# A Contrasting Look at Network Formation Models and Their Application to the Minimum Spanning Tree*

Deanne B. McPherson
United States Navy
deanne.mcpherson@navy.mil

David L. Alderson
Naval Postgraduate School
Operations Research Department
Monterey, CA, U.S.A.
dlalders@nps.edu

## Abstract

*We review network formation models, contrast their behavior, and conduct numerical experiments to investigate the structural features of the networks they generate. We focus primarily on problems related to minimum spanning trees and consider the cost of selfish behavior, more commonly known as the price of anarchy, in network formation. We also explore differences between local, decentralized methods for network formation and their global, centralized counterparts.*

## 1. Introduction

Networks are prevalent in man-made and natural systems throughout the world. Despite recent efforts to characterize and catalog networks of all kinds (e.g., [24, 25]), considerably less is known about the forces that drive network formation. More specifically, significant effort has been devoted to studying the "what" of these networks, but much less is known about the "how" and the "why." A fundamental question is whether networks form to achieve some overarching global objective, or if network structure is just a byproduct of local, selfish, or even random decisions, or a combination of all these. It is also not clear how differences between global, centralized network formation and local, decentralized formation affect the properties of the resulting network.

Understanding the forces that drive network formation is becoming increasingly important. Of particular interest is the Internet. This is an extremely complex network that has managed to evolve and grow at an amazing pace. To some researchers, the Internet exemplifies a system that has self-organized. They argue that the network was not built by a central designer, but arose rather as a result of the localized actions of the users and service providers. In spite of its ad-hoc construction, the Internet is still relatively robust [34]. The ability to model such a complex network and to understand its underlying properties is relevant for the study of both man-made and natural systems.

Research in the "how" of network formation has ranged from random graph generation to system design. Erdös and Rényi pioneered the exploration of random graphs models [11], which generated interest in graph and network theory. More recently, the study of network science has focused attention on "small-world networks" and "scale-free networks" (see [3] for a review).

In this paper, we review some of the recent models used for network formation and conduct numerical experiments to contrast the structural features of the graphs they generate. We focus on problems related to the formation of minimum spanning trees and consider the cost of selfish behavior, more commonly known as the "price of anarchy", in network formation. We then contrast some of the local, decentralized methods for network formation to the global, centralized methods. Our results help clarify the tensions in network formation problems for both man-made and natural systems.

## 2. Network Formation Models

A *graph*, $G = (N, A)$ consists of a set $N$ of nodes and a set $A$ of arcs. The number of nodes is $n = |N|$ and the number of arcs is $m = |A|$. An arc from node $i$ to node $j$ is denoted as $(i, j)$ where $i, j \in N$. If $G$ is a directed graph then $(i, j) \neq (j, i)$, but if $G$ is an undirected graph then $(i, j) = (j, i)$.

A subgraph of $G$ is a graph $G'(N', A')$ if $N' \subseteq N$ and $A' \subseteq A$. It is a spanning subgraph of $G$ if $N' = N$. A *tree* is a connected graph that contains no cycles. A *spanning*

*This paper highlights the Masters thesis work presented in [23]. This work was supported by Office of Naval Research MURI Award # N0001408WR20242.

*tree* of $G$ is a tree that is a spanning subgraph of $G$ and has exactly $n-1$ arcs.

A *cut* is a partition of node set $N$ into two parts, $K$ and $\bar{K}$. A cut defines the set of arcs that have one node in $K$ and the other in $\bar{K}$. The *degree* of node $i$, denoted $deg_i$, is the total number of arcs connected to node $i$.

A *network* is a graph with additional data attributes. For example, we use $c_{ij}$ to denote the *cost* of arc $(i, j)$. We measure the *total cost* of a network as $\sum_{(i,j)\in A} c_{ij}$.

In many of the problems that we consider, we associate each abstract node $i$ with a location $x^i = (x^i_1, x_2,^i, \ldots, x^i_d)$ in the $d$-dimensional Euclidean space $\Re^d$. In such cases, the cost of arc $(i, j)$ is simply the Euclidean distance, $c_{ij} = \sqrt{\sum_{k=1}^{d}(x^i_k - x^j_k)^2}$.

A *minimum spanning tree (MST)* is a spanning tree having minimum network cost.

## 2.1. Classical Random Graph Models

The modern treatment of networks was forged by Paul Erdös and Alfréd Rényi [11], who examined a class of random graphs denoted $G(n, p)$. In this construction, there are $n$ nodes and each node has an independent probability $p$ of connecting to each other node in the graph. By altering the parameter $p$, the measurable properties of the connectedness of random graphs change quite suddenly (see [7] for an in depth review). For small values of $p$, the graph demonstrates low connectivity with several isolated nodes. Interestingly though, as $p$ approaches $1/n$, a majority of nodes form a cluster and the graph becomes almost completely connected. For values $p \approx 1$, the graph becomes highly connected with several cycles. This phenomenon is known as the "emergence of the giant component."

