From Local Network Formation Game to Peer-to-Peer Protocol

Julian Nickerl
julian.nickerl@uni-ulm.de
Institute of Theoretical Computer Science
Ulm University, Germany

David Mödinger
david.moedinger@uni-ulm.de
Institute of Distributed Systems
Ulm University, Germany

Jan-Hendrik Lorenz
jan-hendrik.lorenz@uni-ulm.de
Institute of Theoretical Computer Science
Ulm University, Germany

ABSTRACT

Peer-to-peer protocols often take longer, are less efficient or can't complete lookup queries with increasing network diameter. Peers could mitigate this by increasing their degree, i.e., their amount of open connections, but this increases the operational cost for each peer.

We propose a novel peer-to-peer network formation protocol based on a game-theoretic approach, guaranteeing that diameter and maximum degree do not surpass given thresholds throughout the network. The game generalizes the local network formation game with more versatile strategies and cost functions. This allows for a trade off between operational cost and efficiency based on the individual interest of peers.

We show that for any given diameter k and maximum degree d a Nash equilibrium, i.e., a graph with the desired properties, can be reached by $O(|\text{players}|^2)$ improvement steps. We validate the practical applicability of these theoretical results on networks of 5–50 participants with various strategies and configurations. The experimental results show a fast approximation of the desired properties while taking some time to reach a stable state. We make out several strategies with which the protocol performs well. In particular, a stable state is found quickly when the initial network was already close to a stable state. This property enables the efficient dynamic treatment of the in practice often occurring scenario of nodes joining or leaving the network.

CCS CONCEPTS

• **Networks** → **Network properties**; • **Theory of computation** → *Theory and algorithms for application domains.*

KEYWORDS

game theory, network formation, network protocol, peer-to-peer, small-world

ACM Reference Format:
Julian Nickerl, David Mödinger, and Jan-Hendrik Lorenz. 2020. From Local Network Formation Game to Peer-to-Peer Protocol . In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Peer-to-peer networks are used in many places, from classical file sharing [7] over distributed file storage [4] to cryptocurrencies using blockchain systems [9]. These systems run various algorithms such as information dissemination of blockchain transactions or lookup queries to find previously stored data [16]. The runtime, efficiency, or even fairness and robustness [9] of these algorithms often depends on the diameter of the network, i.e. the maximum distance between two nodes. This so-called small-world property is an essential goal for peer-to-peer networks.

Usually, this small-world property is only approximated, e.g., by having long-range and short-range routing buckets in Freenet [7]. While experiments verified that this approach approximates a small-world property, there are no guarantees. To improve the probability of reaching a desired diameter, protocols can increase minimum node degree, i.e. the number of connections a node creates. This creates new problems [10], as more connections create more cost for participating nodes.

Other attempts, such as algorithms relying on knowledge of the full connection graph of the network, are impractical for peer-to-peer networks, as participants can not be forced to follow central guidance. Game-theoretic approaches that model individual best interest behavior and incentives do not yet cover the desired scenario of diameter and degree restrictions. In this paper, we propose a novel game-theoretic approach to reach stable network configurations, where diameter and node degree do not surpass given thresholds.

1.1 Contributions

As a summary, this paper will provide these contributions over the state of the art:

(1) We provide a game-theoretic model for network formation as a generalization of the local formation game. The game applies to peer-to-peer networks and respects custom degree and diameter thresholds.
(2) We show that Nash equilibria coincide with networks with the desired properties and can be reached from all states with a number of improvement steps quadratic in the number of players.
(3) We provide various update strategies for a protocol that require different amounts of information about the network.
(4) We show that our approach leads to sensible results based on simulations of our strategies, e.g., approximations of the

desired parameters are reached quickly, and networks close to a stable state (after a node joining or leaving) reach a stable state again after few steps.

1.2 Roadmap

This paper is structured in the following way: Section 2 introduces the relevant background for this paper and work related to our contributions. In Section 3 we describe our game-theoretic model, which we analyze in Section 4 to show that global properties are reached in Nash equilibria. We introduce several strategies to transform the game-theoretic model into a peer-to-peer protocol and describe them in Section 5. To evaluate the practicality of our approach, we simulate our strategies, presenting the results in Section 6.

2 BACKGROUND AND RELATED WORK

In this section, we discuss related approaches as well as the scenario and background of our work.

2.1 Network Formation Games

Two prominent game-theoretic approaches to simulate the formation of networks are the local and global network formation games, by Fabrikant et al. [12] and Anshelevich et al. [2, 3] respectively. Roughly summarized, in the local network formation game by Fabrikant et al., every player is a node. A player can establish an undirected edge between herself and any other node. The cost of a player depends on the number of edges she establishes, and the distance to the other nodes, which she both tries to minimize. In the global game by Anshelevich et al., each player is associated with a set of nodes she tries to connect. Any player can pay any amount for any edge in the network. However, an edge is only established once the sum of the payments reaches a threshold. In addition to the payments for the edges, a player suffers cost if she is unable to connect her associated nodes. For a survey of both game-theoretic and non-game-theoretic approaches to network formation, see e.g. [13] by Jackson. For a more recent and extensive overview, see [14] by Jackson.

Both games, and extensions thereof (see, e.g. [6, 8, 15]), try to simulate how networks with rational participants naturally form, with the focus on analyzing the games' price of anarchy. However, the graphs representing Nash equilibria in these approaches are usually very restricted, failing to guarantee interesting properties of the topology. For example, most Nash equilibria in the local network formation game are either the complete graph, the empty graph, or trees (especially stars), with a loose upper bound on the diameter of $2\sqrt{\alpha}+1$, where α is the cost for establishing an edge. This is analyzed, e.g. by Fabrikant et al. in [12] and Albers et al. in [1].

In this work, we are especially interested in two values: the network's diameter, and the nodes' maximum degree. A lack of guarantees on these two values can make the resulting networks unsatisfactory for several applications. For example, Mödinger et al. [17] propose a privacy motivated broadcast protocol for blockchains consisting of three phases. If it could be guaranteed, that the respective network had diameter at most a given d, the third phase could be neglected, significantly improving the protocol's performance. Decker and Wattenhofer [9] explore the problem of blockchain forks, and link their emergence, among others, to the network's diameter. Restricting the degree is often necessary for practical reasons, as Dekker et al. [10] noted. Apart from physical restraints, high degrees can lead to security and privacy issues. As a result of these issues, the motivation for this work sharply differs from the network formation games mentioned above. We are interested in modeling a game that guarantees, that its Nash equilibria correspond to networks with diameter at most d, and maximum degree at most k, for given d and k.

