

Capacity Allocation Games for Network-Coded Multicast Streaming

Elliot Anshelevich, Bugra Caskurlu, Koushik Kar, and Hang Zhang

Abstract—In this paper we formulate and study a capacity allocation game between a set of receivers (players) that are interested in receiving multicast data (video/multimedia) being streamed from a server through a multihop network. We consider *fractional* multicast streaming, where the multicast stream from the source (origin-server) to any particular receiver (end-user) can be split over multiple paths. The receivers are selfish and non-cooperative, but must collaboratively purchase capacities of links in the network, as necessary for delivery of the multicast stream from the source to the individual receivers, assuming that the multicast stream is *network coded*. For this multicast capacity allocation (network formation) game, we show that the Nash equilibrium is guaranteed to exist in general. For a 2-tier network model where the receivers must obtain the multicast data from the source through a set of relay nodes, we show that the price-of-stability is at most 2, and provide a polynomial-time algorithm that computes a Nash equilibrium whose social cost is within a factor of 2 of the socially optimum solution. For more general network models, we show that there exists a 2-approximate Nash equilibrium, whose cost is at most 2 times the social optimum. We also give a polynomial time algorithm that computes a $(2 + \epsilon)$ -approximate Nash equilibrium for any $\epsilon > 0$, whose cost is at most 2 times the social optimum. Simulation studies show that our algorithms generate efficient Nash equilibrium allocation solutions for a vast majority of randomly generated network topologies.

I. INTRODUCTION

The last decade has witnessed an explosive growth in the number of *streaming* video (multimedia) applications. Some of these involve live video streaming, while others stream video that is already available in stored format but too large for download-and-play. Stored video streaming applications like YouTube [4] are already contributing to a large fraction of the Internet traffic today¹, and IPTV and similar other efforts are likely to boost live video streaming through the Internet in the coming years [3]. Streaming stored video may be unicast or multicast, depending on the application: while IPTV [2] video streaming may mostly be multicast (broadcast), receiver-driven video streaming (like video streaming from YouTube [4]) will typically be unicast. Streaming live video will typically be multicast to possibly many receivers.

E. Anshelevich is with the Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180 (email: eanshel@cs.rpi.edu).

B. Caskurlu is with the Lane Department of Computer Science & Electrical Engineering, West Virginia University, Morgantown, WV 26506 (email: caskurlu@gmail.com).

K. Kar and H. Zhang are with the Department of Electrical, Computer & Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180 (emails: koushik@ecse.rpi.edu, zhangh10@rpi.edu).

This work was supported in part by NSF awards CNS-1017932, CCF-0914782, and CCF-1101495.

¹It was estimated that in 2007 YouTube consumed as much bandwidth as the entire Internet in 2000 [1].

For multicast data delivery, use of *network coding* allows individual receivers to simultaneously attain data rates that equal their maxflow capacities [6], which in general may not be achievable through a routing-only approach. Naturally, this makes network coding ideally suited for multicast data delivery over a multi-hop network.

In this paper, we consider a capacity allocation game that end-users will play in buying resources for multicast streaming data delivery. More specifically, receivers (users) buy capacities on the links of the distribution network at fixed (possibly different for different links) per-unit cost, so as to ensure delivery of a multicast stream (with a given source rate) from its source to the receiver. The receivers are selfish and non-cooperative and are only interested in minimizing their individual costs, but must collaboratively pay for capacities bought on network links, as necessary for network coded multicast data delivery from the data source to the individual receivers. We consider *fractional* multicast streaming, i.e., the multicast data between the source and any particular receiver can be split across multiple paths that exist between the source-receiver pair. The problem we consider is a *network formation* game where the amount of capacity collaboratively bought on the different links in the network must be such that the maxflow from the source to each receiver is no less than the desired multicast data rate. Using network coding [6], this ensures that all receivers are able to obtain their full data rate from the source. This paper focuses on the questions of the existence, efficiency, and computation of the equilibria of this game. Initially, we focus on the 2-tier network model where the receivers must obtain the multicast data from the source through a set of relay nodes, and derive some strong results by exploiting structural properties of such topologies. Later we consider arbitrary topology networks for multicast data distribution, and study the existence and efficiency of approximate equilibria for that case.

To measure efficiency of equilibrium, we use the common measures of the *price of anarchy* and the *price of stability* [27] — the supremum of the ratios between the costs of the worst and best pure Nash equilibrium, respectively, and that of the globally optimal solution over all instances of the game.

Our Contributions.: The specific technical contributions of this paper are as follows. For our fractional multicast network formation game, we show that pure Nash equilibrium is guaranteed to exist in general. For the 2-tier model, defined in Section IV, we show a tight bound of 2 on the price-of-stability, and provide a polynomial-time algorithm that returns a Nash equilibrium whose social cost is within a factor of 2 of the socially optimum solution. The 2-tier model is essentially

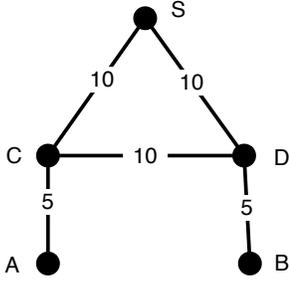


Fig. 1. Example: Fractional multicast streaming is better than integral multicast streaming. Source S sends multicast data at rate 1 to receivers A and B . All link capacities are 1 unit in each direction; numbers across links are per-unit capacity (bidirectional) purchase costs. Optimal fractional solution involves purchase of 1 unit capacities on links CA and DB , and 0.5 unit capacities on links SC , CD , SD ; total cost = 25. Optimal integral solution involves purchase of 1 unit capacities along SC , SD , CA , DB ; total cost = 30.

equivalent to the case where all nodes in the network are receivers, which is itself an important special case (e.g. [17]). For more general network topologies, we show that there always exists a 2-approximate Nash equilibrium, whose cost is at most 2 times the social optimum. We also give a polynomial time algorithm that computes a $(2 + \epsilon)$ -approximate Nash equilibrium for any $\epsilon > 0$, whose cost is at most 2 times the social optimum. Simulation studies show that our algorithms generate efficient Nash equilibrium allocation solutions for a vast majority of randomly generated networks.

While network formation games have been studied in other contexts (see Section II), the questions we consider are new for the context of network-coded fractional multicast streaming. Unlike integral multicast, data distribution networks for socially optimal or Nash equilibrium solutions for network-coded multicast need not be trees (Figure 1), and techniques for integral multicast do not extend to this context (see Section II). Interestingly, however, we show that there exist solutions based on tree topologies that are at Nash equilibrium (exactly or approximately) and attain a near-optimal social cost. For the 2-tier network model, the solution is based on the minimum spanning tree; for more general network models, it is based on the minimum Steiner tree or polynomial-time approximations of it. Despite the complexity of the problem, our results show that there exist easily-computable exact or approximate distribution networks where receivers have no motivation to deviate from it unilaterally, and yet results in the set of receivers paying near-minimal cost as a group for multicast data delivery.

The paper is structured as follows. In Section II, we outline related work on this topic. In Section III we describe the model and problem formulation. In Section IV we state and prove our main results on the existence, efficiency, and properties of Nash equilibrium distribution topology solutions for the 2-tier network model. We extend these results to arbitrary topology networks in Section V. In Section VI we describe the results of experiments conducted on randomly generated network topologies.

II. RELATED WORK

In contrast to the work presented in this paper, the models in most network formation game literature do not allow players to reserve an arbitrary amount of bandwidth on the links; rather, a link can either be constructed and be utilized to full extent or won't be constructed at all. Our game represents a more realistic scenario by allowing players to buy certain amount of capacity (bandwidth) on that link. This becomes particularly relevant for network coded multicast streaming [6], [18], [21], [28] that we consider in this paper, where the capacity that needs to be bought on a link for successful multicast streaming is often less than the link capacity, as well as the source data rate.

There have been several variants of “integral” network formation games where instead of allocating capacity, a link (edge)² can be either fully present or non-existent. One of the most important decisions when modeling network design involving strategic agents is to determine how the total cost of the solution is going to be split among the players. Among various alternatives [13], the most popular one in the literature is the “fair sharing” mechanism [8], [11], [12], [17]. In this cost sharing mechanism, the cost of each edge of the constructed network is shared equally by the players using that edge. Since in our model, each player is allowed to purchase any amount of capacity on an edge, the “arbitrary sharing” model of network formation [9], [7], [15], [19], [20] is closer to being an “integral” version of our game. In this model, players contribute to the cost of an edge, and an edge is present in the network if the player contributions are larger than its cost. This model has many differences from the “fair sharing” model: e.g., the game is not a congestion game, but the price of stability is much better than with fair sharing, etc.

While many interesting results have been proven for network formation games with arbitrary sharing, most do not extend to our “fractional” context where players (multicast data users/receivers in our case) are allowed to reserve an arbitrary amount of bandwidth on the edges. For example, [9] proved that the minimum-cost Steiner tree is always a Nash equilibrium in this integral network formation game with arbitrary sharing, but the same does not hold for the fractional version. Consider, for example, a graph with n receiver nodes, node s , and an extra node v . All receivers wish to receive a rate of 1 from node s . For any receiver node u , the cost of allocating x capacity on an edge (u, v) is x , and the same is true for edge (v, s) . The cost of allocating x capacity on an edge (u, s) is $x(2 - \epsilon)$. No other edges exist in the graph. Then, the minimum-cost Steiner tree has cost $n + 1$ (for this example, it is also the optimum fractional solution, although Figure 1 shows that this is not always the case). However, the min-cost Steiner tree in this example is not a Nash equilibrium: there must be some receiver u who is paying for 1 capacity of edge (u, v) , and at least for $1/n$ capacity of edge (v, s) . This receiver could reduce its payments to both edges by $1/n$, and instead pay for $1/n$ capacity on edge (u, s) , costing it $(2 - \epsilon)/n$, and thus strictly decreasing its cost.

²We use the terms ‘link’ and ‘edge’ interchangeably in this paper.

Our game assumes that there is no central authority that can dictate the network cost-sharing mechanism, and thus players simply purchase edge capacities directly. Other directions in multicast games include cooperative games and mechanism design (see, e.g. [16] and [27, Chapters 14-15] and references therein), where the goal is to come up with a cost-sharing mechanism with good properties that could be implemented by a central authority.