Another important property is the distribution of the node degrees. The degree of node $i$, $deg_i$, follows a binomial ($n$-1, $p$) distribution. For large values of $n$, this distribution can be approximated with the Poisson distribution [2].

## 2.2. Random Geometric Graph Models

Another approach to generating random graphs builds on the notion of a *geometric graph.* Given a set of nodes $N$ indexed $i = 1, 2, \ldots, n$, having locations $\{x^1, x^2, \ldots, x^n\}$, and a positive parameter $r$, the geometric graph $G(N, r)$ is the undirected graph induced by all arcs $(i, j)$ having distance $c_{ij} \leq r$. When the locations $\{x^1, x^2, \ldots, x^n\}$ are the result of an independent and identically distributed (IID) random process, the resulting graph is called a *random geometric graph.*

Most of the theoretic results for random geometric graphs are cumbersome, especially in higher dimensions (see [26], for an in-depth treatment). Unlike classical Erdös-Rényi graphs in which the presence of arcs is independent, the role of proximity in random geometric graphs makes the appearance of (nearby) arcs dependent. However, these graphs share remarkably similar behavior in the emergence of the giant component (see [17], and references therein).

Random geometric graphs are often used in classification problems in statistics. For example, suppose that individuals have $d$ characteristics and each can be represented by a continuous variable. Using an appropriately defined measure of distance in this $d$-dimensional space, one can classify two individuals as being "similar" if their distance is less than some constant parameter $r$. This makes it possible to identify clusters of similar individuals, which can be useful in many practical applications.

## 2.3. Preferential Attachment Models

Unlike the graphs produced in the Erdös-Rényi model, the degree distribution of many real world networks does not follow a Poisson distribution. Albert and Barabási [2] observe that many networks have a skewed distribution, in which the majority of nodes have small degrees while very few nodes have high degrees. The connectivity of these networks can be characterized by a *power law distribution*, in which the probability that a node has a degree distribution $k$ is $P(k) \approx k^{-\gamma}$, where typically $2 \leq \gamma \leq 3$.

Research in a multitude of disciplines has demonstrated power law distributions within networks. Price [27] demonstrates that the network of bibliographic citations in scientific publications has a connectivity distribution whose tail is "heavier" than an exponential distribution. West [33] argues that several characteristics of biological systems, such as metabolic rate, sizes and time scales can be modeled with a power law for several species. Faloutsos, Faloutsos and Faloutsos [14] argue that power laws could be used to predict characteristics of the Internet topology. In finance and economics, Gabaix [15] provides a good summary of power law distributions exhibited in a variety of areas such as firm size, city size, and the distribution of income and wealth.

Because the random graph model does not produce the power law distribution of node degrees as observed in real-world networks, Barabsi and Albert [4] present an alternative model based on *preferential attachment.* This model explicitly incorporates network growth, and it assumes that newly added nodes are more likely to attach to nodes with high connectivity. The probability $p$ that the new node will attach to node $i$ depends on the connectivity of node $i$, specifically in proportion to the ratio of node $i$'s degree to the sum of the degrees of all other nodes, such that $p \propto deg_i / \sum_j deg_j$. This model of network formation produces a *scale-free network*, a graph whose resulting node degree distribution follows a power law. For some systems, the scale-free network produced by the model is more similar in its connectivity than a graph generated from the random graph model.

## 2.4. Optimization-Based Models

Carlson and Doyle [8] propose a different mechanism, called *highly optimized tolerance (HOT)*, that produces power law distributions. They suggest that complex networks are optimized for robust performance and that the observed power law distributions are a result of the tradeoffs that must be made due to system constraints. Key features of their HOT model include "(1) high efficiency, performance and robustness to designed-for uncertainties; (2) hypersensitivity to design flaws and unanticipated perturbations; (3) nongeneric, specialized, structured configurations; and (4) power laws" [8].

Fabrikant, Koutsoupias and Papadimitriou [12] suggest a simple model, which we will refer to as the *FKP model*, for network formation that is based on the tradeoff concept present in the HOT model. Like the Barabási-Albert model, they grow a network one node at a time, but they also give each node a location in the unit square. When deciding to which node in the network the new node should attach, they propose two logical considerations. First, they assume the node would want to minimize its connection "cost" (represented by the Euclidian distance between itself and the node it attaches to). Second, the node would desire to connect to one that is more centrally located. These objectives can be weighted in order to alter the relative importance between the two. Specifically in their model, node $i$ will attach to node $j$ according to:

$$\min_{j:j<i} \alpha \, c_{ij} + h_j \tag{1}$$

where $c_{ij}$ is the Euclidean distance between nodes $i$ and $j$ and $h_j$ is a measure of centrality for node $j$. The weighting factor, $\alpha \geq 0$, is usually defined as a function of the final number of nodes $n$. The centrality $h$ can be defined in several ways, such as the average number of hops to all other nodes, the average Euclidian distance to all other nodes or the distance to some predefined central node [12].

