Profit-Maximizing Multicast Pricing by Approximating Fixed Points

Aranyak Mehta[*] *Scott Shenker*[†] *Vijay V. Vazirani*[*]

Abstract

We describe a fixed point approach for the following stochastic optimization problem: given a multicast tree and probability distributions of user utilities, compute prices to offer the users in order to maximize the expected profit of the service provider. We show that any optimum pricing is a fixed point of an efficiently computable map. In the language of classical numerical analysis, we show that the non-linear Jacobi and Gauss-Seidel methods of coordinate descent are applicable to this problem. We provide proof of convergence to the optimum prices for special cases of utility distributions and tree edge costs.

1 Introduction

We study a "stochastic optimization" problem for pricing multicast transmissions. A single service provider is sending multicast packets to a large number of users. For efficiency, multicasting is achieved over a multicast tree [DC90, DEF$^+$94], with the service provider at the root of the tree and one user per node of the tree. Packets are sent down the tree and are duplicated at every branch node. We assume that this duplication is achieved free of cost. However, there is a cost involved in transmitting over each edge of the tree. User i has a utility u_i for the service. The service provider does not know u_i, but he does know that u_i comes from a distribution F_i. The service provider offers to sell the service to user i at price p_i. The user accepts iff $u_i \geq p_i$. The service is then provided to all the users who accept, through a *subtree* of the original, universal multicast tree, at a cost equal to the cost of the subtree and a revenue equal to the sum of the prices accepted. The goal of the service provider is to find prices p_i so as to maximize his *expected profit*. We describe the problem in detail in Section 2.

This problem is motivated by the recent work on mechanism design in general [MS01, NR99, JV01] and mechanism design for multicasting in particular [FPS01, GHW01]. The standard mechanism design problem involves a mechanism designer (in the case of multicasting, the service provider) and a set of users. Each user is assumed to be selfish, and tries to maximize some private utility value for some service that the designer can offer him. This utility is unknown to the designer. The goal of the designer is (roughly) to use the selfishness and rationality of the users to design a protocol that will elicit truthful responses (regarding the utilities) from the users and hence fulfill some goal of the designer himself.

The goal of the designer in mechanism design problems is usually related to social welfare - either maximizing the efficiency of the protocol (the difference between the sum of the utilities of the

[*]Address: College of Computing, Georgia Institute of Technology, Atlanta, GA 30332–0280, E-mail: {aranyak,vazirani}@cc.gatech.edu. Research supported by NSF Grant CCR-9820896.

[†]ICSI, 1947 Center Street, Suite 600, Berkeley, CA 94704–1198, E-mail: shenker@icsi.berkeley.edu

users who get the service and the cost of providing the service to these users) or maintaining a balanced budget (such that the service provider retrieves exactly the cost of providing the service). For multicasting these questions were investigated in [FPS01] (in terms of the network complexity of achieving these goals). The more recent work of Fiat et.al. [FGHK02] asks a different question - are there mechanisms which maximize the profit to the service provider? This is a *profit maximizing* question as opposed to the *cost sharing* questions above. Fiat et.al. find mechanisms which are competitive in terms of maximizing profit. In fact, they study the profit maximization question for a broad class of problems (which includes multicasting). They find a constant factor mechanism, in the restricted case that there is more than one user at every node of the multicast tree and no user's bid "dominates" the bids of other users in the same node. Furthermore, the constant factor depends critically on the assumption that the optimal profit is itself a constant factor more than the cost involved.

In this paper we also address the profit maximization question, but in a different context. We remove all restrictions on the number of agents at each node and on the optimal profit. However, we assume that while the individual values of the utilities are not known to the service provider, their distribution is. This is not the typical assumption made in the mechanism design literature, but it might apply to settings, such as what we consider here, where there are a random set of users drawn from a pool whose overall characteristics are well-known from market research or other means.

On assuming knowledge of the distributions of the utilities, the problem changes to a stochastic optimization problem. Recent work in the same vein is that of Kleinberg, Rabani and Tardos [KRT97] - they study the stochastic versions of minimum makespan, bin-packing and knapsack, i.e. versions in which the item/job sizes are not known but only the distributions from where they are picked. However, in their case the non-stochastic problems are well-known NP complete problems. Hence they look for approximation algorithms. In our case, the non-stochastic problem is polynomial time solvable: knowing the utility of a user, the only options are to set the price to the utility or to not offer the service at all. Which users to make that offer to can be figured out by dynamic programming (e.g. trickling a value up the tree which says "if serving my parent then the maximum extra profit you can get through the subtree rooted at me is so much.")