Games involving network-coded traffic have received significant attention in recent literature, although the game model/formulation considered in existing work differ substantially from ours. Some of these consider games played between unicast flows (sessions) [22], [23], [24]; thus the model and concerns in these are quite different from our game model that assumes and utilizes the multicast nature of the traffic. Among prior work, [25], [26] are closest to ours, and also consider multicast traffic. However, unlike our work, [25] deals with a traffic routing (forwarding) game; [26] analyzes a cooperative game whereas our game is non-cooperative. Moreover, another major difference is that in the games considered in [25], [26], a player's payment to a link is determined by a fixed sharing scheme. In our game, however, players are free to pay as much as they like for any link; thus the players in our game have much more freedom in how they behave.

III. SYSTEM MODEL AND FORMULATION

We consider a network modeled by an undirected graph $G = (V, E, b)$, where vertices V denote the set of nodes, and (undirected) edges E denote the set of (bidirectional) links in the network. For each edge (link) $e \in E$, we assume that the capacity of the corresponding edge is at least 1 in each direction. In other words, at full capacity at least 1 unit of traffic can travel on edge e in one direction, and simultaneously 1 unit of traffic can travel on e in the opposite direction. This network is to be used for delivery of a given traffic stream of rate 1. One of the nodes, $s \in V$ is designated as the source of the traffic stream, and a subset of the nodes, $R \subseteq V$ are receivers (users) of the traffic stream. In our model, all receivers must collaboratively pay for capacities of the edges that are used for carrying the traffic stream, for shared use by all receivers in R . We assume that an edge e is associated with a cost of $c(e)$ that buys the corresponding edge, i.e., $c(e)$ is the cost of buying capacity of 1 of the edge in each direction. If each receiver $i \in R$ pays $p_i(e)$ for edge e , then **the total purchased capacity on the corresponding edge is $\theta(e) = \min\{1, (\sum_{i \in R} p_i(e))/c(e)\}$** . In other words, if the receivers R pay a fraction of edge e 's cost in total, then the same fraction of e 's desired capacity becomes available in both directions.

To formally define the game, we note that the receivers R is the set of players, and $\mathbf{p} = (p_i(e), i \in R, e \in E)$, or the prices paid by the receivers for the edges, constitute the player strategies. (Formally, a strategy of receiver $i \in R$ is a function $p_i : E \rightarrow R_{\geq 0}$ that determines how much i is offering to pay for each edge.) Once the strategies are given, the network used for multicast traffic streaming, or the *distribution network*, is (V, E, θ) , where the purchased capacities $\theta(\cdot)$ depend on the

strategy vector \mathbf{p} . We assume that each node in the network is capable of network-coding. Therefore, in our context each node (other than those that act solely as receivers) has to perform operations that go beyond traditional relaying (i.e., replication and forwarding). The exact set of operations that each node must be capable of would depend on the network coding algorithm used; in general, this may require "mixing" of packets coming from different upstream nodes to create coded packets that are then sent further downstream.

With network coding, a necessary and sufficient condition for delivery of the given traffic stream to all receivers is that the maxflow from the source s to each receiver $i \in R$ in distribution network (V, E, θ) is no less than the stream rate of 1 [6]. In other words, a feasible strategy vector \mathbf{p} is one that ensures that

$$f_i(\mathbf{p}) \geq 1, \quad (1)$$

holds for each receiver $i \in R$, where $f_i(\mathbf{p})$ denotes the maxflow from source node s to receiver node i in distribution network (V, E, θ) . The goal of each receiver i is to receive the full rate of 1, but pay as little as possible. Formally, the cost of receiver i with strategy vector \mathbf{p} is $\sum_e p_i(e)$ if $f_i(\mathbf{p}) \geq 1$, and is very large otherwise.

It is worth re-stating that the purchased capacity does not become split or shared among the receivers; it does not "belong" to any receiver; instead network coding is used to send traffic to all the receivers using this capacity. Thus, the goal of each receiver is to make sure that enough capacity is purchased to guarantee that $f_i(\mathbf{p}) \geq 1$, and the more of this capacity is purchased by *other* receivers, the happier receiver i will be!

Among all strategies that form feasible networks, we are specifically interested in those that are Nash equilibria. A solution is a Nash equilibrium if each receiver does not have an incentive to unilaterally deviate from it. To state this formally, consider a strategy vector \mathbf{p}^* that is feasible for all receivers, and let $\mathbf{p}^*_{-i} = (p_j^*(e), j \in R \setminus i, e \in E)$, denote the strategies of (purchase prices paid by) all receivers other than i . Then the strategy vector \mathbf{p}^* is said to be a Nash equilibrium if for any $\mathbf{p}_i = (p_i(e), e \in E)$ such that $(\mathbf{p}_i, \mathbf{p}^*_{-i})$ satisfies the feasibility condition (1) for receiver i , we have that $\sum_{e \in E} p_i^*(e) \leq \sum_{e \in E} p_i(e)$. In other words, given the strategies of (prices paid by) the other receivers, and subject to maintaining the feasibility condition $f_i(\mathbf{p}) \geq 1$ that is necessary for receiver i to receive the streamed data at the full rate, the total price paid by receiver i is minimized at Nash equilibrium. A solution that is not feasible for all receivers will never be a Nash equilibrium, since the infeasible receivers will have very large cost, and will have incentive to purchase more capacity in the network in order to become feasible.

To study the efficiency of the Nash equilibrium, we next define the *social optimum* against which the Nash equilibrium solution will be compared in terms of the total capacity purchase cost. A price vector \mathbf{p} is said to be a social optimum if it minimizes $\sum_{i \in R, e \in E} p_i(e) = \sum_e \theta(e)c(e)$, subject to satisfying the feasibility constraints (1) for all receivers $i \in R$. We will denote this solution as *OPT*. The supremum of the

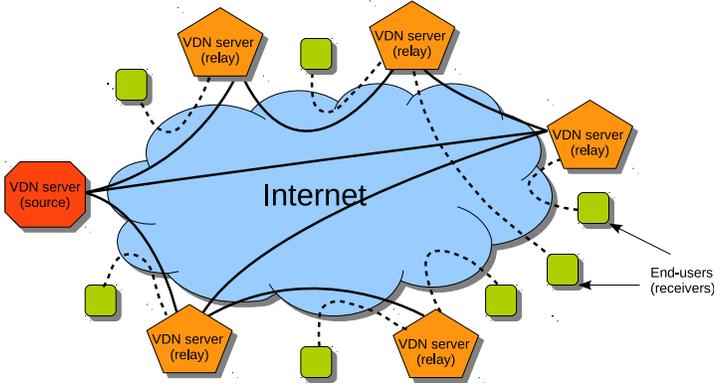


Fig. 2. Streaming video delivery in the 2-tier network model.

ratio of the overall cost of the *worst* Nash equilibrium to that of the social optimum over all instances of a game is defined as the *price of anarchy* of the game. Similarly, the supremum of the ratio of the overall cost of the *best* Nash equilibrium to that of the social optimum over all instances of a game is defined as the *price of stability* of the game. In this paper, we use these two notions to characterize the efficiency of the worst and best Nash equilibria for the game.

IV. CAPACITY ALLOCATION GAMES ON THE 2-TIER NETWORK MODEL

It has been envisioned that in the near future, streaming of video over the Internet will be done through the use of multiple, dedicated servers that would relay video from the source server to the end-users [5]. This network of video relay servers is expected to play a role similar to that of content distribution networks (CDN) for delivery of various kinds of non-streaming and non-real time data (content). This relay network will deliver video from the source to the receiver through multiple relay-hops, possibly performing network coding at the intermediate (relay) nodes/servers. This is illustrated in Figures 2-3.

In this section, we restrict our attention to the 2-tier network model. In this model, $V = \{s\} \cup L \cup R$, i.e., the vertex set of the graph is composed of the source vertex s , the set of receivers $R = \{r_1, r_2, \dots, r_n\}$, and the set of relay nodes $L = \{l_1, l_2, \dots, l_k\}$. Each receiver node r_i of G has *exactly* one incident edge and is adjacent to a relay node of G , i.e., for each $r_i \in R$ there exists $l_j \in L$ such that $(r_i, l_j) \in E$, and r_i has no other incident edges in G . Each relay node is adjacent to one or more receiver nodes, i.e., for each $l_j \in L$ there exists $r_i \in R$ such that $(r_i, l_j) \in E$. These assumptions imply that each receiver is directly connected to exactly one relay node, from which it must receive the data being streamed (this relay node can obtain the data from the source through other relay nodes, however). In addition, each relay node serves at least one receiver. This assumption can be interpreted in the following way: if a relay node has no receiver to serve, then it does not participate in the multicast data distribution.

Note that the 2-Tier model is equivalent to the model where all nodes are receivers, in the following sense. If G is an instance of our game in the 2-Tier model, let G' be a game

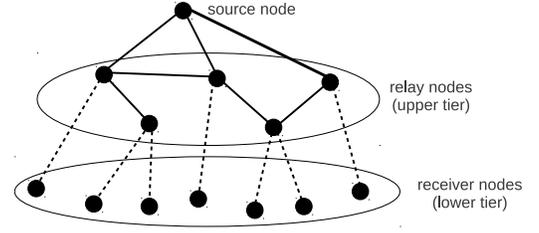


Fig. 3. Graph representation of the 2-tier network model in Figure 2.

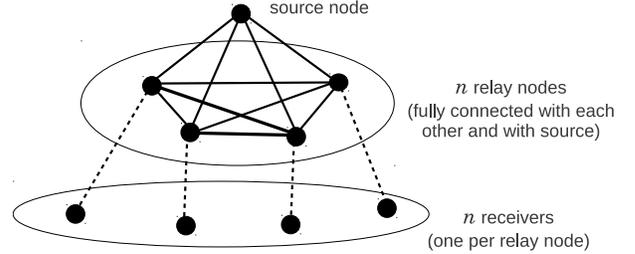


Fig. 4. Example 2-tier topology to show that price of stability is at least 2.

obtained by contracting all the edges incident to a receiver node in G . Since every relay node in G must be adjacent to a receiver, then all nodes in G' except s are receivers, although there may be several receiver players located at the same node. Since every Nash equilibrium in G must have each receiver r_i purchasing 1 capacity on the edge (r_i, l_j) , then it is easy to see that there is a one-to-one correspondence between the Nash equilibria in G and in G' . Below we argue about the price of stability for 2-Tier networks, but all the arguments can easily be extended to the model where the network has arbitrary topology, but every node except s is a receiver/player.

We prove below that if G has a 2-tier topology as described above, then there exists a Nash equilibrium solution that does not cost much more than the cost of the socially optimal solution. We first show, however, that there are examples where all Nash equilibria cost a factor of 2 more than the social optimum.

