2016

Steiner Tree Games

Samuel Rossin
Oberlin College

Steiner Tree Games

Sam Rossin
Advisor: Kevin Woods

May 24, 2016

Abstract

Prize-collecting Steiner tree is a network design problem in which a utility provider located at some position in a graph attempts to construct a network (subtree) of maximum profit based on the value of the vertices in the graph and the costs of the edges. I consider three network formation games where the players represent competing providers attempting to build networks in the same market. These games seek to preserve the key feature of Prize-Collecting Steiner tree, namely that players must each build a subtree that attempts to include customers who are of high value or are easy to reach. I analyze the price of anarchy and price of stability of each of these games.

1 Introduction

Network formation problems show up in a wide variety of contexts with a disparate set of goals. In some of these instances it would be unreasonable to expect a single central authority to have the power to enforce the formation of an optimal network. Therefore we often study what happens when a network is formed by a large group of selfish agents, and see how much worse the result is than a centrally enforced solution. To this end we consider Nash Equilibria – states in which no player can improve by unilaterally changing their strategy. In particular we consider the *Price of Anarchy* [6] and *Price of Stability* [8, 14], which are respectively the ratios of the values of the worst and best Nash Equilibria to the optimal value. It has been argued [14] that the Price of Anarchy is a good measure of the worst that could happen with no regulation, and the Price of Stability is a measure of what happens when some third party can suggest a solution to the selfish agents. There are three broad categories of network formation games that have been widely studied: facility location games [7, 9], in which providers attempt to place facilities to maximize the customers they can serve, social network formation games [12], in which individuals attempt to expand their own network reach, and classic network design games [8, 14, 15, 17], in which each player is attempting to connect a specific pair of points in a shared network.

One important network formation problem is Prize-Collecting Steiner Tree [2]. In this problem a single utility provider is attempting to create a network connecting customers to their service. Each potential user has an associated profit, and each edge in the network has a cost, so the provider must decide which users to include based on their location and value. It is natural to ask how this sort of problem would behave in a decentralized setting. However, none of the categories of network formation games really capture the key idea of this problem, namely that a decision must be made about whether to add a potential user to a network based on their value and how difficult it is to add them. With this in mind, I consider a setup in which there are multiple utility providers based at different points in the graph. I consider three variations of this setup based on how competing providers interact with each other.

2 Backgound

In this section I discuss important background information and related work. In particular, I describe Prize-Collecting Steiner tree in detail, give some background on algorithmic game theory, and explore some games that are structure related to the problem at hand or have features that are needed later.

2.1 Prize-Collecting Steiner Tree

Prize-collecting Steiner Tree (PCST) is a classic NP-complete network formation problem, first defined in this form by Bienstock et al. a[2]. An instance of PCST is a graph $G = (V, E)$ where each edge e has a cost c_e and each vertex v has an associate profit p_v. One vertex r is specially designated as the root. The goal of the problem is to form a subtree, or *Steiner Tree*, $T = (V_T, E_T)$ rooted at r that maximizes the total profit:

$$\sum_{v \in V_T} p_v - \sum_{e \in E_T} c_e$$

This problem is usually thought of in terms of a utility provider installing a network, where vertices represent potential customers and edges represent location for lines to be installed. The profit of a vertex is the value to the provider of having that vertex as a customer, and the cost of an edge represents how costly (or difficult) it is to install a line there. There is also another more general version of the problem, prize-collecting Steiner forest (PCSF), in which the vertices no longer have an associated profit, but instead there are n source sink pairs, each with an associated profit p_i, and we want to choose a subgraph that maximizes the sum of the profits of the pairs we connect, minus the edge costs. Clearly PCST is a special case of PCSF in which all sources are a single vertex r and every other vertex is a sink.

There are several important approximation results for this problem. The simplest is a deterministic rounding algorithm from the same paper that originally defined the problem [2]. The most used result is a general primal-dual algorithm due to Goemans and Williamson which gives a 2-approximation for PSCT [3]. Finally, there is a somewhat more recent result from Archer et al. that uses a modification of the Goemans-Williamson algorithm to give a (2-ε)-approximation [16].

2.2 Algorithmic Game Theory

Sometimes there is an optimization problem in which it is unreasonable to assume that we can just enact the optimal solution, as there are various selfish agents involved, each trying to maximize their own personal utility. In this case, it is often interesting to see how much worse the solutions arising from this selfish behavior can be. To do this, we will model the problem as a game. A game consists of n players, where each player i has a set of possible strategies A_i. We let $A = A_1 \times, ..., \times A_n$ be the set of possible strategy vectors. Given a strategy vector $S = (S_1, ..., S_n) \in A$, we designate the utility of player i by $u_i(S)$. Finally, we designate the social utility, the thing we are trying to optimize, as $u(S)$.

