Graph Analysis of Blockchain P2P Overlays and Their Security Implications

Aristodemos Paphitis[1(✉)], Nicolas Kourtellis[2], and Michael Sirivianos[1]

[1] Cyprus University of Technology, Limassol, Cyprus
am.paphitis@edu.cut.ac.cy, michael.sirivianos@cut.ac.cy
[2] Telefonica Research, Barcelona, Spain
nicolas.kourtellis@telefonica.com

Abstract. In blockchain systems, similar to any distributed system, the underlying network plays a crucial role and provides the infrastructure for communication and coordination among the participating peers. As a result, the properties of the network define the level of security, availability, and fault tolerance within a blockchain system. This study aims to improve our understanding of the structural properties of peer-to-peer overlay networks that underpin blockchain applications. Our objective is to gain insights into the security and resilience of these systems. By analyzing seven distinct blockchain overlay networks and evaluating a comprehensive set of graph characteristics, we draw important conclusions about their overall robustness. Our findings reveal that major blockchain networks have vulnerabilities that make them susceptible to exploitation by malicious actors. Furthermore, despite relying on similar protocols for node discovery and network formation, we observe dissimilar characteristics among these blockchains.

Keywords: Blockchain · P2P Networks · Resilience

1 Introduction

Blockchain (BC) technology has garnered significant attention in recent years for its potential to revolutionize various industries and enhance trust in digital transactions [5,12,13,66]. The decentralized and immutable nature of blockchain systems has introduced novel solutions to long-standing problems, such as secure and transparent transactions, efficient supply chain management, and decentralized finance. However, while the benefits of blockchain technology have been widely discussed, the underlying peer-to-peer (P2P) networks that power these systems have received comparatively little scrutiny [25,27].

The P2P networks that support blockchain systems serve as the backbone of their operation, facilitating consensus, data propagation, and transaction validation. Understanding the structural properties, topological characteristics, and vulnerabilities of these networks is crucial for realizing the full potential of blockchain technology and ensuring its robustness against emerging threats [17,30]. Yet, the research community's attention has predominantly

B. Arief et al. (Eds.): SocialSec 2023, LNCS 14097, pp. 167–186, 2023.
https://doi.org/10.1007/978-981-99-5177-2_10

focused on the cryptographic and consensus aspects of blockchain systems, leaving the underlying P2P networks relatively unexplored.

This research paper aims to bridge this gap by delving into the largely uncharted territory of blockchain's P2P networks. By investigating the structure and behavior of these networks, we can gain valuable insights into their limitations, vulnerabilities, and potential improvements. This exploration is critical for devising effective strategies to enhance network resilience, scalability, and security in blockchain systems.

1.1 Research Question and Objectives

In this work, we aim to analyze the graph properties of underlying P2P overlays in blockchain networks to gain insights into their network robustness. Our goal is two-fold: First, we would like to understand the resilience properties of blockchain overlay networks, by uncovering potential vulnerabilities that might be exploited by adversaries to compromise the security of blockchain systems. Second, we would like to look into their structural properties to examine whether they are structured in a similar fashion and whether they exhibit properties similar to other well-known networks like the Web, the Internet, or Social Networks.

To address these questions, we conducted a study on the most important structural properties of seven distinct BC networks. Specifically, we continuously probed and crawled these BC networks over a period of 28 days to gather information about all available peers. We analyzed 335 network snapshots per BC network, resulting in a total of 2345 snapshots. At regular intervals, we constructed connectivity graphs for each BC network, consisting all potential connections between peers. We then analyzed the structural graph properties of these networks and compared them across the seven BC networks.

2 Background and Related Work

The following seven networks are included in our study: Bitcoin, Bitcoin Cash, Dash, Dogecoin, Ethereum, Litecoin, and ZCash. These networks were chosen based on their importance and high market capitalization as indicated by [15]. All networks use similar overlay implementations [19]. Two exceptions are Dash and Ethereum. Dash uses similar network messages as Bitcoin but employs a two-tier network consisting of mining nodes (peers) and master nodes that facilitate network discovery and message dissemination. Ethereum uses a different set of protocols based on the Kademlia [44] P2P architecture for network discovery.

2.1 Bitcoin Overlay Network

In the Bitcoin overlay network, nodes communicate through unencrypted TCP connections to create a random P2P network. The security of Bitcoin is achieved through its Proof-of-Work consensus protocol, ensuring that all nodes see the same version of the blockchain. The protocol is outlined in the Bitcoin developer

guide [24]. To better understand its intricacies, we alse studied previous research papers [7,34,48] and analyzed the source code of Bitcoin's reference client [23].

When a node joins the network, it queries a set of hardcoded DNS seeds in the Bitcoin Core client to obtain the IP addresses of full nodes that accept new connections. Once connected, a node receives unsolicited `addr` messages from its peers, containing IP addresses of other nodes in the network. The client can also proactively request additional addresses using `getaddr` messages. The response to a `getaddr` message can include up to 1000 peer addresses. All known addresses are stored in-memory by the address manager (`ADDRMAN`) and periodically saved to disk in the `peers.dat` file. This allows the client to directly connect to these peers on future launches without relying on DNS seeds.

In terms of connections, when Alice initiates a connection to Bob, it is considered an outbound connection from Alice's perspective and an inbound connection for Bob. Each peer is permitted to establish up to eight outbound connections to active Bitcoin nodes and maintain a maximum of 125 active connections in total.

2.2 Ethereum Overlay Network

Ethereum's network protocols utilize both UDP for node discovery and TCP TLS channels for other communication, as described in the Ethereum Developer's Guide to the P2P network [26]. Node discovery in Ethereum is based on the Kademlia routing algorithm, which employs a distributed hash table (DHT) [44]. Each peer in Ethereum has a unique 512-bit node ID, and the XOR operation is used to compute the distance between two node IDs.

Ethereum nodes maintain internally 256 buckets, with each bucket containing a number of Etehreum-peers node IDs. Peers assign known nodes to specific buckets based on their XOR distance from themselves. To find peers, a new node initially adds a pre-defined set of bootstrap node IDs to its routing table. It then sends a `FIND_NODE` message to these bootstrap nodes, specifying a random target node ID. In response, each peer provides a list of 16 nodes from its routing table that are closest to the target. The node subsequently attempts to establish a certain number of connections (typically 25 or 50) with other peers.

2.3 Related Work

Delgado-Segura *et al.* [19] emphasize that blockchain P2P networks present unique characteristics and challenges compared to previously known P2P networks. Similarly, Dotan *et al.* [25] recognize the distinct requirements of blockchain overlay networks and highlight the lack of understanding of their fundamental design aspects. Their work identifies differences and commonalities between blockchains and traditional networks, emphasizing open research challenges in network design for distributed decentralized systems.