Theorem 1: The price of stability of the capacity allocation game with the 2-Tier topology is at least 2.

Proof: Consider the following example: Figure 4 shows G , the graph representation of a 2-tier topology that we consider for showing that the price of stability is lower bounded by 2. In G , there are n receiver nodes and n relay nodes, and therefore there is a matching between the set of receiver and relay nodes. The source node s and n relay nodes form a complete subgraph of $n+1$ nodes. The capacity of each edge in G is 1. The cost of reserving 1 unit of capacity on an edge $e = (i, j)$ of G is defined as follows:

$$c(i, j) = \begin{cases} 0 & \text{if } i \in R \text{ and } j \in L \\ (1 + \epsilon) & \text{if } i, j \in L \\ 1 & \text{if } i \in R \text{ and } j = s \end{cases}$$

Consider a distribution network (V, E, θ) , where the receivers reserve 1 unit of capacity on all edges incident to a receiver, and $\frac{1}{n}$ units of capacity on all other edges of G . First, let us argue that all receivers satisfy the feasibility condition given by Equation (1). Each relay node l_i can receive a flow of size $\frac{1}{n}$ from s through the edge (s, l_i) , and a flow of size $\frac{1}{n}$ from s through the edges $(s, l_j), (l_j, l_i)$ for all $j \neq i$. Since all these flows are disjoint, l_i can receive a flow of size 1 from s on (V, E, θ) . Therefore, each receiver can receive a flow of size 1 as well.

The total cost of the distribution network (V, E, θ) is

$$\frac{n(n-1)}{2} \frac{(1+\epsilon)}{n} + n \frac{1}{n} = \frac{(n-1)(1+\epsilon)}{2} + 1 \quad (2)$$

In a Nash equilibrium solution, however, no receiver can make a payment for any of the edges between the relay nodes. For the purpose of contradiction, assume receiver r_i pays a strictly positive amount x for the cost of the edge (l_j, l_k) in a Nash equilibrium solution. Let l_i be the relay node adjacent to r_i . Then, notice that r_i satisfies Equation (1) if she sets her payment on (l_j, l_k) to 0 and increases her payment on (l_i, s) by $\frac{x}{(1+\epsilon)}$. Since r_i can reduce her cost by unilaterally deviating, then we have a contradiction with this being a Nash equilibrium. Therefore, no receiver makes a payment for any of the edges between the relay nodes in a Nash equilibrium solution.

In a Nash equilibrium solution 1 unit of capacity is to be reserved on all the edges between relay nodes and the source to satisfy Equation (1), since no capacity is reserved on the edges between the relay nodes. Therefore, the Nash equilibrium for the above example is unique, and the cost of the Nash equilibrium solution is n . Notice that the ratio

$$\frac{n}{\frac{(n-1)(1+\epsilon)}{2} + 1} \quad (3)$$

can be made arbitrarily close to 2 by assigning large values to n and small values to ϵ . Therefore, the price of stability in the 2-tier network model is at least 2. Since there always exists a Nash equilibrium solution whose social cost is within a factor of 2 of the socially optimal solution by Theorem 2 (see below), the example given above is indeed the worst case example, and the price of stability is 2. ■

Theorem 1 states that the supremum of the ratio of the cost of the best Nash equilibrium to the cost of the socially optimal solution is at least 2. In the proof of Theorem 1, we show that for any $\epsilon > 0$ we can construct an instance of the capacity allocation game with the 2-Tier topology such that the social cost of all Nash equilibria are at least $(2 - \epsilon)$ times the social cost of OPT .

Consider the cheapest solution satisfying Condition (1) for our game, which we denoted by OPT . Notice that OPT is exactly the optimum fractional solution to the LP-relaxation of the classic Steiner tree problem (see, e.g., Sec 3.1 of [29] for the definition of the Steiner Tree problem, and Sec 22.1 of [29] for the LP-relaxation for the more general Steiner Forest problem, which also applies to the Steiner Tree problem), with

the terminal nodes being $R \cup \{s\}$ and edge costs being $c(e)$. Since the integrality gap of this LP is at most 2 ([29], Chapter 22), this implies that a minimum-cost Steiner tree has cost at most twice the cost of OPT . More precisely, if we take a Steiner tree T of graph G with terminals $R \cup \{s\}$ and edge costs $c(e)$, and set the capacity of each edge in T to be 1, and each edge not in T to be 0, then the cost of this tree is at most twice the cost of OPT . We show in Section V that we can always form a 2-approximate Nash equilibrium on a Steiner tree, and there are simple examples where there is no Nash equilibrium that buys the minimum-cost Steiner tree. For the 2-Tier network topology, however, notice that the minimum-cost Steiner tree is simply the minimum spanning tree (MST) of G . This is because every node r_i of R has *exactly* one incident edge and is adjacent to a relay node of G , and each relay node is adjacent to one or more receiver nodes. Notice that in the example depicted in Figure 4, the set of edges that has 1 unit of capacity on the unique Nash equilibrium solution is actually the minimum spanning tree of G . We next prove that this is not a coincidence, i.e., there always exists a Nash equilibrium solution that buys the minimum spanning tree, which gives the result stated by Theorem 2 since the cost of the minimum spanning tree is at most twice the cost of the socially optimal solution, as argued above.

The proof of Theorem 2 gives a polynomial-time algorithm that returns a strategy profile $\mathbf{p} = (p_i(e), i \in R, e \in E)$ that is a Nash equilibrium, and reserves 1 unit of capacity on the edges of the MST. Since the cost of MST is at most 2 times the cost of OPT , the price of stability is at most 2. Theorem 1 and Theorem 2 together imply that the price of stability is 2.

Theorem 2: There is a polynomial-time algorithm that returns a Nash equilibrium of the capacity allocation game for the 2-tier topology whose social cost is within a factor of 2 of the cost of OPT , and thus the price of stability is at most 2.

Proof: We prove the result by showing that there always exists a Nash equilibrium solution that reserves 1 unit of capacity on the edges of the minimum spanning tree and 0 units of capacity on the remaining edges. Our proof is constructive, i.e., we explicitly form payments that purchase the minimum spanning tree. In our payment scheme, for each edge e of minimum-cost spanning tree T , there is a corresponding receiver r_i that reserves 1 unit of capacity on it. Notice that even though the 'arbitrary-sharing' cost-sharing scheme allows receivers to share the cost of reserving 1 unit of capacity on the edges of T , our payment scheme does not use this property. Therefore, in order to fully specify the payment scheme all we need to do is to assign one receiver for each edge e of T .

Without loss of generality, for an edge $e = (i, j)$, we will assume that j is the node that is closer to s on T than i . If $e = (i, j)$ is an edge incident to a receiver, i.e., $i \in R$, then the receiver i makes a payment on e that is sufficient to reserve 1 units of capacity on it. Let $e = (i, j)$ be an edge between two relay nodes, or a relay node and the source, i.e., $i \in L$. Let r_k be an arbitrary receiver that has a direct edge to the relay node i , i.e., (r_k, i) is an edge of T . Then, r_k makes a payment of $c(e)$ on e .

Since we have fully specified the payment scheme on all the edges of T , let us now prove that this payment scheme is indeed a Nash equilibrium. Notice that in our payment scheme each receiver is reserving 1 units of capacity on either one or two edges of T . More precisely, each receiver r_i paying for the cost of the edge (r_i, l_j) incident to her node, and possibly she is also paying for the cost of the other incident edge of l_j .

Recall that any solution that is feasible for r_i must satisfy Inequality (1) for r_i . If a receiver r_i is only paying for the cost of her incident edge e in T , she trivially does not have an incentive of unilateral deviation since r_i does not have any other incident edges in G , and therefore any solution where r_i satisfies Equation 1 reserves 1 unit of capacity on e . Suppose r_i is paying for the cost of both $e = (r_i, l_j)$ incident to her node, and the cost of another incident edge f of l_j . Since the payments of the receivers $R - \{r_i\}$ reserve 1 unit of capacity on $T - \{e, f\}$, the best response of r_i must include enough capacity for a flow of size 1 between r_i and $T - \{e, f\}$. The cheapest way to obtain this capacity is to reserve 1 unit of capacity along a single path between r_i and $T - \{e, f\}$, and the cost of this path cannot be more than the total cost of e and f , since otherwise T would not be a minimum spanning tree. Thus, any deviation of r_i that results in a feasible solution for r_i is at least as expensive as $c(e) + c(f)$. Since no receiver has an incentive for unilateral deviation, the resulting payment scheme is a Nash equilibrium.

Running Time: The runtime to find this payment scheme is just the time to find a minimum spanning tree, plus a linear amount of time to form the payment assignment. Thus, the runtime is $O(|E| \log |V|)$ using standard MST algorithms. ■

V. GENERALIZATIONS FOR ARBITRARY NETWORK MODELS

In this section, we consider our general game, with the graph G having arbitrary topology, and an arbitrary subset of receiver nodes. The capacity allocation game is guaranteed to have a Nash equilibrium by Theorem 3, however, the cost of some Nash equilibria can be prohibitive by Theorem 5.

Theorem 3: Nash equilibrium in pure strategies is guaranteed to exist in the capacity allocation game.

Proof:

For each receiver $i \in R$, let S_i denote the strategy space of receiver i . Receiver i selects a strategy $s_i \in S_i$ when she plays the game. Let $S = \prod_{i \in R} S_i$ denote the strategy space of the game. Notice that S is the product space of the strategy spaces of the receivers. A *strategy profile* $s \in S$ is an n -tuple $s = (s_1, \dots, s_n)$ such that each entry s_i of s is a strategy of receiver i . We use the common notational convenience and write a strategy profile as $s = (s_i, s_{-i})$ where $s_{-i} \in \prod_{j \in R - \{i\}} S_j$. Notice that a strategy $s_i \in S_i$ of receiver i is a vector of size m (with m being the number of edges in the graph), since a strategy for a receiver consists of a nonnegative payment for each edge e of G . Without loss of generality, we will assume $s_i(e) \leq \max_e c(e)$. Notice that S is a nonempty, convex, and compact set since S is a cube in $R^{n \times m}$: specifically it is just the cross product of the closed interval $[0, \max_e c(e)]$ taken nm times.

In order to prove the result, we use the technique used in Nash's proof for showing existence of mixed Nash equilibrium in finite games, that uses Kakutani's fixed point theorem. Our proof uses standard techniques, except for the part showing that the graph $\Gamma(F)$ is closed, which requires somewhat different arguments due to the fact that our cost functions are not continuous.