We say that a strategy vector S is a Pure Nash Equilibrium if no player can improve his utility by unilaterally deviating from S. That is, if we let S' be a strategy vector that differs only in the strategy of some player i (we write $S' = (S_1, ..., S'_i, ..., S_n)$ or $S' = (S'_i, S_{-i})$), then we will have $u_i(S) \geq u_i(S')$. Note that there is no guarantee that a Pure Nash Equilibrium exists. Similarly, we define a Strong Nash Equilibrium to be a solution in which there is no coalition of players $C \subseteq \{1, ..., n\}$ such that there is a solution $S' = (S'_C, S_{-C})$ in which only the members of C deviate from S, no player in C has worse utility in S', and at least one player in C has strictly better utility.

Finally, we use Nash Equilibria to determine the cost of selfishness. That is, we define the *Price of Anarchy* [6] to be the ratio of the social utility of the worst Nash Equilibrium to the social utility of the optimal solution. That is, $\frac{u(S)}{u(S^*)}$ where S^* is the solution that maximizes the social utility, and S is the Nash Equilibrium that minimizes the social utility. The Price of Anarchy represents the worst effect selfish players can possibly have. Similarly we define the *Price of Stability* [8, 14] to be the ratio of the *best* Nash Equilibrium to the optimal. This represents how bad it can be for players to be selfish, if we allow some third party to suggest a solution.

2.3 Congestion Games and Potential Games

One general game type that occurs repeatedly is the class of *congestion games*, first described by Rosenthal [1]. A congestion game consists of n players and a set of resources R. Each player has a finite set of strategies $A_i \subseteq 2^R$ where each strategy $S_i \in A_i$ is a subset of the resources. Given a strategy vector $S = (S_1, ..., S_n)$, let $x_r = |\{i : e \in S_i\}|$ be the number of players using resource r. Then each resource r has a cost $f_r(x_r)$ which is a function only of x_r. The cost to player i of a particular strategy vector S is $c_i(S) = \sum_{r \in S_i} f_r(x_r)$. Given a congestion game, we define Rosenthal's potential function Φ as follows:

$$\Phi(S) = \sum_{r \in R} \sum_{x=1}^{x_r} f_e(x)$$

Rosenthal showed [1] that if a single player deviates in strategy, the change in their personal cost will be reflected exactly in the potential function. That is, given a strategy vector $S' = (S_1, ..., S'_i, ..., S_n)$ that differs from S only in the strategy of player i, then we will have $\Phi(S) - \Phi(S') = c_i(S) - c_i(S')$. As a result, congestion games always have a pure Nash Equilibrium, as the strategy S^* that maximizes Φ necessarily has no improving deviations. Note that this result holds even if the cost functions f_r are negative, so it works for profit maximization games as well as cost minimization games. A related class of game, *potential games*, defined by Monderer and Shapley [5], arises when we just require a potential function. That is, a potential game is any game in which there exists a potential function Φ with the property (described above) that if 2 strategies differ in the strategy of player i, then the change in the personal utility of player i is reflected in Φ. Monderer and Shapley [5] showed that any potential game $\mathcal{P}$ is isomorphic to a congestion game, in the sense that there exists a congestion game whose strategy set is bijective with $\mathcal{P}$ in such a way that the corresponding strategy vectors have the same utility for all players. This isomorphism, however, is often not particularly useful in practice as it uses an exponential number of resources. Additionally, Monderer and Shapley [5] showed that the potential function Φ is unique up to an additive constant for any given potential game, and as a result it is referred to as *the* potential function.

It turns out that potential games and congestion games show up quite often when dealing with network formation. We will illustrate this with an example that is important as background for the work here. *Shapley network design games*, defined by Anshelevich et al. [14] consist of a graph with edge costs c_e. There are k players, each with a source-sink pair (s_i, t_i) that it wishes to connect using an $s_i - t_i$ path. The cost of each edge is divided up among the players that use it. That is, given a vector of strategies $(S_1, ..., S_k)$, where S_i is an $s_i - t_i$ path, then the cost to player i is:

$$C_i(S_1, ..., S_k) = \sum_{e \in S_i} \frac{c_e}{|\{j : e \in S_j\}|}$$

This game is a congestion game, since strategies are subsets of the edges, the cost functions are just the sum of the costs of the edges chosen, and the costs of the edges only depend on the number of players using that edge. Anshelevich et al. [14] use the potential function to show that the Price of Stability is at most H_k, the harmonic number in k, and they give an example to show that if edges are directed this bound is tight. Further, they argued that the Price of Stability is a more important number than the Price of Anarchy in the case of network design games, since there will often be a third party to suggest the best Nash Equilibrium to all players. In the case with undirected edges, Disser et al. were able to reduce the H_k bound on the Price of Stability slightly [18], and in the more restricted case of Undirected Broadcast Games (in which all source vertices are the same, and every other node is a sink) Bilò et al. [17] showed that the Price of Stability is constant.