Fabrikant et al. [12] show that by varying the value of $\alpha$, graphs with very different properties result. They prove that when centrality is measured as the number of hops to a defined node, $n_0$, then for $\alpha < 1/\sqrt{2}$, distance is relatively insignificant compared to centrality, and the resultant network is a star with the center at $n_0$. As $\alpha$ approaches $\sqrt{n}$, there is a closer trade-off between distance and centrality, and the node degree distribution can be represented by a power law. (Berger et al. [5] later argue that the resulting distribution is not a strict power law, but has an exponential cutoff.) Once $\alpha$ exceeds $\sqrt{n}$, distance becomes the overriding factor, and a form of a Euclidean spanning tree results.

The FKP model introduces a novel idea for network formation. Unlike the Erdös-Rényi graphs that are entirely based on a random selection of arcs, this model suggests a highly organized, locally optimized model that still produces a power law in the node degree distribution.

## 2.5. Game Theoretic Models

Fabrikant et al. [13] propose a network formation game to explore how an undirected network created from selfish-acting nodes would affect the network performance as a whole. The game is as follows: There are $n$ players, each representing a node in the network. The entire set of players is $N$, with $|N| = n$. Each player $i \in 1, 2, \dots, n$ chooses a strategy set $s_i = \{s_{i1}, s_{i2}, \dots, s_{ij}, \dots, s_{in}\}$, which defines the network edges to build from $i$ to other nodes $j$. The set $s = \{s_1, s_2, \dots, s_n\}$ denotes the collective strategy of all players.

Let $A(s)$ be the set of arcs resulting from strategy $s$. Therefore, $A(s) = \{(i,j) : i \neq j, s_{ij} = 1 \text{ or } s_{ji} = 1\}$ and $G(s) = (N, A(s))$ is the undirected graph that results from strategy $s$. Once a strategy is chosen, each player $i \in 1, 2, \dots, n$ incurs a cost $c_i(s) = \alpha \cdot |s_i| + \sum_{j \in N} d_{(i,j)}(G(s))$ where $\alpha$ is the fixed cost of forming a single connection between two players, and $d_{(i,j)}(G(s))$ is the distance, measured in hop count, between nodes $i$ and $j$ in the resulting graph $G(s)$. If no path exists between $i$ and $j$, then $d_{(i,j)}(G(s)) = \infty$. This is called the *Unilateral Connection Game (UCG)* because each node is able to use an arc, regardless of who paid for it.

An extension of this game is the *Bilateral Connection Game (BCG)* described by Corbo and Parks [10]. The major difference from the UCG is that in the BCG an arc only forms if both node strategies contain that arc. So in this game $A(s) = \{(i,j) : i \neq j, s_{ij} = 1 \text{ and } s_{ji} = 1\}$ and any connection cost is incurred equally by the two nodes.

In both cases, the interest is in the network that forms from selfish node decisions. In this context, the *Nash equilibrium* is a strategy $s$ that satisfies $c_i(s) = c_i(s_i, s_{Ni}) \leq c_i(s'_i, s_{Ni}), \forall i \in N, s_i \in S_i$. In other words, at the Nash equilibrium, no node has incentive to change its strategy. The *social cost* of the network is defined as:

$$\begin{aligned} C(G(s)) &= \sum_{i \in N} c_i(s) \\ &= \begin{cases} \alpha \cdot |A(s)| + \sum_{i,j \in N} d_{(i,j)}(G(s)) & \text{(UCG)} \\ 2\alpha \cdot |A(s)| + \sum_{i,j \in N} d_{(i,j)}(G(s)) & \text{(BCG).} \end{cases} \end{aligned}$$

The term *price of anarchy* is the ratio of the social cost of the worst-case Nash equilibrium and the social optimum [20, 29] and is used to measure the lack of coordination when the nodes act selfishly.

Fabrikant et al. [13] show that the results of the UCG vary based on the value of the parameter $\alpha$ . When $\alpha < 1$, the social optimum is a complete graph, and this is the only Nash equilibrium. When $1 \leq \alpha < 2$, the complete graph still results in a Nash equilibrium, but it is no longer unique. The worst Nash equilibrium is the star, leading to a price of anarchy of $C(star)/C(complete\ graph) \leq 4/3$. When $\alpha \geq 2$, the social optimum is a star, although there can be worse Nash equilibria.

These game theoretic models focus on different properties of the networks formed from the UCG and BCG than the previously reviewed models. Similar to the HOT model, the connection cost is associated with Euclidian distance and the number of arcs in the network and by tuning the weighting factor $\alpha$, networks ranging from the complete graph to a star can be produced. However, unlike the previous models, these concentrate on quantifying the cost associated with selfish behavior to compare it to the social optimum. Also, the BCG introduces a unique feature of restricting the arcs in network to those formed through the agreement of the two nodes. Both models provide an interesting way of looking at network formation.

