Topological Analyses of Unstructured Peer-to-Peer Systems

Hend Alrasheed

King Saud University, Riyadh, Saudi Arabia

Summary
Understanding the structural properties of complex peer-to-peer (P2P) networks is essential for their future improvement. It was shown that the performance of a P2P protocol is directly related to its underlying topology. Therefore, multiple efforts have analyzed the topological properties of several P2P systems, including their degree distribution and clustering coefficient. Some obtained results contradict while others coincide. In this work, we closely analyze the topologies of three unstructured P2P systems: BitTorrent, Gnutella, and Freenet and we shed some light on their implications[1].

Key words:
Peer-to-peer network topologies, graph structure, unstructured overlay, network diameter, node degree.

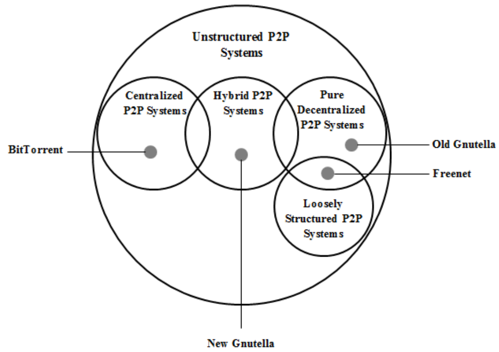

Fig. 1 A taxonomy of multiple P2P systems.

1. Introduction

A P2P network can be represented as an undirected graph $G = (V, E)$, where V is the number of nodes (peers) and E is the number of edges (connections). This representation has become very popular to understand the different phenomena that those graphs exhibit. From a graph theory point of view, a number of network properties can be identified and measured. Each property plays an important role in improving the performance of the overall system. It was shown that the performance of a P2P protocol is directly related to its underlying topology [1, 5]. For example, the small-world property contributes to better data dissemination; therefore, several studies have proposed methods to enhance the performance of the BitTorrent by enforcing its topology to exhibit such property [3]. Furthermore, research is currently investigating how to reduce the clustering in Gnutella to improve its functionality [13]. P2P systems avoid the dependency on a central entity (a server) such that all nodes collaborate to achieve network goals such as data distribution. However, not all systems are purely decentralized. P2P can be centralized (BitTorrent), pure decentralized (Freenet), or hybrid (new Gnutella) according to the degree to which the system depends on the existence of a central object. With respect to their topologies, P2P systems can be classified into two categories (Fig. 1): (1) Structured systems: where the overlay is well-controlled and the content is distributed in a way that enables efficient querying. (2) Unstructured systems: where the files are scattered at random locations, and the general overlay is random. This is mainly because of the lack of specific rules that guide where a new node should be added. Loosely-structured systems go under this category.

Although structured P2P systems can locate rare items more efficiently, unstructured systems have a number of advantages that make them more popular over the Internet [8]. First, they require less overhead compared to structured protocols. Moreover, they are easier to implement, require less maintenance, and more able to handle large populations of transient nodes. Since the overlay of an unstructured P2P network is unpredictable and constantly changing according to peer dynamics (joining, leaving, and failing), a decent amount of effort has been devoted to analyzing their graphs [11]. In this work, we present a detailed review of the topological analyses of three unstructured P2P networks (Table 1). We aim at highlighting the similarities and contradictions in their findings. The aim is to instigate further research to deeply understand those systems and explore the proper approaches to analyzing and evaluating them.

Manuscript received November 5, 2020
Manuscript revised November 20, 2020
https://doi.org/10.22937/IJCSNS.2020.20.11.13

Table 1: Summary of the P2P systems used in this work.

System	Purpose	Centralization	Topology
BitTorrent	File sharing	Centralized	Unstructured
Gnutella	File sharing	Hybrid	Unstructured
FreeNet	Content storage and retrieval	Purely decentralized	Loosely-structured

2. Network Topological Properties

Multiple structural properties can be used to measure and understand network structures. Network structural properties can be classified as global such as the network diameter and clustering coefficient. Other properties are local such as node degrees. Next, we introduce definitions for each network property.