2.4 Competitive Facility Location

In this section I describe another important related problem, *Competitive Facility Location*. This particular version was described by Adrian Vetta [7]. We are given a complete bipartite graph $G = (W \sqcup U, E)$, in which W represents a set of possible facility locations, and U represents a set of users or markets. Each market $u \in U$ has an associated value π_u, and each location $v \in W$ has a fixed cost c_v for which a facility can be built at location v. Each edge vu has an associated marginal cost λ_{vu} that represents the cost to service market u from location v. Finally there are n players that each get to place a single facility at some location in W (or they may choose the empty strategy of not placing their facility at all).

Given a strategy vector $S = (v_1, ..., v_n)$ representing choices of locations for the n players, we calculate player utilities as follows. Each player must pay the fixed cost of their location v_i. Then each market $u \in U$, is serviced by the player that can afford to bid the lowest, that is, the player i such that $\lambda_{v_i u}$ is minimized. Player i then achieves a value p_u from market u equal to the cost that the second lowest bidder would pay (since if they tried to charge more than that, the second lowest bidder could undercut them) , or π_u if there is no other bidder. They must then pay the marginal cost to service $\lambda_{v_i u}$, so they achieve a profit of $p_u - \lambda_{v_i u}$. The social utility is calculated as the sum of the players' personal utilities, plus $\sum_{u \in U}(\pi_u - p_u)$ which is the profit achieved by the users (not the players) due to competition between the players.

Vetta [7] showed that this game always has a Pure Nash Equilibrium, and that the price of anarchy is 2, up to an additive constant that depends on the fixed costs (so in the case where the fixed costs are 0, the price of anarchy is really 2).

2.5 Other Related Work

There are some other methods that have been used in the past to turn prize-collecting Steiner tree into a game. One approach, used for example by Gupta et al. [13] is to let the actual network be constructed by a central authority, and have the players be the customers, whose only power is to report what their utility for being connected to the network is. The game, then, is to design a cost-sharing method that incentives users to be willing to pay as close to their true value as possible. Another approach, considered by Kuipers et al. [4] is to consider a cooperative game in which players are trying to build a network together, and one must then find the best way to distribute the profits to incentivize players to build an optimal network.

An additional type of related game, mentioned in the introduction, is the social network formation type of game, a survey of which was done by Jackson [12]. In these games the players are vertices of a complete graph, and each player wishes to form a social network. Various techniques are used to model the formation of network links, for example, in the simplest model all players can declare who they would like links with at the same time and then all reciprocating pairs are granted links. Then there is a valuation function which decides how much the resulting network is worth, as well as an allocation rule which determines how the value is divided among the players.

Another problem with many features similar to our games is *connected facility location*. In this problem, as in the standard facility location problem, we want to place some number of facilities on a bipartite graph in order to maximize the value of customers served. Connected facility location adds an additional constraint that the facilities must be connected by a Steiner Tree. This leads to solutions that look fairly similar to our problem, though the bipartite graph causes the details to be quite different in the end. This problem has mostly been studied from the perspective of approximation algorithms (see for example Swamy and Kumar [11]), but Leonardi and Schäfer [10] also studied cost-sharing methods, as mentioned above in relation to prize-collecting Steiner tree.

3 Steiner Tree Games

Network formation games, as studied, generally fall into three categories. Facility location games focus on the perspective of the utility providers by using a bipartite graph in which they have to locate facilities. Once facilities are located, the value obtained is determined automatically. Traditional network formation games focus on individuals located at each vertex of a complete graph that wish to connect with each other. Selfish routing games focus on the users of a utility network that want to be connected and must figure out how to do it most cheaply. None of these approaches, however, capture the essence of the way a network is formed in Prize Collecting Steiner tree: that is, the idea that the network provider must decide which users it values enough to connect to the network, in a way that might depend on where they are in relation to each other.

With this in mind I consider games in which n players representing network providers are building networks on an undirected graph $G = (V, E)$. Each player i is rooted at some vertex $r_i \in V$ and wants to build a Steiner tree of the greatest value rooted at r_i. As in Prize Collecting Steiner Tree, each edge $e \in E$ has an associated cost c_e and each vertex $v \in V$ has an associated profit p_v. Simply applying the rules for Prize Collecting Steiner Tree to each player results in no interaction so we must consider ways to set the game up that result in interesting outcomes. In the following sections three games are analyzed that use different strategy sets and utility functions based on this premise.

Note that I will assume without loss of generality that each root node r_i has cost $c_{r_i} = 0$, and that no players share a root node – that is for all $i \neq j$ we have $r_i \neq r_j$. We can do this since if one of these assumptions is violated we can add a new vertex r'_i with 0 profit and an edge (r'_i, r_i) with 0 cost, and then move the root node for player i to r'_i. One can check that this doesn't affect the results that follow. Additionally, all of the following results still hold if we allow the edge costs to depend on the player (as is often done in competitive facility location games), so for each edge e we have a cost function $c_e(i)$ that depends on the player. However, necessary modifications to the proofs to make this work are very small, and not worth the additional notational complexity, so we will stick with symmetrical edge costs.