Miller *et al.* [45] were the first to successfully infer Bitcoin's public network topology. They discovered links between nodes using the timestamps included in `addr` messages. In their work, they found indications that the Bitcoin network is

not purely random, having a skewed degree distribution. Biryukov *et al.* [8], proposed sending fake addresses to reachable nodes and then monitor their propagation to the network to infer connections among peers. Delgado-Segura *et al.* [18] inferred Bitcoin's network topology using orphaned transactions. Their method relies on subtleties of Bitcoin's transaction propagation behavior. Their results also indicate that Bitcoin's testnet does not resemble a random graph. Neudecker *et al.* [49] used timing analysis of transaction propagation delays, as observed by a monitoring node, to infer the topology. Their approach requires a highly connected monitoring node and the creation of transactions. Grundmann *et al.* [34], proposed mechanisms for Bitcoin topology inference based on double–spending transactions. However, this method was not intended to perform a complete network topology inference due to the high incurred cost of fabricated transactions. Taking advantage of block-relay mechanisms, Daniel *et al.* [16] presented a passive method to infer the connections of mining nodes and their direct neighbors in the ZCash network. Neudecker and Hartenstein [50] surveyed the network layer of permissionless BCs, simulated a passive method to infer the network topology with substantial accuracy, and highlighted that keeping the network topology hidden is an intermediate security requirement.

To hinder attacks that utilize topology inference, Bitcoin Core developers implemented a series of changes to the network protocol. To mitigate the methods described in [8], the Bitcoin client now rejects `getaddr` requests from inbound connections [22]. To address adversarial methods proposed by Miller et al. [45], nodes stopped updating the timestamp field in the address manager, making it impossible to infer active connections [52]. Neudecker's timing analysis is also rendered impractical due to code changes [21].

Works like [31,61] shed light on the unreachable side of Bitcoin. More recently, Grundmann *et al.* calculated the degree distribution of reachable peers in the Bitcoin network, by leveraging a spam wave of IP addresses [32].

Despite previous efforts, little is known regarding the structure and topological properties of BC overlay networks. Past studies have mainly focused on methods for inferring the well-hidden topology of Bitcoin, either against the whole network or a specific peer. With the exception of [45], these studies were validated against the Bitcoin testnet [18], or against selected nodes [34,49].

Graph Analysis and Its Applicability to Blockchain Networks. Graph analysis is a powerful tool for understanding network resilience. It has been widely used to characterize complex networks and investigate resilience in various fields and applications in a variety of network types, such as technological, social, infrastructure, transportation, and biological. A recent survey highlights the prevalence of graph analysis with respect to network resilience research [29]. Graph analysis has also been used extensively to study the transaction graphs of major BCs, namely Bitcoin and Ethereum [4,11,37,41,53,64,65]. Using similar methods, Lee *et al.* analyze Bitcoin's Lightning Network [39]. In their work, they found that it exhibits strong scale-free network characteristics, implying that the

Lightning Network can be vulnerable to DDoS attacks targeting some central nodes in the network.

Although it is an indispensable tool for assessing network robustness, graph analysis has not been applied to BC networks. We believe that a contributing factor to this omission in the literature is mainly the lack of topological information on the underlying networks.

A recent work by Paphitis *et al.* [55], examines the partition resistance of these networks against random failures and targeted attacks, as well as the potential for malicious attacks facilitated by the presence of common entities across different networks and their placement in Autonomous Systems.

To our knowledge, this is the first study to focus on the structural properties of P2P networks of multiple blockchains. By crawling the reachable nodes in the network, we circumvent the challenges of topology inference and build a simple network monitor that can probe seven different BC networks in parallel to uncover all potential connections. Our implementation does not require high connectivity in each network and is free of transaction processing costs, allowing greater scalability. Finally, we analyze the graph properties of BC overlay networks to compare their structure and investigate how their characteristics affect their security properties.

3 Methodology

To analyze a graph, information is needed about the graph topology, i.e., how the vertices are connected to each other. Acquiring exact topological information on a dynamic P2P network is a challenge. More so in blockchain overlays, where this information is considered paramount for the security of the network, and, as previously discussed, a variety of topology hiding techniques are used [34,45,50].

3.1 Data Collection Process

To mitigate the challenges associated with acquiring a precise snapshot of the overlay network, as discussed in Sect. 2.3, we employ the same approach that the authors introduced in a related research study conducted by Paphitis *et al.* [55]. In more detail, we collect all known peers for each reachable node in the P2P network. We achieve this by repeatedly sending `getaddr` messages to each connected node. Nodes receiving the message respond with an `addr` message that contains a number of IP addresses known to the replying peer. Each BC is assigned to a process that creates hundreds of user-level threads. Intermediate data collected during crawling are stored in an in-memory key-value store, each process having its own instance. Following the protocols of each BC, each process connects to its assigned network and recursively asks each discovered node for its known peers. Each new discovered node is stored in a `pending` set. Threads constantly poll their `pending` set for a new node, initiate a connection, and retrieve a list of its known peers.

Upon successful connection to a peer, its entry is removed from the `pending` set. On each response received to a `getaddr` message, the process makes an entry, mapping the originating node (N_{or}) to the peer list it knows of: $N_{or} \rightarrow \{P_0, P_1, ..., P_n\}$, where P_{0-n} are the peers included in the reply of N_{or}. In effect we draw an outgoing edge from N_{or} to each peer in the reply. This entry is stored in the `edges` set. When the `pending` set becomes empty, the crawler starts over. The `edges` set remains intact and is updated in subsequent rounds. Replies from nodes that are already mapped in the `edge` set are appended to the respective entry. After a period of approximately two hours, all processes synchronize and dump their `edge` set to storage.[1] After the dump, all sets are emptied and each process restarts and repeats the same procedure.

In this fashion, we construct *connectivity graphs*, i.e., graphs that contain all possible connections that could be made in the network. Our methodology is presented in more detail in [55], where we also show that this method is capable of reconstructing the contents of the address manager (`ADDRMAN`). In the same work, the accuracy of the collected data is validated against a controlled monitoring node, as well as against external data sources. The collected data set is available at [54]. The observed graphs were analyzed using the SNAP [40] and NetworkX [35] packages.

Ethical Considerations. We emphasize that we only collected and processed publicly available data, with no intention of deanonymizing users or establishing connections between individuals or organizations and their IP addresses. No personally identifiable information was collected during the study. We have gathered IP addresses known to each node using the node discovery mechanism of the protocol. We only established short-lived connections with discovered peers and responded only to the expected initial handshake. Finally, we have refrained from frequent retransmissions and requests to avoid exhausting a peer's network resources.