Thus our problem is (in terms of difficulty) between (1) knowing the utilities (easy) and (2) not knowing anything about the utilities (hard). We do not know whether this stochastic problem is polynomial time solvable even for "nice" distributions. A naive approach would be to assume the utility of each user to be his expected utility and offer that as the price. However, it can be shown that this approach can give an arbitrarily large approximation factor, in fact it may result in a loss even if the situation is profitable.

The main difficulty in the problem is that the probability that a user accepts a price depends on the price, but the price in turn depends on the probability that the service is routed through that user. A direct numerical method would not work because the function we want to optimize takes exponential space to even write down. We suggest an indirect fixed point approach to the problem - we find a map which is easy to compute such that the optimum prices are a fixed point of this map. Fixed point methods for stochastic optimization problems have been used elsewhere, e.g. in traffic route guidance systems [BBABI98, BC01].

Results: We describe a quadratic time computable map such that the optimal prices form a fixed point of this map, and hence a simple fixed point approach may lead to the optimal solution. Fixed point problems have been well studied, and several general purpose methods are known in the literature (see e.g., [OR70]). We analyze the simplest iteration method for special distributions.

These iterations are known in the numerical analysis literature as the Jacobi or the Gauss-Seidel method. We show that the simple iterations converge geometrically to the optimum prices (and hence yield a fully polynomial approximation) in special cases for the uniform distribution and for the exponential distribution.

This approach can be used in any multi-variate optimization problem in which the expected cost of adding one variable when the rest of the variables are fixed at arbitrary values can be computed easily.

This paper has three sections. In Section 2 we define the expected profit maximization problem. In Section 3 we describe a map for which solutions to the problem are fixed points. In Section 4 we study two particular distributions (the exponential and the uniform) and show that for special cases of edge costs the simplest iterative process does converge to a solution of the profit maximization problem.

2 The Expected Profit Maximization Problem

To motivate the problem we study in this paper, consider first the simple situation in which there is a service provider who offers a service and a single user who requires this service. The cost to the service provider to serve the user is c. The user has a utility value u which is chosen from a (cumulative) probability distribution F. The provider does not know u, but he does know F. F has non-negative support (i.e. $F(0) = 0$). We also assume throughout that computing $F(x)$ is a unit time operation for all $x \in \mathbf{R}$.

The service provider has to set the price p at which he offers the service to the user. If $u \geq p$ then the user accepts the offer and the provider's profit is $(p - c)$; if $u < p$ then the user rejects the offer, the service is not provided, and there is no profit. The goal of the provider is to maximize his expected profit $(1 - F(p))(p - c)$. We assume throughout that the distribution F is such that there is precisely one value of p which maximizes the expected profit. Thus we have the following definition:

Definition 1 *The* single user profit maximizing function *for a distribution F is the function $\Psi_F : \mathbf{R}_+ \to \mathbf{R}_+$ such that $\Psi_F(c)$ is the price which maximizes the expected profit in serving a single user at a cost of c, when the user's utility comes from F.*

We restrict attention to distributions F such that $\Psi_F(c)$ can be computed in polynomial time, and assume throughout that computing $\Psi_F(c)$ is a unit time operation for all $c \in \mathbf{R}$. We also assume that Ψ_F is a monotonically increasing function.

For example, if F is the exponential distribution E with parameter λ then the expected profit is $e^{-\lambda p}(p - c)$, and this is maximized at $\Psi_E(c) = c + \frac{1}{\lambda}$. If F is the uniform distribution U on $[0, 1]$ then the expected profit is $(1 - p)(p - c)$ when $0 \leq p \leq 1$ and 0 when $p > 1$, and this is maximized at $\Psi_U(c) = min\{1, \frac{1+c}{2}\}$.

A subtle point to be noted here is that the protocol must be *price fixing*, in the sense that if the user rejects the offer and offers in return another price at which he is willing to buy the service, the provider should not accept this counter offer, *even if* it is greater that c. This is essential, otherwise the user will claim a lower utility than the true value, and the protocol may be suboptimal.