Network diameter and characteristic path length. The diameter of a given network measures the length of a longest shortest path between any two peers (number of hops in unweighted networks). If the network is not connected, the value of the diameter will be undefined since some parts of the network are not reachable. In this case, the diameter is measured for every connected component independently. Generally, smaller diameter values are preferred to support better content distribution, but the shape of the overlay must be taken into consideration. For example, a star-shaped network, where all nodes are connected to a single central node, has a diameter of two. However, the central node represents a bottleneck for traffic exchange [5]. The characteristic path length (CPL) measures the average distance between two nodes. It gives a sense of how far the peers are from each other [7]. Generally, the average path length is used as a performance measure of any routing system.

Clustering coefficient. The clustering coefficient measures the tendency of the nodes to form clusters. The clustering coefficient of a node u is the proportion of the number of edges between all u's neighbor nodes to the number of the largest possible edges that can exist between all u's neighbor nodes. The network clustering coefficient C, where $0 \leq C \leq 1$, is the average of the clustering coefficients of all of its nodes. The closer the value to one indicates that the network is showing higher clustering behavior among the peers. Higher clustering is important for efficient look-ups in unstructured networks since it implies better reachability and smaller CPL. However, clustering may adversely affect the robustness of the network, especially when few links exist between different clusters.

Manuscript received August 2005.
Manuscript revised August 22, 2005.

Network connectivity. Network connectivity indicates if all nodes are reachable from one another. That is, starting at any peer, is it possible to reach every other peer in the network through a sequence of intermediate peers? The connectivity of the network is critical to its content distribution and robustness.

Network topology. The network topology shows how peers are connected. To some extent, the topology is highly affected by how new nodes are added and linked to the rest of the network. Generally, unstructured P2P networks have no specific rules. However, since the topology highly impacts the network performance, a general classification of the topology is crucial. The goal is to decide if the topology admits one or more of the following known graphs.

(1) Random. Generally, random graphs exhibit low CPLs and clustering coefficients. The average path length of large random graphs is about $\log n/\log k$, where n is the number of vertices and k is the average number of edges per vertex. Their clustering coefficient is k/n [7]. Two methods can be used to decide if a network represents a random graph. First, to plot the peer connectivity matrix (Fig. 2) and decide if the network admits a special organization. Second, to construct an equivalent random graph and calculate the clustering coefficients of both networks. The network represents a random graph if the two values are close.

(2) Regular: in a regular graph, all peers have almost equal number of neighbors. Under this topology, each neighbor has some closer neighbors and some farther ones. This increases clustering which is important for connectivity in most networks. The average path length is about $n/2k$ (much larger compared to random graphs).

(3) Small-world: a network has this property when the network CPL is relatively small. This implies that any two non-neighbor nodes can reach one another with few hops, which indicates better reachability and content distribution. Generally, small-world graphs have high clustering coefficients. To examine if a network is small-world, it is compared to an equivalent (with respect to the number of vertices and edges) random graph. The network has the small-world property if (a) Its CPL is close to the CPL of the equivalent random graph and (b) Its clustering

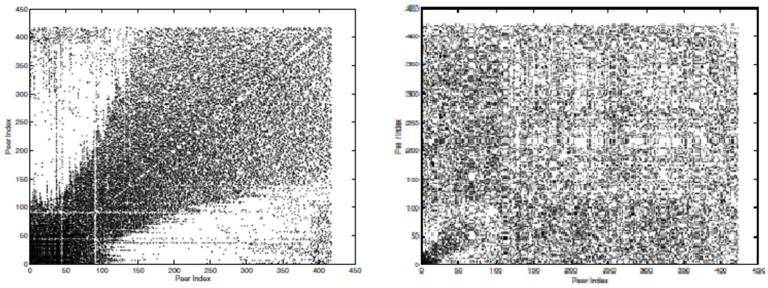

Fig. 2 The connectivity matrix of the swarm [3]. Both axes have the number of nodes (1-450), and each point (*x, y*) represents a connection between a pair of peers. (a) The connectivity matrix at hour 4. (b) The connectivity matrix at hour 32.

coefficient is larger than the clustering coefficient of the equivalent random graph.