3.2 Limitations

Arguably, the observed connectivity graphs contain a number of false edges in the graph, i.e., they contain edges that do not exist in the real network. To understand how much the network properties are affected by these errors, we turn to an area of research that deals with measurement errors in network data. Wang *et al.* [60] studied the effect of measurement errors on node-level network measures and found that networks are relatively robust to false positive edges. Similarly, Booker described the effects of measurement errors on the attack vulnerability of networks [9]. Booker also finds that false positive edges have the least impact on the effectiveness of random and targeted attacks.

To investigate the accuracy of the observed graphs compared to real networks, we adapt the methods used by Booker and Wang [9,60]. In particular, we

[1] Two-hour periods were chosen, to allow future analysis of longitudinal evolution of the networks. We believe that a larger window would not capture enough of the evolution dynamics.

construct a random graph G_{real} consisting of $N = 1000$ vertices, assigning to each vertex k outgoing links, so that k is drawn from the real Bitcoin degree distribution, as calculated by Grundmann *et al.* in [32]. Then, starting with G_{real}, we add random edges with the constraint that the resulting observable graph, G_{obs}, has a degree sequence drawn from the observed degree distribution we obtain using the methodology described above (see Sect. 3.1), by probing peers for their known addresses. Since Grundman's calculated degree distribution applies only to reachable peers, we also use the degree sequence of reachable peers, ignoring any unreachable nodes. In this way, the resulting *observable* graph G_{obs} contains a number of real links plus an additional number of edges that correspond to the known peers of each node (false positive edges in [9]). To inspect the effects of false edges on the observed network characteristics, we calculated a set of graph metrics for both graphs G_{real} and G_{obs} and compared them.

The average values calculated from 20 simulations are presented in Table 1. G_{obs} exhibits more robust characteristics, evident by a higher clustering and a lower average betweenness. This is expected as it contains much more edges than G_{real}. On the other hand, the average shortest-path values are very close in both sets of graphs. The results of this simulation show that the differences in the calculated metrics are consistent and almost constant. Thus, the calculated properties of the observed graphs can serve as a bound to the properties of the real graphs. The Chebyshev distance in the last row indicates the maximum absolute distance between the corresponding values.

Table 1. Measurement error simulation results. *Betweenness not normalized.

Metric→	Avg. Shortest Path	Average Degree	Clustering	Assortativity	Avg Betweenness*
G_{real}	1.89	114.6	0.21	-0.02	447,893
G_{obs}	1.56	437.7	0.63	0.07	280,648
Chebyshev Distance	0.34	333.9	0.43	0.12	172,904

4 Analysis of P2P Overlays

We aim to answer the following questions about BC overlay networks: a) What are their structural properties and network characteristics? b) Are they all structured similarly? c) Do they share common properties? d) Do their properties relate to other networks such as the Internet topology, Web or social networks, or are they random? e) How do their characteristics affect security? This section presents metrics, adapted from previous research [1,2,29,36,56,62], to assess the resilience of a blockchain network. These metrics are considered standard for analyzing networks and understanding non-obvious properties [62], and can be used to evaluate network resilience to errors and attacks. In this section, we use the following notation for clarity and conciseness: each set of edges corresponds to a graph, denoted S_c^t, representing a snapshot of the BC network c, on date t.

Other Online Networks. Online social networks, the Web and the Internet/AS topology are the most studied online networks [10,42,46,58]. This section shares much of the methodology used in such studies. It is reasonable to compare the structure of blockchain networks with the structure of other known technological and information networks. Nevertheless, we are aware that: a) the studied graphs do not represent the actual network topology, and b) the P2P structure of blockchain networks is fundamentally different from the aforementioned networks. The comparisons made throughout this section serve as a reference point for the results collected. However, we note that useful conclusions can be drawn about blockchain overlays, especially when comparing the different networks between them, since they implement similar protocols [19] and we follow the same measurement methodology.

4.1 Fundamental Graph Properties

The most important properties of the derived graphs are summarized in Table 2. The metrics were individually calculated on each graph S_c^t and then averaged. The values extracted from the collected data sets match the values reported in related measurement work [16,20,38]. Specifically, each day, the monitoring node was able to discover 120081 nodes in Bitcoin, 19543 in Ethereum, and 4132 in Zcash (reporting median values). On average, the monitoring node made more than 1.3 M requests per day, covering all networks.

The diameter of a connected graph is defined as the longest shortest path between all pairs of nodes. A smaller diameter usually indicates better robustness, as adding edges would shorten the longest shortest path between distant nodes, making the network more tightly coupled. The Average Shortest Path (ASP) is closely related to network connectivity. Smaller average shortest paths imply increased robustness, since the distance between any pair of nodes is reduced. All networks appear to be well connected, given the size of their largest connected component, their low diameters, and short ASP. Moreover, we observe that Dash is markedly the most dense network and is almost fully connected. It has a strongly connected component (SCC), i.e., a subgraph in which all nodes are reachable from all other nodes. The SCC comprises 75% of the total network nodes. Larger blockchain networks have a smaller SCC compared to smaller ones. Networks differ mainly in size, but this is independent of their protocols; in a free market, user perception of value determines a network's popularity.

4.2 Degree Distribution

The degree (number of links with other nodes) distribution affects many network phenomena, such as network robustness and efficiency in information dissemination [6]. In addition, random networks have binomial degree distributions, while in real systems, we usually encounter highly connected nodes that the random network model cannot account for. In Fig. 1, we plot the complementary cumulative distribution (CCDF) of the out-degree of all snapshots collected for all networks in our study.

Table 2. Basic network graph metrics per BC network (average values across all collected snapshots) For each metric we highlight the value that indicates less resilience. * Normalized Betweenness using the min-max method.

Network:	Bitcoin	Bitcoin Cash	Dash	Dogecoin	Ethereum	Litecoin	Zcash
Nodes	120k	33k	9k	2.1k	17.5k	11.7k	4.1k
Edges	37M	748k	29M	330k	556k	3.7M	231k
Connected Component	1	1	1	1	0.99	1	1
Strogngly Connected Component	0.06	**0.03**	0.75	0.2	0.13	0.14	0.06
Diameter	4	4	3	3	**5**	3	4
Density	0.004	**0.001**	0.5	0.11	0.004	0.047	0.024
Avg. Degree	254.16	**20.22**	2370.88	126.45	31.14	278.85	48.84
Assortativity	-0.2	**–0.64**	–0.06	–0.13	–0.02	–0.01	–0.22
Reciprocity	0.32	0.21	0.49	0.34	0.02	0.27	0.25
Global Clustering Coefficient	0.049	0.011	0.166	0.28685	**0.0022**	0.0735	0.3094
Avg. Shortest Path	2.55	2.82	1.93	1.77	**3.78**	1.96	1.72
Average Betweenness	**2.40e+07**	1.95e+06	2.74e+06	1.62e+04	1.12e+06	5.35e+05	1.43e+04
Normalized Betweenness*	**49727**	23018	8666	1257	8871	8160	1462