Recall that Kakutani's fixed point theorem is defined as follows:

Theorem 4 (Kakutani's Fixed Point Theorem): Let S be a non-empty, compact and convex subset of some Euclidean space R^n . Let $F : S \rightarrow 2^S$ be a set-valued function on S with a closed graph $\Gamma(F)$, and the property that $F(s)$ is nonempty and convex for all $s \in S$. Then F has a fixed point.

A set-valued function $F : S \rightarrow 2^S$ is some rule that maps each element $s \in S$ to a subset of S , i.e., $F(s) \subset S$. Notice that each element of $F(s)$ is a strategy profile of the capacity allocation game. Since S is nonempty, compact, and convex, Kakutani's fixed point theorem states that if the function graph $\Gamma(F) = \{(s, t) | s \in S, t \in F(s)\}$ (which is a subset of the product space $S \times S$) is a closed set, and $F(s)$ is a nonempty and convex set for all $s \in S$, then there exists $s \in S$ such that $s \in F(s)$, i.e., a fixed point.

For a strategy profile $s = (s_i, s_{-i})$, let $\chi_i(s_{-i})$ denote the set of best responses of receiver i to the strategies s_{-i} of other receivers. Given the strategies s_{-i} of other receivers (which correspond to some capacity reservation on the edges of G), each element $s'_i \in \chi_i(s_{-i})$ is a minimum cost strategy of receiver i that will ensure that a flow of size 1 can be sent from the source to i in the distribution network purchased by (s'_i, s_{-i}) . It is easy to see that for each s_{-i} we can express $\chi_i(s_{-i})$ as the set of optimal solutions of a linear program, with only non-strict inequalities. Therefore, $\chi_i(s_{-i})$ is a closed and convex subset of S_i for all $s_{-i} \in \prod_{j \in R - \{i\}} S_j$. Moreover, $\chi_i(s_{-i})$ is non-empty, since receiver i always has at least one best response.

We define the mapping $F : S \rightarrow 2^S$ as follows. Given a strategy profile $s, t \in F(s)$ if t is a strategy profile that can be obtained if each receiver $i \in R$ deviates from her strategy s_i to one of her best responses, i.e., to an element of $\chi_i(s_{-i})$. Formally, we define F as $F(s_1, \dots, s_n) = \{(t_1, \dots, t_n) | t_i \in \chi_i(s_{-i})\}$. In other words, $F(s_1, \dots, s_n) = \prod_{i \in R} \chi_i(s_{-i})$. Since $F(s)$ is the product space of nonempty, closed, and convex sets, then $F(s)$ is nonempty, closed, and convex for all $s \in S$. Therefore, if the graph $\Gamma(F) = \{(s, t) | s \in S, t \in F(s)\}$ is a closed set, then by Kakutani's fixed point theorem there exists $s \in S$ such that $s \in F(s)$. Notice that $s \in F(s)$ if and only if $s_i \in \chi_i(s_{-i})$ for all receivers i , i.e., the strategy of all the receivers is a best response of them to the strategies of the other receivers. Hence, a fixed point of F is a Nash equilibrium of the capacity allocation game. Therefore, in order to complete the proof all we need to show is that $\Gamma(F)$ is a closed set.

Let $(x^1, y^1), (x^2, y^2), \dots$ be an arbitrary convergent sequence of points in $\Gamma(F)$, and denote its limit by (x^*, y^*) . To show that $\Gamma(F)$ is closed, all we need to show is that $(x^*, y^*) \in \Gamma(F)$. Recall that $\Gamma(F) \subset S \times S$, and $S \times S$ is closed, so $(x^*, y^*) \in S \times S$. Therefore, all we need to show is

that $y_i^* \in \chi_i(x_{-i}^*)$ for all $i \in R$. In order to do this, fix some arbitrary receiver i .

Let $C_i(s)$ denote the cost of receiver $i \in R$ for strategy profile s , and let $\hat{C}_i(s)$ be the cost of i 's best response to s , i.e., $\hat{C}_i(s) = \min_{s'_i \in S_i} \{C_i(s'_i, s_{-i})\}$. Since $C_i(s'_i, s_{-i})$ is minimized if and only if $s'_i \in \chi_i(s_{-i})$, then we can equivalently define \hat{C}_i as $\hat{C}_i(s) = C_i(s'_i, s_{-i})$ for some $s'_i \in \chi_i(s_{-i})$.

We next define a function $\overline{C}_i : S \rightarrow \mathcal{R}_{\geq 0}$ for each $i \in R$ as follows: $\overline{C}_i(s) = C_i(s) - \hat{C}_i(s)$. Notice that $\overline{C}_i(s) \geq 0$ for all $s \in S$ and $\overline{C}_i(s) = 0$ if and only if $s_i \in \chi_i(s_{-i})$. In other words, $\overline{C}_i(s) = 0$ if and only if s is a stable strategy profile for receiver i , i.e., receiver i does not have an incentive of unilateral deviation from s . Notice that for any point $(s, t) \in \Gamma(F)$, we have that $\overline{C}_i(t_i, s_{-i}) = 0$ since $t_i \in \chi_i(s_{-i})$ by definition of F . Therefore, $\overline{C}_i(y_i^k, x_{-i}^k) = 0$ for all $k > 0$. In the usual argument about the existence of Nash equilibrium, \overline{C}_i is continuous over S and therefore this completes the proof, since this implies that $\overline{C}_i(y_i^*, x_{-i}^*) = 0$, and thus that $y_i^* \in \chi_i(x_{-i}^*)$. In our game, however, \overline{C}_i is not continuous.

Recall that \overline{C}_i is defined as the difference of two functions, i.e., $\overline{C}_i(s) = C_i(s) - \hat{C}_i(s)$. The function $\hat{C}_i(s)$ is continuous on S , since for any strategy profile s' such that $\|s - s'\| \leq \epsilon$, we have that $|\hat{C}_i(s) - \hat{C}_i(s')| \leq \epsilon$. However, $C_i(s)$ is not necessarily continuous on S , since when the mincut between the source and i becomes less than 1 in the distribution network, then the cost for receiver i suddenly becomes unbounded. Let Δ_i be the set of strategy profiles where receiver i is feasible, i.e., $\Delta_i = \{s | f_i(s) \geq 1\}$. For any $s \in \Delta_i$, the cost $C_i(s)$ is simply equal to $|s_i| = \sum_e s_i(e)$. Thus C_i is clearly continuous on the domain Δ_i . Notice that Δ_i can be formulated as a set of linear constraints with non-strict inequalities and therefore, Δ_i is a closed (and convex) set. Since both \hat{C}_i and C_i are continuous on Δ_i , then \overline{C}_i is also continuous on Δ_i .

Notice that for any $k > 0$, we have that $(y_i^k, x_{-i}^k) \in \Delta_i$, since y_i^k is a best response of receiver i to x_{-i}^k , and thus results in a solution feasible for receiver i . Since Δ_i is closed, then (y_i^*, x_{-i}^*) is also in Δ_i . Thus, since $\overline{C}_i(y_i^k, x_{-i}^k) = 0$ for all k , and \overline{C}_i is continuous on Δ_i , then $\overline{C}_i(y_i^*, x_{-i}^*) = 0$. This implies that $y_i^* \in \chi_i(x_{-i}^*)$. Since this is true for all i , then $(x^*, y^*) \in \Gamma(F)$, as desired. ■

Theorem 5: The price of anarchy for the capacity allocation game is N and this bound is tight.

Proof: We will first establish that the price of anarchy cannot be more than N . For the purpose of contradiction, assume the price of anarchy is more than N , i.e., there exists a Nash equilibrium solution whose social cost is more than N times the cost of OPT . Then, by pigeonhole principle, there exists a receiver i whose total payment is more than the cost of OPT . Observe that receiver i has an improving deviation since she can reduce her cost simply by buying OPT instead of her existing strategy. Therefore, the price of anarchy cannot be more than N .

In Figure 5, all the receivers are at the node to the left. There are 2 paths between the terminal node and s . The upper path is composed of a single edge and the lower path is composed of N edges. The cost of all edges in the network is 1. Consider the strategy profile where each receiver reserves 1 unit of

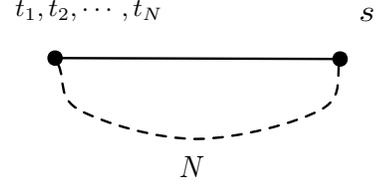


Fig. 5. An example where price of anarchy is N .

capacity on one of the edges of the lower path. Observe that this strategy profile is a Nash equilibrium since the cost of the best response of all receivers is 1. The cost of this Nash equilibrium solution is N . In the optimal solution the upper path will be bought, so the cost of OPT is 1. Therefore, the price of anarchy is N and this bound is tight. ■

In network formation games on undirected graphs, bounding the price of stability is usually a lot more challenging than bounding the price of anarchy [8], [10]. The price of stability is known to be 1 for arbitrary sharing in the discrete model, i.e., there exists a Nash equilibrium that buys the Steiner Tree [9]. However, the analysis for the discrete model does not carry over for capacity allocation games (i.e., the fractional model), for which the price of stability is shown to be at least 2 in Section IV. In fact, the example in Section II shows that the Steiner tree is not necessarily a Nash equilibrium for our game, and so it is not the case that this factor of 2 arises simply because of the gap between integral and fractional solutions. Thus, even though there always exists a Nash equilibrium that buys the cheapest integral solution in any 2-tier topology (which is the minimum spanning tree in this case), this is not true for general undirected networks.

Though we do not have an upper bound for the price of stability in general undirected networks and therefore cannot guarantee the existence and efficient computation of cheap Nash equilibrium, we prove that there always exists a cheap *approximate* Nash equilibrium that can be efficiently computed. By an α -approximate Nash equilibrium, we mean a solution where no receiver can reduce her cost by a factor of α by unilateral deviation [9].

Similar to the 2-tier topology in spirit, we start with a cheap integral feasible solution T and form payments on the edges of T . The cheapest integral solution T is the min-cost Steiner tree that connects the source and the receiver nodes. Given the Steiner tree T , the proof of Theorem 6 gives a polynomial-time algorithm that returns a strategy profile $\mathbf{p} = (p_i(e), i \in R, e \in E)$ that is a 2-approximate Nash equilibrium, and reserves 1 unit of capacity on the edges of T . Since the cost of the Steiner tree T is at most 2 times the cost of OPT (as discussed in the previous section), we obtain a 2-approximate Nash equilibrium, whose social cost is at most twice the cost of OPT . However, since there is no polynomial-time algorithm to compute the min-cost Steiner tree unless $P = NP$, we cannot obtain this 2-approximate Nash equilibrium solution in polynomial-time. Fortunately, in Theorem 7 we give a polynomial-time algorithm to compute a $(2 + \epsilon)$ -approximate Nash equilibrium, whose social cost is at most twice the cost of OPT .