(4) Scale-free: a network is scale-free if its degree distribution follows a power-law degree distribution. In power-law distributed network, there are several nodes with many connections compared to the majority of nodes. Having such hubs in the overlay increases the robustness of the network against random node failures [3, 14].

3. Topological Analysis of Unstructured Peer-to-peer Systems

For each of the three P2P systems (BitTorrent, Gnutella, and Freenet), we briefly describe its main features. Then we discuss the related overlay analyses.

3.1 BitTorrent

The BitTorrent is a centralized P2P system that uses a central entity (tracker) to maintain a list of active peers who are all interested in exchanging the same content. The group of peers is called a swarm. The BitTorrent differentiates between two types of peers: seeder and leecher. A seeder is a peer that has a complete copy of the content being exchanged; whereas, a leecher is a peer that is still downloading the file.

As peers interact through the BitTorrent protocol, four types of networks can be recognized: the interest network, the unchoked network, the download network, and the connectivity network [3]. Here we are interested in the connectivity network of the swarm, which is constructed as follows. Before a new node joins the swarm, it contacts the tracker which responds by providing the peer with a set of addresses of currently active peers in the swarm. Then the peer randomly selects a subset of addresses to build its neighborhood and starts the process of exchanging the content. The size of the neighborhood is fixed to a specific value by the client. The swarm's life cycle comprises two stages. First, the initial stage when the swarm first started and the number of seeders is much smaller than the number of leechers. During this stage, the number of joining peers increases rapidly. Second, the steady stage that comes after the initial stage when most peers have joined the system.

Al-Hamra et al. [5] use a simulator to capture various stages of the swarm evolution as peers join and leave to study the generated overlay. They conduct a series of experiments to investigate how the efficiency of the swarm is affected by each of the following parameters: the peer's neighbourhood size, the convergence speed of the peer (to reach its maximum size), and the diameter of the overlay. They conclude that (1) A larger peer's neighborhood improves the download time of the piece, and the average size of the neighborhood is not affected by the overall size of the swarm. (2) The larger the size of the swarm, the longer it takes a new joining peer to reach its maximum neighborhood size. Moreover, peers tend to group into clusters according to their joining time (Fig. 2(a)). This suggests that the generated overlay is not random. The convergence speed decreases as the size of the swarm increases. (3) A small network diameter with proper structure (no bottleneck peers) is necessary for better data dissemination. The swarm's diameter increases slowly as the size of the network increases; however, it slowly decreases as the neighborhood size of peers increases. The authors also show how PEX (PEX or peer exchange is a protocol that allows peers to periodically exchange their peer lists) contributes in generating a chain-like overlay with a large diameter which may negatively impact content distribution. The network disconnects towards the end of the swarm's lifetime. As many peers complete downloading and depart, the existing peers lose many of their connections (no more joining peers to connect to). By default, a peer contacts the tracker in such a case, but if the minimum

Manuscript received August 2005.
Manuscript revised August 22, 2005.

number of peers before contacting the tracker *m* is set to a lower value, the peer stays with very few connections until this number drops to *m*. This results in a disconnected network.

In [3], Dale et al. analyze the topological aspects of the BitTorrent swarm through a modified BitTorrent client such that each peer logs its connections to other peers. They identify four networks that result from peer interactions: the interest network, the unchoked network, the download network, and the connectivity network. Unlike [5], they find the connectivity network to be random and the fan-out topology found in the connection matrix towards the beginning of the swarm's lifetime is only temporary (Fig. 2). Therefore, the behaviour of the swarm at the early stages is not an enough indication of the overall pattern followed by the network.