Finally, it should be noted that all of the following games are profit maximization games with utility functions that can be positive or negative. Since the empty set is always a valid solution, no optimal solution or Nash Equilibrium will have negative values for any utility function (this is not completely trivial, but follows directly from the definitions of the utility functions). However, it is possible for the optimal social utility of an instance to be 0, in which case we cannot calculate the Price of Anarchy or Price of Stability. Therefore, all PoA and PoS results below assume nonzero optimal value.

4 Equal Division of Shared Resources

This first game has the simplest basic idea. Players each construct their desired Steiner tree, and then must pay in full for each edge they used and get the full value of any vertex that they service alone. However, if multiple players choose the same vertex, they must split its value evenly between them. The idea behind this game is that vertices represent collections of customers (or some other dividable resource) and if multiple players service the same vertex, the customers will on average split their business between the providers.

More formally, let G be a graph with the properties described above. A strategy for player i is a subtree $S_i = (V_i, E_i)$ rooted at r_i. Given a vector of players' strategies $S = (S_1, ..., S_n)$

let $x_v = |\{j : x_v \in V_j\}|$ be the number of players j that use vertex v. Then the personal utility of player i is:

$$u_i(S) = \sum_{v \in V_i} \frac{p_v}{x_v} - \sum_{e \in E_i} c_e$$

We will consider the social utility to be just the sum of the individual utilities of all the players:

$$u(S) = \sum_{i=1}^{n} u_i(S) = \sum_{v : x_v \geq 1} p_v - \sum_{i=1}^{n} \sum_{e \in E_i} c_e$$

Theorem 4.1. *This game always has a Pure Nash Equilibrium.*

Proof. The game can be restated as a congestion game as follows. Lets $R = V \sqcup E$ be the set of resources, and let the value of a resource $r \in R$ be a function of the number of players using the resource, x_r:

$$v_r(x_r) = \begin{cases} \frac{p_r}{x_r} & \text{if } r \in V \\ -c_r & \text{if } r \in E \end{cases}$$

Then each strategy S_i for player i is a subset of R, and $u_i(s)$ is just the sum of the values of the resources used:

$$u_i(s) = \sum_{v \in V_i} \frac{p_v}{x_v} - \sum_{e \in E_i} c_e = \sum_{r \in S_i} v_r(x_r)$$

Thus our game is a congestion game, and therefore always has a Pure Nash Equillibrium [1]. □

Theorem 4.2. *The game, however, has arbitrarily large Price of Stability.*

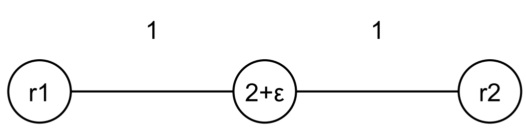

Figure 1: A Counterexample.

Proof. Observe the graph in Figure 1. Here the optimal solution is to have player 1 take the central node, for a social utility of $1 + \varepsilon$. However, the only Nash equillibrium has both players using this node, for a social utility of 2ε. So the PoS is $\frac{1+\varepsilon}{\varepsilon}$ which goes to infinity as ε goes to 0. □

It may be possible to modify this game slightly into a game with better properties by giving a bonus to the social utility based on the number of players serving a vertex. This bonus would reflect the value that users gain from having a choice of providers. The most natural way to do this would be to add $\frac{x_v - 1}{x_v} p_v$ to the social utility for each vertex v. This is natural as it is the difference between the value of v and the actual value achieved by each of the players serving v. If we add this term, the social utility becomes:

$$u(S) = \sum_{v:x_v \geq 1} p_v - \sum_{i=1}^{n} \sum_{e \in E_i} c_e + \sum_{v:x_v \geq 1} \frac{x_v - 1}{x_v} p_v = \sum_{v:x_v \geq 1} (2p_v - \frac{p_v}{x_v}) - \sum_{i=1}^{n} \sum_{e \in E_i} c_e$$

Conjecture 4.1. *The Price of Anarchy of this version is 2.*

5 Mafia

Is this second game, we simply modify what happens in the case where multiple players use the same vertex. Players still must pay the full cost of any edges they use, and still achieve the full reward of any vertices they service alone. However, instead of dividing up the cost of a shared vertex evenly, all players simply receive 0 reward for a shared vertex. This describes a case where competition has the potential to completely remove all value, and should therefore be avoided at all costs. We then add an additional component to the social utility (in a way that is commonly done in competitive facility location): since it is possible for there to be extra profit at a vertex unclaimed by the service providers, this profit is going the utility of the customers, and therefore should still be added to the social utility.