2.2 Scenario

We assume a context close to peer-to-peer networks, where every node, or player, acts rationally, but has selfish interests, and is not allowed to cooperate with other players. The diameter in such networks gives an upper bound on the time a message travels from the source to the destination node. However, the diameter is a global property, while the actions and interests of each player are locally motivated.

This work generalizes the previously discussed local network formation game by Fabrikant et al. [12] by allowing more versatile cost functions and strategies, s.t. a state of the game is a Nash equilibrium if and only if the corresponding network fulfills the desired properties. The game is designed to allow the deletion of edges without the necessity of players cooperating, which has been an issue of previous designs, as discussed in Chapter 11 of [14] by Jackson.

2.3 Approach in Peer-to-Peer Systems

In previous works on peer-to-peer systems, such as the work by Zhang et al. [19], the property of network diameters is discussed in the form of the so-called small-world property. Derived from the Milgram experiment, it describes the desired feature of routing messages between participants in relatively few steps, i.e., having a small network diameter.

We examine Freenet [7] as an example of a peer-to-peer network that tries to reach such a small-world network, but other systems follow similar strategies. Freenet is a structured peer-to-peer network and uses a circular identifier space $[0, 1)$ for its participants and the files stored in the system. Each node is responsible for files with an identifier close to its node identifier, which overlap for redundancy. When a node wants to store a file, it forwards the file to the neighbor with the closest identifier to the file identifier.

To help this structure succeed and reach all desired nodes in a few steps, the routing table is split into two parts. The majority of routing entries contain short-range routes: Nodes that have identifiers close to the nodes own id. This local clustering helps to find the explicit close targets. The smaller part of the routing table contains long-range entries, i.e., connections that lead to nodes far away in the identifier space.

Empirically, it was shown that this construction exhibits small-world properties [19], but no strong guarantees exist, and it grows over a considerable amount of time. This system requires an identifier space of a structured or semi-structured peer-to-peer network. In our game-theoretic approach, no such identifier space or structure is required.

3 MODEL

We base the model on the local network formation game by Fabrikant et al. [12], extended with different cost functions and the possibility to reject and accept edges. In the course of this paper, sets of the form $\{1, ..., m\}$ will be abbreviated by $[m]$ for any $m \in \mathbb{N}$. Even though we are interested in undirected graphs, the following model is easier understood if all edges are interpreted as directed. A previously undirected edge is then replaced by two directed edges, guaranteeing the symmetric connection. Later on, we will see, that all "stable" connections are symmetric, i.e., either edges exist in both directions, or none.

Definition 3.1. A generalized local network formation game is a tuple $\Gamma = (n, (c_i)_{i \in [n]})$ with $n \in \mathbb{N}$ the number of nodes of a directed graph. Each node is a player of the game. A strategy of player i is a vector $s_i = (s_{i\sim 1}, ..., s_{i\sim(i-1)}, s_{i\sim(i+1)}, ..., s_{i\sim n})$, with $s_{i\sim j} \in \{\square, +, -, \times\}$. The symbol $\square$ stands for standing idle, $+$ for establishing a connection, $-$ for rejecting an incoming edge, and $\times$ for accepting an incoming edge. A state is a tuple $S = (s_1, ..., s_n)$ representing strategies of each player in that state. The cost of a player i in a state S is given by $c_i(S)$. The cost functions c_i project from the set of states of the game to the real numbers.

The major differences to the original local network formation game are the introduction of two new strategies, $-$ and $\times$, and the generalization of the cost functions. This generalized game can be seen as a meta-game. Different interpretations of strategy combinations and definitions of cost functions lead to vastly different games. In the following, we introduce interpretations and definitions that lead to a game whose Nash equilibria form graphs fulfilling the desired restrictions on the diameter and the maximum degree. A *Nash equilibrium* is an essential concept in game theory and describes a state where no player can decrease her cost by changing her strategy. These states are stable under the assumption of rational (the player prefers states with low cost over states with high cost) and selfish (the player prefers states with low cost for herself over states with high cost for herself but low cost for others) players. Both are reasonable assumptions in the context of peer-to-peer networks. For more details on the game-theoretic terminology, we refer the reader to [18] by Nisan et al.

First, we explain how different combinations of strategies translate to a graph. Every node represents a player (*player* and *node* will be used interchangeably in the rest of the work). Figure 1 illustrates the resulting connection between nodes i and j for the different combinations of strategies. The missing combinations are symmetric cases of the ones shown in the figure.

We call a connection between player i and player j *established* if there exists at least one edge between them according to Figure 1. We say player i establishes a connection between i and j if $s_{i\sim j} = +$. Furthermore, player i *rejects* an (incoming) edge if $s_{i\sim j} = -$, and *accepts* it if $s_{i\sim j} = \times$. We call a connection *stable*, if one of the players chooses $+$ and the other $\times$, or both players choose $\square$.

Note, that rejection, i.e., choosing strategy $-$, only refers to the incoming edge, not the connection as a whole. This leads to an unintuitive interaction: Following Figure 1b), the strategy pair $(s_{i\sim j}, s_{j\sim i}) = (+, \square)$ leads to a directed edge from i to j. However, j rejecting the incoming edge results in the strategy pair

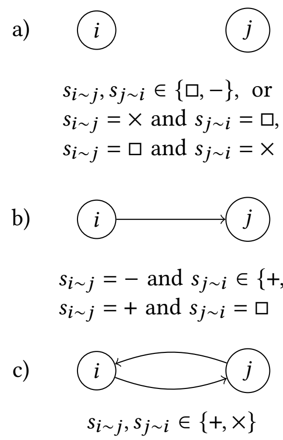

Figure 1: Resulting connections for different strategy combinations.

$(s_{i\sim j}, s_{j\sim i}) = (+, -)$, leading to a directed edge from j to i while removing the edge from i to j.

The properties in-degree, distance, neighborhood, and diameter are defined as follows:

Definition 3.2. We define $G_\Gamma(S) = (V, E)$ as the directed graph that represents state S of Γ according to Figure 1.

The function $deg_i(S)$ is the in-degree of node (or player) i in $G_\Gamma(S)$:

$$deg_i(S) = |\{(j, i) \in E\}|.$$

The distance $dist_{i\sim j}(S)$ from player i to player j is the minimum number of edges necessary on a path from i to j in $G_\Gamma(S)$. If a node j is unreachable from i, we agree on the distance n from i to j.

The m-neighborhood $\mathcal{N}_i^m(S)$ is the set of nodes with distance at most m from i:

$$\mathcal{N}_i^m(S) = \{j \mid dist_{i\sim j}(S) \leq m\}.$$

The diameter is the smallest integer d, such that for all players i, $\mathcal{N}_i^d(S) = V$.