Now consider the situation in which there are several users, each with some utility, independent of the other users. If these users have to be provided the service independent of each other (say by direct links, one to each user) then the maximum expected profit is computed by simply computing the Ψ values for each user. Of course, service is never delivered in this manner.

For our problem we assume that we are given a *fixed* multicast tree. The service provider is at the root of the tree and there are n other nodes. Each node (other than the root) corresponds to a user, and we use the terms node i and user i interchangeably. The edges of the tree have specified edge costs, which could be different for different edges. The service can be provided to the nodes only through the edges of the tree. We assume that there is no cost of duplicating the service at the branching nodes of the tree, whenever such duplication is required. Each node has a utility value which determines how much it is willing to pay for the service. Node i's utility u_i is distributed according to the (cumulative) distribution F_i, independent of the utilities of the other nodes. These distributions are known to the service provider. The provider has to name the prices at which he is willing to provide the service to the users. The prices need not be the same for all users. Again, the goal of the provider is to set prices so as to maximize his expected profit. Let $\mathbf{p} = (p_1, p_2, ..., p_n)$ be the offered price vector (node i is offered price p_i). Node i accepts the offer if $u_i \geq p_i$ and rejects otherwise. Let A be the (random) set of nodes that accept the offer.

Definition 2 *The* accepting subtree *for a given price vector* $\mathbf{p}$ *is the smallest subtree of the multicast tree that contains all the nodes in* A*.*

The service is routed through the accepting subtree to the nodes in A. Nodes which are not in the accepting subtree have rejected the offer and are not provided the service. Again, we insist that a node which is not in A but is in the accepting subtree is also not provided the service, even though the service is routed through it. This is essential, otherwise the users may claim a utility lower than the true value and hope to end up in the accepting subtree and get the service at lesser price.

Definition 3 *The* cost *of providing the service is the cost of the accepting subtree. The* revenue *is the sum of the prices which were accepted.* Profit = revenue – cost.

Every price vector $\mathbf{p}$ results in some expected profit (the expectation is over the outcomes of the utility values drawn from the distributions).

The Expected Profit Maximization Problem: Given the multicast tree with edge costs and given the distributions from which the utilities of the users are picked, find a price vector $\mathbf{p}^*$ which maximizes the expected profit.

Optimum prices exist due to the extreme value theorem for continuous functions (we can restrict the function to a bounded region, outside which the profit is always negative). The profit function is random, as it depends on which nodes accept. To write down the expected profit itself would take exponential number of terms. Hence it is not possible to use numerical methods on the expected profit function itself – we need to use more indirect methods.

We approximate the optimum prices offered to the nodes, rather than the optimum profit. In fact, it is not clear how we can calculate the expected profit from the prices, as the function is exponentially big.

3 A Fixed Point Method

In this section we suggest a solution to the Expected Profit Maximization Problem using fixed points of a certain map. In general, finding fixed points of maps is not known to be in polynomial time. We suggest a very simple iterative method whose correctness and efficiency we analyze for special cases in the next section.

3.1 Relatively Optimum Pricing

Given a price vector $\mathbf{p} = (p_1, p_2, ..., p_n)$ we denote by $\mathbf{p}_{-i}$ the vector $(p_1, p_2, ...p_{i-1}, p_{i+1}, ...p_n)$, i.e. $\mathbf{p}_{-i}$ is a price vector without the ith coordinate. A price vector $\mathbf{p}_{-i}$ is interpreted as fixing the prices for all the nodes except node i. For $\mathbf{p}_{-i}$ fixed, different prices offered to node i will result in different profits.

Definition 4 *Given a vector $\mathbf{p}_{-i}$, the relatively optimum price for node i (or the optimum price for node i relative to $\mathbf{p}_{-i}$) is the price p_i^* for node i which maximizes the expected profit conditional on the fact that prices for the other nodes are fixed at $\mathbf{p}_{-i}$.*

The definition assumes that the relatively optimum price is unique. We show next that this is indeed the case, and that p_i^* can be computed in $O(n)$ time.

Fix $\mathbf{p}_{-i}$, i.e. prices for all nodes except node i. Ignoring node i, consider all possible outcomes for the other $(n-1)$ nodes. The probability space Ω is the set of all possible 2^{n-1} events: whether node j accepts or rejects the offer p_j, for $j \neq i$. For an $\omega \in \Omega$ there is a corresponding accepting subtree. This is a subtree of the multicast tree which may or may not contain i depending on whether some descendant of i in the multicast tree accepted the offer made in $\mathbf{p}_{-i}$ or not.