### 2.6. Discussion

The key insight of Fabrikant et al. [12] is that the power laws observed in the structure of many man-made and natural systems can result from design tradeoffs that can be captured in simple optimization models. Their model was inspired by tensions perceived in the Internet—a desire to minimize the cost of connecting while also wanting to have low delay (i.e., be central) when communicating. But their model reflects a local, myopic decision process. It is unclear how, if at all, this local process relates to the global behavior of the network.

The price of anarchy in the study of network formation games explicitly addresses the difference between the social optimum for some system (as achieved, for example, by a central decision maker) with the aggregate outcome of local agents. The focus in [12] is on the global connectivity properties (e.g., degree distributions) of the graph, but is there an interpretation for a system-wide objective?

In the case of large $\alpha$ the Eq. (1) objective emphasizes only the local connection cost, and it is possible to interpret the collective behavior as trying to minimize the distance of the resulting tree, albeit in a heuristic manner. With this in mind, we now consider the classic minimum spanning tree problem.

## 3. The Minimum Spanning Tree

A classic case in the study of networks is the minimum spanning tree (MST) with the first algorithm for finding a MST published by Otakar Borüvka in 1926 (see [18], for history of MST). Minimum spanning trees have several practical applications such as the minimum amount of wire to connect several electrical components, or the minimum amount of piping required to connect houses in a neighborhood to a water system.

### 3.1. Finding a MST

Finding a MST for a network $G(N, A)$ can be formulated as a global optimization problem. This formulation is an integer linear program, where $H \subseteq N$ and the arc set for $H$, $A(H) \subseteq A$.

**Indices**

| | |
|---|---|
| $i \in N$ | node ($i = 1, 2, \ldots, n$) (alias $j$) |
| $(i, j) \in A$ | undirected arc between node $i$ and node $j$ |

**Data**

| | |
|---|---|
| $c_{ij} \in A$ | cost of arc $(i, j)$ |

**Decision Variable**

| | |
|---|---|
| $Z_{ij} \in \{0, 1\}$ | indicates if arc $(i, j)$ is in tree |

**Formulation**

$$\min_{Z} \quad \sum_{(i,j)\in A} c_{ij} Z_{ij} \tag{2}$$

$$s.t. \quad \sum_{(i,j)\in A} Z_{ij} = n - 1 \tag{3}$$

$$\sum_{(i,j)\in A(H)} Z_{ij} \leq |H| - 1, \forall \text{ sets } H \subseteq N \tag{4}$$

$$Z_{ij} \in \{0, 1\} \tag{5}$$

The objective function (2) sums the costs of the arcs chosen. Eq. (3) is a cardinality constraint that ensures that exactly $n-1$ arcs are selected, while Eq. (4) ensures that there are no resulting cycles. Although this problem is simple to formulate, solving it with linear programming is nontrivial for large $n$. The number of sets $H \subseteq N$ grows exponentially with $n$, so the total number of constraints arising from Eq. (4) becomes exponential, making the problem increasingly difficult to solve as $n$ grows.

Fortunately, the MST has a special tree structure that allows for customized algorithms that can solve it efficiently.

**Kruskal's algorithm** [21] makes use of a necessary and sufficient condition known as the *cut optimality condition*, which states:

> A spanning tree $T$ is a MST if and only if for every tree arc $(i, j) \in T$, $c_{ij} \leq c_{kl}$ for every arc $(k, l)$ contained in the cut formed by deleting arc $(i, j)$ from $T$.

Kruskal's algorithm uses this condition to build the MST one arc at a time. The algorithm initializes by sorting all the arcs in increasing order of cost, and it then iterates over the list to add new arcs to the tree. This algorithm, as presented, requires $O(m \log n)$ time to sort the arcs and $O(nm)$ time to detect a cycle, although Ahuja et al. [1] provides a more efficient algorithm that operates in $O(m + n \log n)$ time.

**Prim's algorithm** [28] is based on the *path optimality condition*, which states:

> A spanning tree $T$ is a MST if and only if for every nontree arc $(k, l) \in A$ of $G$, $c_{ij} \leq c_{kl}$ for every arc $(i, j)$ contained in the path in $T$ connecting nodes $k$ and $l$.

This algorithm initiates with a cut, in which an arbitrary start node of the network is in subset $K$, while the remainder of the nodes are in subset $\bar{K}$. The minimum-weight arc from the start node is then added to the list of MST arcs, and the node at the other end of that arc is removed from $\bar{K}$ and placed in $K$, creating a new cut. This method continues to create cuts between the two subsets until all nodes have been placed into $K$, and the resulting MST list will contain $n - 1$ arcs. Prim's algorithm requires $O(mn)$ time because of the work required to search for the minimum arc in the cut. Ahuja et al. [1] also present a more efficient data structure that can reduce the time to $O(m + n \log n)$.

