skeletons not only preserve the community structure of empirical networks, but at the same time make the structure more pronounced. In the remaining of the section, we further investigate this in the case of a coauthorship network.

(c) Network backbones of coauthorship network

Convex skeleton seem to provide a particularly reasonable abstraction of different social collaboration networks such as coauthorship networks [56], since these are unions of cliques by construction. In the following, we analyse convex skeletons extracted from the Slovenian computer scientists coauthorship network [19] and compare them with other network backbone techniques.

The coauthorship network was collected from the Slovenian national research database SICRIS [57] 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 scientists 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 [58]. Further details about the network are given in appendix A.

Table 3. 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 realisations.

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

Table 3 shows the standard statistics of the network, and the extracted spanning trees and convex skeletons. The results are consistent with section 4. 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, the convex skeletons retain the node degree distribution p_k of the network (\diamond in the middle row of figure 9).

Table 3 further shows the statistics of two alternative network backbone techniques based on edge betweenness [2] and saliency [12]. The betweenness of an edge is defined as the fraction of shortest paths between all nodes that traverse the edge, while the saliency 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 saliency of the edges was computed using the implementation provided in references [11,42], while we use our own implementation for edge betweenness [59].

The distribution of edge saliency is bimodal [12], which gives a natural definition of a high-saliency skeleton. On the other hand, the distribution of edge betweenness is heavy-tailed in the concerned coauthorship network. In table 3, 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-saliency 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 $X_s \approx 0.63$. 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 figure 9).

A realisation of a convex skeleton is shown in the top of figure 12, 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 figure 12 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 [60] 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

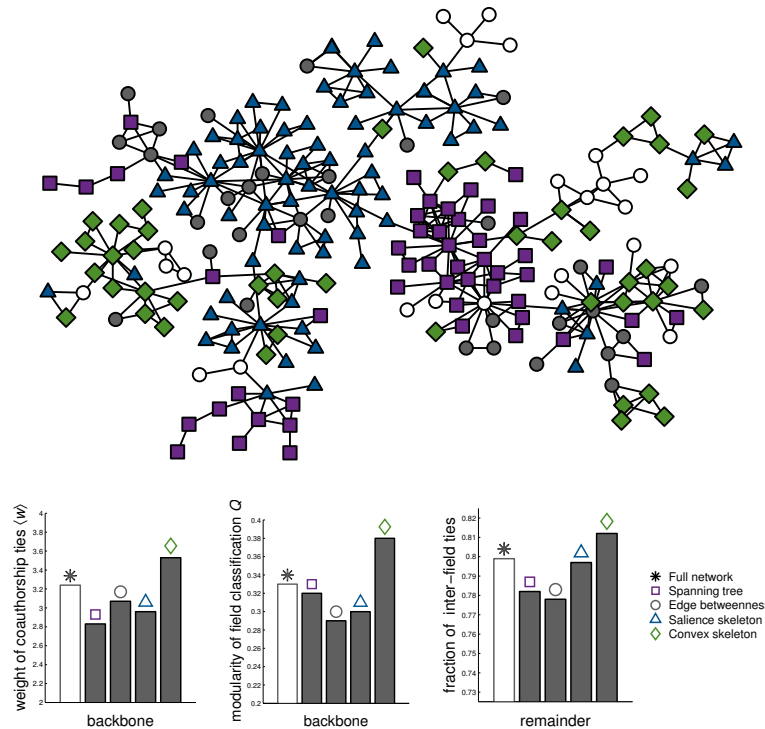


Figure 12. (top) A convex skeleton of the computer scientists coauthorship network. The node symbols represent primary research fields of the authors: computer theory (◆), information systems (■), intelligent systems (▲), programming technologies (○) and other (●). (bottom) Comparison of network backbones 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 realisations.

the backbones. In all cases, these properties are enhanced in a convex skeleton (◇) compared to the full network, differently from a spanning tree (□), high-betweenness backbone (○) and also high-saliency skeleton (△). Preliminary results on coauthorship networks from the SICRIS database for other scientific disciplines are consistent with above, while detailed results will be reported elsewhere.