As before, we are given a graph $G = (V, E)$ where each edge $e \in E$ has an associated cost c_e and each vertex $v \in V$ has an associated profit p_v. Again we have n players with root nodes r_i whose strategies $S_i = (V_i, E_i)$ are subtrees rooted at r_i. Given a strategy set $S = (S_1, ..., S_n)$ let x_v be the number of players using v, as before, and let $X_i \subseteq V_i = \{v \in V_i : x_v = 1\}$ be the set of vertices only used by player i. Then the utility of player i is:

$$u_i(S) = \sum_{v \in X_i} p_v - \sum_{e \in E_i} c_e$$

Unlike in the previous game, it is now possible for some of the profit from some vertex to be awarded to none of the players. In this case, this excess profit is counted as value for the consumer, and therefore is still a part of the social utility. This means that any vertex that is serviced will have its full value included in the social utility function, so the social utility function isn't merely the sum of the utilities for all the players. Instead it is:

$$u(S) = \sum_{v:x_v \geq 2} p_v + \sum_{i=1}^{n} u_i(S) = \sum_{v:x_v \geq 1} p_v - \sum_{i=1}^{n} \sum_{e \in E_i} c_e$$

Note that this game still always has a Pure Nash Equillibrium, as the same proof can be applied.

Lemma 5.1. *$u(S)$ is an exact potential function for this game.*

Proof. As with the previous game, this game is a potential game if we let $R = V \sqcup E$ be the set of resources with value functions:

$$v_r(x_r) = \begin{cases} p_r & \text{if } r \in V, x_r = 1 \\ 0 & \text{if } r \in V, x_r > 1 \\ -c_r & \text{if } r \in E \end{cases}$$

Then Rosenthal's potential function is:

$$\sum_{r\in R}\sum_{x=1}^{x_r} v_r(x) = \sum_{v:x_v\geq 1} p_v - \sum_{e\in E}\sum_{i=1}^{x_e} c_e$$

But the second sum here counts each edge used by each player exactly once, so it can be rearranged into:

$$\sum_{v:x_v\geq 1} p_v - \sum_{i=1}^{n}\sum_{e\in E_i} c_e$$

Which is exactly $u(S)$.

□

Theorem 5.2. *This game has Price of Stability 1.*

Proof. Let S^* be a social optimal, that is let S^* be a strategy set that maximizes $u(S)$. Then, since $u(S)$ is an exact potential function by Lemma 5.1, S^* is also a Pure Nash Equillibrium. Therefore, the social optimal is always a Pure Nash Equilibrium, so we have a price of stability of 1. □

Recall that there is a stronger type of equilibrium than a Pure Nash Equilibrium, namely a Strong Nash Equilibrium. A strategy set S is a Strong Nash Equilibrium if there is no coalition C such that there is some deviation (S'_C, S_{-C}) in which only the members of C deviate, such that no member of C has decreased utility, and at least one member has strictly increased utility. Clearly this is a much stabler situation, so it is desirable if possible.

Theorem 5.3. *There is always a Strong Nash Equilibrium and the price of stability is still 1 among Strong Nash Equilibria.*

Proof. Let S^* be an optimal strategy set. Then, without loss of generality, we can assume that no vertex is used by more than 1 player in S^*. To see why, suppose we have an optimal solution S in which there is some vertex v that is used by multiple players. Then we can modify S as follows. We choose some player i such that $v \in V_i$ and remove v from all other players' strategies. If this results in some player j having a disconnected graph, we then add all vertices and edges from from S_j that are no longer connected to r_j to S_i (unless the addition of an edge creates a cycle in S_i, in which case we just ignore that edge). We then repeat this process until no vertex is used by multiple players. Note that this process can only increase the social utility: no vertices are removed from S, and no edges are added.

Now observe that for any strategy S we have $\sum_{i=1}^{n} u_i(S) \leq \sum_{v:x_v\geq 2} p_v - \sum_{i=1}^{n} u_i(S) = u(S)$. Note further that if $x_v \leq 1$ for all $v \in V$ then we have equality. Therefore, we have equality in the case of S^*. Now suppose there is some coalition C with some deviation S'_C such that no player in C is worse off as a result of the deviation, and at least one player is strictly better off. In this case, we have:

$$\sum_{i \in C} u_i(S^*) \leq \sum_{i \in C} u_i(S'_C, S^*_{-C})$$

But since the sum of the personal utilities in S^* is equal to the social utility, the sum of the social utilities in (S'_C, S^*_{-C}) cannot be greater than it is in S^* (since this is a lower bound for the social value, and S^* is optimal). Therefore some other player must have his social value decreased to compensate. But the only way for a non-deviating player to have his utility decrease is if someone takes one of his vertices. This decrease, however, does not effect the social utility (since the vertex is still used), so the increase in personal utility in C will result in an increase in social utility. That is $u(S'_C, S^*_{-C}) > u(S^*)$, which contradicts the fact that S^* is optimal. So S^* is a strong Nash equilibrium, which proves the claim. □

Theorem 5.4. *This game has arbitrarily large Price of Anarchy, even among Strong Nash Equillibria.*

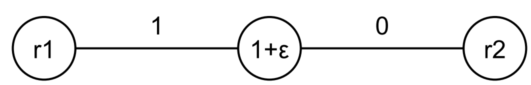

Figure 2: A Counterexample.