**Decentralized "GHS" Algorithm.** Gallager, Humblet and Spira [16] present a distributed algorithm that also uses the path optimality condition to solve this problem in an undirected network. Their algorithm relies on a node's localized information and its ability to receive, process and send "messages." The algorithm works by combining separate graph "fragments" together into a final MST. Initially each node is its own fragment. Nodes pass messages to one another along their minimum-weight edges and then combine with other nodes to create new fragments, finally combining into a final fragment containing the MST. By passing messages, each node eventually discovers which of its arcs are in the MST. See [23] for a detailed description. The complexity of this algorithm is therefore measured by the number of messages that are passed, which is at most $2m + 5n \log 2n$.

### 3.2. Why Study the MST?

Finding a MST is a simple problem whose objective corresponds to a social optimum (i.e., the minimum cost for the entire network). This is simple to state and solve as a global optimization problem. This problem can also be solved by incremental graph formation, specifically by adding one node at a time, as in Kruskal's algorithm. Moreover, a simple myopic FKP-style network construction can yield an optimal solution *if nodes are added in the right order*. However, finding a MST using only local, decentralized, and asynchronous activities is considerably more complicated. The GHS algorithm shows that individual nodes, making local decisions, can solve this problem correctly and efficiently.

These features make finding the MST an ideal problem with which to experiment on the interaction of local, possibly selfish decisions of nodes during graph formation.

## 4. Numerical Experiments

In the sequential selection models considered here, the network grows incrementally by the addition of one node and arc at a time. A basic question is: *What is the role of precedence and node ordering in these network formation models?*

In the classical Erdös-Renyí random graph and the random geometric graph, nodes "arrive" at the same time. Similarly, the UCG and BCG network formation games are simultaneous-play games. For these models, node precedence plays no role in their structure.

In preferential attachment models, nodes are added to the network one at a time. Nodes that arrive "earlier" have an advantage in obtaining more connections—indeed the high-connectivity "hubs" tend to be the "oldest" nodes in the network [2]. However, the nodes are essentially interchangeable, so if the order of the nodes is different, their relative placement within the network would change, but the statistical properties of the global network would remain the same, including the power law distribution of the node degrees.

In models where network growth is incremental and where nodes have unique locations, precedence plays a role. In the FKP model, because an arriving node can only attach to a preceding node, changing that arrival order can lead to different (and possibly improved) networks. We seek to explore the extent to which this kind of precedence plays a role in the overall cost of the network.

### 4.1. Reordering Experiments

We generate an FKP-style network of $n = 100$ nodes, each having a location in the unit square, and where each node has the local objective function $\min_{j:j<i} \alpha\, c_{ij}$, with $\alpha = 1$ (see Fig. 1a). We then compute the total network cost. This is essentially a greedy heuristic applied to the problem of finding a MST.

We then generate other networks using the same nodes, but where the order of node "arrivals" is different. We consider two variations. The first uses the same node sequence, but chooses a different start node (Fig. 1b). The second completely randomizes the order of the nodes (Fig. 1c). In both cases, the network that forms is different than the initial network. Although the nodes have the same locations,