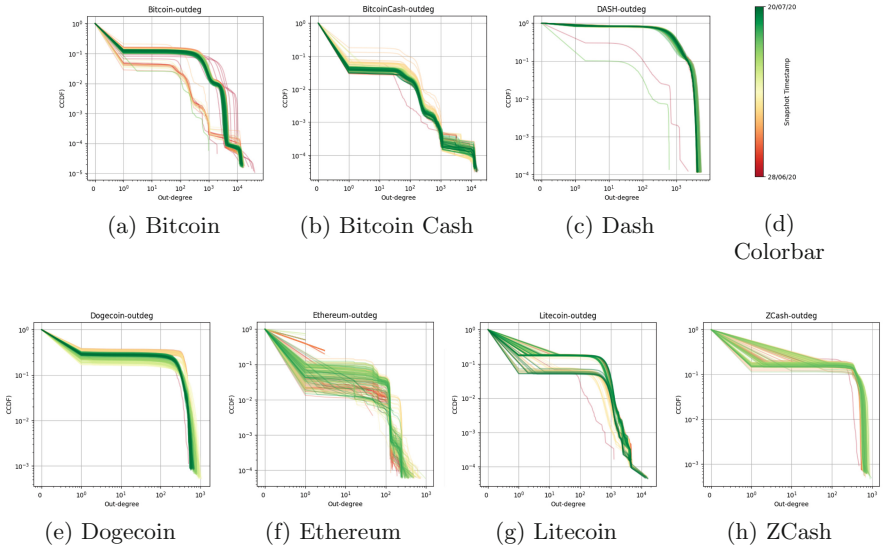
We color the snapshots according to their timestamp. Our first observation is that networks such as Bitcoin and Ethereum manifest considerable variability in degree distribution between snapshots. In contrast, the degree distributions in Dash and Dogecoin have less variability (seen by the distance between snapshots). Another interesting observation is that in most networks we have a high fraction of unreachable nodes, either because they are offline or behind NATs. This observation confirms the findings of Wang and Pustogarov [61] who studied the prevalence and deanonymization of unreachable peers. The presence of unreachable peers is discussed in a following paragraph.

Our results also suggest that these blockchain networks have heavy-tailed degree distributions. We further discuss their best distribution fit and their scale-free property in a following paragraph. Finally, we observe significant deviations from the network protocols. In Bitcoin, for instance, one would expect that reachable nodes would have at least 1K out-degree, since Bitcoin clients with the default parameters are set to respond with 1K known peers. In contrast, we observe a number of nodes with an out-degree less than 100, i.e., nodes reply with fewer addresses than the default parameter. We note that this behavior along with network churn could be leveraged to amplify eclipsing or network attacks similar to the SyncAttack [57].

Comparing the network densities, we observe that DASH has a very tight network, while Bitcoin, BitcoinCash, and Etherum are much less dense. This result indicates that DASH and Dogecoin have a more resilient structure than other networks.

4.3 Degree Assortativity

In general, a network shows degree correlations if the number of links between the high- and low-degree nodes is systematically different from what is expected



(a) Bitcoin (b) Bitcoin Cash (c) Dash (d) Colorbar

(e) Dogecoin (f) Ethereum (g) Litecoin (h) ZCash

Fig. 1. Out-degree complementary cumulative distribution function of collected graphs. Snapshots are colored according to the colorbar.

by chance. In some types of networks, high-degree nodes (or hubs) tend to link to other such hubs, while in other types, hubs tend to link to low-degree nodes, i.e., what is known as a hub-and-spoke pattern. Assortativity, or assortative mixing, is a preference for nodes in a network to attach to others that are similar in some property; usually a node's degree.

The assortativity coefficient, ρ, is the Pearson's correlation coefficient of degree between pairs of linked nodes and lies in the range $-1 \leq \rho \leq 1$. A network is said to be assortative (ρ tends to 1) when the high-degree nodes tend to link to each other and avoid linking to the low-degree nodes, while the low-degree nodes tend to connect to other low-degree nodes. A network is said to be disassortative (ρ tends to -1) when the opposite happens. A random network has ρ close to zero and can be characterized as neutral. Incorporating this feature into network models improves the accuracy of the model in simulating the behavior of real-world networks. Disassortative networks tend to exhibit greater vulnerability to targeted attacks [36,43,51].

Correlations between nodes of similar degree are common in various observable networks. Social networks tend to exhibit assortative mixing, while technological and biological networks often show disassortative mixing, with high-degree nodes connecting to low-degree nodes. In disassortative networks, low-degree nodes, particularly those that have recently joined the network, can be discovered more quickly when connected to hubs. Removing these hubs can impact node discovery, graph connectivity, and potentially facilitate attacks such as eclipsing. Adversaries with high connectivity can exploit this knowledge to

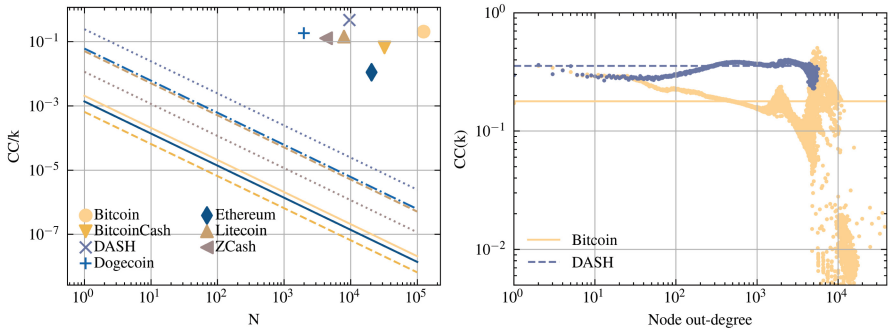
advertise malicious peer addresses, compromising the `ADDRMAN` of benign peers. We compute the assortativity coefficient for each snapshot, reporting the average values in Table 2. The networks analyzed exhibit negative assortativity, with DASH, Dogecoin, and Litecoin being closer to neutral (assortativity close to 0). Conversely, Bitcoin Cash, Zcash, and Bitcoin display more pronounced disassortativity. The negative assortativity indicates a hub-and-spoke structure in these networks, suggesting the presence of central peers that are crucial to the network and susceptible to targeted DDoS attacks.