Right away, there are two interesting properties to mention:

Property 1. *A game $\Gamma = (n, (c_i)_{i \in [n]})$ with its states interpreted as defined above has the following properties:*

1. *Initially establishing a connection does not increase the in-degree of the establishing player.*
2. *Rejecting an incoming edge does not decrease the distance of the rejecting player to the other nodes.*

Proof. The properties follow directly from Definition 3.2 and the interpretation of strategy combinations from Figure 1.

1. Player i initially establishing a connection with j leads to the strategy pair $(s_{i\sim j}, s_{j\sim i}) = (+, \square)$. Following Figure 1b), this only creates a directed edge from i to j, not affecting the in-degree of i.
2. Player i rejecting an incoming edge leads to either $(-, +)$ or $(-, \times)$ or $(-, -)$. In all cases, an outgoing edge from i is either preserve or did not exist in the first place. Hence, the distance from i to all other nodes does not decrease.

□

The goal is now to define the cost functions, s.t. every Nash equilibrium represents a graph with maximum diameter at most d and maximum degree at most k, with d and k as given constants. The cost functions consist of four components, each fulfilling a particular role:

- $\kappa_i(S,k)$ generates cost whenever the in-degree of player i in $G_\Gamma(S)$ is above the desired maximum degree k.
- $\delta_i(S,d)$ generates cost whenever there is at least one player whose distance from i is more than the required maximum diameter d.
- $sat_i(S,k,d)$ is non-zero if $\kappa_i(S,k) = \delta_i(S,d) = 0$. This indicates that player i locally fulfills the requirements.
- $\zeta_{i\sim j}(S,k,d)$ ensures that all Nash equilibria only have stable connections. Additionally, the cost function links a cost of 1 to every established connection to the player establishing it. Another important property is that ζ uses the *sat* function to create cost whenever a player rejects an edge even though she locally already fulfills the required properties.

Formally, the components are defined as follows:

$$\begin{aligned}
\kappa_i(S,k) &:= \begin{cases} 0, & \text{if } deg_i(S) \le k \\ N \cdot deg_i(S), & \text{else} \end{cases} \\
\delta_i(S,d) &:= N \sum_{j \in [n] \setminus \mathcal{N}_i^d(S)} dist_{i\sim j}(S) \\
sat_i(S,k,d) &:= \begin{cases} 1, & \text{if } \kappa_i(S,k) = \delta_i(S,d) = 0 \\ 0, & \text{else} \end{cases}
\end{aligned}$$

With $\zeta_{i\sim j}(S,k,d)$ given by the table:

$s_{i\sim j} \setminus s_{i\sim j}$	$+$	$-$	$\square$	$\times$
$+$	N	N	1	1
$-$	$sat_i(S,k,d)$	N	N	$sat_i(S,k,d)$
$\square$	N	0	0	N
$\times$	0	N	N	N

Here, N is a large number. For example, $N = n^2$ suffices. The final cost functions are

$$\begin{aligned}
c_i(S) &= c_i^{k,d}(S) \\
&= \kappa_i(S,k) + \delta_i(S,d) + \sum_{j \in [n] \setminus i}^{n} \zeta_{i\sim j}(S,k,d).
\end{aligned}$$

4 NASH EQUILIBRIA GUARANTEEING GLOBAL PROPERTIES

As mentioned before, we are interested in undirected graphs with maximum degree at most k and maximum diameter at most d. However, the above interpretation of the generalized local network formation game features only directed graphs. Still, we show that if a state is a Nash equilibrium, then all connections must be stable. Since stable connections are those where either no edge exists or edges exist in both directions, this coincides with an undirected graph. Additionally, the in-degree of a node in the directed graph with only stable connections coincides with the degree of the node in the undirected graph. Accordingly, if we speak of the *degree* of a node, we typically refer to its in-degree. A Nash equilibrium is reached when no player can decrease her cost by changing her strategy.

Theorem 4.1. *If a state S is a Nash equilibrium of $\Gamma = (n, (c_i^{k,d})_{i\in[n]})$, all connections are stable.*

Proof. We consider two cases. In the first, we assume, that $G_\Gamma(S)$ fulfills the desired property. We know, that then $sat_i(S,k,d) = 1$ for all players i. This means additionally, that for every player, the only non-zero component of the cost function is $\zeta_{i\sim j}(S,k,d)$. We can update the cost function by setting $sat_i(S,k,d) = 1$.

$\zeta_{i\sim j}(S,k,d) :=$

$s_{i\sim j} \setminus s_{i\sim j}$	$+$	$-$	$\square$	$\times$
$+$	N	N	1	1
$-$	1	N	N	1
$\square$	N	0	0	N
$\times$	0	N	N	N

Together with $\zeta_{j\sim i}(S,k,d)$, this cost function can be interpreted as a two-player symmetric cost-minimization game. The game has three pure Nash equilibria, marked in green, that exactly coincide with the states representing stable connections. There exists a player that can reduce her cost in any other state of the game.

In the other case, $G_\Gamma(S)$ does not fulfill the properties, i.e., there exists a player i with $sat_i(S,k,d) = 0$. We can update $\zeta_{i\sim j}(S,k,d)$ for player i accordingly.

$\zeta_{i\sim j}(S,k,d) :=$

$s_{i\sim j} \setminus s_{i\sim j}$	$+$	$-$	$\square$	$\times$
$+$	N	N	1	1
$-$	0	N	N	0
$\square$	N	0	0	N
$\times$	0	N	N	N

The new cost function defines another two-player cost-minimization game. This game is not necessarily symmetric since $sat_j(S,k,d)$ can be either 1 or 0. However, the only Nash equilibria are again stable states, marked in green.

Therefore, S can only be a Nash equilibrium if all connections are stable. □

An additional property of the two-player games used in Theorem 4.1 is that there always exists a sequence of improvement steps leading to a stable state:

Corollary 4.2. *Given a state S of $\Gamma = (n, (c_i^{k,d})_{i\in[n]})$, either all connections in S are stable or there exists a sequence of improvement steps to a state S^*, s.t. all connections in S^* are stable.*

Using Theorem 4.1, we can ensure, that a state of the game is a Nash equilibrium, if the respective graph fulfills the desired restrictions on the maximum diameter and maximum degree.