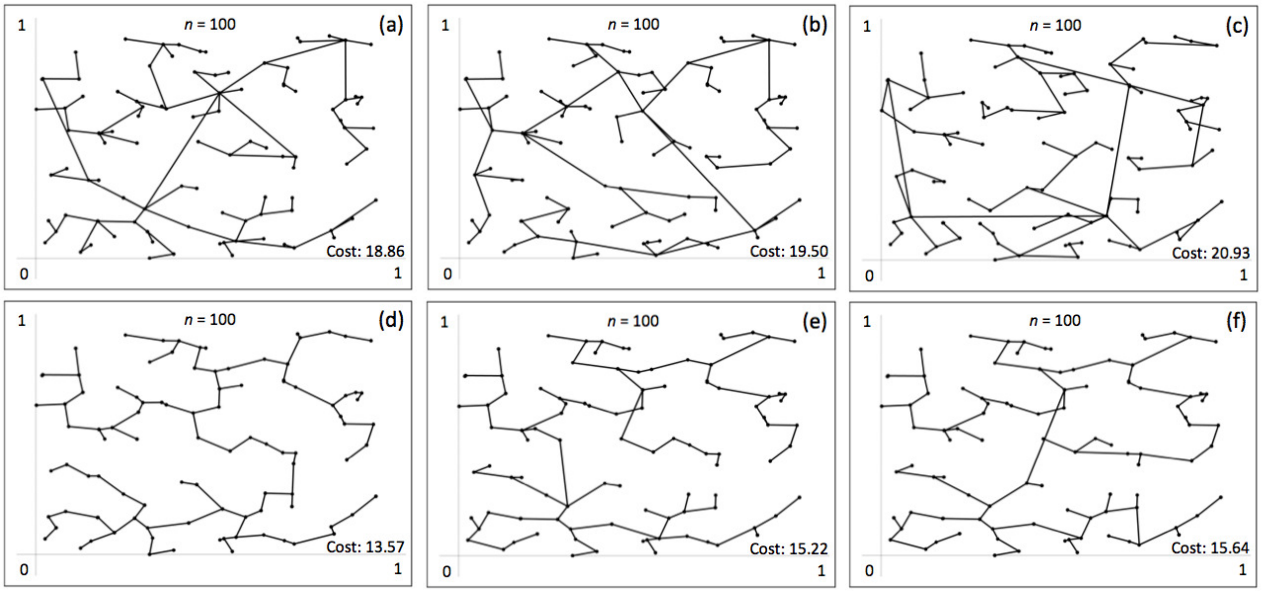


**Figure 1. Spanning trees of size $\mathbf{n = 100}$. (a) The initial graph generated from FKP-style construction. (b) FKP-style construction for the same nodes in the same sequence, but with a different start point. (c) FKP-style construction for the same nodes but in a different sequence. (d) The minimum spanning tree for these nodes. (e) Equilibrium network based on rewiring the graph based on the original node sequence. (f) Equilibrium network based on rewiring the graph using arbitrary node sequence.**

and the result is always a spanning tree, the arcs can differ greatly. We compare this result to MST for these nodes (Fig. 1d).

### 4.2. Rewiring Experiments

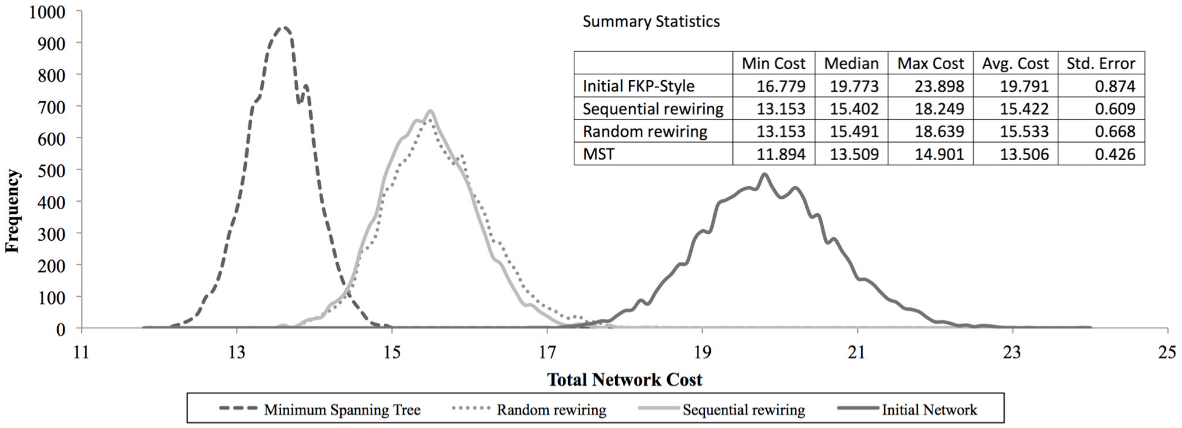
In a second numerical experiment, we examine the effects of allowing the nodes to change their connections, a process that we call *rewiring*. Once all nodes have been added to the network, we allow each the opportunity to change its initial connection and thereby (possibly) improve its local cost in Eq. (1). We continue this until none of the nodes in the network can benefit from connecting to a different node, at which point we say the network is *in equilibrium*. We then compare the cost of the rewired network with that of the original graph and the MST.

We generate networks of $n = 100$ nodes as described above. Once all nodes are added, we permit the nodes to rewire, subject to two constraints: (1) a node is only able to rewire the arc it formed when it joined the network and not any of the arcs from other nodes that attached to it; and (2) the rewiring must preserve the connectivity of the entire network.

We alter, in two ways, the order that the nodes rewire. We first give the nodes the opportunity to rewire in the same sequence they arrived in the network (Fig. 1e). In the second case, we randomly select the order in which nodes can rewire (Fig. 1f).

We compare the cost of the equilibrium network to that of the MST. We repeat the rewiring experiments 10,000 times, generating the initial network, rewiring it to equilibrium, and then determining the MST. Fig. 2 shows the distribution of network costs associated with each type of network and compares them to the cost of the MST. For networks with $n = 100$, the initial network cost is 46.5% greater than the MST. However, the sequential rewiring process and the random rewiring process substantially improve the network cost, with the sequential rewiring providing a lower cost in 51.4% of cases; the improvement in total network cost had a 99% confidence interval of $[0.090, 0.132]$. These sequential and random rewiring schemes result in networks that are on average still 14.2% and 15.0% costlier than the MST, respectively. In 26% of the experiments, both the sequential rewiring and the random rewiring methods produce equilibrium networks with the same costs.