4.4 Clustering Coefficient

The global clustering coefficient C is based on the number of triplets of nodes in the graph and provides an indication of how well the nodes tend to cluster together. A triplet is defined as three nodes connected by two edges. A triangle is a closed triple, i.e., three nodes connected by three edges. The global clustering coefficient is the number of closed triplets (or 3 x triangles) over the total number of triplets (both open and closed). A higher clustering coefficient indicates the presence of redundant pathways between nodes (due to the higher number of triangles), increasing the overall robustness of the network. The global clustering values are presented in Table 2. We observe that larger networks, tend to have lower clustering than smaller networks with Ethereum having the lowest value. This indicates that larger networks exhibit less robust characteristics. We suspect that this is closely related with the presence of unreachable peers, which is addressed in a following paragraph.

Unlike global clustering, the local clustering coefficient CC_i measures the density of links in the immediate neighborhood of node i: $CC_i = 0$ means that there are no links between i's neighbors, while $CC_i = 1$ implies that each of i neighbors also links to each other. In a random network, the local CC is independent of the node's degree, and average CC, i.e., $< CC >$, depends on the size of the system with respect to the number of nodes, N. On the contrary, measurements indicate that for real networks, e.g., the Internet, the Web, science collaboration networks, CC decreases with the degree of the node and is largely independent of the size of the system [6]. The local CC in a random network (CC_{rand}) is calculated as the average degree $< k >$ over N, i.e., $CC_{rand} = \frac{<k>}{N}$. The average degree of a network is $\frac{2L}{N}$, where L is the number of links. The average CC of a real network is expected to be much higher than that of a random graph.

In Fig. 2(a), we compare the average CC of the collected graphs with the expected CC for random networks of similar size. As in other real networks, we observe a higher CC than expected for a random network, indicating that the synthesized graphs deviate significantly from random networks. In Fig. 2(b), we plot the dependence of CC on the degree of the node for two of the networks studied, where we make some remarkable observations. Although the empirical rule of Barabasi [6] states that higher-degree nodes have lower CC, in Bitcoin we observe a significant fraction of high-degree nodes with high CC. The same finding was observed in the Ethereum and Zcash graphs. Another deviation



(a) All Networks (b) Bitcoin and Dash

Fig. 2. Analysis of Clustering Coefficient (CC) results. (a) $\frac{<CC>}{<k>}$ vs. network size; Size and CC averaged across snapshots $S_c^t \forall t \in T$. Markers correspond to the networks of Table 2. Lines correspond to the prediction for random networks, $CC = \frac{<k>}{N}$, with constant $< k >$ and varying size N. Similar to other known networks, the average CC appears to be independent of the network size N. (b) The dependence of the local CC on the node's degree for each network. $CC(k)$ is measured by averaging the local CC of all nodes with the same degree k (showing results of aggregating all snapshots of a given network). Horizontal lines correspond to the average CC of the network.

from the same empirical rule is observed in Dash, where all nodes appear to have an almost constant CC, independent of the node degree. We attribute this behavior to its temporal characteristics, previously discussed in the results related to Fig. 1. Further inspection reveals that Dash has very low churn and that most nodes are always online. The observed CC distributions indicate that the collected graphs are governed by rules that are rarely encountered in other known network systems. Note that the actual networks represented by these synthesized graphs are likely to have lower CCs, since we would expect fewer edges (see also Table 1).

As explained in Sect. 3, synthesized graphs are constructed by node advertisements. From Fig. 2 we can say that almost all nodes in the Dash network know and advertise almost all other peers. This is not surprising given the size of the network and the strongly connected component being very high. In contrast, the Bitcoin network exhibits variations in the clustering coefficient, indicating that not all nodes know and advertise all other peers. This is partly explained by the size of the network and the high presence of unreachable peers (see also Sect. 4.8). The temporal dynamics of the network could also affect peer announcements.

4.5 Average Betweenness Centrality

Average betweenness centrality measures how many short paths between vertices in the network pass through a given vertex. The betweenness centrality of a node v is given by the expression: $g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$ where σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(v)$ is the number of those

paths that pass through v. Nodes with high betweenness centrality act as bridges between parts of the network and therefore have a great control in the connectivity and information propagation of the network. It has been demonstrated that attacking or removing highly central nodes is one of the most effective strategies to partition a network or diminish its largest connected component.

The average node betweenness is the sum of node betweenness centrality for all nodes in a graph. Betweenness centrality of a node v is the sum of the fraction of the shortest paths of all pairs that pass through v [28]. A smaller average betweenness indicates that shortest paths are more evenly distributed among nodes; thus, it implies greater robustness. Nodes with high betweenness centrality tend to play a prominent role in networks, as they act as a bridge between groups of other nodes. Nodes with fewer connections than others may still have high betweenness, allowing them to fulfill a broker role and facilitate communication and information flow throughout the network. In effect, high average betweenness implies that network connectivity relies on a few central nodes, and such networks are more susceptible to targeted attacks. High variance in the betweenness centrality distribution is also an indication of lower robustness, as observed in [1,63]. Bitcoin and BitcoinCash have very high values of average betweenness, which further suggests that these networks are less resilient.

4.6 Scale-Free Property

One network property, tightly related with the degree distribution of a network, is the scale-free property. A scale-free network is defined as a network whose degree distribution follows a power law, i.e., having a probability distribution $p(k) \propto k^{-\alpha}$. The exponent α is known as the scaling parameter and typically lies in the range $2 < \alpha < 3$. The scale-free property strongly correlates with the network's robustness to random failures and has received tremendous attention in the scientific literature (e.g., see [6]). Many real-world networks have been reported to be scale-free, although their prevalence is questioned [14]. To test how well the degree distribution of each network snapshot can be modeled by a *power-law* (PL), *log-normal* (LN), *power-law with exponential cutoff* ($PLEC$) or *stretched exponential* (SE), we calculate the best fit using the *powerlaw* package available by Alstott *et al.* [3].

In Table 3, we report the number of times each type of distribution was the best fit, for all snapshots of the same network The calculated results indicate the dynamic nature of blockchain networks. Such networks that change over time may fit different distributions depending on the snapshot collected, something that is also visible in Fig. 1. These results suggest that blockchain overlays are not structured in the same way. However, in general, the degree distributions of the collected graphs belong to the exponential family of distributions. According to sources [17,18,25] Bitcoin's network formation procedure is intended to induce a random graph. Previous research [18,45] showed that the Bitcoin network does not resemble a random graph. Our results indicate that the synthesized graphs are also substantially different from random networks.

Table 3. Degree distributions of graphs best-fit for different types of exponential distributions. PL:power-law; LN: log-normal; $PLEC$: power-law with exponential cutoff; SE: stretched exponential.