Theorem 4.3. *If S is a Nash equilibrium of the game $\Gamma = (n, (c_i^{k,d})_{i\in[n]})$, then $G_\Gamma(S)$ has maximum degree at most k and diameter at most d.*

Proof. We conduct a proof by contradiction. Assume S is a Nash equilibrium that does not meet the diameter or degree restrictions. Due to Theorem 4.1, we can assume that all connections in $G_\Gamma(S)$ are stable. We consider two cases:

(1) There exists a node i with in-degree above k, i.e., $deg_i(S) > k$. Player i can reduce her in-degree and therefore her cost by rejecting any edge directed at her. According to Property 1, the player reduces her cost for the in-degree, while leaving the costs for distances unchanged.
(2) The degree of every node is at most k. However, there exists a tuple (i, j) with $dist_{i\sim j}(S) > d$. Player i can reduce her cost for the distances by establishing a connection to j. According to Property 1, the cost for the in-degree is unaffected. Additionally, reducing the distance reduces player i's cost by at least N, while establishing an edge has cost only 1.

In all cases, we can find a player that can reduce her cost. Therefore, S cannot be a Nash equilibrium, and the theorem holds. □

While Theorem 4.3 shows, that every Nash equilibrium of the game represents a graph fulfilling the desired properties, it does not guarantee the converse.

Theorem 4.4. *Given a state S of the game $\Gamma = (n, (c_i^{k,d})_{i\in[n]})$. If $G_\Gamma(S)$ has maximum degree at most k and diameter at most d, and all connections are stable, then S is a Nash equilibrium.*

Proof. If $G_\Gamma(S)$ is already a graph with the desired properties, the cost functions $\kappa_i(S, k) = \delta_i(S, d) = 0$, and therefore, $sat_i(S, k, d) = 1$ for all players i. The only way a player could potentially reduce her cost is by reducing the number of edges she establishes. According to the $\zeta_{i\sim j}$, this requires rejecting a number of incoming edges, since any other approach to undo establishing the connection is linked to even higher cost. However, since $sat_i(S) = 1$, rejecting an edge has at least the same cost in $\zeta_{i\sim j}(S, k, d)$ as maintaining the connection, i.e., the the player cannot further reduce her cost. □

Theorems 4.3 and 4.4 form an equivalence:

Corollary 4.5. *Given a state S of the game $\Gamma = (n, (c_i^{k,d})_{i\in[n]})$, S is a Nash equilibrium if and only if $G_\Gamma(S)$ has maximum degree at most k and diameter at most d, and all connections are stable.*

An additional direct consequence of Corollary 4.5 is the following:

Corollary 4.6. *Given n, k, and d, the game $\Gamma = (n, (c_i^{k,d})_{i\in[n]})$ has at least one Nash equilibrium if and only if there exists a graph with n nodes, maximum degree at most k and diameter at most d.*

So far, we have guaranteed the existence of Nash equilibria representing graphs that satisfy the desired properties. We can furthermore show, that from any state, a Nash equilibrium can be reached by a sequence of improvement steps of quadratic length in the number of players. However, these steps do not have to be best responses. We accompany the steps of the upcoming proof with an example, illustrated by Figure 2.

Theorem 4.7. *Given a state S of $\Gamma = (n, (c_i^{k,d})_{i\in[n]})$. If there exists a graph with n nodes, maximum degree at most k and diameter at most d, a Nash equilibrium is reachable from S with a sequence of improvement steps of length $O(n^2)$.*

Proof. According to Corollary 4.6, there exists a Nash equilibrium S^* in this game. For $n = 5$, $k = 2$, and $d = 2$, this is, for example, the graph from Figure 2a). Let E^* be the edge set of $G_\Gamma(S^*)$. For simplicity, assume that $G_\Gamma(S^*)$ represents the only possible graph that fulfills the properties. If a different equilibrium is reached during the following steps, directly terminate on that equilibrium. In four main steps, we show that we can reach a state where all established connections are stable and correspond to the edges from E^* and vice versa.

(1) **Reduce all degrees to at most k:** According to Corollary 4.2, we can assume, that all connections are stable, or can be made stable with a sequence of $O(n^2)$ improvement steps since every edge has to be visited at most twice.
Any player i with $deg_i(S) > k$ can reject any $deg_i(S) - k$ incoming edges, thus reducing her cost. Afterward, we "clean up" all unstable connections, s.t. both players choose □, which is possible, again, due to Corollary 4.2. Once this is done, we reach a state $S^{(1)}$, where all players have degree at most k, and again all connections are stable.
In our example, Figure 2b) is the initial graph. The degree of nodes c and d is too high. In order to reduce their degrees, c can reject the edge from e, and d the edges from b and c. Stabilizing the connections leads to Figure 2c).
(2) **Create a disconnected graph:** In general, $G_\Gamma(S^{(1)})$ does not have diameter at most d yet. Since all connections in $S^{(1)}$ are stable, they are symmetric. Hence, there exists a pair (i, j) with $dist_{i\sim j}(S^{(1)}) = dist_{j\sim i}(S^{(1)}) > d$. Clearly, $s_{i\sim j} = s_{j\sim i} = \square$, since otherwise $dist_{i\sim j}(S^{(1)}) = dist_{j\sim i}(S^{(1)}) = 1$. We set $s_{i\sim j} = +$, and reject all incoming edges to i, thus reducing i's cost (record that due to Property 1, rejecting edges does not affect the player's distances). However, j can reject all incoming edges, including the one from i. According to Figure 1b), the resulting strategy combination $(+, -)$ between players i and j represents a directed edge from j to i. This decreases the distance from j to i to 1, reducing j's cost in return.
Returning to a state where all connections are stable leaves a disconnected graph with at least three components. Players i and j are isolated nodes, while all other nodes are possibly connected. We call this final state $S^{(2)}$, and $\hat{V} = [n]\backslash\{i, j\}$.
In our example, we can choose nodes b and e as i and j respectively, since $dist_{b\sim e}(S^{(1)}) = dist_{e\sim b}(S^{(1)}) = 5 > 2$. Player b establishing a connection to e creates a directed edge from b to e. However, b is already disconnected, i.e., no edges have to be rejected by her. Node e, on the other hand, can reject the incoming edge from b, thus creating a directed edge to b herself. Simultaneously, e can reject the incoming edge from d. Stabilizing the connections leads to Figure 2d).
(3) **Establish connections in $\hat{V}$:** Let $\mathcal{A}$ and $\mathcal{B}$ be the isolated nodes from the previous step. For every node $i \in \hat{V}$, establish a connection to $\mathcal{A}$ and establish all connections that correspond to edges $(i, j) \in E^*$ with $j \neq \mathcal{A}$ and $j \neq \mathcal{B}$. Reject all incoming edges that are not in E^*. Since $\mathcal{A}$ was unreachable beforehand, the total cost of i is guaranteed to lower. Player $\mathcal{A}$, on the other hand, can now decrease her cost by rejecting the edge from i.
After considering every node in $\hat{V}$, all edges between nodes from $\hat{V}$ are also in E^*, and $\mathcal{A}$ and $\mathcal{B}$ remain isolated nodes.