Proof. Observe the graph in Figure 2. The optimal solution here is for player 2 to take the middle vertex, for a total social value of $1+\varepsilon$. This is, of course, a Nash Equilibrium (in fact it is a strong Nash). However, there is another strong Nash Equilibrium, in which player 1 takes the middle vertex for a social value of ε. In this case no player has an improving deviation, and additionally player 1 will not be willing to deviate to optimal in a coalition with player 2, since he loses his current personal utility of ε. So the price of anarchy in this case is $\frac{1+\varepsilon}{\varepsilon}$, which is unbounded.

□

6 Competition

In this third game each player will again pick a starting subtree rooted at their root node. However players will then try to connect any nodes directly adjacent to their starting tree. Each node will only be serviced by one player, the player that can afford to charge the least for it (based on the cost of the connecting edge, or 0 if it is in the starting tree), and the value they can get from the node will be the price of the second lowest bidder.

More precisely, we again have a graph $G = (V, E)$ where each edge $e \in E$ has an associated cost c_e and each vertex $v \in V$ has an associated profit p_v. Again we have n players with root nodes r_i. Player i picks a base tree, $S_i = (V_i, E_i)$ rooted at r_i. Now consider some strategy $S = (S_1, ..., S_n)$. For each vertex v, let B_v be the set of players i such that $v \in V_i$ or v is adjacent to some $w \in V_i$. For each player $i \in B_v$, let $\gamma_{i,v}$ be the value of the cheapest edge (call it $e_{i,v}$) connecting v to some vertex in V_i, or 0 if $v \in V_i$. Let $B^*_v = \arg\min_i\{\gamma_{i,v}\}$ be the player who bid the lowest on vertex v (ties are broken arbitrarily, as in the case of a tie the winner gets exactly 0 profit out of the vertex anyway), and let $\pi_v = \min_{j \neq i}\{\gamma_{j,v}, p_v\}$ be the value profit that the winner gets. Finally let $\Gamma_i = \{v : B^*_v = i\}$ be the set of vertices that player i is the winner of, and let $\eta_i = \{e_{i,v} : v \in \Gamma_i\}$ be the set of edges used to connect these vertices to S_i. Then the utility for player i is:

$$u_i(S) = \sum_{v \in \Gamma_i} (\pi_v - \gamma_{i,v}) - \sum_{e \in E_i} c_e$$

This can also be thought of in a slightly different way. If we let $\hat{S}_i = (\hat{V}_i = V_i \cup \Gamma_i, \hat{E}_i = E_i \cup \eta_i)$ be the augmented subtree resulting from adding all the vertices that player i won the bid on to the base tree, then the utility for player i becomes:

$$u_i(S) = \sum_{v \in \hat{V}_i} \pi_v - \sum_{e \in \hat{E}_i} c_e$$

In this game, we again note that it is possible for there to be some profit for the users at the nodes. So again when calculating the social utility, we include the full value of any vertices that are in any players subtree. That is, if we let x_v be the number of players i with vertex v in $\hat{S}_i$, we have a social utility function that is basically the same as the previous game:

$$u(S) = \sum_{v : x_v \geq 1} p_v - \sum_{i=1}^{n} \sum_{e \in \hat{E}_i} c_e$$

Lemma 6.1. *This game is a potential game, with potential function $u(S)$.*

I will delay this proof to the appendix as it is a bit messy. The idea is to consider strategy vectors S and S' that differ only in the strategy of player i. One can then divide up the change in utility of player i into edges gained and lost, vertices gained and lost that no other player was bidding on, and vertices gained and lost in which player i was competing with another player. It is then possible to see that these categories together exactly make up the change in $u(S)$.

Theorem 6.2. *This game always has a Nash equilibrium and has Price of Stability 1.*

Proof. By Rosenthal, all potential games have a Pure Nash Equilibrium [1].

Let S^* be a social optimal, that is let S^* be a strategy set that maximizes $u(S)$. Then, since $u(S)$ is an exact potential function by Lemma 6.1, S^* is also a Pure Nash Equillibrium. Therefore, the social optimal is always a Pure Nash Equilibrium, so we have a price of stability of 1. □

Theorem 6.3. *It has arbitrarily large PoA.*

Proof. Observe the graph in Figure 3 (unlabeled edges cost 0). In this graph the social optimal occurs if player takes the left half of the graph and player 2 takes the right half for a total social utility of $2 + 2\varepsilon$. However, if player 1 instead takes the vertices necessary to