Disrtibution	Bitcoin	Bitcoin Cash	Dash	Dogecoin	Ethereum	Litecoin	Zcash
LN	6.29%	76.90%	–	49.40%	21.90%	40.10%	0.60%
PL	0.60%	16.20%	1.80%	4.80%	24.60%	12.60%	18.90%
PLEC	93.11%	6.90%	57.20%	–	18.30%	46.40%	–
SE	–	–	41%	45.80%	35.30%	0.90%	80.50%

4.7 Small-World Property

The small-world phenomenon states that if you choose any two individual nodes in a small-world graph, the distance between them will be relatively short and definitely orders of magnitude smaller than the size of the network. We examined all collected snapshots to see if they satisfy the small-world property, by calculating the ω metric proposed in [59]. The metric is defined as $\omega = \frac{L_r}{L} - \frac{C}{C_l}$ where L and C are the average shortest path and the average clustering coefficient of the snapshot, respectively. L_r is the average shortest path for an equivalent random network, and C_l is the average clustering coefficient of an equivalent lattice network. The value of ω ranges between -1, when the network has lattice characteristics, to $+1$ when the network has random graph characteristics, with values near 0 interpreted as evidence of small worldliness. The average shortest path of a random network, L_r, is given by $\frac{ln(n)}{ln(k)}$ [6]. The Clustering Coefficient of the lattice, C_l is calculated as $\frac{3}{4}\frac{k-1}{k-2}$ [47]. The parameter k is the average degree.

We did not find evidence that the networks under study satisfy this property. Although we observe low average distances in all graphs, they do not have high enough clustering coefficients to be considered as small-world. Indicatively, the ω values we calculated are greater than 0.5 for Dash and Zcash. The rest of the networks have values greater than 0.8. According to Table 1 we would expect the real networks to exhibit lower clustering coefficients but similar average shortest path length, therefore driving ω even higher. Thus, we do not expect that the real BC networks would satisfy the small-world property.

4.8 Presence of Unreachable Nodes

It is well known that the vast majority of nodes on the Bitcoin overlay network are unreachable [33,61]. Our collected data verify this and also suggest that unreachable peers are present in all blockchain overlays. In Table 4 we list our findings. The in-degree indicates how many reachable peers advertise an unreachable address. Notably, a high percentage of unreachable nodes appears in all networks, leading to the observation that blockchain networks have a strongly connected core and a high number of unreachable nodes that lie on the fringe of the network. DASH stands out for having much less unreachable peers.

Table 4. Presence and median in-degree of unreachable peers in each overlay.

Network	% of unreachable nodes	Median in-degree
Ethereum	98%	4
BitcoinCash	96%	3
Bitcoin	88%	3
Litecoin	86%	**75**
ZCash	84%	4
Dogecoin	73%	**68**
DASH	18%	**984**

Unreachable nodes were previously known to exist in the Bitcoin and Ethereum networks. Our results indicate that they are also present in all blockchain networks, although at different percentages. The existence of unreachable peers is long known, but this class of peers has received little attention from the research community. It has been demonstrated that they play an important role in blockchain systems [61].

The presence of unreachable peers, which can affect the properties of a network, is not related to the network protocols used. Their presence is more likely influenced by socioeconomic factors such as the popularity of a cryptocurrency, its value, and the availability of compatible wallet software. Many blockchain clients, such as cryptocurrency wallets, appear as unreachable peers in a network, and the number of these peers depends on the factors mentioned above. However, we observe that networks with a high percentage of unreachable nodes exhibit rather less robust properties (see Table 2) such as high average betweenness, lower density, and lower clustering.

5 Discussion

In this study, we analyze the structure of seven blockchain networks and evaluate their resilience based on the computed graph properties. Our results are summarized below:

- Major blockchain networks have characteristics that indicate towards a less resilient structure. In particular, Bitcoin, BitcoinCash, and Ethereum display lower density and higher average betweenness than other networks, suggesting increased vulnerability to targeted attacks.
- Among the networks studied, BitcoinCash appears to be the most vulnerable, demonstrating lower density, a dissassortative nature, and high average betweenness.
- Despite utilizing similar protocols (excluding Ethereum), the networks exhibit distinct structural properties and resilience traits. Possible explanations for these differences include variations in network size, temporal characteristics, and the presence of unreachable peers.

– The networks' degree distribution per snapshot demonstrates significant variation. While some snapshots align with power-law distributions, others exhibit better fits with log-normal, power-law with exponential cut-off, or stretched exponential distributions.
– Their clustering coefficient distributions are similar to other real networks, and differ from random networks with similar size and average degree. They have low diameters and short average shortest path lengths, but we did not observe evidence of satisfying the small-world property.

It is important to note that our results are derived from connectivity graphs constructed using P2P address propagation, rather than representing the real topology of the networks. As a result, the networks studied may not accurately reflect precise network properties. Table 1 illustrates how these results can establish limits for the properties of real networks. Our simulations in Sect. 3 indicate that real networks are likely to exhibit lower clustering and higher betweenness, rendering them less resilient than our observations suggest.

6 Conclusions

To conclude, we have presented a comprehensive examination and analysis of the structural properties of seven distinct blockchains, focusing on their resilience. By leveraging selected graph metrics, we extract valuable insights into the resilience properties of these overlay networks. To achieve this, we employ custom crawlers to probe 32 million blockchain peers, capturing each node's list of known peers and extracting their potential connections. Our dataset is made available for future research purposes.

Through graph analysis, we have discovered that blockchain networks exhibit a distinct structure compared to traditional networks such as the Web. Surprisingly, we have observed significant variations in the graph characteristics among the studied blockchain networks, despite their similar protocols. Our findings highlight a concerning vulnerability in major blockchains: they heavily rely on a limited number of central nodes for connectivity, making them susceptible to targeted denial-of-service (DoS) attacks. While blockchains are renowned for their decentralized nature, it is crucial to acknowledge that vulnerabilities at the network layer can introduce significant risks. These vulnerabilities may lead to network partitioning, leaving the blockchain exposed to various attacks, including user deanonymization, node eclipsing, consensus breaches, and double spending [27].

Acknowledgements. This project has received funding from the European Union's Horizon 2020 Research and Innovation program under the Marie Skłodowska-Curie INCOGNITO project (Grant Agreement No. 824015), CONCORDIA project (Grant Agreement No. 830927), SPATIAL project (Grant Agreement No. 101021808) and the Cyprus's Research and Innovation Foundation (Grant Agreement: COMPLEMENTARY/0916/0031). The authors bear the sole responsibility for the content presented in this paper, and any interpretations or conclusions drawn from it do not reflect the official position of the European Union nor the Research Innovation Foundation.