Let X be the random variable which denotes the cost of connecting node i to the accepting subtree. If some descendant of node i in the original multicast tree accepts the offer made in $\mathbf{p}_{-i}$ then $X = 0$, otherwise X is the distance of node i to the closest ancestor in the multicast tree which accepts the offer in $\mathbf{p}_{-i}$. Recall the definition of the single user profit maximizing function $\Psi_F(c)$.

Lemma 1 $p_i^* = \Psi_{F_i}(E[X])$.

Proof : For an outcome $\omega \in \Omega$, let $r(\omega)$ be the revenue in that outcome, i.e. the sum of the prices $p_j, j \neq i$ that are accepted. Let $c(\omega)$ be the cost of the accepting subtree. The expected value of the profit as a function of p_i is:

$$(1 - F_i(p_i)) \sum_{\omega \in \Omega} P[\omega] \, (r(\omega) - c(\omega) + p_i - X(\omega)) + F_i(p_i) \sum_{\omega \in \Omega} P[\omega] \, (r(\omega) - c(\omega))$$

Here the first summation is the expected profit given that node i accepts, which happens with probability $(1 - F_i(p_i))$; and the second summation is the expected profit given that node i rejects, which happens with probability $F_i(p_i)$. Splitting the first summation, and then simplifying, we get:

$$\begin{aligned}
& (1 - F_i(p_i)) \sum_{\omega \in \Omega} P[\omega] \, (r(\omega) - c(\omega)) + (1 - F_i(p_i)) \sum_{\omega \in \Omega} P[\omega] \, (p_i - X(\omega)) + F_i(p_i) \sum_{\omega \in \Omega} P[\omega] \, (r(\omega) - c(\omega)) \\
= \; & (1 - F_i(p_i)) \sum_{\omega \in \Omega} P[\omega] \, (p_i - X(\omega)) + \sum_{\omega \in \Omega} P[\omega] \, (r(\omega) - c(\omega)) \\
= \; & (1 - F_i(p_i))(p_i - E[X]) \; + \; \Pi(\mathbf{p}_{-i})
\end{aligned}$$

where $\Pi(\mathbf{p}_{-i})$ is the expected profit in the absence of node i. Since $\Pi(\mathbf{p}_{-i})$ is independent of p_i, it follows (from the definition of the single user profit maximizing function) that the value for p_i which maximizes the above function is $p_i^* = \Psi_{F_i}(E[X])$. $\square$

Lemma 2 *For any given* $\mathbf{p}_{-i}$ *we can compute the relatively optimum price* p_i^* *in* $O(n)$ *time.*

Proof : By Lemma 1 it suffices to prove that $E[X]$ can be computed in $O(n)$ time. If node i is at a distance of j edges from the root, then X takes $j+1$ distinct values: X equals zero if some descendant of node i accepts the offer, X equals the cost of the edge connecting node i to its parent if no descendant of i accepts and the parent of i accepts, and so on. For any given $\mathbf{p}_{-i}$ the probabilities of all of these events can be computed in $O(n)$ time, hence also the relatively optimum price. □

3.2 A Fixed Point Map

Lemma 1 suggests the following subroutine. It takes as input a price vector $\mathbf{p}$ and an index i between 1 and n, and it modifies the price vector.

RELATIVE-OPT$(\mathbf{p}, i)$:
Obtain $\mathbf{p}_{-i}$ from $\mathbf{p}$ by deleting the ith coordinate.
Compute the optimum price p_i^* for node i relative to $\mathbf{p}_{-i}$.
Replace the ith coordinate of $\mathbf{p}$ by p_i^*.

This motivates the following map which takes a price vector to another price vector.

REFINE$(\mathbf{p})$:
for $i = 1$ to n
  RELATIVE-OPT$(\mathbf{p}, i)$

We know from Lemma 2 that **RELATIVE-OPT** is a linear time subroutine, and so **REFINE** can be implemented in $O(n^2)$ time.

Definition 5 *A price vector* $\mathbf{p}$ *is a* fixed point *of the map* **REFINE** *if* $\mathbf{p} = $ **REFINE**$(\mathbf{p})$.