Before stating Theorem 6, we first define the term *edge block* which we use in the proof of Theorem 6.

Definition 1: Given a minimal (with respect to edge removal) tree T that connects the source and the receiver nodes, a maximal length path of T whose interior nodes are degree 2 nonreceiver nodes is called an *edge block* of T .

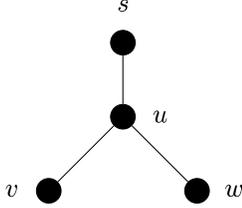


Fig. 6. A simple example with 4 nodes s, u, v , and w . s is the source, and v and w are receiver nodes. The network only has three edges: (s, u) , (v, u) , and (w, u) . According to Definition 1, each of these three edges is a separate edge block, so there are 3 edge blocks in this tree.

Figure 6 gives a simple example, where there are 2 receivers and 3 edges. In this example every edge is also an edge block. If instead on a single edge (w, u) , there were two edges (w, w') and (w', u) , then this path of 2 edges $(w, w'), (w', u)$ would still be a single edge block, since it is a path with all internal nodes being non-receiver nodes of degree 2 in the tree. The number of edge blocks is at most $2n - k$, as proven in Lemma 1.

Lemma 1: The number of edge blocks in a tree T is at most $2n - k$, where n is the number of receiver nodes, and k is the number of non-leaf receiver nodes of T .

Proof: Consider the tree T rooted at s , but with each edge block replaced by a single edge. All leaves of this tree are receiver nodes (otherwise the edges leading to them would not be needed). The number of edges adjacent to the leaves is thus $n - k$. The only nodes of degree two are also receiver nodes, since otherwise these nodes would have been removed when we replaced each edge block by a single edge. The number of edges directly above these nodes of degree 2 is at most k . Finally, the number of nodes of degree at least 3 in a tree is at most the number of leaves. Thus, there are at most $n - k$ such nodes, and thus the number of edges directly above these nodes is at most $n - k$. Thus, the total number of edge blocks is at most $(n - k) + k + (n - k) = 2n - k$. ■

Theorem 6: There always exists a 2-approximate Nash equilibrium solution that buys the minimum-cost Steiner tree T that connects the source and the receiver nodes.

Proof: Notice that the removal of an edge block \mathbf{e} from T will divide T into 2 connected components $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$. A critical property of the Steiner tree T is that \mathbf{e} constitutes the cheapest path between $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$, since otherwise we can obtain a cheaper tree than T that connects the source and the receiver nodes. This would lead to a contradiction since T , by definition, is the cheapest such tree.

Our payment algorithm assigns each edge block \mathbf{e} to a receiver r_i and asks r_i to reserve 1 unit of capacity on *all* the edges of \mathbf{e} . We use the notation $p_i = \{\mathbf{e}\}$ if edge block \mathbf{e} is assigned to receiver r_i and all other edge blocks of T are

assigned to the other receivers. Similarly, we say $p_i = \{\mathbf{e}, \mathbf{f}\}$ if edge blocks \mathbf{e} and \mathbf{f} are assigned to receiver r_i and all other edge blocks of T are assigned to the other receivers. Let $\chi_i\{\mathbf{e}\}$ and $\chi_i\{\mathbf{e}, \mathbf{f}\}$ denote the cheapest deviations of receiver r_i under the two possible strategies of her described above. Let $|\chi_i\{\mathbf{e}\}|$ and $|\chi_i\{\mathbf{e}, \mathbf{f}\}|$ denote the cost of these deviations to receiver i . We can now show the following lemmas.

Lemma 2: For an edge block \mathbf{e} between s and r_i on T , we have that $\chi_i\{\mathbf{e}\} = \{\mathbf{e}\}$.

Proof: For an edge block \mathbf{e} between s and r_i on T , the cheapest deviation $\chi_i\{\mathbf{e}\}$ of receiver r_i to strategy $p_i = \{\mathbf{e}\}$, where other receivers are reserving 1 unit of capacity on all other edge blocks of T , must reserve enough capacity to send 1 unit of traffic between $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$. The cheapest way to do this is to reserve 1 unit of capacity along the cheapest path between $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$, and thus $|\chi_i\{\mathbf{e}\}|$ is at least the cost of this path. Since \mathbf{e} constitutes the cheapest path between $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$, we have that $\chi_i\{\mathbf{e}\} = \{\mathbf{e}\}$. ■

Lemma 3: For any two edge blocks \mathbf{e} and \mathbf{f} between s and r_i on T , we have $|\chi_i\{\mathbf{e}, \mathbf{f}\}| \geq \max\{|\chi_i\{\mathbf{e}\}|, |\chi_i\{\mathbf{f}\}|\}$.

Proof: By Lemma 2, we know that $\chi_i\{\mathbf{e}\}$ is the cheapest path between s and r_i if all the edge blocks of T other than \mathbf{e} are bought by other receivers, and similarly for $\chi_i\{\mathbf{f}\}$. By the same argument as in the previous lemma, we know that $\chi_i\{\mathbf{e}, \mathbf{f}\}$ costs at least as much as the cheapest path between s and r_i if all the edge blocks of T other than \mathbf{e} and \mathbf{f} are bought by other receivers. The lemma holds since r_i can find a cheaper path between r_i and s if other receivers buy more edges. ■

Lemma 2 and Lemma 3 imply that we will form a 2-approximate Nash equilibrium if for each receiver r_i , we assign at most 2 edge blocks between r_i and s to it. This is due to the fact that if r_i is assigned edge blocks \mathbf{e} and \mathbf{f} , then $|\chi_i\{\mathbf{e}, \mathbf{f}\}| \geq \max\{|\chi_i\{\mathbf{e}\}|, |\chi_i\{\mathbf{f}\}|\}$ (by Lemma 3), which is exactly the maximum of the cost of \mathbf{e} and \mathbf{f} by Lemma 2. Thus, the cost of r_i 's best deviation $|\chi_i\{\mathbf{e}, \mathbf{f}\}|$ to its current strategy costs at least half of what r_i is currently paying. Since this is true for every receiver, this forms a 2-approximate Nash equilibrium. All that is left to show is that we can form such an assignment.

Recall that there are at most $2n$ edge blocks of T and we can make an assignment where each receiver is assigned at most 2 edge blocks. However, we cannot make an arbitrary assignment of edge blocks to receivers, since Lemma 2 and Lemma 3 hold only if a receiver r_i is assigned edge blocks that are between r_i and s . In order to make an assignment with this desired property, we will root T at s , and loop through the edge blocks of T in the reverse BFS order from s . For each edge block \mathbf{e} in this order, we select an arbitrary receiver r_i under \mathbf{e} that is not assigned 2 edge blocks yet, and assign \mathbf{e} to r_i . It is easy to show by induction that at the time the algorithm decides the assignment of an edge block \mathbf{e} , there is always a receiver r_i that is not assigned 2 edge blocks yet. By the above argument, this assignment creates a 2-approximate Nash equilibrium that purchases 1 unit of capacity on edges of T , and thus has cost at most twice that of OPT . ■

Since the Steiner Tree problem is not efficiently computable, in order to obtain a polynomial-time algorithm, we use an

approximation to Steiner Tree that costs at most twice the cost of OPT. We obtain the integral solution by starting with the primal-dual approximation algorithm for minimum-cost Steiner Forest problems (see, e.g. [29]). We use T to denote the tree returned by the primal-dual algorithm with $R \cup \{s\}$ being terminal nodes, and root it at s . Since the primal-dual algorithm uses the fractional optimal solution OPT as its lower bound, the cost of the integral solution returned by the primal-dual algorithm is not only within a factor of 2 of the cost of the minimum-cost Steiner Tree, but also the cost of OPT. The proof of Theorem 7 gives a polynomial-time algorithm that returns a strategy profile $\mathbf{p} = (p_i(e), i \in R, e \in E)$ that is a $(2 + \epsilon)$ -approximate Nash equilibrium, and reserves 1 unit of capacity on the edges of T' , a tree that connects all the receivers to the source and is cheaper than T .

Theorem 7: There is a polynomial-time algorithm that returns a $(2 + \epsilon)$ -approximate Nash equilibrium solution, whose cost is at most twice the cost of the social optimum solution.

Proof: As described above, we begin by constructing a tree T using the primal-dual 2-approximation algorithm for min-cost Steiner tree. In the proof of Theorem 6, we used the fact that each edge block \mathbf{e} of a minimum-cost Steiner Tree is the shortest path between the connected components $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$ that will arise upon removal of \mathbf{e} . However, this property does not necessarily hold for the integral solution T obtained as above. One strategy would be to go through all the edge blocks \mathbf{e} of T , and check whether \mathbf{e} is the shortest path between $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$. If \mathbf{e} is not the shortest path between $T_1(\mathbf{e})$ and $T_2(\mathbf{e})$, then we can obtain a tree T' cheaper than T by replacing \mathbf{e} with this shortest path. We can then repeat the same procedure starting with T' . It is trivial to see that this algorithm will allow us to obtain an approximation to the Steiner Tree such that for every edge block \mathbf{e} of the tree, \mathbf{e} is the shortest path between the connected components that will arise upon removal of \mathbf{e} . However, this algorithm may not terminate in polynomial-time. Instead, we proceed as follows.

Recall from the proof of Theorem 6, that we can cluster the edges into edge blocks and assign the edge blocks to the receivers in such a way that each receiver r_i is assigned at most 2 edge blocks, both of which are between r_i and s . Starting from T , our polynomial-time algorithm will generate cheaper trees by replacing some edge blocks with the shortest paths between the connected components that will arise upon removal of these edge blocks. However, in order to ensure a polynomial time bound, we will make such a replacement only if the cost of the tree is *substantially* decreased. Specifically, we classify the edge blocks into two categories. We will say that an edge block of a tree T' is *heavy* if its cost is at least $\frac{c(T')}{n^2}$ (where $c(T')$ is the cost of tree T' , and n is the number of receiver nodes), and we will say that an edge block is *light* if its cost is less than $\frac{c(T')}{n^2}$. When we are generating cheaper trees by replacing some edge blocks with the shortest paths between the connected components that will arise upon their removal, we will only consider the heavy edge blocks, as explained next in detail.