References

1. Alenazi, M.J.F., Sterbenz, J.P.G.: Comprehensive comparison and accuracy of graph metrics in predicting network resilience. In: 2015 11th International Conference on the Design of Reliable Communication Networks (DRCN), pp. 157–164 (2015)
2. Alenazi, M.J.F., Sterbenz, J.P.G.: Evaluation and comparison of several graph robustness metrics to improve network resilience. In: 2015 7th International Workshop on Reliable Networks Design and Modeling (RNDM), pp. 7–13 (2015)
3. Alstott, J., Bullmore, E., Plenz, D.: powerlaw: a python package for analysis of heavy-tailed distributions. PLoS ONE **9**(1) (2014). https://doi.org/10.1371/journal.pone.0085777
4. Atish Kulkarni, M.: Leeuwen: graph pattern mining for blockchain networks (2021)
5. Azaria, A., Ekblaw, A., Vieira, T., Lippman, A.: Medrec: using blockchain for medical data access and permission management. In: OBD. IEEE Computer Society (2016)
6. Barabási, A.L., et al.: Network Science. Cambridge University Press, Cambridge (2016)
7. Biryukov, A., Tikhomirov, S.: Deanonymization and linkability of cryptocurrency transactions based on network analysis. In: IEEE European Symposium on Security and Privacy (EuroS&P) (2019). https://doi.org/10.1109/EuroSP.2019.00022
8. Biryukov, A., Khovratovich, D., Pustogarov, I.: Deanonymisation of clients in bitcoin P2P network. In: CCS, ACM (2014)
9. Booker, L.B.: The effects of observation errors on the attack vulnerability of complex networks: technical report, defense technical information center, Fort Belvoir, VA, November 2012. https://doi.org/10.21236/ADA576235, http://www.dtic.mil/docs/citations/ADA576235
10. Broder, A.Z., et al.: Graph structure in the web. Comput. Networks **33**(1–6), 309–320 (2000)
11. Casale-Brunet, S., Ribeca, P., Doyle, P., Mattavelli, M.: Networks of ethereum non-fungible tokens: a graph-based analysis of the ERC-721 ecosystem. In: 2021 IEEE International Conference on Blockchain (Blockchain), IEEE, December 2021. https://doi.org/10.1109/Blockchain53845.2021.00033
12. Chen, W., Xu, Z., Shi, S., Zhao, Y., Zhao, J.: A survey of blockchain applications in different domains. In: ICBTA, ACM (2018)
13. Christidis, K., Devetsikiotis, M.: Blockchains and smart contracts for the internet of things. IEEE Access **4**, 2292–2303 (2016)
14. Clauset, A., Shalizi, C.R., Newman, M.E.J.: Power-law distributions in empirical data. SIAM Rev. **51**(4), 661–703 (2009)
15. CoinMarketCap: Coinmarketcap (2021). https://coinmarketcap.com
16. Daniel, E., Rohrer, E., Tschorsch, F.: Map-z: exposing the zcash network in times of transition. In: LCN, IEEE (2019)
17. Decker, C., Wattenhofer, R.: Information propagation in the bitcoin network. In: 13th IEEE International Conference on Peer-to-Peer Computing, IEEE P2P 2013, IEEE (2013). https://doi.org/10.1109/P2P.2013.6688704
18. Delgado-Segura, S., et al.: TxProbe: discovering bitcoin's network topology using orphan transactions. In: Goldberg, I., Moore, T. (eds.) FC 2019. LNCS, vol. 11598, pp. 550–566. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32101-7_32

19. Delgado-Segura, S., Pérez-Solà, C., Herrera-Joancomartí, J., Navarro-Arribas, G., Borrell, J.: Cryptocurrency networks: A new P2P paradigm. Mob. Inf. Syst. **2018**, 2159082:1–2159082:16 (2018)
20. Deshpande, V., Badis, H., George, L.: Btcmap: mapping bitcoin peer-to-peer network topology. In: 2018 IFIP/IEEE International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN), IEEE (2018)
21. Developers, B.C.: Replace global trickle node with random delays (2015). https://github.com/bitcoin/bitcoin/pull/7125
22. Developers, B.C.: Ignore getaddr msg from inbound connections (2020). https://github.com/bitcoin/bitcoin/blob/37e9f07996d3a7504ea54180d188ca91fdf0c884/src/net_processing.cpp#L3567
23. Developers, B.C.: Bitcoin core integration/staging tree (2021). https://github.com/bitcoin/bitcoin
24. Developers, B.C.: Bitcoin p2p network (2021). https://developer.bitcoin.org/devguide/p2p_network.html
25. Dotan, M., Pignolet, Y.A., Schmid, S., Tochner, S., Zohar, A.: Sok: cryptocurrency networking context, state-of-the-art, challenges. In: Proceedings of the 15th International Conference on Availability, Reliability and Security, ARES 2020, ACM (2020). https://doi.org/10.1145/3407023.3407043
26. Ethereum: Ethereum peer-to-peer networking specifications (2014). https://github.com/ethereum/devp2p
27. Franzoni, F., Daza, V.: Sok: network-level attacks on the bitcoin p2p network. IEEE Access **10**, 94924–94962 (2022). https://doi.org/10.1109/ACCESS.2022.3204387
28. Freeman, L.C.: A set of measures of centrality based on betweenness. Sociometry **40**(1), 35–41 (1977). http://www.jstor.org/stable/3033543
29. Freitas, S., Yang, D., Kumar, S., Tong, H., Chau, D.H.: Graph vulnerability and robustness: a survey. IEEE Trans. Knowl. Data Eng. 1 (2022). https://doi.org/10.1109/TKDE.2022.3163672
30. Gervais, A., Karame, G.O., Wüst, K., Glykantzis, V., Ritzdorf, H., Capkun, S.: On the security and performance of proof of work blockchains. In: CCS, pp. 3–16. ACM (2016)
31. Grundmann, M., Amberg, H., Hartenstein, H.: On the estimation of the number of unreachable peers in the bitcoin P2P network by observation of peer announcements. CoRR abs/2102.12774 (2021)
32. Grundmann, M., Baumstark, M., Hartenstein, H.: On the peer degree distribution of the bitcoin p2p network. In: 2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), pp. 1–5 (2022)
33. Grundmann, M., Baumstark, M., Hartenstein, H.: On the peer degree distribution of the bitcoin P2P network. In: ICBC, pp. 1–5. IEEE (2022)
34. Grundmann, M., Neudecker, T., Hartenstein, H.: Exploiting transaction accumulation and double spends for topology inference in bitcoin. In: Zohar, A., et al. (eds.) FC 2018. LNCS, vol. 10958, pp. 113–126. Springer, Heidelberg (2019). https://doi.org/10.1007/978-3-662-58820-8_9
35. Hagberg, A.A., Schult, D.A., Swart, P.J.: Exploring network structure, dynamics, and function using networkx. In: Varoquaux, G., Vaught, T., Millman, J. (eds.) Proceedings of the 7th Python in Science Conference (2008)
36. Iyer, S., Killingback, T., Sundaram, B., Wang, Z.: Attack robustness and centrality of complex networks. PLoS ONE **8**, e59613 (2013)