The two studies analyzed simulated BitTorrent swarms. Kryczka et al. [6] examined more than 250 live BitTorrent swarms through a measurement tool that uses the PEX protocol for collecting peer routing tables. They target the network's randomness, dynamicity, and the effect of the swarm size on the topology. They show that the swarm's network is not random nor a small world. Moreover, although the peer neighborhoods are changing rapidly, the overall overlay attributes such as the characteristic path length and the clustering coefficient remain unchanged.

They also conclude that the increase in the network size affects its topology making it deviate from a random graph, which contradicts the result in [5]. They also report that the clustering coefficient is high at the initial stage; then it gradually decreases until the swarm gets to a steady state. This clustering is especially apparent among leecher peers, but this clustering fades as leechers develop into seeders. Generally, the network clustering is always present (this is the opposite of what has been reported in [3]).

Su et al. [12] use snapshots captured from real swarms; as well as PEX messages to verify the topological measurements presented in previous works. Their results show that the degree distribution of the peers follows a Gaussian distribution rather than a power-law distribution. Moreover, similar to [5], they find the clustering coefficient of the network to be high during the initial stage and drops gradually until the network reaches its steady stage. On the other hand, they report that the distance between peer pairs to be increasing during the steady stage, and that the PEX extension does not increase the diameter in the initial stage (unlike [5]). They find the diameter in this stage to be small (6 hops). They agree with [3] and [6] that the topology represents a random graph and the distances between the majority of peer pairs (and the diameter) are small. However, since the network is disconnected, it does not represent a small-world network. They also conclude that the connection limitations imposed by the protocol affect more peers in the initial stage than in the steady stage. This is the reason most peers appear to have fewer connections.

3.2 Gnutella

Gnutella is a pure decentralized file sharing system. Every peer (known as servant) acts as a client and as a server. When a peer joins the network, it will be connected to the neighbor whose address was given during bootstrapping[2]. Then the new joining peer starts announcing its presence and exploring its neighborhood through the initial neighbor (or neighbors) via exchanging Ping/Pong messages. Ping/Pong messages have multiple types and purposes, but the one we are interested in are

- Ping: the request for other servants to acknowledge the presence of the new peer.
- Pong: the response to the Ping which contains the IP addresses and port numbers of the replying servants.

For a query, the flooding method is used as follows. A Query is submitted by a peer to its neighbors and then to the neighbors of the neighbors and so on until the query's TTL (Time to live) becomes zero or the peer runs out of servants to ask. The Query encloses a list of keywords, and a servant with a matching content will send a QueryHit. Afterwards, an HTTP connection will be established between the querying servant and one of the peers that sent the QueryHit packet to start downloading. High clustering is not preferable for Gnutella when TTL value is small because message flooding will be more than desired.

To scale with the rapid increase in the network size, a new Gnutella protocol has been developed. It uses a two-tier overlay which distinguishes between two types of peers (Fig. 3). (1) Ultrapeers which are peers with higher capabilities (processing power, bandwidth, etc.) This set of peers forms a well-connected top-level overlay and is considered the core plane of the Gnutella network. (2) Leaf peers which are connected to one or more ultrapeers. There are no constraints on selecting the parent peers; however, leaves are usually connected to a small number of ultrapeers due to bandwidth limitations. A high-bandwidth leaf peer may be promoted to an ultrapeer to maintain good a balance between the numbers of ultrapeers and leaf peers. This upgrade may happen in some special cases; for example, when no ultrapeer has open slots to accept the new leaf peer [9, 10]. Ultrapeers manage the querying for their leaves.

[2] Bootstrapping is a function that Gnutella peers perform to discover and connect to other on-line peers.

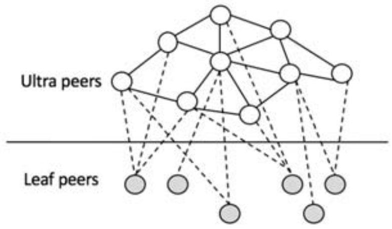

Fig. 3 The overlay topology of the two-tier Gnutella.