Let \mathbf{e} be an arbitrary heavy edge block of a tree T' , and let r_i be the receiver that \mathbf{e} is assigned to. We define $\chi_i\{\mathbf{e}\}$ as in the proof of Theorem 6, i.e., it is the best deviation of receiver

r_i if \mathbf{e} were removed and all other edges of T' are paid for by other receivers. If the cost of \mathbf{e} is at least $|\chi_i\{\mathbf{e}\}|(1 + \frac{\epsilon}{4})$, then we will replace \mathbf{e} with $\chi_i\{\mathbf{e}\}$ and obtain a cheaper tree. We will then restart the algorithm on this cheaper tree. This gives us a sequence of trees T_1, T_2, \dots , with $T_1 = T$ and each tree being cheaper than the previous one. Since the cost of a heavy edge block in a tree T_j is at least $\frac{c(T_j)}{n^2}$, the cost of a new tree T_{j+1} is at most $c(T_j)(1 - \frac{\epsilon}{n^2(4+\epsilon)})$. Thus, since with every new tree we reduce the cost by at least a factor of $1 - \frac{\epsilon}{n^2(4+\epsilon)}$, we know that this process can only continue for a polynomial number of steps. After this number of steps, we obtain a tree T' which is cheaper than T (and thus within a factor of 2 of OPT), and such that for every heavy edge block \mathbf{e} of T' , we have that $|\mathbf{e}| < |\chi_i\{\mathbf{e}\}|(1 + \frac{\epsilon}{4})$.

Once we have completed the tree generation process described above, we ask the receivers to pay for the entire cost of the heavy edge blocks that are assigned to them. Notice that if a receiver r_i is assigned only 1 heavy edge block, then $|p_i| = |\mathbf{e}| \leq (1 + \frac{\epsilon}{4})|\chi_i\{\mathbf{e}\}|$ by construction. If a receiver r_i is assigned 2 heavy edge blocks \mathbf{e} and \mathbf{f} , then we have $|p_i| = |\mathbf{e}| + |\mathbf{f}| \leq (2 + \frac{\epsilon}{2}) \max\{|\chi_i\{\mathbf{e}\}|, |\chi_i\{\mathbf{f}\}|\}$. Therefore, no receiver is paying more than $2 + \frac{\epsilon}{2}$ of the cost of her best deviation, since $|\chi_i\{\mathbf{e}, \mathbf{f}\}| \geq \max\{|\chi_i\{\mathbf{e}\}|, |\chi_i\{\mathbf{f}\}|\}$ by the arguments in Lemma 3.

Since we are done with deciding the payment for the heavy edge blocks, we will next decide the payment on the light edge blocks. For each light edge block \mathbf{e} , we will ask *all* the receivers to contribute to the cost of \mathbf{e} in proportion to what they have paid for on the heavy edge blocks. Notice that while each heavy edge block is paid by only one receiver, each light edge block is paid for by many receivers. Moreover, if a receiver is not assigned a heavy edge block, our algorithm does not ask her to contribute to the cost of light edge blocks at all, and this receiver is a “free-rider”. We prove below that this payment scheme is a $(2 + \epsilon)$ -approximate Nash equilibrium for the case when the number of receiver nodes is high, i.e., $n \geq 4 + \frac{8}{\epsilon}$. We will later discuss how we handle the case when the number of receiver nodes is small, i.e., $n < 4 + \frac{8}{\epsilon}$.

Since the cost of each light edge block is at most $\frac{c(T')}{n^2}$ and there are less than $2n$ edge blocks in total, then the total cost of the light edge blocks is at most $\frac{2 \cdot c(T')}{n}$. The total cost of the heavy edge blocks is, therefore, at least $\frac{(n-2) \cdot c(T')}{n}$. Since the total cost of the light edge blocks is small compared to $c(T')$, and the payments on the light edge blocks are made in proportion to what is paid for the heavy edge blocks, then the payment of each receiver is increased by a factor of at most $\frac{n}{n-2}$ due to their paying for the light edge blocks. Therefore, no receiver pays more than $(2 + \frac{\epsilon}{2}) \frac{n}{n-2}$ times her best response. Since $n \geq 4 + \frac{8}{\epsilon}$, we have $(2 + \frac{\epsilon}{2}) \frac{n}{n-2} \leq (2 + \epsilon)$. Therefore, the solution returned by the algorithm is a $(2 + \epsilon)$ -approximate Nash equilibrium solution as desired.

We have completed the proof for the case where the number of receivers is high, i.e., $n \geq 4 + \frac{8}{\epsilon}$. If the number of receivers is small, i.e., $n < 4 + \frac{8}{\epsilon}$, we can simply obtain the minimum-cost Steiner tree T in polynomial-time by using a brute-force technique and obtain a 2-approximate Nash equilibrium solution on it as described in the proof of Theorem 6. To do

this, we simply enumerate all the possible sets of nonreceiver nodes of degree 3 or more in the tree. Suppose that we know exactly what set of non-receiver nodes is going to be included in the min-cost Steiner tree with degree 3 or higher. Then, the min-cost Steiner tree is simply a minimum spanning tree on these nodes (together with the receiver nodes and the source) using the shortest-path metric, since these are the only nodes that may have degree higher than 2 in our tree. To find the min-cost Steiner tree, it is thus enough to enumerate all such sets of nodes, find the minimum spanning tree as described above, and then take the best of these trees. The number of non-receiver nodes of degree at least 3 in the Steiner tree is bounded by the number of receiver nodes (this is just because the number of nodes with degree at least 3 in a tree is at most the number of leaves). The number of receiver nodes is in turn bounded by $4 + \frac{8}{\epsilon}$. Thus, the number of sets which we must enumerate is polynomial if $n < 4 + \frac{8}{\epsilon}$, and so we can obtain the minimum-cost Steiner tree in polynomial time if $n < 4 + \frac{8}{\epsilon}$. Once we have this tree, we can obtain a 2-approximate Nash equilibrium (don't need the ϵ) for this simpler case using the algorithm in Theorem 6.

Running Time: The time for the initial primal-dual algorithm is $O(|E|^2)$. The algorithm then reduces the cost of the tree by a factor of $(1 - \frac{\epsilon}{n^2(4+\epsilon)})$ at each stage, with each stage taking $O(|E| \log |V|)$ time. Thus, for any constant ϵ , this process takes $O(|E| \log^2 |V| \log C)$, where $C = \sum_e c(e)$ is the total cost of the network. For the case when the number of receivers is at most $4 + \frac{8}{\epsilon}$, the runtime is constant for any constant ϵ (since we have to do a brute-force search here, the runtime depends on $2^{8/\epsilon}$). Thus, the total running time is $O(|E|^2 + |E| \log^2 |V| \log C)$. ■

VI. EXPERIMENTAL RESULTS

A. Experimental results for uniformly random networks

In this section, we run experiments on random networks to evaluate the tightness of our worst case bounds on the price-of-stability (PoS) and the equilibrium approximation factors, as outlined in the previous sections. We first present the experimental results for the 2-tier network model. We run the MST-based polynomial time algorithm as described in the proof of Theorem 2 to compute a Nash equilibrium (which guarantees a PoS of 2), and calculate β , the ratio of the cost at NE and the socially optimal cost (the latter obtained by solving a linear program). In the representative results shown in Table I, the total number of non-receiver nodes, i.e., the relay nodes plus the source, is varied from 5 to 25. For each value of the number of nodes, we compute the maximum and average values of β , namely β_{avg} and β_{max} , over 200 random runs (network samples). Edges are drawn between the non-receiver nodes randomly, in the following manner: the nodes are picked one by one, and the node picked at any step is connected to each of the nodes already included (in previous steps) with probability $p = 0.5$. If the chosen node remains unconnected at the end of the step, to maintain connectivity, an edge is drawn between this node and a randomly chosen node that is already included. The cost of these edges follows a uniform distribution between 1 and 100. Finally receivers

are assigned to the relay nodes randomly, such that each relay node is associated with at least one receiver: first we assign one receiver to each relay node, and then assign the remaining receiver nodes to the relay nodes randomly. The cost of edges between the receivers and their peers follows a uniform distribution between 1 and 5. Table I(a) and (b) shows the results for two different numbers of (receiver nodes/non-receiver nodes) ratios. From the results, we observe that the β value of the solution computed by our algorithm is very close to 1 on the average, and less than 1.5 in the worst case. From these results it seems, therefore, that the PoS for random 2-tier networks is quite small, and close to 1 on an average.

Next we present the experimental results for the general network model. We use the primal-dual min-cost Steiner tree approximation, as described in Theorem 7, to compute an α -approximate Nash equilibrium that attains a cost that is within a factor of β of the socially optimal cost. Theorem 7 implies that $\alpha \leq 2 + \epsilon$ and $\beta \leq 2$. Our experimental study on these random networks shows that the observations on β are similar to those observed for the 2-tier model, so we only show the α values in Table II. In these experiments, the total number of nodes (receivers, relays, and the source) is varied from 20 to 100. For each value of the number of nodes, we compute the average and maximum values of α , namely α_{avg} and α_{max} , over 500 random runs (network samples). Edges are drawn randomly as in the 2-tier model, but across all nodes instead of only the non-receiver nodes. The cost of these edges follows a uniform distribution between 1 and 100, as before. From Table II, we observe that α value of the solution computed by our algorithm is very close to 1 on average, and equals 1 for a large fraction of the networks. Therefore, for general networks, our algorithm generates a solution that is an exact equilibrium or extremely close to an equilibrium, and has low cost (typically within 1.5 times the socially optimal cost). For (receiver/non-receiver) node number ratio of 2 (not shown in the table), α was observed to be 1 in all 500 runs (network samples). Therefore, from these results we conclude that a larger (receiver/non-receiver) node number ratio creates better (smaller) α values.

B. Experimental results for scale-free networks

Now we consider scale-free networks generated according to the Barabasi-Albert algorithm [30]. In the algorithm, new nodes are added to the network one at a time. Each new node is connected to the existing nodes with a probability that is proportional to the number of edges that the existing nodes already have. As in the previous subsection, we present the experimental results for the 2-tier network model first. The parameters and conditions for the simulation are the same as for the previous one. We compute the maximum and average values of β , namely β_{avg} and β_{max} . As shown in Table III(a) and (b), we observe similar results of β of scale-free networks to those of uniformly random networks: β is very close to 1 in general, and less than 1.5 in the worst case.