After returning to all stable connections, we call this state $S^{(3)}$.
In the example, consider player c. The procedure for the other players functions accordingly. Node c establishes a connection to $\mathcal{A}$. Simultaneously, she can reject the incoming edge from a, since (a, c) is not an edge in the desired Nash equilibrium. Additionally, she establishes a connection to d, which is part of the Nash equilibrium. $\mathcal{A}$ rejects the edge from c, keeping the node disconnected after stabilizing the edges. After performing a similar procedure for the other nodes, Figure 2e) is reached.

(4) **Reconnect the graph:** Finally, $\mathcal{A}$ can establish all its connections corresponding to edges from E^* apart from the one to $\mathcal{B}$ (if it is part of E^*). Similarly, $\mathcal{B}$ can establish all its connections from E^* including the one to $\mathcal{A}$ if $(\mathcal{B}, \mathcal{A}) \in E^*$. Accepting all incoming edges leads to state $S^{(4)}$.
In the example, player $\mathcal{A}$ establishes the connections to a and c, and similarly, $\mathcal{B}$ establishes the connections to a and d. Stabilizing all edges leads to the final graph from Figure 2f).

The established connections in $S^{(4)}$ exactly correspond to the edges from E^*. Since S^* is a Nash equilibrium, and according to Theorem 4.3, a graph with edges as in E^* has maximum degree at most k and maximum diameter at most d. Due to Theorem 4.4, $S^{(4)}$ is a Nash equilibrium of the game.

Each of the above phases requires $O(n^2)$ improvement steps, since every edge is visited at most a constant number of times, and the number of edges is bounded from above by $O(n^2)$. □

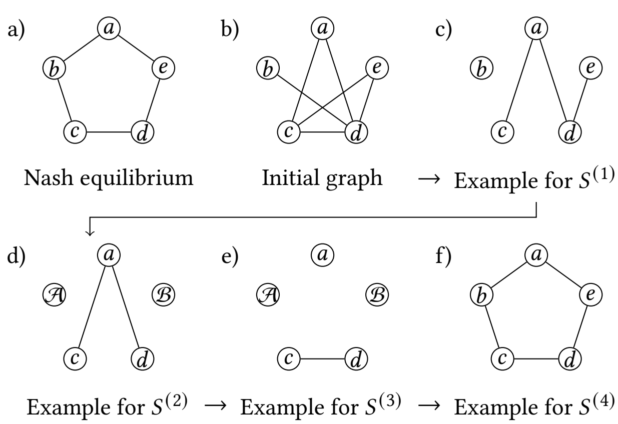

Figure 2: Illustration of the proof of Theorem 4.7 for $n = 5, k = 2, d = 2$.

An interesting property is the size of the graphs in a Nash equilibrium. Since these states are equivalent to the possible graphs that fulfill the desired properties concerning the maximum degree and diameter, it suffices to analyze the size of graphs with these properties. A simple upper bound to the number of edges of a graph with diameter at most d and maximum degree at most k is $m \leq \frac{nk}{2}$. Based on a result by Erdős, Rényi, and Sós [11], a lower bound can be found in Chapter 4 of [5] by Bollobás:

Theorem 4.8. *([5]) The number of edges in an undirected graph with n nodes, diameter at most d, and maximum degree at most k is at least* $\frac{n(n-1)(k-2)}{2((k-1)^d-1)}$.

Lastly, it is not hard to see, that the game still behaves in the intended way, if each node chooses its own maximum degree and maximum eccentricity. This is a useful property, since, e.g. different peers in a peer-to-peer network have access to varying qualities of hardware, and can thus maintain different amounts of connections.

5 PRACTICALS

The previous section has introduced a game-theoretical model that, if transferred into a network protocol, would lead to states with beneficial properties. However, this transformation faces some hurdles.

An assumption in the game from the previous section is that it is played only exactly once. The improvement steps mentioned, e.g. in Theorem 4.7 are considerations regarding the choice of strategy *before* applying them. However, the core concept of these improvement steps can be used to create update rules for the protocol. Out of the many options we have considered the following rules:

Regarding establishing a new edge (in the following also called the *add type*), we considered

- *bounded by degree, furthest first*: Establish edges to nodes that are too far away, until the maximum desired degree is reached or an edge has been established to all nodes in question. Prioritize further away nodes over closer ones.
- *all*: If there is an edge out of reach, establish an edge to it (if possible).
- *bounded by degree, random*: Uniformly at random establish edges to nodes that are too far away, until the maximum desired degree is reached or an edge has been established to all nodes in question.

Regarding rejecting an established edge (in the following also called the *delete type*), we considered

- *least important first*: Reject connections until the maximum desired degree is reached, prioritizing connections to nodes that are seldom the next step on a shortest path to another node.
- *oldest first*: Reject connections until the maximum desired degree is reached, prioritizing connections that have been established a long time ago.
- *youngest first*: Reject connections until the maximum desired degree is reached, prioritizing connections that have been established recently.
- *random*: Uniformly at random reject connections until the maximum desired degree is reached.

Note, that one could think about each time choosing the actions that would minimize the respective player's cost regarding the game's rules, however finding these actions is likely to be a hard problem [12].

Now we have some options on update rules; however it is not clear which player should apply them. When we considered an improvement step in the game, it always only concerned a single player. However, it is not usual that only a single peer in a network may update its state. Again, we compared several different approaches to choosing who may apply her updates (in the following also called the *choose type*):

- *highest cost*: Choose the player with the highest cost with respect to the game's rules.
- *highest eccentricity*: Choose the player with the highest eccentricity.
- *highest degree*: Choose the player with the highest degree.
- *one chosen at random*: Choose a player at random, weighted by the number of changes she would apply. A high number of changes results in a higher chance.
- *all simultaneously*: All players apply their changes simultaneously.

Clearly, not all of the above rules can be applied directly in a network protocol. One of the biggest issues is that of lack of information: In the game model, complete information is assumed. However, we are aiming for a peer-to-peer network; hence there should not be a central agent supplying the peers with information. The different rules presented above each require a different amount of global information. We have listed them roughly in descending order with respect to the required information. The less information a rule requires, the more practical it is in a peer-to-peer network setting. It should be noted, however, that for each approach, the complete topology of the network remains unknown.

6 EVALUATION

The previous sections introduced a theoretical model and proposed different approaches to a peer-to-peer network protocol. In this section, we describe and present experiments we have conducted to compare said approaches.