This addition added some structure to the overlay [10]. Stutzbach and Rejaie [10] studied the underlying topology of Gnutella. They develop a new crawler that uses handshake messages to quickly get the list of neighbors of each servant instead of exchanging Ping/Pong messages like older crawlers. Their crawler interacts with top-level peers since almost every other peer is connected to at least one top-level peer. This helps reducing the number of contacted peers while collecting the information. The crawler is able to capture a snapshot of 300-400K peers in less than four minutes. Their results show that different servants use different Gnutella clients. This is apparent by the various results collected for every parameter. For example, different results exist for the average node degree (Table 2). Generally, to study the average peer degree in Gnutella, the following need to be considered: the average number of ultrapeers an ultrapeer node (top-level) is connected to, the average number of leaves an ultrapeer node is connected to, and the average number of ultrapeers a leaf node is connected to (the results are summarized in Table 2). The network size growth does not seem to have an impact on the characteristic path length, which is found to be about 4 hops in 2004 compared to 4-5 hops in 2001 (small world network). This phenomenon is due to the logarithmic effect of population increase in randomly connected graphs. Additionally, as the population of the network increases, more and more leaf peers become ultrapeers. As a result, a decent number of ultrapeers cannot establish new connections. This leads to a sparser top-level network with a lower clustering coefficient.

In [9], Rasti et al. use the same crawler introduced in [10] to collect snapshots and analyze the growth of the new Gnutella network. First of all, they show how the population increase affects the degree distribution. They find the degree distribution among top-level peers to increase slightly and among ultrapeers with fewer than 20 connections, i.e., ultrapeers with open slots. Moreover, the size increase does not affect the degree distribution of ultrapeers to leaves. The increase was found to be in the number of ultrapeers with fewer leaves. This is a result of the growth of the number of ultrapeers. Moreover, the population increase increases the number of leaf nodes who have 1 to 3 parents; however, the degree distribution of those peers remains the same. Second, they find that the size increase decreases the clustering coefficient because as the number of ultrapeers grows, the top-level overlay becomes sparser. The reason as mentioned above is that the number of connections among ultrapeers remains the same despite the rise in the number of peers. Finally, similar to [10], they find the increase in the size of the network to not affect the characteristic path length among peer pairs. Also, the diameter of the network remains small.

In [13], Wang et al. studied the degree distribution and the small-world property of each of the two levels of the Gnutella network. They use a crawler named D-crawler. D-crawler is adaptable based on the feedback collected from previous snapshots. Each snapshot contains the following data about some known peers: the peer's version, the peer's type (ultrapeer or peer), a list of neighbors, and a list of leaves. Each snapshot takes a minute and includes 160K peers. They find the degree distribution of top-level peers to follow a double peeks Gaussian distribution. The two peaks are 32 and 25. The degree distribution of ultrapeers to leaves exhibits Gaussian distribution; whereas, the leaves to ultrapeers follows power-law distribution. Although leaves can connect to any number of ultrapeers, most leaves connect to only about three ultrapeers due to bandwidth limitations. Moreover, the clustering coefficient is found to increase with the size of the network, making the Gnutella overlay highly connected.

3.3 Freenet

Freenet is a decentralized anonymous information P2P protocol that is used to store and retrieve information. Each file is given a location-independent unique identifier. Each peer holds a set of <key, pointer> pairs, where the key is the hashed binary file name and the pointer points to a node that has a copy of the file denoted by the key. To increase the system efficiency, each peer has a data store where it keeps local copies of the files that rout through it. Joining nodes use a similar mechanism to the one used in Gnutella.

In loosely-structured P2P systems, the overlay is created based on hints or probability. Consequently, searches are done based on the hints used for the network construction. Freenet uses hints for network construction and searching. Peers use a query technique similar to the flooding query method used by the default Gnutella protocol: queries are forwarded from the query initiator node to the rest of the network through its neighbors. However, in Freenet, it is impossible (to some extent) to know if a peer is forwarding the query or if it is the peer who initiated the query.