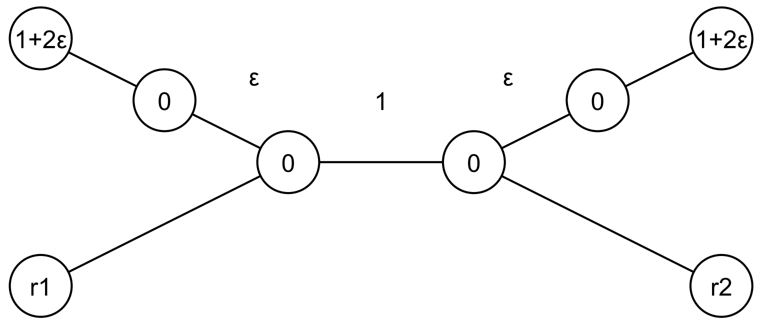

Figure 3: A Counterexample.

connect to the top right, and player 2 connects to the top left, we are left with a total value of 2ε, but the solution is still a Pure Nash Equilibrium. Thus the price of anarchy is $\frac{1+\varepsilon}{\varepsilon}$ which is unbounded. □

7 Conclusions and Future Directions

These Steiner Tree formation models often seem to have a very wide spectrum of Equilibria, with the best case being actually optimal and the worst case arbitrarily bad. As a consequence, one good direction for follow up will be to continue tweaking the models to see if a more stable situation arises. The first step in this regard is to settle the issue of Conjecture 4.1, which if true would result in a fairly promising model.

Since prize-collecting steiner tree is a NP-complete problem, all of these games suffer from the issue that players often can't reasonably be expected to compute what their best strategies really are. Therefore, another future direction would be to examine what happens if players can only use an α-approximation to determine the what strategy they use, or if players can only use strategies that are restricted to a smaller area of the graph (in which computation is more feasible).

Finally, it would potentially also be interesting to study the convergence time of the Nash Dynamics. That is, if players start from an arbitrary solution and take turns deviating to a better strategy, how long does it take to converge to a Nash Equilibrium? As the answer to this is probably exponential, one could also ask the same with slightly relaxed parameters by using approximate equilibria instead.

A Appendix

Here is a proof of Lemma 6.1.

Proof. Let $S = (S_1, ..., S_n)$ be a strategy vector and let $S' = (S_1, ..., S_i', ..., S_n)$ be strategy vector that differs only in the strategy of player i. Also, denote by x_v' and π_v' the number of players using v and the profit of v . Now consider $\hat{S}_i$ and $\hat{S}_i'$. We can characterize the difference between these trees as follow: some edges have been added or removed, and some vertices have been added or removed. Further, we can divide the vertices added or removed into two catagories: either player i is the only player bidding on them, or at least one other player has bid on them. That is we define the following sets:

$$E^+ = \{e \in \hat{E}_i' : e \notin \hat{E}_i\}, \qquad E^- = \{e \in \hat{E}_i : e \notin \hat{E}_i'\}$$
$$N^+ = \{v \in \hat{V}_i' : v \notin \hat{V}_i, x_v = 0\}, \qquad N^- = \{v \in \hat{V}_i : v \notin \hat{V}_i', x_v = 1\}$$
$$C^+ = \{v \in \hat{V}_i' : v \notin \hat{V}_i, x_v > 0\}, \qquad C^- = \{v \in \hat{V}_i : v \notin \hat{V}_i', x_v > 1\}$$

Then we can express the change in the personal utility of player i as follows, by canceling the edges and vertices that appear in both strategies, and grouping what remains. Note that we can use p_v instead of π_v in the middle terms since by definition there is no competition:

$$\begin{aligned} u_i(S) - u_i(S') &= \sum_{v \in \hat{V}_i} \pi_v - \sum_{e \in \hat{E}_i} c_e - \sum_{v \in \hat{V}'_i} \pi_v + \sum_{e \in \hat{E}'_i} c_e \\ &= \sum_{e \in E^-} c_e - \sum_{e \in E^+} c_e + \sum_{v \in N^+} p_v - \sum_{v \in N^-} p_v + \sum_{v \in C^+} \pi_v' - \sum_{v \in C^-} \pi_v \end{aligned}$$

Now consider the change in the social utility. Note that every change in edges used by player i will effect the social utility directly. Also any vertex that i had to himself and no longer is serving will now have no service, so its value will be lost. Similarly, any unused vertex that player i adds to the network will be added to the social utility. However, any vertex that was serviced by another player in S will still be serviced by someone, so its social value will remain unchanged (since any serviced vertex contributes its full value). The only other way that the social value can change will be if other players gain or lose edges. To track these changes we define two more sets:

$$F^+ = \{e \in E : e \in \hat{E}_j', e \notin \hat{E}_j \text{ some } j \neq i\}$$
$$F^- = \{e \in E : e \in \hat{E}_j, e \notin \hat{E}_j' \text{ some } j \neq i\}$$