Thus $\mathbf{p}$ is a *fixed point* of **REFINE** if for every i, p_i is the optimum price for node i relative to $\mathbf{p}_{-i}$. It is clear that any optimal price vector is a fixed point of **REFINE**. Consider a very simple iterative scheme:

SIMPLE-ITERATIONS:
Initialize $\mathbf{p}$ to all coordinates at infinity.
Repeat:
  REFINE$(\mathbf{p})$

The next lemma proves that in these simple iterations the price for each node is decreasing monotonically. Suppose that $\mathbf{p}_{-i}$ and $\mathbf{q}_{-i}$ are price vectors such that $\mathbf{p}_{-i}$ is greater than or equal to $\mathbf{q}_{-i}$ in every coordinate. Let p_i^* be the optimum price for node i relative to $\mathbf{p}_{-i}$ and let q_i^* be the optimum price for node i relative to $\mathbf{q}_{-i}$.

Lemma 3 $p_i^* \geq q_i^*$.

Proof : It can be shown that $E[X]$ for $\mathbf{p}_{-i}$ is greater than or equal to $E[X]$ for $\mathbf{q}_{-i}$. Since Ψ_{F_i} is an increasing function, Lemma 1 implies that $p_i^* \geq q_i^*$. □

While the SIMPLE-ITERATIONS cause the prices to decrease monotonically (or increase monotonically if we initialize all prices at 0), it is not clear whether they will converge to an optimum pricing:

Question 1: Do SIMPLE-ITERATIONS converge to an optimum price? If so, do they converge rapidly?

SIMPLE-ITERATIONS are essentially Jacobi or Gauss-Seidel iterations, which have been well studied in numerical analysis. However, there are no general convergence theorems that seem to directly apply to this problem. Note that we have to also exclude the case that the iterations converge to some point which is not an optimum price, but is a sub-optimal fixed point. Again there are general theorems that guarantee this, but they require convexity of the function being maximized, and that may not be true for our function for general utility distributions.

If the answer to Question 1 is in the negative then the immediate question is whether we can use other methods like Newton's Methods or Scarf's algorithm [Sca67] to find the optimum pricing fast.

Question 2: Are there other numerical methods which find rapidly the optimum prices as a fixed point of **REFINE**?

4 Analysis for particular distributions

In this section we provide a partial answer to Question 1 (from Section 3.2). We consider two distributions for the utilities - the exponential and the uniform. With certain assumptions on the parameters of the distributions (or alternatively, on the cost of the tree edges) we prove that **REFINE** is a contractive map. This means that the iterations converge geometrically to a unique fixed point (which is the optimal pricing). Such contraction results are well known to be difficult to prove. However, we believe that the simple iterations are a good technique to solve the general problem, and may be applied to other maximization problems of a similar nature. In case they do not converge fast, more complicated fixed point methods can be used. These may run much longer but may guarantee a good result.

4.1 The Exponential Distribution

In this section we assume that the utility of each node comes independently from the exponential distribution E with parameter λ.

As noted in Section 2, the single user profit maximizing function is $\Psi_E(c) = c + \frac{1}{\lambda}$.

The probability that node i accepts price p_i is $e^{-\lambda p_i}$. Let $q_i = 1 - e^{-\lambda p_i}$, the probability that node i rejects. Using Lemma 1 we can obtain expressions for the relatively optimum price for each node.

For example, if the tree is simply n nodes in a line with all edge costs c, we have the following expressions (the nodes are in order with node 1 closest to the root and node n the farthest). To see the equation S_i observe that the cost of joining i to the accepting subtree of the rest of the nodes is ic if no node accepts, $(i-1)c$ if only node 1 accepts, $(i-2)c$ if node 2 accepts and no node j ($j > 2$) accepts, etc. The cost is zero if any node $j > i$ accepts.

$$\begin{aligned}
S_1 : p_1 &= \Psi_E(c\, q_2\, q_3\, q_4 \ldots q_n) \\
S_2 : p_2 &= \Psi_E(2c\, q_1\, q_3\, q_4 \ldots q_n + c\,(1-q_1)\, q_3\, q_4 \ldots q_n)
\end{aligned}$$

$$\vdots$$

$$\begin{aligned} S_i : p_i &= \Psi_E(ic\, q_1\, q_2 ... q_{i-1}\, q_{i+1} ... q_n + (i-1)c\,(1-q_1)\, q_2\, q_3 ... q_{i-1}\, q_{i+1} ... q_n + \\ &\quad (i-2)c\,(1-q_2)\, q_3\, q_4 ... q_{i-1}\, q_{i+1} ... q_n + ... + c\,(1-q_{i-1})\, q_{i+1} .... q_n) \\ &\vdots \\ S_n : p_n &= \Psi_E(nc\, q_1\, q_2 ... q_{n-1} + (n-1)c(1-q_1)\, q_2\, q_3 ... q_{n-1} + (n-2)c(1-q_2)\, q_3\, q_4 ... q_n + ... \\ &\quad +c(1-q_{n-1})) \end{aligned}$$