We observe that both rewiring schemes can improve the overall cost of the network. Moreover, sequential rewiring seems to perform slightly better. We speculate that this could be because sequential rewiring guarantees that every node gets an chance to rewire following each change in the



Summary Statistics

| | Min Cost | Median | Max Cost | Avg. Cost | Std. Error |
|---|---|---|---|---|---|
| Initial FKP-Style | 16.779 | 19.773 | 23.898 | 19.791 | 0.874 |
| Sequential rewiring | 13.153 | 15.402 | 18.249 | 15.422 | 0.609 |
| Random rewiring | 13.153 | 15.491 | 18.639 | 15.533 | 0.668 |
| MST | 11.894 | 13.509 | 14.901 | 13.506 | 0.426 |

**Figure 2. Distribution of costs for initial FKP-style network, rewired equilibrium networks, and MST for $n = 100$. We conduct 10,000 experiments where we (1) generate a FKP-style network, (2) conduct rewiring experiments using "sequential" and "randomized" heuristics, and (3) solve for the corresponding MST. The rewiring heuristics improve the cost from the initial construction, but typically fall short of the MST. Inset: summary statistics for each of these distributions.**

network. In contrast, under the random rewiring scheme, the interval between successive rewiring opportunities for each node will vary. In some cases, this could mean more rapid improvement, while in other cases this could mean that some nodes must wait a long time before they can improve their cost.

Both rewiring schemes fall short of the socially optimal MST solution, which is not surprising. Because all spanning trees for the same nodes have the same number of arcs, there exists *some* rewiring scheme that would allow us to convert any spanning tree into the MST, this scheme could require multiple rewiring operations in parallel, something that we have restricted here. Such rewiring operations would require coordination between the nodes, possibly at the global level.

These experiments focus on incremental network construction based on local, myopic decisions. The major difference between this model and the FKP model is that we remove the tradeoff aspect by not using the centrality term in the local objective function. We simplify the objective function so we can compare the results of the formed networks to the optimal network, the MST.

There is obviously room for more complex numerical experiments and deeper analyses. Heuristic rewiring improves the total network cost substantially, but does not achieve a MST. Developing heuristics based on more sophisticated rewiring schemes could lend insight into the information sharing and incentive requirements needed to solve this problem as a decentralized game.

## 5. Conclusions and Future Work

Understanding the drivers of network formation is nontrivial. In this paper, we focus on models of network formation with emphasis on both centralized and localized algorithms for finding the minimum spanning tree. We develop a local, heuristic model based on a FKP-style construction, which uses rewiring to produce networks in equilibrium. Although heuristic rewiring does not typically produce a MST, we know that for any spanning tree there exists some rewiring that will transform it into a MST. This leads to the question: *Is there an interpretation of the local, myopic decision process of the FKP-style construction that lends itself to an equivalent global optimization problem?* If the answer is yes, then the local and global methods would provide the optimal solution and the price of anarchy would be zero. This could have significant implications for the formation of real network systems when global information and central decision processes are not possible.

Is there evidence to suggest that such an interpretation is possible? Here, we appeal to the notion of *dual decomposition* in network optimization problems and note that there is a considerable literature in the use of duality arguments for the development of decentralized algorithms (see [6], for an in-depth treatment).

The Internet is an example where duality arguments have recently enhanced our understanding of network behavior. The Transmission Control Protocol (TCP) is fundamental to the operation of the Internet. It guarantees end-to-end delivery of data packets by recognizing and retransmitting packets that are lost, and it also controls the rate at which individual computers inject packets into the network. Like most of the protocols used in the Internet, TCP was developed in an ad hoc manner, based on engineering intuition and trial-and-error more than mathematical theory. To researchers in network science, the behavior of TCP seemed like a case of self-organization [31]. However, research over the last decade has shown that TCP and its complementary protocol Active Queue Management (or AQM, which runs in routers to manage the size of their limited buffers) work together as a primal-dual algorithm to solve a global resource allocation problem in a decentralized and asynchronous manner [19, 22]. This type of analysis is not only bringing greater understanding to the way that the existing Internet works [30], but it is also helping to influence the design of future network protocols [9].

While considerable work remains to understand the forces governing network behavior, it is clear that optimization is an important tool for exploring the tradeoffs at work in network formation. Identifying the precise mechanisms at work in specific applications, as well as how to improve them, will be a topic of future research.

## References

[1] R. K. Ahuja, T. L. Magnanti, J. B. Orlin. *Network Flows*. Prentice Hall, Inc., Upper Saddle River, NJ, 1993.

[2] R. Albert, A.-L. Barabási, A.-L. Statistical mechanics of complex networks. *Reviews of Modern Physics* 74:47–97, 2002.