6.1 Methodology

Based on network size, desired diameter, and desired maximum degree, we consider six different network configurations with five to fifty participants. The initial networks were generated in different ways (in the following also called the *generation type*). The simplest cases are the empty and full network, where, respectively, no edges exist at all or all edges exist. Additionally, we consider initial networks where the existence of each directed edge is decided based on a small constant probability and networks in which we simulate a participant leaving and joining again after reaching a stable state. For this last type of network, we generate a stable network and then remove all connections to and from a randomly chosen node.

Adding the options from the previous section (of choice of next player, choice of which edges to establish, and choice of which edges to reject) to network size and initial network generation type, we total in 1440 different parameter combinations which we exhaustively apply. Twenty runs are started with each of the parameter combinations. The simulation is interrupted once a stable state is reached or after a maximum of 250.000 improvement steps, indicating problems with convergence.

6.2 Discussion of Results

The experiments yield several interesting results. A first look at the number of instances that failed to find a solution even after 250.000 steps seemed devastating. However, we were able to make out four strategies that lead to almost all of these fails. These strategies are the choose types *highest eccentricity* and *highest degree*, and the delete types *least important first* and *youngest first*. After removing results from instances that use one of these types, only a total of six runs failed to find a stable solution in time, compared to the previous thousands, as visualized by Figure 3. In the following discussion of results, these four strategies are omitted.

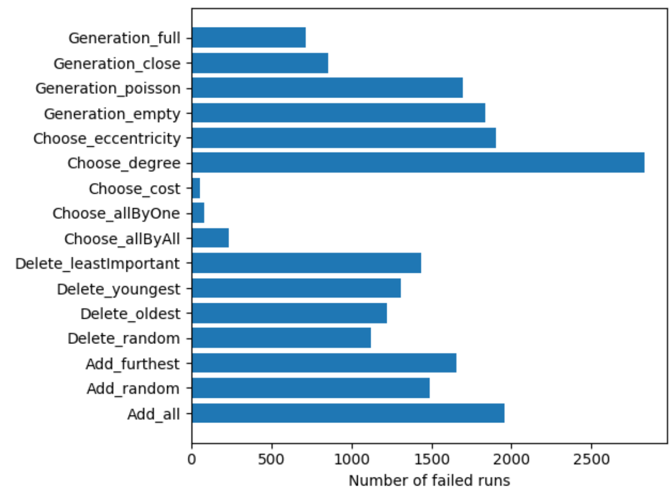

Figure 3: For each parameter, the number of runs that, if that parameter was present, failed to find a stable solution even after 250.000 steps. After removing Choose_eccentricity, Choose_degree, Delete_leastImportant, and Delete_youngest, only a total of six of the remaining runs failed.

Certain behaviors can be observed over almost all parameter combinations. The network quickly reaches a state that is close to an equilibrium but requires a longer time for final convergence. This effect can be seen, for example, in Figure 4 and Figure 5, where after an initial steep improvement, all strategies struggle to finish. Still, reaching almost stable solutions very fast is already a desirable property, especially in networks with a lot of fluctuation. Additionally, especially the average distance between the participants is and remains low almost immediately, as can be seen in Figure 6.

Lastly, some single parameters show noteworthy behavior. Initializing the network by simulating the leaving and joining of a node finds a stable state very fast (see Figure 7), and never moves far away from an equilibrium (see again Figure 4 or Figure 5). This is a valuable property since it indicates a stable convergence. Also, it enables the efficient dynamic treatment of the in practice often occurring scenario when the network is already close to a stable solution.

A negative mention is the add type *all*. Figure 8 demonstrates that compared to the other add types, *all* severely struggles to converge to an equilibrium. This is also clearly displayed in the runtime comparison of Figure 9.

A last interesting parameter is the choose type *highest cost*, as it most closely respects the underlying game, even if it may not

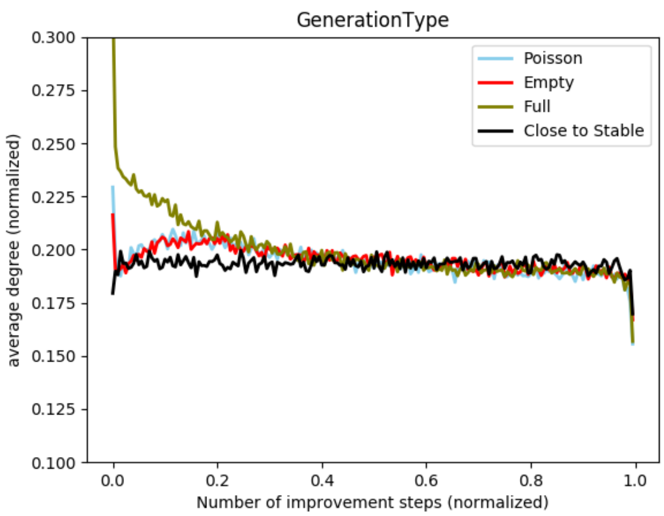

Figure 4: Comparison of the development of the average degree for the generation types. All strategies quickly settle on a value close to the final result.

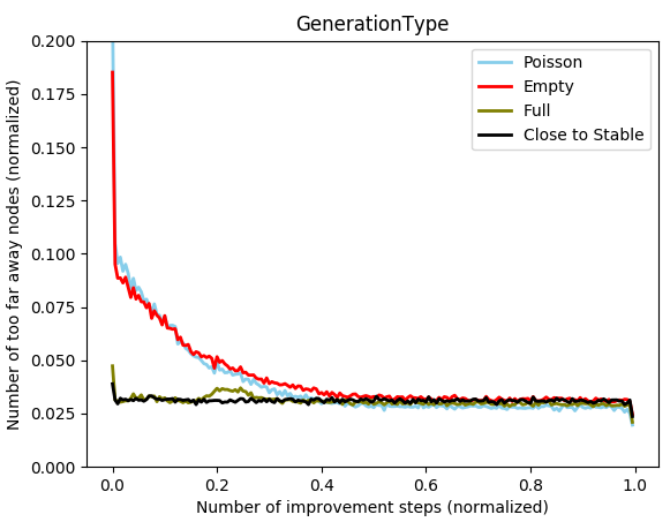

Figure 5: Comparison of the development of the number of nodes that are still too far away from at least one other node for the generation types. All strategies quickly settle on a value close to the final result. The *close to stable* strategy performs particularly stable.

be realistic in practice as it requires too much information. It can be seen as a confirmation of the design of the original game, that runs with this parameter seem to have little problem converging (see Figure 10), and perform well with respect to the runtime (see Figure 11).