6. 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 generalisation 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, distances, node position and also community structure, 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-saliency skeleton [12]. A convex skeleton seems to provide a particularly reasonable abstraction of social collaboration networks such as coauthorship networks, with applications in network analysis, modelling, visualisation, navigation and possibly also elsewhere. Still, convex skeletons are not the most reasonable abstraction of networks that are a priori non-convex like food webs or random graphs.

A convex skeleton can be seen as a particular case of a network sampling or sparsification technique [10,38,61], but with diverging goals. At the one hand, sparsification techniques try

to simplify the network by removing redundant or spurious edges [11,62], 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 efficient and effective techniques exist. The important fact is that one does not have to estimate convexity to extract a convex skeleton. Consequently, the most expensive part of the approach is ensuring that convex skeletons remain connected, which could be ensured in lesser time by using recent techniques for detecting bridges in networks [37]. Besides, we were here 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.

As stated above, 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.

Data Accessibility. The networks used in this work are freely available online within the SNAP datasets at <http://snap.stanford.edu/data>, the Pajek datasets at <http://vlado.fmf.uni-lj.si/pub/networks/data>, the KONECT repository at <http://konect.uni-koblenz.de> and the BioGRID repository at <https://thebiogrid.org>. The codes used for network convexity are available at <https://github.com/t4c1/Graph-Convexity>, the codes for convex skeletons at <https://github.com/t4c1/Convex-Skeleton>, the codes for network backboneng at http://www.michelecoscia.com/?page_id=287 and the codes for community detection at <http://www.mapequation.org/code.html>.

Competing Interests. The authors have no conflicting interests to declare.

Funding. 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.

Acknowledgements. The authors thank Tilen Marc, Vincent Traag, Ludo Waltman, Zoran Levanjić, Matjaž Perc, Dalibor Fiala and two anonymous reviewers for helpful comments and suggestions, Tadej Ciglaric for implementations of network convexity and convex skeletons, Michele Coscia for implementations of network backbones and Martin Rosvall for implementations of community detection algorithms. The authors also acknowledge a stimulating environment at the Centre for Science and Technology Studies of the Leiden University where part of this paper was written.

A. Networks and synthetic graphs

In this appendix, we provide the details of the empirical networks and synthetic graphs analysed 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 [17], coauthorships between network scientists parsed from the bibliographies of two review papers in 2006 [18] and coauthorships between Slovenian computer scientists up to 2010 extracted from the SICRIS database [19]. 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 [20]. The autonomous systems graphs are the

Table 4. Statistics of empirical networks and synthetic graphs analysed 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 realisations.

Class	Network	n	$\langle k \rangle$	$\langle \ell \rangle$	$\langle C \rangle$	X
Collaboration	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
Protein interactions	<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
Autonomous systems	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
Food webs	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
Synthetic graphs		2500	10.00	3.65	0.00	0.00
	Random graph	1000	10.00	3.26	0.01	0.01
		225	10.00	2.60	0.05	0.03
		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
		1000	5.97	8.40	0.90	1.00
	Convex graph	225	6.01	5.44	0.90	1.00

Internet maps at the level of autonomous systems [21] on the first day of 1998, 1999 and 2000 reconstructed from the University of Oregon Route Views project [63]. The food webs represent observed predator-prey relationships between the species of Little Rock Lake [22] and Florida Bay in wet and dry season [23].

Table 4 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 $\langle C \rangle$ [32] 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 equation (2.1) in section 2.

Table 4 also shows the statistics of the analysed synthetic graphs. First, these include triangular and rectangular lattices with the side of 15 nodes and the Erdős-Rényi random graphs [24] with $n = 225, 1000$ or 2500 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 in section 4 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 equation (3.2), 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 [64]. Note, however, that this construction process cannot generate all possible convex graphs. The graphs also lack scale-free degree distribution [30], although this could be imposed by an appropriate preferential attachment mechanism [65].

B. Convexity under full randomisation

In section 3, we study the robustness of convexity in different empirical networks and synthetic graphs under degree-preserving randomisation by edge rewiring [27]. In this appendix, we consider also full randomisation, 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 [24] that are non-convex.