Study	**No. of networks**	**No. of nodes**	**Duration**	**Topology**	**Peer deg distrib**	***R***	**Avg peer deg**	**CPL**	**Diame**	**Conn**	**Clust Coef**
						BitTorrent					
[5]	1	1867	70 min.	Not random			65[a]		2-4	Disconn	
[3]	1	> 400	100 hours.	Random Not small-world			65[a]	I: 1-2 S: 2			0.1
[6]	250		15 days.	Random Not small-world				I: 1.6 S: 1.8 -1.9			I: 0.6 S: 0.1
[12]	106	> 300 in each	45 days.	Random	Gaussian	0.5 - 0.97	≈ 50[b]	I: < 3 S: < 4	I: < 6 S: < 10	I: Conn S: Disconn	< 0.15
						Gnutella					
[10]	1	800000 (some not reachable)	6 months	Small-world	Not power-law		U to U: 30 U to L: 30, 45, 75. L to U: < 3	4	9	Conn	0.012-0.014
[9]	1		15 months	Random			U to U: 20-30 U to L: 30, 45 L to U: 1-3	4-7	8	Conn	0.011-0.02
[13]	1	1,738,773	8 months	Small-world Not scale-free	U to U: Double Peeks Gaussian U to L: Gaussian L to U: Power-law	0.933 0.252	U to U: 25, 32 U to L: 30 L to U: 1-3	2-5	6	Conn	0.03-0.05
						FreeNet					
[7]	1	1000	5000 time steps	Small-world Scale-free			50	6		Conn	0.22
	1	200000						log(n)			

Similarly, no way of deciding if a node is taking the content or forwarding it. Generally, each peer locally decides to forward the message to the neighbor that can make most progress towards the destination. The destination is the peer that most likely has a copy of the requested content. Attempts to analyze Freenet networks are very limited since it is based on anonymity. This makes measuring the network's global parameters very challenging. Therefore, as far as we know, all studies that have been conducted on Freenet were on simulated networks.

In his book, O'Reilly [7] reviews a number of P2P systems. Then he analyzes their performance and whether or not they admit the small-world property. At the beginning of the simulation, the author gives each node the address of two other nodes in the network. This makes the overlay similar to a regular graph. Then every 100 time steps, he checks the state of the network by analyzing the data stores of the nodes. Because the network at the beginning is regular, it has a large CPL and high clustering coefficient. However, both attributes decrease with time as the topology becomes increasingly random. At the end, the network has CPL of a random graph, but has a clustering coefficient close to a regular graph. He also finds the graph to exhibit the small-world property. Most routes require about six hops; even though, the optimal CPL was found to be between 2 and 4. Despite the fact that the network is connected, its diameter could not be identified. This can be justified by the fact that in Freenet networks, the produced routes can be much longer than the optimal (shortest) routes. Overall, Freenet has a good average performance but a bad worst-case performance. In another simulation, the author measures the scalability of the Freenet; especially the routing paths under large network size. He finds that the average routing path takes log(n) hops. This means that the Freenet network is highly scalable.

4. Final Remarks and Conclusions

Understanding the different properties that unstructured P2P network overlays have is an essential step for improving their performance [1]. Several analytical studies present results for one or more topological aspects of each system. Here we introduce some of those results. The absence of central entities in P2P networks forces researchers to use other methods to collect the necessary information needed for their analysis. Even with the availability of those methods, there are multiple factors that make analyzing P2P networks very challenging. Those factors can be classified as: network related, tool related, and P2P protocol client related. Generally, a 100% accuracy of the collected data and results is hard to achieve. Some of the major challenges that are related to the network are its size and dynamicity. Peers constantly join, leave, and fail, which requires high speed data collection. Moreover, a P2P network is heterogeneous. Some peers use firewalls and NATs, which makes them difficult to reach. In [12], for example, it was reported that about 10%-15% of the peers were unreachable. The other set of challenges is related to the developed measurement tool. Due to the dynamic nature of the network, using a high-speed tool to collect data is crucial. Although longer windows (such as snapshots) may collect more data, the results can be less accurate [13]. Moreover, most tools require connecting to every peer to collect its routing information. However, connecting to peers who already have reached their maximum number of neighbors will not be possible. This results in incomplete data. The third class of challenges is related to the P2P protocol client. For example, one way of collecting peer data is by exchanging PEX messages with the peer. However, peers who have the µTorrent client do not send PEX messages if they have fewer than three connections.