Each equation can be considered to be a surface (the surface S_i indicates the optimum price for node i relative to fixed prices $\mathbf{p}_{-i}$ for the other nodes). A price vector $\mathbf{p}$ is a fixed point of the map **REFINE** iff it is a point of intersection of all n surfaces.

For this paper a function $g(p_1, p_2, ..., p_{i-1}, p_{i+1}, ..., p_n)$ is called a *multiexponential* function if it is multilinear in the $q_j = 1 - e^{-\lambda p_i}$, $j \neq i$.

For a general tree the system of equations obtained from Lemma 1 is similar to the particular example above:

$$S_i : p_i = \Psi_E(h_i(p_1, p_2, ..., p_{i-1}, p_{i+1}, ..., p_n)) \qquad i = 1, ..., n \tag{1}$$

where the h_i are some multiexponential functions with positive coefficients.

Convergence

As noted before, we are interested in finding the following (l_∞) approximation to the optimum pricing

Definition 6 *For $\epsilon > 0$, a vector $\mathbf{p}$ is said to be an ϵ-approximation of the fixed point $\mathbf{p}^f$ if $|p_i - p_i^f| \leq \epsilon$, $\forall\ i = 1, ..., n$.*

We use a general method based on the norm of the partial derivatives matrix to prove that **REFINE** is contractive in a restricted case. This will imply that the optimum pricing is the unique fixed point, and that the iterations converge geometrically to it. A sketch of the proof is given in the appendix.

Theorem 4 *For any multicast tree with each edge of cost c, if the utilities of the nodes are taken independently from the exponential distribution with parameter λ, and if the costliest path has cost less than $\frac{1}{\lambda}$, then* **SIMPLE-ITERATIONS** *converge geometrically to the optimum price vector, i.e. we reach an ϵ-approximation of the profit maximizing price vector in $poly(n, log\frac{1}{\epsilon})$ time.*

4.2 The Uniform Distribution

In this section we assume that the utility of each node comes independently from the uniform distribution U on $[0, 1]$. Since the utilities come from the distribution U, we restrict our attention to price vectors $\mathbf{p} \in [0, 1]^n$.

As noted in Section 2, the single user profit maximizing function is $\Psi_U(c) = min\{1, \frac{(1+c)}{2}\}$.

The probability that node i accepts price p_i is $(1 - p_i)$. Using Lemma 1 we can obtain expressions for the relatively optimum price for each node.

For a general tree the system of equations obtained from Lemma 1 looks similar to that for the exponential distribution case:

$$S_i : p_i = \Psi_U(h_i(p_1, p_2, ..., p_{i-1}, p_{i+1}, ..., p_n)) \qquad i = 1, ..., n \tag{2}$$

where the h_i are some multilinear functions.

Each equation can be considered to be a surface (the surface S_i indicates the optimum price for node i relative to fixed prices $\mathbf{p}_{-i}$ for the other nodes). A price vector $\mathbf{p}$ is a fixed point of the map **REFINE** iff it is a point of intersection of all n surfaces.

Convergence

Consider an n node multicast tree with the utilities of the nodes drawn independently from the uniform distribution on $[0, 1]$. Suppose the edge costs are such that for some constant $K < 1$

$$S_i(1, 1, ...1) \leq K < 1, \quad i = 1, ..., n \tag{3}$$

Let the optimum price vector be $\mathbf{p}^f = (p_1^f, p_2^f, ..., p_n^f)$.

We prove that under assumption 3, **REFINE** is a contractive map and hence the iterations converge geometrically to the optimum pricing. A sketch of the proof is given in the appendix.