Figure 13 shows the evolution of corrected convexity X_s in equation (3.1) under full randomisation. The results are conceptually the same as for degree-preserving randomisation in figure 4, with the only difference that convexity X is now no longer bound by equation (3.2).

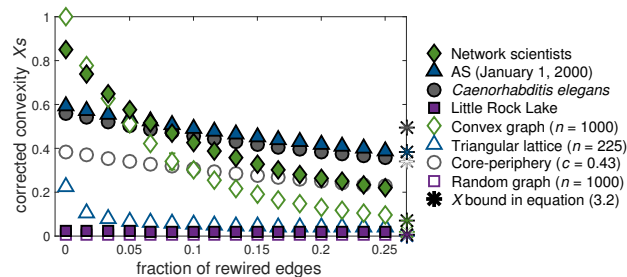


Figure 13. Evolution of convexity in empirical networks and synthetic graphs under full randomisation by edge rewiring, while other details are the same as in figure 4.

C. Node distributions of convex skeletons

In section 4, 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 figure 14 shows the node degree distributions p_k of full networks (asterisks) and the extracted convex skeletons (Δ and \circ). These are largely consistent and seem to follow a power-law $k^{-\gamma}$ with $\gamma \approx 2.3$ [47]. Furthermore, the convex skeletons preserve the distributions of distances between the nodes p_d shown in the bottom row of figure 14. On the other hand, the distances between the nodes increase significantly in the spanning trees (\blacktriangle and \bullet).

References

1. Milgram S. 1967 The small world problem. *Psychol. Today* **1**, 60–67.

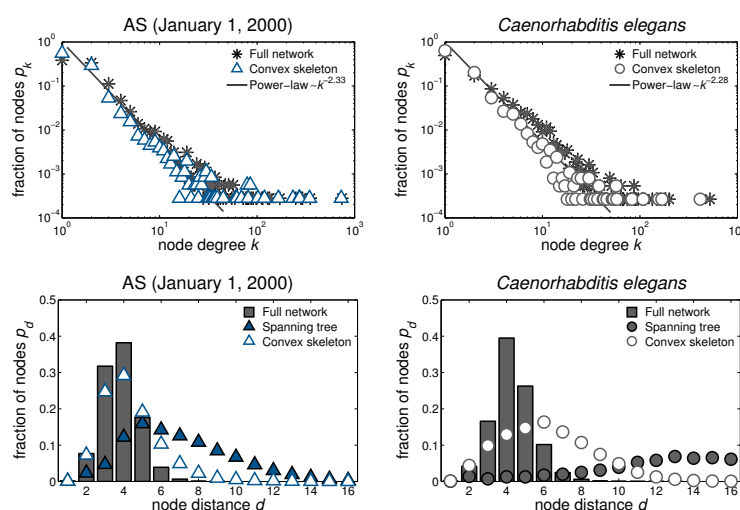


Figure 14. Node distributions of core-periphery networks and particular realisations of spanning trees and convex skeletons, while other details are the same as in figure 9.