[3] D.L. Alderson. Catching the 'network science' bug: insight and opportunity for the operations researcher. *Operations Research* 56: 1047–1065, 2008.

[4] A.-L. Barabási, R. Albert. Emergence of scaling in random networks. *Science* 286:509–512, 1999.

[5] N. Berger, B. Bollobás, C. Borgs, J. Chayes, O. Riordan. Degree distribution of the FKP network model. *Lecture Notes in Computer Science* 2719: 725–738, 2003.

[6] D.P. Bertsekas, J.N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods.* Athena Scientific, Belmont, MA, 1997.

[7] B. Bollobás. *Random Graphs*. Academic Press, London, 1985.

[8] J.M. Carlson, J. Doyle. Highly optimized tolerance: A mechanism for power laws in designed systems. *Physical Review E* 60(2):1412–1427, 1999.

[9] M. Chiang, S.H. Low, A.R. Calderbank, J.C. Doyle. Layering as optimization decomposition. *Proceedings of IEEE* 95:255–312, 2007.

[10] J. Corbo, D. Parkes. The price of selfish behavior in bilateral network formation. *ACM PODC '05*, Las Vegas, NV, 2005.

[11] P. Erdös, A. Rényi, A. On random graphs. *Publicationes Mathematicae* 6, 290–297, 1959.

[12] A. Fabrikant, E. Koutsoupieas, C.H. Papadimitriou. Heuristically optimized trade-offs: A new paradigm for power laws in the Internet. *Lecture Notes in Computer Science* 2380:110–122, 2002.

[13] A. Fabrikant, A. Luthra, E. Maneva, C.H. Papadimitriou, S. Shenker. On a network creation game. *ACM PODC '03.* Boston, MA, 2003.

[14] M. Faloutsos, P. Faloutsos, C. Faloutsos. On power-law relationships of the Internet topology. *ACM SIGCOMM '99.* Cambridge, MA, 1999.

[15] X. Gabaix. Power laws in economics and finances. *Annual Review of Economics* 1:255–294, 2009.

[16] R.G. Gallager, P.A. Humblet, P.M. Spira. A distributed algorithm for minimum-weight spanning trees. *ACM Transactions of Programming Languages and Systems* 5(1): 6677, 1983.

[17] A. Goel, S. Rai, B. Krishnamachari. Monotone properties of random geometric graphs have sharp thresholds. *Annals of Applied Probability*15(4):2535–2552, 2005.

[18] R.L. Graham, P. Hell. On the history of the minimum spanning tree problem. *Annals of the History of Computing* 7(1): 43–57, 1985.

[19] F.P. Kelly, A. Maulloo, D. Tan. Rate control in communication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society* 49:237–252, 1998.

[20] E. Koutsoupias, C.H. Papadimitriou. Worst-case equilibria. *Lecture Notes in Computer Science* 1563:404–413, 1999.

[21] J.B. Kruskal. On the shortest spanning tree of graph and the traveling salesman problem. *Proceedings of the American Mathematical Society* 7:48–50, 1956.

[22] S.H. Low. A duality model of TCP and queue management algorithms. *IEEE/ACM Transactions on Networking* 11(4):525–536, 2003.

[23] D.B. McPherson A Contrasting Look at Network Formation Models and Their Application to the Minimum Spanning Tree, M.S. Thesis in Operations Research, Naval Postgraduate School, Monterey, CA, U.S.A., September 2009.

[24] National Research Council. *Network Science*. Washington, D. C.: The National Acadamies Press, 2005.

[25] M.E. Newman, A.-L. Barabási, D.J. Watts. *The Structure and Dynamics of Networks*. Princeton University Press, Princeton, NJ, 2006.

[26] M. Penrose. *Random Geometric Graphs*. Oxford University Press, Oxford, UK, 2003.

[27] D.J. Price. Network of scientific papers. *Science* 149(3683):510–515, 1965.

[28] R.C. Prim. Shortest connection networks and some generalizations. *Bell System Technical Journal* 36:1389–1401, 1957.

[29] T. Roughgarden. *Selfish Routing and the Price of Anarchy*. MIT Press, Cambridge, MA, 2005.

[30] R. Srikant. *The Mathematics of Internet Congestion Control*. Birkhauser, Boston, MA, 2004.

[31] A. Veres, M. Boda. The chaotic nature of TCP congestion control. *Proceedings of IEEE INFOCOM*, Tel-Aviv, Israel, 2000.

[32] D.J. Watts, S. Strogatz. Collective dynamics of "small world" networks. *Nature* 393, 1998.

[33] G.B. West. The origin of universal scaling laws in biology. *Physica A* 263:104–113, 1999.

[34] W. Willinger, J.C. Doyle. Robustness and the Internet: Design and evolution. In E. Jen (Ed.), *Robust-Design: A Repertoire of Biological, Ecological, and Engineering Case Studies* (pp. 231-272). Oxford University Press, New York, NY, 2004.