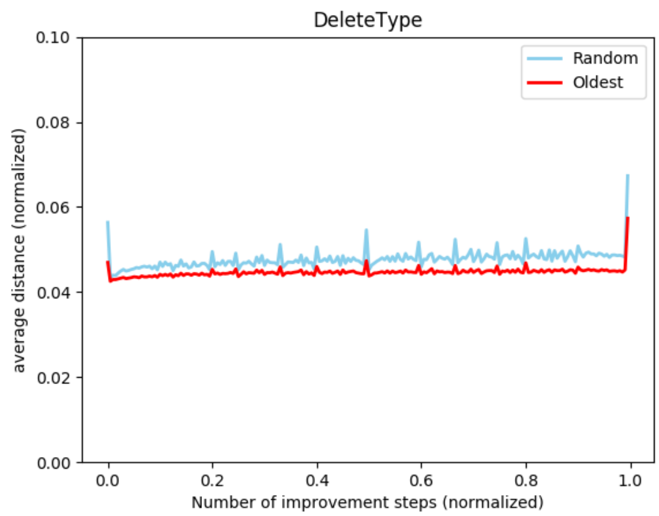

Figure 6: Comparison of the development of the average distances for the delete types. Besides minor fluctuations, both strategies remain stable around a small value that is close to the final result.

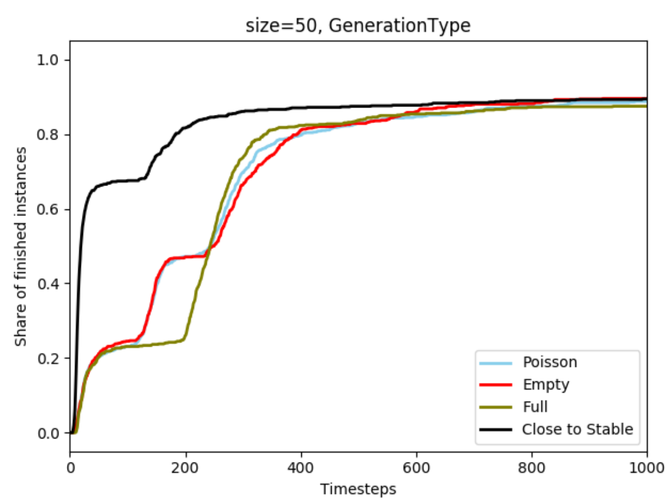

Figure 7: Comparison of runtimes for the generation types. *Close to stable* performs particularly well, solving most instances with few steps. The other strategies perform similarly.

On the negative side, the current protocol prototype suffers from rather slow convergence, and still requires unrealistic amounts of information. Nonetheless, these initial experiments are promising. For most parameter combinations, the distances between the nodes quickly decrease without drastically increasing the participants' degrees. Generating the initial network such that it is close to an already stable state demonstrates the stable and efficient behavior

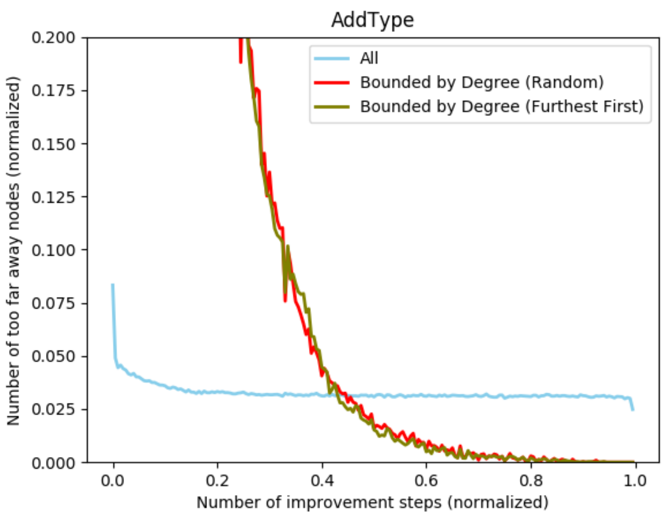

Figure 8: Comparison of the development of the number of nodes that are still too far away from at least one other node for the add types. While adding all edges quickly reduces the amount of too far away nodes, it has severe problems in finding the final solution. The other two strategies perform similarly, quickly finding a stable solution after reducing the number of far away nodes.

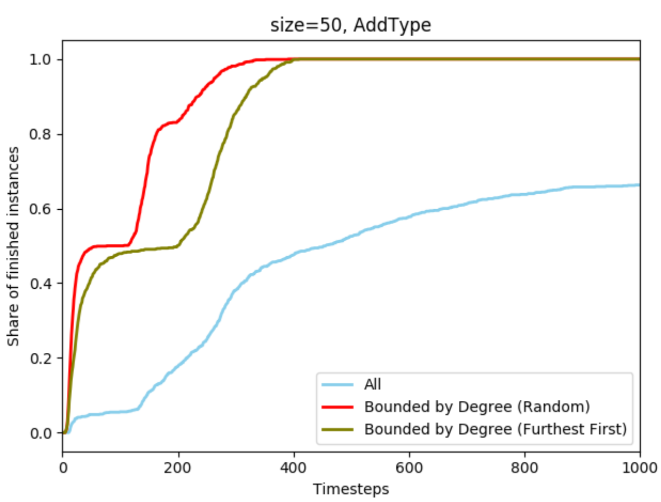

Figure 9: Comparison of runtimes for the add types. Adding all edges struggles to find any stable solution. The other two strategies perform similarly, solving all given instances after only about 400 steps.

of the protocol for an in practice regularly occurring scenario. It is not unreasonable to assume that the protocol performs well with a good choice of parameters.

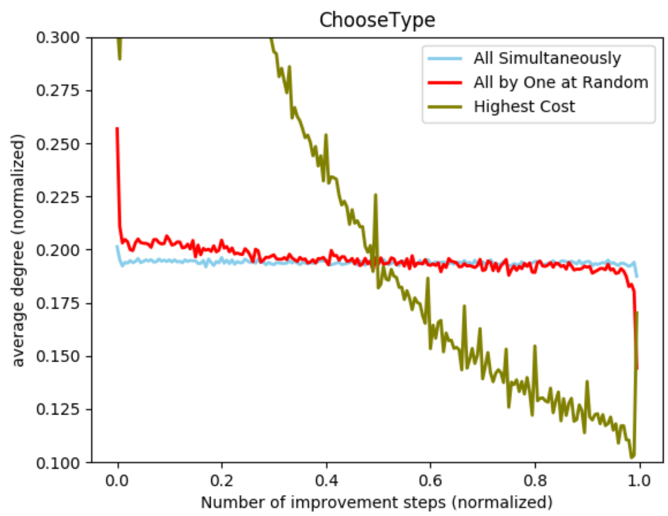

Figure 10: Comparison of the development of the average degree for the choose types. *Highest cost* does not seem to have any problem finding the final stable solution.