2. Freeman L. 1977 A set of measures of centrality based on betweenness. *Sociometry* **40**, 35–41.
3. Harary F, Nieminen J. 1981 Convexity in graphs. *J. Differential Geom.* **16**, 185–190.
4. Farber M, Jamison R. 1986 Convexity in graphs and hypergraphs. *SIAM J. Algebra. Discr.* **7**, 433–444.
5. Van de Vel MLJ. 1993 *Theory of Convex Structures*. Amsterdam: North-Holland.
6. Pelayo IM. 2013 *Geodesic Convexity in Graphs*. New York: Springer.
7. Everett MG, Seidman SB. 1985 The hull number of a graph. *Discrete Math.* **57**, 217–223.
8. Marc T, Šubelj L. 2018 Convexity in complex networks. *Netw. Sci.* **6**, 176–203.
9. Dourado MC, Gimbel JG, Kratochvíl J, Protti F, Szwarcfiter JL. 2009 On the computation of the hull number of a graph. *Discrete Math.* **309**, 5668–5674.
10. Hamann M, Lindner G, Meyerhenke H, Staudt CL, Wagner D. 2016 Structure-preserving sparsification methods for social networks. *Soc. Netw. Anal. Min.* **6**, 22.
11. Coscia M, Neffke F. 2017 Network backboning with noisy data. In *Proceedings of the IEEE International Conference on Data Engineering* pp. 425–436 San Diego, CA, USA.
12. Grady D, Thiemann C, Brockmann D. 2012 Robust classification of salient links in complex networks. *Nat. Commun.* **3**, 864.
13. Batagelj V. 1988 Similarity measures between structured objects. In *Proceedings of the International Conference on Interfaces between Mathematics, Chemistry and Computer Science* pp. 25–39 Dubrovnik, Yugoslavia.
14. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U. 2002 Network motifs: Simple building blocks of complex networks. *Science* **298**, 824–827.
15. Pržulj N, Corneil DG, Jurisica I. 2004 Modeling interactome: Scale-free or geometric?. *Bioinformatics* **20**, 3508–3515.
16. Ciglaric T. 2017 Network Convexity. <https://github.com/t4cl/Graph-Convexity>.
17. Gleiser P, Danon L. 2003 Community structure in jazz. *Adv. Complex Syst.* **6**, 565–573.
18. Newman MEJ. 2006 Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E* **74**, 036104.
19. Blagus N, Šubelj L, Bajec M. 2012 Self-similar scaling of density in complex real-world networks. *Physica A* **391**, 2794–2802.
20. Stark C, Breitkreutz BJ, Reguly T, Boucher L, Breitkreutz A, Tyers M. 2006 BioGRID: A general repository for interaction datasets. *Nucl. Acids Res.* **34**, 535–539.
21. Leskovec J, Kleinberg J, Faloutsos C. 2007 Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data* **1**, 1–41.

22. Martinez ND. 1991 Artifacts or attributes? Effects of resolution on the Little Rock Lake food web. *Ecol. Monogr.* **61**, 367–392.
23. Batagelj V, Mrvar A. 2006 Pajek Datasets. <http://vlado.fmf.uni-lj.si/pub/networks/data>.
24. Erdős P, Rényi A. 1959 On random graphs I. *Publ. Math. Debrecen* **6**, 290–297.
25. Borgatti SP, Everett MG. 2000 Models of core/periphery structures. *Soc. Networks* **21**, 375–395.
26. Holme P. 2005 Core-periphery organization of complex networks. *Phys. Rev. E* **72**, 046111.
27. Maslov S, Sneppen K. 2002 Specificity and stability in topology of protein networks. *Science* **296**, 910–913.
28. Newman MEJ, Strogatz SH, Watts DJ. 2001 Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E* **64**, 026118.
29. Ray J, Pinar A, Seshadhri C. 2015 A stopping criterion for Markov chains when generating independent random graphs. *J. Complex Netw.* **3**, 204–220.
30. Barabási AL, Albert R. 1999 Emergence of scaling in random networks. *Science* **286**, 509–512.
31. Adai AT, Date SV, Wieland S, Marcotte EM. 2004 LGL: Creating a map of protein function with an algorithm for visualizing very large biological networks. *J. Mol. Biol.* **340**, 179–190.
32. Watts DJ, Strogatz SH. 1998 Collective dynamics of ‘small-world’ networks. *Nature* **393**, 440–442.
33. Bavelas A. 1950 Communication patterns in task-oriented groups. *J. Acoust. Soc. Am.* **22**, 725–730.
34. Freeman LC. 1979 Centrality in social networks: Conceptual clarification. *Soc. Networks* **1**, 215–239.
35. Ciglaric T. 2017 Convex Skeleton. <https://github.com/t4cl/Convex-Skeleton>.
36. Barabási AL. 2016 *Network Science*. Cambridge: Cambridge University Press.
37. Wu AK, Tian L, Liu YY. 2018 Bridges in complex networks. *Phys. Rev. E* **97**, 012307.
38. Blagus N, Šubelj L, Bajec M. 2017 Empirical comparison of network sampling: How to choose the most appropriate method?. *Physica A* **477**, 136–148.
39. Coscia M. 2018 Using arborescences to estimate hierarchicalness in directed complex networks. *PLoS ONE* **13**, e0190825.
40. Zeng A, Cimini G. 2012 Removing spurious interactions in complex networks. *Phys. Rev. E* **85**, 036101.
41. Van Mieghem P, Stevanovicacute D, Kuipers F, Li C, van de Bovenkamp R, Liu D, Wang H. 2011 Decreasing the spectral radius of a graph by link removals. *Phys. Rev. E* **84**, 016101.
42. Coscia M. 2017 Network Backboning. http://www.michelecoscia.com/?page_id=287.
43. Newman MEJ, Watts DJ. 1999 Scaling and percolation in the small-world network model. *Phys. Rev. E* **60**, 7332–7342.
44. Newman MEJ. 2010 *Networks: An Introduction*. Oxford: Oxford University Press.
45. Meir A, Moon JW. 1970 The distance between points in random trees. *J. Comb. Theory* **8**, 99–103.
46. Barrat A, Barthélemy M, Vespignani A. 2008 *Dynamical Processes on Complex Networks*. Cambridge: Cambridge University Press.
47. Clauset A, Shalizi CR, Newman MEJ. 2009 Power-law distributions in empirical data. *SIAM Rev.* **51**, 661–703.
48. Brin S, Page L. 1998 The anatomy of a large-scale hypertextual Web search engine. *Comput. Networks ISDN* **30**, 107–117.
49. Rosvall M, Bergstrom CT. 2008 Maps of random walks on complex networks reveal community structure. *P. Natl. Acad. Sci. USA* **105**, 1118–1123.
50. Schaub MT, Delvenne JC, Rosvall M, Lambiotte R. 2017 The many facets of community detection in complex networks. *Appl. Netw. Sci.* **2**, 4.
51. Lancichinetti A, Fortunato S. 2009 Community detection algorithms: A comparative analysis. *Phys. Rev. E* **80**, 056117.
52. Rosvall M, Bergstrom CT. 2011 Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. *PLoS ONE* **6**, e18209.