Our experimental study on scale-free general network topologies shows that the observations on β are similar to those observed for the 2-tier model. Furthermore, under similar

#non-receivers	5	10	15	20	25
β_{avg}	1.093	1.199	1.201	1.195	1.194
β_{max}	1.49	1.48	1.37	1.35	1.36

(a) #receiver nodes = $2 \times$ #non-receiver nodes

#non-receivers	5	10	15	20	25
β_{avg}	1.079	1.174	1.162	1.147	1.130
β_{max}	1.37	1.42	1.34	1.29	1.28

(b) #receiver nodes = $4 \times$ #non-receiver nodes

TABLE I
PRICE-OF-STABILITY VALUES (β) FOR UNIFORMLY RANDOM 2-TIER NETWORKS.

#nodes	20	40	60	80	100
α_{avg}	1.0003	1.0003	1.0002	1.0009	1.0007
α_{max}	1.1250	1.1429	1.0833	1.2000	1.3333

(a) #receiver nodes = $0.5 \times$ #non-receiver nodes

#nodes	20	40	60	80	100
α_{avg}	1.0000	1.0009	1.0000	1.0000	1.0000
α_{max}	1.0000	1.1667	1.0000	1.0000	1.0000

(b) #receiver nodes = #non-receiver nodes

TABLE II
EQUILIBRIUM-APPROXIMATION VALUES (α) FOR UNIFORMLY RANDOM GENERAL NETWORKS.

#nodes	5	10	15	20	25
β_{avg}	1.0609	1.0698	1.0835	1.0905	1.1013
β_{max}	1.4136	1.2329	1.2628	1.2117	1.2869

(a) #receiver nodes = $2 \times$ #non-receiver nodes

#nodes	5	10	15	20	25
β_{avg}	1.0430	1.0622	1.0731	1.0771	1.0875
β_{max}	1.2857	1.2970	1.2354	1.1904	1.2363

(b) #receiver nodes = $4 \times$ #non-receiver nodes

TABLE III
PRICE-OF-STABILITY VALUES (β) FOR SCALE-FREE 2-TIER NETWORKS.

parameters and number of runs (network samples) as those considered in Table II, both the α_{avg} and α_{max} values were 1, implying that the equilibrium-approximation factor α was observed to be 1 in all networks simulated. Thus the observations for the scale-free general topology model are largely similar to the other models discussed before, and therefore the detailed results for this case are omitted here.

VII. PAYMENT STRATEGIES FOR SOCIAL OPTIMUM

In Section VI we showed results of simulations in which we form integral solutions which cost at most twice as much as the social optimum. Instead, in this section we consider several approaches of paying for the true (fractional) social optimum. While these payment strategies result in minimum overall cost (total price paid by the receivers for multicast data delivery), they may not be at NE. We evaluate these payment strategies on the basis of the following two performance metrics. The first one is α which describes how close a payment assignment is to a pure NE. The second metric is motivated by considering a third party that, in order to facilitate the formation of an equilibrium, pays the difference between the receivers' payments and their best deviations to all the receiver nodes. By doing this, i.e., giving subsidies to the receivers as encouragement for them not to change their strategies, this third party can make any solution, including OPT, become an equilibrium. Our metric, which we call the *subsidiary ratio*, measures how much this third-party facilitator needs to pay to the receivers as a fraction of the socially optimum cost.

As defined in Section III, a price vector \mathbf{p} is said to be a social optimum if it minimizes $\sum_{i \in R, e \in E} p_i(e) = \sum_e \theta(e)c(e)$, subject to satisfying the feasibility constraints (1) for all receivers $r_i \in R$. The optimal value is unique, but the payment

assignment solutions are various and they need not be a Nash equilibrium. Let p_i be the payment vector for receiver r_i , then the total payment of r_i is $|p_i| = \sum_{e \in E} p_i(e)$. Then α is calculated by comparing p_i with χ_i which is the best deviation of r_i , $\alpha = \max\{|p_i|/|\chi_i|\}$ for $r_i \in R$. Thus, α is the maximum factor by which any receiver could improve its cost if it changed its strategy. The subsidiary ratio is defined as:

$$\gamma = \frac{\sum_{r_i \in R} (|p_i| - |\chi_i|)}{\sum_{r_i \in R} |p_i|} \quad (4)$$

Thus, γ is the total amount that a third-party (e.g., the government) needs to subsidize the receivers in order to make the socially optimum solution become stable, divided by the total cost of the social optimum. For example, if γ were 0.2, this would mean that the third party needs to pay one fifth of the cost of OPT to the receivers to ensure that OPT become a Nash equilibrium.

A. Payment Schemes and Theoretical Analysis

We would like the receivers to pay for the social optimum in a way that α (equilibrium approximation factor) is minimized. For the general network model we would like all *witnessing* nodes (the nodes that at least use some amount of capacity of an edge) to pay for the edge. Suppose that for receiver $r_i \in R$, the capacity it uses on edge $e \in E$ in the social optimum is $b_i(e)$. More precisely, if less than $b_i(e)$ fraction of e is purchased, then constraint (1) for i would be violated, and so it needs at least $b_i(e)$ capacity on edge e .

We consider the following payment method. Each edge e has a vector of amount of capacities all receivers use on it: $b_1(e), b_2(e), \dots, b_{|R|}(e)$. Without loss of generality, assume $b_1(e) \leq b_2(e) \leq \dots \leq b_{|R|}(e)$. Then edge e is paid in

this way: r_1 pays $p_1(e) = \frac{b_1(e)}{n}c(e)$; r_i pays $p_i(e) = [\frac{b_1(e)}{n} + \sum_{j=2}^i \frac{b_j(e)-b_{j-1}(e)}{n-j+1}]c(e)$ for $2 \leq i \leq |R|$. Thus, the cost of each fraction of capacity is equally shared among the receiver nodes that need it. The total payment of r_i is $|p_i| = \sum_{e \in E} p_i(e)$. Then we can calculate α and γ by comparing p_i and the best deviation of r_i .

For the 2-tier network model, we consider two methods to pay for the social optimum. The first one is the same as that for the general network model. The second method is that for edge e , instead of forming the payment by all the witnessing nodes, we consider the witnesses that are the closest to the edge. In particular, in a 2-tier network $V = \{s\} \cup L \cup R$, $R = \{r_1, r_2, \dots, r_n\}$ is the set of receivers, and $L = \{l_1, l_2, \dots, l_k\}$ is the set of relay nodes. For an edge $e = e(i, j)$, there are 3 cases: (1) $i = s$ and $j \in L$ or vice visa; (2) $i \in R$ and $j \in L$ or vice visa; (3) $i \in L$ and $j \in L$. In the first case, without loss of generality, let $i = s$, then receivers adjacent to j pay for the edge; in the second case, without loss of generality, let $i \in R$ and $j \in L$, then only receiver i pays for the edge; in the third case, receivers adjacent to i as well as those adjacent to j pay for the edge.

Before presenting the experimental results, we show theoretically that α can be very large in the worst case. Consider a general network model of circular topology with N nodes and N edges. Node 1 is the source s , and the other $N - 1$ nodes are receivers r_1, \dots, r_{N-1} . The cost of the edge connecting s and r_1 is c ; the cost of other $N - 1$ edges is Kc , for some $K > 1$. The social optimum is to use half unit of capacity of all the edges, which leads to a total cost of $C = \frac{c}{2} + (N - 1)\frac{Kc}{2}$. Then every receiver gets half unit of flow from both clockwise and counterclockwise directions. Therefore every receiver pays the same amount, which is $|p_i| = \frac{C}{N-1} = \frac{c}{2(N-1)} + \frac{Kc}{2}$, $i \in 1, 2, \dots, N - 1$. Now we consider the deviation of r_1 . Since it can get $\frac{N-2}{N-1}$ unit of flow from the source through the available capacity paid by other receivers, its best strategy is to pay $\frac{1}{N-1}$ unit of flow on the edge connecting itself to the source which gives a cost of $\frac{c}{N-1}$. Then $\alpha = (\frac{c}{2(N-1)} + \frac{Kc}{2}) / \frac{c}{N-1} = \frac{1}{2}[K(N-1) + 1]$. If K is large, α will be large as well. This example easily extends to show that α can be very large for 2-tier networks as well.

B. Experimental results for NE approximation factor α

We present experimental results for the NE approximation factor α in the following. We have considered both general and 2-tier network models, and each of them can be scale-free (generated by Albert-Barabasi model) or uniformly random. For the general network model, we use the method that all witnessing nodes of an edge pay for the edge. For the 2-tier network model, we consider two payment methods. In the first method all witnessing nodes pay, and in the second method only the closest witnessing nodes pay. Therefore we run experiments on a total of 6 cases: (1) scale-free general networks (SG); (2) uniformly random general networks (UG); (3) scale-free 2-tier networks with all witnesses to pay (S2A); (4) uniformly random 2-tier networks with all witnesses to pay (U2A); (5) scale-free 2-tier networks with closest witnesses to

pay (S2C); (6) uniformly random 2-tier networks with closest witnesses to pay (U2C).

In general network model the total number of nodes is 20. In the 2-tier network model, the number of relay nodes plus the source is 20, and each relay node may have up to 3 receivers. We also run the experiments on other total numbers of nodes and the results are similar; thus we only present the results for the case of 20. We compute the average of α and γ over 100 random runs (network samples). The cost of edges and the probability to pick up an edge in network generation stay the same as in Section VI.

As shown in Figure 7, the average of α is very large in the first 2 cases of scale-free and uniformly random general networks, which means they are far from the pure Nash equilibrium. In the following 4 cases of 2-tier network model, the average of α is less than 2 (or less than 1.5 more accurately). So the method performs better in the 2-tier model than in the general network model.

It is clear from our findings, that in order to form an approximately-stable solution for the general network model, it is far preferable to relax the requirement that the cost of the solution is socially optimum (i.e., that $\beta = 1$). With this constraint, solutions generated are far from stable, with receivers often being able to decrease their cost by a factor of 5 or 10 after switching their strategy. If we relax this constraint and allow for slightly more expensive solutions (e.g., $\beta = 1.5$), then as we saw in Section VI we can use our algorithms to obtain almost exact Nash equilibrium ($\alpha < 1.001$ on average) which will also be close to the optimum in cost.