Theorem 5 *For any multicast tree if the utilities of the nodes are drawn independently from the uniform distribution on* $[0, 1]$*, and the edge costs are such that* $S_i(1, 1, ..., 1) \leq K < 1,\ i = 1, .., n$*, for some constant* K*, then, for any given* $\epsilon > 0$*,* **SIMPLE-ITERATIONS** *reach an* ϵ*-approximation of the profit maximizing price vector in time* $poly(n, log(1/\epsilon))$*.*

5 Discussion

Profit maximization in multicasting attempts to tread the fine line between over-charging, which could lead to loss of customers and under-charging, which results in lower revenues. A naive approach of assuming the utility of each user to be his expected utility and offering that as the price, can give an arbitrarily large approximation factor.

The main contribution of this paper is to suggest a fixed point approach for this optimization problem. Although we can prove convergence to an optimum pricing only in very special cases, simulations show that in fact simple iterations converge extremely fast to optimum pricing even for cases in which we have no proofs. Fixed point iterations have been studied extensively in the numerical analysis literature. In cases where we cannot prove fast convergence of the simple iterations, we can resort to more powerful techniques for finding fixed points of **REFINE**, e.g. Newton's methods or the algorithm in [Sca67].

The fixed point approach suggested here for multicast trees, can be used for any multivariate optimization problem in which we can compute the expected additional cost of adding one variable when the rest are fixed at any fixed values (i.e. when some result like Lemma 1 holds). Of course, convergence of the scheme will need a proof. However, computing the expected additional cost does not seem to be easy for multicasting on general DAGs.

A different approach to the problem on multicast trees is to discretize the prices and use dynamic programming. However, even in that approach it is not clear that we get an FPAS for reasonable distributions.

Finally we should point out that while the expected profit is an important quantity to maximize, other quantities have been investigated : e.g. minimizing the probability that the quantity we want to maximize is less than a certain value ([KRT97] use the expected value for the makespan problem, but have overflow probability as a parameter for bin-packing and knapsack). Other measures of stochastic optimization can be found in [Pin95].

It would be interesting to get simple proofs of convergence for a large class of distributions. Profit maximization for other problems can also be studied using the approach proposed here. Two problems that seem particularly attractive are minimum spanning tree and facility location.

Acknowledgments: We wish to thank Saugata Basu for help on uniqueness of the fixed point. We also wish to thank Milena Mihail, Christos Papadimitriou and Craig Tovey for discussions on understanding the complexity of our problem, as also Parikshit Gopalan and Amin Saberi for a careful reading and for several important comments on the results.

References

[BBABI98] J. Bottom, M.E. Ben-Akiva, M. Bierlaire, and Chabini I. Generation of consistent anticipatory route guidance. *Presented at the Tristan III Conference, San Juan, Puerto Rico*, 1998.

[BC01] M. Bierlaire and F. Crittin. Newalgorithmic approaches for the anticipatory route guidance generation problem. *Proceedings of the 1st Swiss Transportation Research Conference*, 2001.

[DC90] Stephen E. Deering and David R. Cheriton. Multicast routing in datagram internetworks and extended LANs. *ACM Transactions on Computer Systems*, 8(2):85–110, May 1990.

[DEF+94] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C.-G. Liu, and L. Wei. An architecture for wide-area multicast routing. 24(4), October 1994. SIGCOMM '94 Symposium.

[EWK90] Lars Eldeń. and Linde Wittmeyer-Koch. *Numerical Analysis - An Introduction.* Academic Press, 1990.

[FGHK02] A. Fiat, A. Goldberg, J. Hartline, and A. Karlin. Competitive generalized auctions. *STOC*, 2002.

[FPS01] J. Feigenbaum, C. Papadimitriou, and S. Shenker. Sharing the cost of multicast transmissions. *J. Computer an System Sciences*, 63, 2001.

[GHW01] A. Goldberg, J. Hartline, and A. Wright. Competitive auctions and digital goods. *SODA*, 2001.

[JV01] Kamal Jain and Vijay V. Vazirani. Applications of approximation algorithms to cooperative games. In *ACM Symposium on Theory of Computing*, 2001.

[KRT97] J. Kleinberg, Y. Rabani, and E. Tardos. Allocating bandwidth for bursty connections. *STOC*, 1997.

[MS01] H. Moulin and S. Shenker. Strategyproof sharing of submodular costs: Budget-balance vs. efficiency. *Economic Theory*, 18, 2001.

[NR99] N. Nisan and A. Ronen. Algorithmic mechanism design. In *Proceedings of STOC*, 1999.