53. Peixoto TP, Rosvall M. 2017 Modelling sequences and temporal networks with dynamic community structures. *Nat. Commun.* **8**, 582.
54. Danon L, Díaz-Guilera A, Duch J, Arenas A. 2005 Comparing community structure identification. *J. Stat. Mech.* p. P09008.
55. Meila M. 2007 Comparing clusterings: An information based distance. *J. Multivariate Anal.* **98**, 873–895.
56. Newman MEJ. 2001 The structure of scientific collaboration networks. *P. Natl. Acad. Sci. USA* **98**, 404–409.
57. IZUM, ARRS. 2010 SICRIS Database. <http://www.sicris.si>.
58. Rodela R. 2016 On the use of databases about research performance: Comments on Karlovčec and Mladenčić (2015) and others using the SICRIS database. *Scientometrics* **109**, 2151–2157.
59. Brandes U. 2001 A faster algorithm for betweenness centrality. *J. Math. Sociol.* **25**, 163–177.
60. Girvan M, Newman MEJ. 2002 Community structure in social and biological networks. *P. Natl. Acad. Sci. USA* **99**, 7821–7826.
61. Leskovec J, Faloutsos C. 2006 Sampling from large graphs. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* pp. 631–636 Philadelphia, PA, USA.
62. Guimerà R, Sales-Pardo M. 2009 Missing and spurious interactions and the reconstruction of complex networks. *P. Natl. Acad. Sci. USA* **106**, 22073–22078.
63. Leskovec J, Krevl A. 2014 SNAP Datasets. <http://snap.stanford.edu/data>.
64. Slater N, Itzhack R, Louzoun Y. 2014 Mid size cliques are more common in real world networks than triangles. *Netw. Sci.* **2**, 387–402.
65. Hébert-Dufresne L, Allard A, Marceau V, Noël PA, Dubé LJ. 2011 Structural preferential attachment: Network organization beyond the link. *Phys. Rev. Lett.* **107**, 158702.