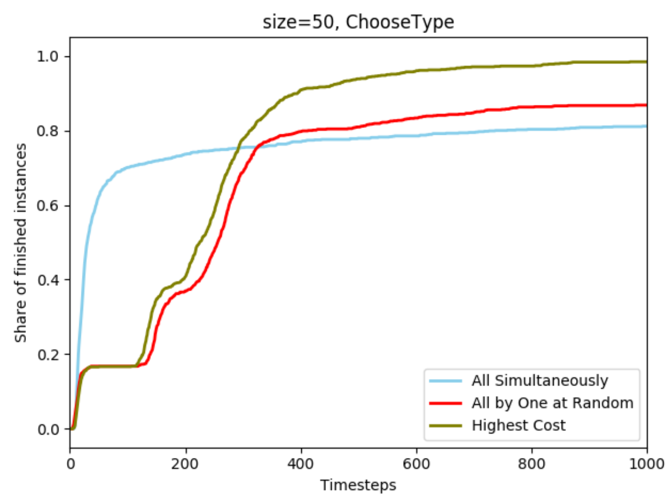

Figure 11: Comparison of runtimes for the choose types. Letting all players update simultaneously finds stable solutions to most instances significantly faster than the other strategies. However it seems to struggle with the remaining instances.

7 CONCLUSION

This work introduced a novel peer-to-peer protocol dedicated to forming and maintaining networks with a given upper bound on diameter and maximum degree. For this purpose, we generalized the local network formation game, allowing for more strategies and cost functions, and used it as the basis for the protocol. We show that the game's Nash equilibria exactly coincide with networks with the desired properties. We proved, that, if at least one such Nash equilibrium exists, it can be reached in a sequence of $O(|\text{players}|^2)$ improvement steps. From these results follows, that reaching a Nash equilibrium guarantees global properties, despite the players being locally motivated and uncooperative.

The rationality and selfishness of the players allow for easy conversion of cost-reducing strategy changes to actions in a peer-to-peer protocol. We introduced several strategies for establishing edges, rejecting edges, and choosing the next player to apply these updates. The strategies require different amounts of global information. Those that need more information tend to more closely follow the game's rules, while those with less information tend to be more realistic for a practical peer-to-peer protocol.

The strategies are exhaustively tested in experiments on networks of five to fifty participants. The simulations show that our approach is promising. Most strategies quickly approximate the desired parameters; however, they can take longer to reach the final stable result. It stands out that initializing the network close to a stable state performs particularly well. This property enables the efficient dynamic treatment of the in practice often occurring scenario of nodes joining or leaving the network.

These results show the applicability of our approach for real networks, guaranteeing the desired properties for all stable networks. This is unlike previous methods which only approximate these concepts and fail to give similar guarantees.

REFERENCES

[1] Susanne Albers, Stefan Eilts, Eyal Even-Dar, Yishay Mansour, and Liam Roditty. 2006. On Nash equilibria for a network creation game. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*. Society for Industrial and Applied Mathematics, 89–98.

[2] Elliot Anshelevich, Anirban Dasgupta, Jon Kleinberg, Eva Tardos, Tom Wexler, and Tim Roughgarden. 2008. The price of stability for network design with fair cost allocation. *SIAM J. Comput.* 38, 4 (2008), 1602–1623.

[3] Elliot Anshelevich, Anirban Dasgupta, Eva Tardos, and Tom Wexler. 2003. Near-optimal network design with selfish agents. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*. ACM, 511–520.

[4] Juan Benet. 2014. Ipfs-content addressed, versioned, p2p file system. *arXiv preprint arXiv:1407.3561* (2014).

[5] Béla Bollobás. 2004. *Extremal graph theory*. Courier Corporation.

[6] Byung-Gon Chun, Rodrigo Fonseca, Ion Stoica, and John Kubiatowicz. 2004. Characterizing selfishly constructed overlay routing networks. In *IEEE INFOCOM 2004*, Vol. 2. IEEE, 1329–1339.

[7] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W Hong. 2001. Freenet: A distributed anonymous information storage and retrieval system. In *Designing Privacy Enhancing Technologies*. Springer, 46–66.

[8] Jacomo Corbo and David Parkes. 2005. The price of selfish behavior in bilateral network formation. In *Proceedings of the twenty-fourth annual ACM symposium on Principles of distributed computing*. ACM, 99–107.

[9] Christian Decker and Roger Wattenhofer. 2013. Information propagation in the bitcoin network. In *Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on*. IEEE, 1–10.

[10] Anthony Dekker, Hebert Pérez-Rosés, Guillermo Pineda-Villavicencio, and Paul Watters. 2012. The maximum degree & diameter-bounded subgraph and its applications. *Journal of Mathematical Modelling and Algorithms* 11, 3 (2012), 249–268.

[11] Paul Erdős, Alfréd Rényi, and Vera T. Sós. 1962. On a problem in the theory of graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences* 7 (1962), 215–235.

[12] Alex Fabrikant, Ankur Luthra, Elitza Maneva, Christos H. Papadimitriou, and Scott Shenker. 2003. On a network creation game. In *Proceedings of the twenty-second annual symposium on Principles of distributed computing*. ACM, 347–351.

[13] Matthew O. Jackson. 2005. A survey of network formation models: stability and efficiency. *Group Formation in Economics: Networks, Clubs, and Coalitions* (2005), 11–49.

[14] Matthew O. Jackson. 2010. *Social and economic networks*. Princeton university press.

[15] Mina Kamel, Caterina Scoglio, and Todd Easton. 2007. Optimal topology design for overlay networks. In *International Conference on Research in Networking*. Springer, 714–725.

[16] Henning Kopp, David Mödinger, Franz J Hauck, and Frank Kargl. 2019. Cryptographic Design of PriCloud, a Privacy-preserving Decentralized Storage with Remuneration. *IEEE Transactions on Dependable and Secure Computing* (2019).

[17] David Mödinger, Henning Kopp, Frank Kargl, and Franz J. Hauck. 2018. A Flexible Network Approach to Privacy of Blockchain Transactions. In *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 1486–1491.

[18] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V Vazirani. 2007. *Algorithmic game theory*. Cambridge university press.

[19] H. Zhang, A. Goel, and R. Govindan. 2002. Using the small-world model to improve Freenet performance. In *Proceedings.Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 3. 1228–1237 vol.3. https://doi.org/10.1109/INFCOM.2002.1019373