We can then express the change in the social utility as follows, again by canceling terms that appear in positive and negative and then grouping using the sets we have defined here:

$$\begin{aligned} u(S) - u(S') &= \sum_{v: x_v \geq 1} p_v - \sum_{i=1}^{n} \sum_{e \in \hat{E}_i} c_e - \sum_{v: x_v' \geq 1} p_v + \sum_{i=1}^{n} \sum_{e \in \hat{E}_i'} c_e \\ &= \sum_{e \in E^-} c_e - \sum_{e \in E^+} c_e + \sum_{v \in N^+} p_v - \sum_{v \in N^-} p_v - \sum_{e \in F^+} c_e + \sum_{e \in F^-} c_e \end{aligned}$$

Now consider some $e \in F^+$. The only way for a nondeviating j player to gain an edge is as the result of winning a vertex that player i had previously outbid them on. Therefore there is some corresponding $v \in C^-$, a vertex that player i is no longer the winner of. But in order for j to now be the winner of v, they must have been the second lowest bidder before, so player i must have had exactly gotten exactly the value of player j's bid out of vertex v. That is, we must have $\pi_v = c_e$. Note further that for every vertex $v \in C^-$ there must be some corresponding edge in F^+ with this property. Therefore we have:

$$\sum_{v \in C^-} \pi_v = \sum_{e \in F^+} c_e$$

Similarly, the only way to lose an edge is to be outbid by player i, resulting in player i gaining value exactly equal to the cost of that edge. Therefore, we can see that for each edge $e \in F^-$ there is some vertex $v \in C^+$ such that $c_e = \pi'_v$, and vice versa. So we also have:

$$\sum_{v \in C^+} \pi_v = \sum_{e \in F^-} c_e$$

Thus, substituting, we get:

$$u(S) - u(S') = \sum_{e \in E^-} c_e - \sum_{e \in E^+} c_e + \sum_{v \in N^+} p_v - \sum_{v \in N^-} p_v + \sum_{v \in C^+} \pi'_v - \sum_{v \in C^-} \pi_v = u_i(S) - u_i(S')$$

So $u(S)$ is an exact potential function.

□

References

[1] Robert W Rosenthal. "A class of games possessing pure-strategy Nash equilibria". In: *International Journal of Game Theory* 2.1 (1973), pp. 65–67.

[2] Daniel Bienstock et al. "A note on the prize collecting traveling salesman problem". In: *Mathematical programming* 59.1-3 (1993), pp. 413–420.

[3] Michel X Goemans and David P Williamson. "A general approximation technique for constrained forest problems". In: *SIAM Journal on Computing* 24.2 (1995), pp. 296–317.

[4] Jeroen Kuipers, Vincent Feltkamp, and Stephanus Hendrikus Tijs. *Vertex weighted Steiner tree games*. Maastricht University, 1996.

[5] Dov Monderer and Lloyd S Shapley. "Potential games". In: *Games and economic behavior* 14.1 (1996), pp. 124–143.

[6] Christos Papadimitriou. "Algorithms, games, and the internet". In: *Proceedings of the thirty-third annual ACM symposium on Theory of computing*. ACM. 2001, pp. 749–753.

[7] Adrian Vetta. "Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions". In: *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*. IEEE. 2002, pp. 416–425.

[8] Elliot Anshelevich et al. "Near-optimal network design with selfish agents". In: *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*. ACM. 2003, pp. 511–520.

[9] Hee-Kap Ahn et al. "Competitive facility location: the Voronoi game". In: *Theoretical Computer Science* 310.1 (2004), pp. 457–467.

[10] Stefano Leonardi and Guido Schäfer. "Cross-monotonic cost-sharing methods for connected facility location games". In: *Proceedings of the 5th ACM conference on Electronic commerce*. ACM. 2004, pp. 242–243.

[11] Chaitanya Swamy and Amit Kumar. "Primal–dual algorithms for connected facility location problems". In: *Algorithmica* 40.4 (2004), pp. 245–269.

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

[13] Anupam Gupta et al. "An efficient cost-sharing mechanism for the prize-collecting Steiner forest problem". In: *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics. 2007, pp. 1153–1162.

[14] Elliot Anshelevich et al. "The price of stability for network design with fair cost allocation". In: *SIAM Journal on Computing* 38.4 (2008), pp. 1602–1623.

[15] Ho-Lin Chen and Tim Roughgarden. "Network design with weighted players". In: *Theory of Computing Systems* 45.2 (2009), pp. 302–324.

[16] Aaron Archer et al. "Improved approximation algorithms for prize-collecting Steiner tree and TSP". In: *SIAM Journal on Computing* 40.2 (2011), pp. 309–332.

[17] Vittorio Bilò, Michele Flammini, and Luca Moscardelli. "The price of stability for undirected broadcast network design with fair cost allocation is constant". In: *Games and Economic Behavior* (2014).

[18] Yann Disser et al. "Improving the Hk-bound on the price of stability in undirected shapley network design games". In: *Theoretical Computer Science* 562 (2015), pp. 557–564.