For the 2-tier model, however, our findings show a more subtle tradeoff. If our goal was to obtain the true social optimum ($\beta = 1$), our simulations in this section suggest that we can usually form a 1.5-approximate Nash equilibrium on this solution. If, on the other hand, we wanted an *exact* Nash equilibrium ($\alpha = 1$), then our findings from Section VI suggest that we can still obtain a very cheap solution ($\beta < 1.2$).

C. Experimental results for subsidiary ratio γ

We now present experimental results for the subsidiary ratio γ . The parameters are the same as those for calculating α . As shown in Figure 8, γ is very large in cases UG and U2A. Notice that cases UG, U2A and U2C are uniformly random networks, and γ in these cases is greater than it is in cases SG, S2A and S2C, which are scale-free networks. Thus we observe that better subsidiary ratios are attained in scale-free networks than in uniformly random networks. Indeed, for the case of scale-free graphs, it is enough to subsidize the receiver nodes with less than 1/5 of the total cost of the social optimum, in order to prevent all receiver nodes from deviating, and thus forming a stable solution.

In addition, through comparing the subsidiary of SG with that of S2A and S2C, and comparing the subsidiary of UG with that of U2A and U2C, we can conclude our algorithm works better for the 2-tier model than for the general network model. We also get this point through the results for α above.

Another interesting point that draws our interest is that γ in cases S2C and U2C is less than it is in cases S2A and

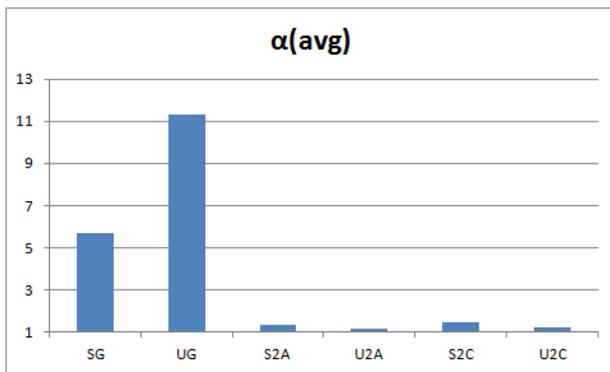


Fig. 7. Average α for the 6 cases. (The minimum possible value of α is 1.)

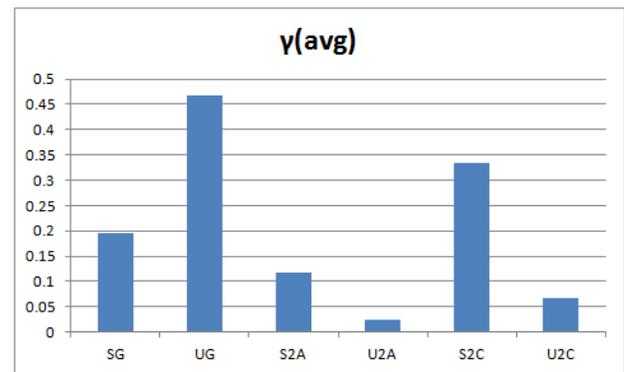


Fig. 8. Average γ for the 6 cases. (The minimum possible value of γ is 0.)

U2A, and γ in cases S2C and U2C is very small and close to zero. Notice that in S2A and U2A all witnesses need to pay for the edge, but in S2C and U2C only closest witnesses pay for it. This is interesting because intuitively if the cost is shared by all witnessing nodes, then it seems to give better performance, but it may require more computation since all the nodes need to be considered. On the other hand, if the closest witnesses of an edge pay for it, we just need to consider the two ends of the edge, which requires less computation. Then based on the experimental results we come to the conclusion that the method of closest witnesses wins over the method of all witnesses on both performance and computation. Therefore for 2-tier networks, whether scale-free or uniformly random, the social optimum can be paid by the closest witnesses of the edges.

Our payment solutions for the social optimum return a payment assignment that only needs a small subsidiary from the government to become stable. Thus, although the social optimum may not be “close” to a Nash equilibrium in terms of α in the case of general networks (as shown in Figure 7), a third-party would only need to subsidize 1/5 of its cost (or 1/2 for non-scale free graphs) in order to make the socially optimum solution become stable. For the 2-tier model, these numbers become much smaller, and we see that by subsidizing less than 3 percent of the socially optimum solution (8 percent for non-scale-free), it is enough to make the socially optimum solution into a stable one.

VIII. CONCLUSION

We show that pure Nash equilibrium is guaranteed to exist in our capacity allocation game. We provide a polynomial-time algorithm that computes a $(2 + \epsilon)$ -approximate Nash equilibrium whose total cost is at most 2 times the social optimum for general network topologies. Besides, we specifically consider 2-tier network topologies and provide a polynomial-time algorithm that computes a pure Nash equilibrium whose total cost is at most 2 times the social optimum. The experimental results show that both of the approximation factor α and the ratio of the cost β are very close to 1. It means our algorithm generates a solution that is an exact equilibrium or extremely close to an equilibrium, and has low cost.

We also consider the problem of paying for the social optimum and provide a polynomial-time algorithm that computes

payment assignments. We evaluate the experimental results based on approximation factor α and the subsidiary ratio γ . The results show that our algorithm returns smaller and acceptable α and γ in the 2-tier network model than in the general network model. We also notice that in the experimental results the subsidiary ratio of scale-free network model is smaller than that of uniformly random model. Moreover, the results show that for 2-tier networks the strategy of closest witnesses to pay provides a better subsidiary ratio than that of all witnesses to pay.

It is worth noting that in our model, the purchased capacity on a link is available for use in both directions. Alternatively, one could consider a model in which link capacities are purchased separately in each direction of the link. Extension of our analysis and results to that model remains an interesting direction for future research.

REFERENCES

- [1] “Web could collapse as video demand soars”, *Daily Telegraph*, 7th April, 2008.
- [2] *ITU IPTV Focus Group*, <http://www.itu.int/ITU-T/IPTV/>
- [3] “Future Looks Bright For IPTV”, *Satellite Today*, 2nd May, 2005.
- [4] <http://www.youtube.com>
- [5] “Cisco Virtual Video Infrastructure Managing Complexity and Scale in a Next-Generation Video Network”, White Paper, Cisco Systems, 2008.
- [6] R. Ahlswede, N. Cai, S. Li, and R. Yeung. “Network information flow,” *IEEE Trans. Inform. Theory*, vol. 46, no. 4, p. 1204-1216, July 2000.
- [7] E. Anshelevich and B. Caskurlu. Price of Stability in Survivable Network Design. In *Theory of Computing Systems*, Volume 49, Number 1 (July 2011), pp. 98–138.
- [8] E. Anshelevich, A. Dasgupta, J. Kleinberg, É. Tardos, T. Wexler, T. Roughgarden. The Price of Stability for Network Design with Fair Cost Allocation. *SIAM Journal on Computing*, Volume 38, Issue 4 (November 2008), pp. 1602-1623.
- [9] E. Anshelevich, A. Dasgupta, É. Tardos, T. Wexler. Near-Optimal Network Design with Selfish Agents. In *Theory of Computing*, Volume 4 (2008), pp. 77-109.
- [10] Vittorio Bilo, Ioannis Caragiannis, Angelo Fanelli, and Gianpiero Monaco. Improved lower bounds on the price of stability of undirected network design games. *SAGT 2010*.
- [11] C. Chekuri, J. Chuzhoy, L. Lewin-Eytan, J. Naor, and A. Orda. Non-cooperative multicast and facility location games. Proceedings of the 7th ACM Conference on Electronic Commerce (EC), Ann Arbor, Michigan (2006), pp. 72-81.
- [12] H. Chen, T. Roughgarden. Network Design with Weighted Players. In *SPAA 2006*.
- [13] H. Chen, T. Roughgarden, and G. Valiant. Designing Networks with Good Equilibria. *SODA 2008*.
- [14] G. Christodoulou and E. Koutsoupias. On the price of anarchy and stability of correlated equilibria of linear congestion games. *ESA*, 2005.

- [15] A. Epstein, M. Feldman, and Y. Mansour. Strong Equilibrium in Cost-Sharing Connection Games. *EC* 2007.
- [16] J. Feigenbaum, C. Papadimitriou, S. Shenker. Sharing the Cost of Multicast Transmissions. *Journal of Computer and System Sciences* 63 (2001), 21–41.
- [17] A. Fiat, H. Kaplan, M. Levy, S. Olonetsky, R. Shabo. On the Price of Stability for Designing Undirected Networks with Fair Cost Allocations. *Proceedings of ICALP* 2006, pp. 608–618.
- [18] T. Ho, M. Medard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," *Proc. 41st Annual Allerton Conf. Comm., Control, & Computing*, Monticello, IL, Oct. 2003.
- [19] M. Hoefer. Non-cooperative Facility Location and Covering Games. In *ISAAC* 2006.
- [20] M. Hoefer, P. Krysta. Geometric Network Design with Selfish Agents. *COCOON* 2005.
- [21] S.-Y. R. Li, R. W. Yeung, and N. Cai. Linear network coding. *IEEE Trans. Inform. Theory*, vol. 49, no. 2, pp. 317-381, Feb. 2003.
- [22] J.R. Marden and M. Effros. The Price of Selfishness in Network Coding. *IEEE Trans. Inform. Theory*, vol. 58, no. 4, pp. 2349-2361, 2012.
- [23] J. Price and T. Javidi. Network Coding Games with Unicast Flows. *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 7, pp. 2349-2361, 2012.
- [24] E.N. Ciftcioglu, Y.E. Sagduyu, R. Berry and A. Yener. Cost sharing with network coding in two-way relay networks. *Proc. 47th Allerton Conference*, pp. 1302-1316, 2008.
- [25] S. Bhadra, S. Shakkottai, and P. Gupta. Min-Cost Selfish Multicast With Network Coding. *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 5077-5087, 2006.
- [26] Z. Li. Cross-Monotonic Multicast. *Proc. Infocom 2008*, pp. 1588-1596.
- [27] N. Nisan, T. Roughgarden, É. Tardos, and V. V. Vazirani (eds.), *Algorithmic Game Theory*, Cambridge University Press.
- [28] P. Sanders, S. Egner, and L. Tolhuizen. Polynomial time algorithms for network information flow. *Proc. of the fifteenth annual ACM symposium on Parallel Algorithms and Architectures*, p. 286–294, 2003.
- [29] V.V. Vazirani, *Approximation algorithms*, Springer-Verlag, Berlin, 2001.
- [30] R. Albert and A.-L. Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 2002.