[OR70] J.M. Ortega and W.C. Rheinboldt. *Iterative solution of nonlinear equations in several variables.* Academic Press, 1970.

[Pin95] M. Pinedo. *Scheduling - Theory, Algorithms, and Systems.* Prentice Hall, 1995.

[Sca67] H. Scarf. The approximation of fixed points of a continuous mapping. *SIAM Journal of Applied Mathematics*, 15, 1967.

A Sketch of Proof of Theorem 4

We use the following standard result (see e.g. [EWK90]) to prove geometric convergence to an optimum pricing under certain assumptions on the edge costs.

Let $\phi : \mathbf{R}_+^n \to \mathbf{R}_+^n$ be a map such that $\phi(\mathbf{p}) = (\phi_1(\mathbf{p}), \phi_2(\mathbf{p}), ..., \phi_n(\mathbf{p}))$. Let $\mathbf{p}^f$ be a fixed point of ϕ, i.e. $\phi(\mathbf{p}^f) = \mathbf{p}^f$. Consider iterations $\mathbf{p^1}, \mathbf{p^2}, ..., \mathbf{p^k}, ...$ of ϕ starting at some initial point $\mathbf{p^1}$. Assume that the partial derivatives $m_{ij} = \frac{\partial \phi_i}{\partial p_j}$ exist in a region R around $\mathbf{p}^f$ in which the iterations run. Let M be the $n \times n$ matrix (m_{ij}).

We say that the iterations converge geometrically to the fixed point if $|\mathbf{p^{k+1}} - \mathbf{p}^f| \leq \beta |\mathbf{p^k} - \mathbf{p}^f|$, for some $\beta < 1$. Geometric convergence implies that we reach an ϵ-approximation of the fixed point in $O(log \frac{1}{\epsilon})$ iterations.

Lemma 6 *If for some matrix norm, $\|M\| \leq m < 1$ for all points in R, then the iterations converge geometrically to the fixed point $\mathbf{p}^f$ in the corresponding vector norm.*

The infinity norm of the matrix M is

$$\|M\|_\infty = max_v \frac{|Mv|_\infty}{|v|_\infty} = max_{1 \leq i \leq n} \left(\sum_{j=1}^{n} |m_{ij}| \right)$$

Suppose that we are given a multicast tree with each edge weight equal to c. Note that for any i if we set all prices in $\mathbf{p}_{-i}$ to ∞, then the relatively optimum price for node i will be less than or equal to $D * c + \frac{1}{\lambda}$, where D is the depth of the tree. On the other hand the iterations will never set the price for a node to less than $\frac{1}{\lambda}$. Thus we can take R to be the region $\{\mathbf{p} | \frac{1}{\lambda} \leq p_i \leq Dc + \frac{1}{\lambda}\}$ and start our iterations with $p_i = Dc + \frac{1}{\lambda}$, $i = 1, .., n$.

By bounding each of the prices by $\frac{1}{\lambda}$ from below and $Dc + \frac{1}{\lambda}$ from above, and assuming that $Dc < \frac{1}{\lambda}$, we can show that $\|M\|_\infty \leq m < 1$, for some constant m. Hence the map is contractive, and the iterations converge to the optimum price vector.

$\square$(Theorem 4)

B Sketch of proof of Theorem 5

By the change of coordinates $\hat{p}_i = p_i - p_i^f,\ i = 1, ..., n$ we can make the equation of each surface have constant term zero, while remaining multilinear. In the new coordinates, the iterations start with the initial price vector $(1 - p_1^f, 1 - p_2^f, ..., 1 - p_n^f)$ and the fixed point is at the origin. We wish to prove that the iterations converge geometrically to the origin.

Due to condition (3), at the end of the first iteration $\hat{p}_i \leq K - p_i^f,\ i = 1, ..., n$, i.e. each $\hat{p}_i$ drops by a multiplicative factor of at least $\frac{K - p_i^f}{1 - p_i^f} \leq K$.

Since each of the equations is multilinear with constant term zero, at the end of the second iteration each variable drops further by a factor of at least K. It is easy to see that every iteration will result in a drop by a factor of at least K, and so each variable converges geometrically to zero, i.e. the iterations converge geometrically to the (unique) fixed point.

Every iteration takes $O(n^2)$ time, and we need $log_{1/K}(1/\epsilon)$ iterations to reach an ϵ-approximation of the fixed point. This proves the theorem. $\square$(Theorem 5)