Acknowledgments

This research project was supported by a grant from the Research Center of the Female Scientific and Medical Colleges, the Deanship of Scientific Research, King Saud University.

References

[1] Andelfinger, P., Neudecker, T., & Hartenstein, H. (2016, July). Timing analysis for inferring the topology of the bitcoin peer-to-peer network. In *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)* (pp. 358-367). IEEE.

[2] Cao, T., Yu, J., Decouchant, J., Luo, X., & Verissimo, P. (2020). Exploring the Monero Peer-to-Peer Network. *Financial Cryptography and Data Security 2020, Sabah, 10-14 February 2020.*

[3] Dale, C., Liu, J., Peters, J., & Li, B. (2008, June). Evolution and enhancement of bittorrent network topologies. In *2008 16th Interntional Workshop on Quality of Service* (pp. 1-10). IEEE.

[4] Essaid, M., Park, S., & Ju, H. T. (2020). Bitcoin's dynamic peer-to-peer topology. *International Journal of Network Management*, e2106.

[5] Hamra, A. A., Legout, A., & Barakat, C. (2007). Understanding the properties of the bittorrent overlay. *arXiv preprint arXiv:0707.1820*.

[6] Kryczka, M., Cuevas, R., Guerrero, C., & Azcorra, A. (2011, August). Unrevealing the structure of live BitTorrent Swarms: methodology and analysis. In *2011 IEEE International Conference on Peer-to-Peer Computing* (pp. 230-239). IEEE.

[7] Lessig, L., Kan, G., & Miller, J. (2001). *Peer-to-peer: Harnessing the Benefits of a Disruptive Technology*. " O'Reilly Media, Inc.".

[8] Park, H., Yang, J., Park, J., Kang, S. G., & Choi, J. K. (2008, February). A survey on peer-to-peer overlay network schemes. In *2008 10Th international conference on advanced communication technology* (Vol. 2, pp. 986-988). IEEE.

[9] Rasti, A. H., Stutzbach, D., & Rejaie, R. (2006, April). On the long-term evolution of the two-tier gnutella overlay. In *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications* (pp. 1-6). IEEE.

[10] Stutzbach, D., & Rejaie, R. (2004). Characterizing today's Gnutella topology. *Technical Report CIS-TR-04–02*.

[11] Stutzbach, D., Rejaie, R., & Sen, S. (2008). Characterizing unstructured overlay topologies in modern P2P file-sharing

systems. *IEEE/ACM Transactions on Networking*, *16*(2), 267-280.

[12] Su, M., Zhang, H., Du, X., Fang, B., & Guizani, M. (2012, December). Understanding the topologies of BitTorrent networks: A measurement view. In *2012 IEEE Global Communications Conference (GLOBECOM)* (pp. 1994-1999). IEEE.

[13] Wang, Y., Yun, X., & Li, Y. (2007, April). Analyzing the characteristics of gnutella overlays. In *Fourth International Conference on Information Technology (ITNG'07)* (pp. 1095-1100). IEEE.

[14] Yang, S. J. (2005). Exploring complex networks by walking on them. *Physical Review E*, *71*(1), 016107.

Hend Alrasheed received the B.E. and M.E. degrees, from King Saud University. She received her PhD in computer Science from Kent State University. in 2018. Currently, she works as an assistant professor in the Department of Information Technology, King Saud University. Her research interests include graph theory, network science, and social network analysis.