37. D Khan, A.: Graph analysis of the ethereum blockchain data: a survey of datasets, methods, and future work. In: 2022 IEEE International Conference on Blockchain (Blockchain), IEEE, August 2022. https://doi.org/10.1109/Blockchain55522.2022.00042
38. Kim, S.K., Ma, Z., Murali, S., Mason, J., Miller, A., Bailey, M.: Measuring Ethereum network peers. In: IMC, ACM (2018)
39. Lee, S., Kim, H.: On the robustness of lightning network in bitcoin. Pervasive Mob. Comput. **61**, 101108 (2020)
40. Leskovec, J., Sosič, R.: Snap: a general-purpose network analysis and graph-mining library. ACM Trans. Intell. Syst. Technol. (TIST) **8**(1), 1–20 (2016)
41. Li, Y., Islambekov, U., Akcora, C., Smirnova, E., Gel, Y.R., Kantarcioglu, M.: Dissecting Ethereum blockchain analytics: what we learn from topology and geometry of the ethereum graph?, pp. 523–531. Society for Industrial and Applied Mathematics, January 2020. https://doi.org/10.1137/1.9781611976236.59
42. Magoni, D.: Tearing down the internet. IEEE J. Sel. Areas Commun. **21**(6), 949–960 (2003)
43. Mahadevan, P., Krioukov, D., Fomenkov, M., Dimitropoulos, X., Claffy, K.C., Vahdat, A.: The internet as-level topology: three data sources and one definitive metric. SIGCOMM Comput. Commun. Rev. **36**(1), 17–26 (2006). https://doi.org/10.1145/1111322.1111328
44. Maymounkov, P., Mazières, D.: Kademlia: a peer-to-peer information system based on the XOR metric. In: Druschel, P., Kaashoek, F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 53–65. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45748-8_5
45. Miller, A., Litton, J., Pachulski, A., Gupta, N., Levin, D., Spring, N., Bhattacharjee, B.: Discovering bitcoin's network topology and influential nodes. University of Maryland, Technical Report (2015)
46. Mislove, A., Marcon, M., Gummadi, K.P., Druschel, P., Bhattacharjee, B.: Measurement and analysis of online social networks. In: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement. IMC 2007, Association for Computing Machinery (2007). https://doi.org/10.1145/1298306.1298311
47. Montana, C.H.S., Huerta-Quintanilla, R.: Generalization of clustering coefficient on lattice networks applied to criminal networks. Int. J. Math. Comput. Sci. **4** (2017)
48. Neudecker, T.: Characterization of the bitcoin peer-to-peer network (2015–2018). Technical Report, 1, Karlsruher Institut für Technologie (KIT) (2019). https://doi.org/10.5445/IR/1000091933
49. Neudecker, T., Andelfinger, P., Hartenstein, H.: Timing analysis for inferring the topology of the bitcoin peer-to-peer network. In: UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld. IEEE Computer Society (2016)
50. Neudecker, T., Hartenstein, H.: Network layer aspects of permissionless blockchains. IEEE Commun. Surv. Tutorials **21**(1) (2019). https://doi.org/10.1109/COMST.2018.2852480
51. Newman, M.E.J.: Mixing patterns in networks. Phys. Rev. E, Stat. Nonlinear Soft Matter Phys. **67**2 Pt 2, 026126 (2002)
52. Nick, J.: Guessing bitcoin's p2p connections (2015). https://jonasnick.github.io/blog/2015/03/06/guessing-bitcoins-p2p-connections/
53. Ozisik, A.P., Andresen, G., Levine, B.N., Tapp, D., Bissias, G., Katkuri, S.: Graphene. In: Proceedings of the ACM Special Interest Group on Data Communication. ACM, August 19 2019. https://doi.org/10.1145/3341302.3342082

54. Paphitis, A., Kourtellis, N., Sirivianos, M.: A first look into the structural properties of blockchain P2P overlays. https://doi.org/10.6084/m9.figshare.23522919
55. Paphitis, A., Kourtellis, N., Sirivianos, M.: Resilience of blockchain overlay networks. In: LNCS. Lecture Notes in Computer Science, vol. 17th International Conference on Network and System Security (NSS 2023). Springer, Cham (2023)
56. Rueda, D.F., Calle, E., Marzo, J.L.: Robustness comparison of 15 real telecommunication networks: structural and centrality measurements. J. Netw. Syst. Manage. **25**(2), 269–289 (2016). https://doi.org/10.1007/s10922-016-9391-y
57. Saad, M., Chen, S., Mohaisen, D.: Syncattack: Double-spending in bitcoin without mining power. In: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, CCS 2021, pp. 1668–1685. ACM, New York, NY, USA (2021). https://doi.org/10.1145/3460120.3484568
58. Siganos, G., Tauro, S.L., Faloutsos, M.: Jellyfish: a conceptual model for the as internet topology. J. Commun. Networks **8**(3), 339–350 (2006)
59. Telesford, Q.K., Joyce, K.E., Hayasaka, S., Burdette, J.H., Laurienti, P.J.: The ubiquity of small-world networks. Brain Connectivity **1**(5) (2011)
60. Wang, D.J., Shi, X., McFarland, D.A., Leskovec, J.: Measurement error in network data: a re-classification. Soc. Networks **34**, 396–409 (2012)
61. Wang, L., Pustogarov, I.: Towards better understanding of bitcoin unreachable peers. CoRR abs/1709.06837 (2017)
62. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Structural Analysis in the Social Sciences, Cambridge University Press, Cambridge (1994). https://doi.org/10.1017/CBO9780511815478
63. Xia, Y., Fan, J., Hill, D.: Cascading failure in watts-strogatz small-world networks. Phys. A Stat. Mech. Appl. **389**(6), 1281–1285 (2010). https://doi.org/10.1016/j.physa.2009.11.037
64. Yap, T.T.V., Ho, T.F., Ng, H., Goh, V.T.: Exploratory graph analysis of the network data of the Ethereum blockchain. F1000Research **10**, 908 (2021). http://dx.doi.org/10.12688/f1000research.73141.1
65. Zhao, C., Guan, Y.: A graph-based investigation of bitcoin transactions. In: Peterson, G., Shenoi, S. (eds.) DigitalForensics 2015. IAICT, vol. 462, pp. 79–95. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24123-4_5
66. Zyskind, G., Nathan, O., Pentland, A.S.: Decentralizing privacy: using blockchain to protect personal data. In: 2015 IEEE Security and Privacy Workshops (2015)